

UNIVERSIDAD PERUANA UNIÓN
FACULTAD DE INGENIERÍA Y ARQUITECTURA
Escuela profesional de Ingeniería de Sistemas



Una Institución Adventista

“Implementación de un sistema de información web bajo la metodología de desarrollo ágil OpenUP y la NTP/IEC 12207 para la gestión y evaluación de los grupos organizados de la Iglesia Universitaria Villa Unión de la Universidad Peruana Unión, filial Tarapoto”

Por:
Persy Quiroz Menor

Asesor:
Mg. Esteban Tocto Cano

TARAPOTO – PERÚ
OCTUBRE, 2016

Cómo citar:

Estilo APA

Quiroz, M. (2016). Implementación de un sistema de información web bajo la metodología de desarrollo ágil OpenUP y la NTP/IEC 12207 para la gestión y evaluación de los grupos organizados de la Iglesia Universitaria Villa Unión de la Universidad Peruana Unión, filial Tarapoto (Tesis de pregrado). Universidad Peruana Unión, Tarapoto – Perú.

Estilo Vancouver

Quiroz P. Implementación de un sistema de información web bajo la metodología de desarrollo ágil OpenUP y la NTP/IEC 12207 para la gestión y evaluación de los grupos organizados de la Iglesia Universitaria Villa Unión de la Universidad Peruana Unión, filial Tarapoto (Tesis de pregrado). Tarapoto – Perú: Universidad Peruana Unión, 2016.

Artículos publicados:

Sistema de información bajo la metodología Ágil OpenUP para la gestión y evaluación de los grupos organizados de la Iglesia Adventista Universitaria Villa Unión Tarapoto. *Revista de Investigación Ciencia, Tecnología y Desarrollo*.

Presentación en eventos científicos:

Ficha catalográfica elaborada por el Centro de Recursos para el Aprendizaje y la Investigación – CRAI – de la UPeU

TIS 2 Q73 2017	Quiroz Menor, Persy Implementación de un sistema de información web bajo la metodología de desarrollo ágil OpenUP y la NTP/IEC 12207 para la gestión y evaluación de los grupos organizados de la Iglesia Universitaria Villa Unión de la Universidad Peruana Unión, filial Tarapoto/ Autor: Persy Quiroz Menor; Asesor: Mg. Esteban Tocto Cano. -- Tarapoto, 2016. 267 páginas: anexos, tablas, figuras Tesis (Licenciatura)--Universidad Peruana Unión. Facultad de Ingeniería y Arquitectura. EP. Ingeniería de Sistemas, 2016. Incluye referencias y resumen. Campo del conocimiento: Ingeniería de sistemas. 1. Metodologías ágiles. 2. OpenUP. 3. Base de datos. 4. Singleton. 5. PostgreSQL
-----------------------------------	---

DECLARACIÓN JURADA DE AUTORÍA DEL INFORME DE TESIS

Esteban Tocto Cano de la Facultad de Ingeniería y Arquitectura, Escuela Profesional de Ingeniería de Sistemas, de la Universidad Peruana Unión.

HACE CONSTAR:

Que el presente informe de investigación titulado: *"Implementación de un sistema de información web bajo la metodología de desarrollo ágil OpenUP y la NTP/IEC 12207 para la gestión y evaluación de los grupos organizados de la Iglesia Universitaria Villa Unión de la Universidad Peruana Unión Filial Tarapoto"*, constituye la memoria que presenta el Bachiller **Persy Quiroz Menor** para aspirar al título de Profesional de Ingeniero de Sistemas, ha sido realizada en la Universidad Peruana Unión bajo mi dirección.

Las opiniones y declaraciones en este informe son de entera responsabilidad del autor, sin comprometer a la institución.

Y estando de acuerdo, firmo la presente constancia en Lima, a los cuatro días de noviembre del 2016.



Mg. Esteban Tocto Cano

Implementación de un sistema de información web bajo la metodología de desarrollo ágil OpenUP y la NTP/IEC 12207 para la gestión y evaluación de los grupos organizados de la Iglesia Universitaria Villa Unión de la Universidad Peruana Unión, filial Tarapoto

TESIS

Presentada para optar el título profesional de ingeniero de sistemas

JURADO CALIFICADOR


Mg. Danny Lévano Rodríguez
Presidente


Ing. Jenson Daniel Chambi Aguilar
Secretario


Ing. Joel Pérez Suárez
Vocal


Ing. Miguel Ángel Valles Coral
Vocal


Mg. Esteban Tocto Cano
Asesor

Tarapoto, 14 de octubre de 2016

Dedicatoria

A mis padres Josué y Edita por su apoyo incondicional, a mis 7 hermanos: Nora, Ruth, Dany, Josué Omar, Keyla, Mélody Brilly y Josué Antony por su confianza en mí como hermano y por ser para ellos una inspiración de lograr sus sueños.

Agradecimientos

En primer lugar agradezco al dador de la vida: Dios el Creador.

A la Universidad Peruana Unión, filial Tarapoto y Bienestar Universitario por su apoyo en la realización de este proyecto y por cederme todas las facilidades para la obtención de requerimientos y por confiar en uno de sus egresados.

A la Iglesia Universitaria Villa Unión y los capellanes de cada facultad por el tiempo dedicado en las reuniones y capacitaciones realizadas, y por organizar las capacitaciones a los líderes de los grupos con el uso del sistema.

A la Facultad de Ingeniería y Arquitectura y a los ingenieros por el apoyo a este proyecto, y todos mis docentes que dieron lo mejor para mi formación profesional.

A cada líder de grupo organizado por el sueño de tener un sistema de información para gestionar un grupo y por su liderazgo.

Al magister Esteban Tocto Cano por la ayuda en el desarrollo del proyecto de investigación y por su asesoramiento en la elaboración del proyecto y el informe de tesis.

Índice general

RESUMEN	xvii
ABSTRACT	xviii
CAPÍTULO I INTRODUCCIÓN	19
CAPÍTULO II MARCO TEÓRICO	25
2.1 Introducción	25
2.2 Grupo organizado y liderazgo	25
2.2.1 Liderazgo.....	27
2.2.2 Trabajo en equipo.....	29
2.3 Sistema de información (SI).....	31
2.3.1 Tipos de sistemas de información	32
2.4 Metodologías ágiles.....	35
2.4.1 El manifiesto ágil	35
2.4.2 XP (Programación Extrema)	42
2.4.3 Enfoque de Reglas de Negocio	46
2.4.4 SCRUM.....	48
2.4.5 Open Unified Process (OpenUP)	52
2.4.6 Comparación de metodologías ágiles.....	65
2.5 Lenguajes de programación	67
2.5.1 Lenguajes de alto nivel.....	68
2.5.2 Comparación entre lenguajes de programación	74
2.6 Bases de datos	76
2.6.1 Sistemas de gestión de Bases de Datos (SGDB / DBMS)	76
2.6.2 Clasificación de motores de base de datos	86
2.7 Servidor web	89
2.7.1 Apache Tomcat	89
2.8 Organización Internacional de Normalización.....	89
2.8.1 NTP ISO/IEC 12207	90
2.9 Integración de la NTP ISO/IEC 12207 y la metodología ágil OpenUP.....	93
CAPÍTULO III MATERIALES Y MÉTODOS.	96
3.1 Introducción	96
3.2 Diseño de la investigación.....	96
3.2.1 Análisis situacional de los grupos	97

3.2.2	Identificación de requerimientos	97
3.2.3	Aplicación de la metodología y la NTP ISO/IEC 12207	97
3.2.4	Implantación de la solución y seguimiento	98
3.3	Tipo de investigación	100
CAPÍTULO IV CONSTRUCCIÓN DE LA PROPUESTA.....		101
4.1	Introducción	101
4.2	Análisis situacional de los grupos	101
4.2.1	Información de los Grupos organizados.....	101
4.2.2	Información de los Grupos organizados.....	104
4.3	Concepción.....	105
4.3.1	Visión del producto	105
4.4	Elaboración.....	126
4.4.1	Introducción	126
4.4.2	Arquitectura de la solución	126
4.5	Construcción.....	131
4.5.1	Introducción	131
4.5.2	Lenguaje de programación	131
4.5.3	IDE Netbeans	131
4.5.4	Bootstrap	132
4.5.5	Librerías Java-Mail	133
4.5.6	Gestor de base de PostgreSQL.....	134
4.5.7	Conexión a base de datos	135
4.5.8	Servidor web	140
4.5.9	Proceso de desarrollo de la Norma Técnica Peruana ISO/IEC 12007	140
4.5.1	Iteraciones de la Construcción	142
4.6	Transición.....	147
CAPÍTULO V RESULTADOS Y DISCUSIÓN.....		148
1.1.	Introducción	148
5.1	Resultados	148
5.1.1	Inicio de sesión.....	148
5.1.2	Menú principal	149
5.1.3	Registro de integrantes.....	150

5.1.4	Administrar datos del grupo.....	155
5.1.5	Cargos del grupo	155
5.1.6	Eventos	156
5.1.7	Sub grupos.....	160
5.1.8	Control de cursos bíblicos	161
5.1.9	Definición de indicadores (metas).....	162
5.1.10	Reportes de gestión	162
5.1.11	Configuraciones	174
5.1.12	Seguridad.....	175
5.1.13	Más opciones.....	177
5.1.14	Encuesta a líderes de grupos pequeños	178
5.2	Conclusiones	189
5.3	Recomendaciones y trabajos futuros.....	190
	REFERENCIAS.....	191
	ANEXOS	195
	APÉNDICES.....	200

Índice de tablas

Tabla 1 <i>Roles de equipo</i>	29
Tabla 2 <i>Diferencias entre metodologías ágiles y tradicionales</i>	41
Tabla 3 <i>Tareas de OpenUP</i>	57
Tabla 4 <i>Artefactos de OpenUP</i>	58
Tabla 5 <i>Subprocesos del ciclo de vida del proceso de desarrollo OpenUP</i>	64
Tabla 6 <i>Entregables del proyecto en OpenUP</i>	65
Tabla 7 <i>Tipo de datos básicos y variables de Java</i>	69
Tabla 8 <i>Nivel de popularidad de los lenguajes de programación</i>	75
Tabla 9 <i>Tipos de sentencias SQL</i>	78
Tabla 10 <i>Ventajas y desventajas de los principales gestores de bases de datos</i>	84
Tabla 11 <i>Nivel de popularidad de Base de Datos</i>	88
Tabla 12 <i>Integración de NTP 12207 y la metodología OpenUP</i>	94
Tabla 13 <i>Requerimientos funcionales del sistema</i>	106
Tabla 14 <i>Nivel de dificultad</i>	111
Tabla 15 <i>Nivel de prioridad</i>	111
Tabla 16 <i>Requerimientos no funcionales del sistema</i>	111
Tabla 17 <i>Costo de recursos humanos del proyecto</i>	118
Tabla 18 <i>Costo referencial del proyecto</i>	119
Tabla 19 <i>Análisis de costo aproximado</i>	119
Tabla 20 <i>Implementación del patrón de diseño Singleton</i>	136
Tabla 21 <i>Actividades y tareas desarrolladas según NTP ISO/IEC 12207</i>	141
Tabla 22 <i>Iteración 1 - Autenticación y registro de Grupos</i>	142
Tabla 23 <i>Iteración 2 - Registro de personas e inscripción de integrantes al grupo</i>	143
Tabla 24 <i>Iteración 3 - Configuración de cargos, datos del grupo, entre otros</i>	144
Tabla 25 <i>Iteración 4 - Registros de eventos, elaboración del programa, asistencia y evaluación</i>	144
Tabla 26 <i>Iteración 5 - Sub grupos, estudiantes de la Biblia, estudios bíblicos e indicadores ...</i>	145
Tabla 27 <i>Iteración 6 - Usuarios, perfiles, módulos y permisos</i>	146
Tabla 28 <i>Iteración 7 - Lista de reportes del sistema</i>	146

Índice de figuras

<i>Figura 1.</i> Modelo general de un sistema de información - Fernández (2010).	31
<i>Figura 2.</i> Diagrama del ciclo de Scrum (Palacio & Ruata, 2011).	48
<i>Figura 3.</i> Capas de OpenUP: micro-incrementos, ciclo de vida y la iteración del ciclo de vida del proyecto. Fuente, Eclipse (2016)	53
<i>Figura 4.</i> Principales roles en OpenUP y sus interacciones (Eclipse, 2016).....	54
<i>Figura 5.</i> Ranking de lenguajes de programación según TIOBE Index for August 2016.....	75
<i>Figura 6.</i> Niveles de popularidad de los gestores de Base de Datos - DB-Engines (agosto 2016)	87
<i>Figura 7.</i> Estructura de la Norma Técnica Peruana 12207. Fuente: NTP ISO/IEC 12207	92
<i>Figura 8.</i> Metodología ágil OpenUP y NTP 12207.....	95
<i>Figura 9.</i> Diseño de la investigación – elaboración propia	96
<i>Figura 10.</i> Organización de los grupos en la Iglesia Universitaria Villa Unión	102
<i>Figura 11.</i> Organización de la iglesia, escuelas sabáticas y grupos pequeños en 2015-1. Fuente: elaboración propia.....	104
<i>Figura 12.</i> Roles de usuario del sistema.....	114
<i>Figura 13.</i> Diagrama de paquetes del sistema	122
<i>Figura 14.</i> Diagrama de clase de análisis – Módulo Seguridad	122
<i>Figura 15.</i> Diagrama de clases de análisis – Módulo Registros.....	123
<i>Figura 16.</i> Diagrama de clases de análisis – Módulo Administrar, gestión de eventos y evaluaciones	124
<i>Figura 17.</i> Diagrama de clases de análisis – Módulo configuraciones	125
<i>Figura 18.</i> Patrón de diseño Modelo – Vista – Controlador.....	128
<i>Figura 19.</i> Diagrama despliegue.....	128
<i>Figura 20.</i> Diagrama de Base de Datos - Parte 1. Fuente: elaboración propia.....	129
<i>Figura 21.</i> Diagrama de base de datos - Parte 2. Fuente: elaboración propia	130
<i>Figura 22.</i> Vista del IDE Netbeans.....	132
<i>Figura 23.</i> Página de inicio, vista en pantalla grande y pequeña con la tecnología Bootstrap...	133
<i>Figura 24.</i> Método para envío de correos, haciendo uso de Java-mail	134
<i>Figura 25.</i> Panel de administración de base de datos PostgreSQL con pgAdmin	135
<i>Figura 26.</i> Página inicio de sesión.....	149
<i>Figura 27.</i> Menú principal al ingresar al sistema (rol: líder de grupo).....	150
<i>Figura 28.</i> Gestión de integrantes del grupo	151
<i>Figura 29.</i> Visualización de los datos del integrante.....	151

<i>Figura 30.</i> Actualización de datos del integrante	152
<i>Figura 31.</i> Búsqueda e inscripción de personas al grupo	153
<i>Figura 32.</i> Lista para impresión de integrantes del grupo	154
<i>Figura 33.</i> Ficha del integrante (reporte de asistencias, participaciones e incidencias).....	154
<i>Figura 34.</i> Opción datos del grupo, módulo administrar.....	155
<i>Figura 35.</i> Gestión de cargos en el grupo.....	156
<i>Figura 36.</i> Registro de eventos por grupo	156
<i>Figura 37.</i> Añadir actividad – registro del programa	157
<i>Figura 38.</i> Copiar programa sugerente – registro del programa	158
<i>Figura 39.</i> Envío de mensajes de participación.....	158
<i>Figura 40.</i> Mensaje de participación que le llega a cada participante.....	159
<i>Figura 41.</i> Formato de impresión del programa.....	159
<i>Figura 42.</i> Registro de asistencia al evento	160
<i>Figura 43.</i> Administrar sub grupos.....	161
<i>Figura 44.</i> Control de estudios biblicos.....	161
<i>Figura 45.</i> Definición de indicadores	162
<i>Figura 46.</i> Lista de grupos organizados	163
<i>Figura 47.</i> Reporte de asistencia a cada grupo	163
<i>Figura 48.</i> Reporte documentos de la persona	164
<i>Figura 49.</i> Monitoreo mi grupo	165
<i>Figura 50.</i> Análisis de asistencias del grupo	165
<i>Figura 51.</i> Termómetro de la Escuela Sabática.....	166
<i>Figura 52.</i> Reporte de grupos e integrantes.....	167
<i>Figura 53.</i> Asistencia a los grupos por fecha	168
<i>Figura 54.</i> Estadísticas generales (reporte)	168
<i>Figura 55.</i> Lista de persona por número de asistencias.....	169
<i>Figura 56.</i> Análisis de asistencias	170
<i>Figura 57.</i> Termómetro de la Escuela Sabtica – Gestión	171
<i>Figura 58.</i> Monitoreo general de los grupos	172
<i>Figura 59.</i> Porcentaje de estados de asistencia por entidad académica.....	174
<i>Figura 60.</i> Perfiles o roles del sistema - Seguridad.....	175
<i>Figura 61.</i> Asignación de accesos a los perfiles.....	176
<i>Figura 62.</i> Usuarios del sistema (seguridad).....	177

Figura 63. Lugares de reunión con puntos de referencia de Google Maps 178

Índice de gráficos

<i>Gráfico 1.</i> Pregunta 1, líderes que no usan el sistema	179
<i>Gráfico 2.</i> Pregunta 1, líderes que usan el sistema	179
<i>Gráfico 3.</i> Pregunta 2, líderes que no usaron el sistema.....	180
<i>Gráfico 4.</i> Pregunta 2, líderes que usan el sistema	180
<i>Gráfico 5.</i> Pregunta 3, líderes que no usan el sistema	181
<i>Gráfico 6.</i> Pregunta 3., líderes que usan el sistema	181
<i>Gráfico 7.</i> Pregunta 4, líderes que usan el sistema	182
<i>Gráfico 8.</i> Pregunta 4, líderes que usan el sistema	182
<i>Gráfico 9.</i> Pregunta 5, líderes que no usan el sistema	183
<i>Gráfico 10.</i> Pregunta 5, líderes que usan el sistema	183
<i>Gráfico 11.</i> Pregunta 6, líderes que no usan el sistema	184
<i>Gráfico 12.</i> Pregunta 6, líderes que usan el sistema	184
<i>Gráfico 13.</i> Pregunta 7, líderes que no usan el sistema	185
<i>Gráfico 14.</i> Pregunta 7, líderes que usan el sistema	185
<i>Gráfico 15.</i> Pregunta 8, líderes que no usan el sistema	186
<i>Gráfico 16.</i> Pregunta 8, líderes que usan el sistema	186
<i>Gráfico 17.</i> Pregunta 9, líderes que no usan el sistema	187
<i>Gráfico 18.</i> Pregunta 9, líderes que usan el sistema	187
<i>Gráfico 19.</i> Pregunta 10, líderes que no usan el sistema	188
<i>Gráfico 20.</i> Pregunta 10, líderes que usan el sistema	188

Índice de anexos

<i>Anexo 1.</i> Instrumento (encuesta) aplicado a líderes, para conocer la situación actual	195
<i>Anexo 2.</i> Instrumento (encuesta) aplicado a líderes, para conocer em mejoramiento de la gestión, tras la implemnetación del sistema de información	196
<i>Anexo 3.</i> Reconocimiento de la Iglesia Universitaria Villa Unión	197
<i>Anexo 4.</i> Reconocimiento de capellanía de la Facultad de Ingeniería y Arquitectura.....	198
<i>Anexo 5.</i> Reconocimiento de capellanía de la Facultad de Ciencias de la Salud.....	199

Símbolos y términos usados

NTP	Norma Técnica Peruana
KPI	Key Performance Indicator
ISO	Organización Internacional de Normalización
IEC	International Electrotechnical Commission
INDECOPI	Instituto Nacional de Defensa de la Competencia y de la Protección de la Propiedad Intelectual
SGO	Sistema de Grupos Organizados
OpenUP	Proceso Unificado Abierto (Metodología ágil)
UPeU FT	Universidad Peruana Unión Filial Tarapoto
MVC	Modelo Vista Controlador (Patrón de diseño)
GPO	Grupo Organizado
SI	Sistema de información
TSP	Transaction Processing Systems
ESS	Executive Support Systems
MIS	Management Information Systems
DSS	Decision Support Systems
AI	Artificial Intelligence
XP	Programación Extrema
SGDB / DBMS	Sistemas de gestión de Bases de Datos
SQL	Structured Query Language
DDL	Data Definition Language
DML	Data Manipulation Language
DCL	Data Control Language
TCL	Transaction Control Language
URL	Uniform Resource Locator
HTTP	Hypertext Transfer Protocol
JSP	Java Server Pages
IDE	Entorno de Desarrollo Integrado
HTML	HyperText Markup Language
CSS	Cascading Style Sheets

RESUMEN

El propósito de la investigación es implementar un sistema de información web bajo la metodología de desarrollo ágil Unified Process (OpenUP) y la NTP/IEC 12207 para la gestión y evaluación de los grupos organizados de la Iglesia Universitaria Villa Unión de la Universidad Peruana Unión, filial Tarapoto. El sistema de información se desarrolló integrando el proceso de desarrollo de la NTP ISO/IEC 12207 en las fases de concepción, elaboración, construcción y transición de la metodología de desarrollo ágil OpenUP del Eclipse Process Framework, con el objeto de obtener un software que responda a estándares de calidad y cumpla con los requerimientos de los usuarios. Se aplicó el patrón de diseño de la arquitectura Modelo Vista Controlador (MVC) con la Programación Orientada a Objetos, y conexión a base de datos PostgreSQL 9.0 mediante el patrón de diseño Singleton, que permite dar una mejor respuesta a las peticiones del sistema. EL sistema de información fue desarrollado en el lenguaje de programación Java, uso de JavaScript, estilos CSS3 y HTML5. El sistema de información contempla las siguientes funcionalidades: registro de grupos organizados, registro de integrantes, gestión de cargos, registro de eventos, elaboración de programas de cada evento, control de asistencias, registros de sub grupos, registro de estudiantes de la Biblia, proceso de evaluación de los miembros del grupo, registro de bautismos, registro de incidencias y registro de publicaciones. La investigación es de tipo descriptiva porque recopiló y sistematizó la información de los grupos organizados, además, es de tipo aplicada, porque está orientada a un giro de negocio. El sistema de información fue validado por los usuarios del sistema que son los líderes de la organización eclesiástica. En conclusión este sistema de información sirve de soporte para la adecuada gestión de los grupos organizados.

Palabras claves: Metodologías ágiles, OpenUP, Bases de datos, Singleton, PostgreSQL.

ABSTRACT

The purpose of this research is to implement a web information system under the development methodology agile Unified Process (OpenUP) and the NTP/IEC 12207 for the management and evaluation of the organized groups from the University Church Villa Union of the Universidad Peruana Unión Tarapoto subsidiary. The information system was developed integrating the process of development of the NTP ISO/IEC 12207 in the conception, elaboration, construction and transition phases of the development methodology agile OpenUP of the Eclipse Process Framework, in order to obtain a software that responds to quality standards and meets the users requirements. The design pattern of the architecture Model View Controller (MVC) was applied with Object Oriented Programming, and connection to the PostgreSQL 9.0 database using the Singleton design pattern, which allows a better response to the system requests. The information system was developed in the Java programming language, using JavaScript, CSS3 and HTML5 styles. The information system includes the following functionalities: organized groups registration, members registration, management of responsibilities, events registration, programs elaboration for each event, attendance control, sub groups registration, Bible students registration, process of Evaluation of group members, baptisms registration, incidents and publications registration. The research is of descriptive type, because it compiled and systematized the information of the organized groups, in addition, it's of the applied type, because it is oriented to a business turnaround. The information system was validated by the system users who are the leaders of the ecclesiastical organization. In conclusion, this information system works as support for the proper management of organized groups.

Keywords. Methodology agile, OpenUP, Databases, Singleton, PostgreSQL.

CAPÍTULO I

INTRODUCCIÓN

En la actualidad existen varios proyectos desarrollados que están orientados a la evaluación de las personas y la gestión de grupos con integrantes como las redes sociales. Usualmente, se usan para la comunicación y publicación de contenidos entre integrantes. La Iglesia Universitaria Villa Unión está organizada en grupos organizados. En entrevistas realizadas a Jimenez, Fernández, & Hanco (Entrevista personal Abril 20, 2015) mencionaron los problemas que existen en el proceso de los grupos organizados tales como:

- Duplicidad de data de los miembros de los grupos organizados: por cada semestre académico los líderes de los grupos organizados vuelven a inscribir a sus integrantes; eso lleva a recabar nuevamente la información de cada miembro, trayendo consigo incomodidad a los miembros y generando pérdida de tiempo en el proceso de inscripción; además, no se logra obtener toda la información que se necesitan para dar seguimiento a cada integrante.
- Miembros de los grupos organizados inscritos en diferentes grupos: existe duplicidad de inscritos; es decir, los miembros se inscriben en diferentes grupos, esto genera malestar a los líderes e integrantes, se pierde el conteo exacto de los miembros a nivel general.
- Los líderes de la iglesia cuentan con reducida información para la asignación de nuevos líderes: no se cuentan con un historial por persona para la selección de nuevos líderes de los grupos organizados por año o ciclo académico.

- Dificultades en la identificación de lugares de reunión de los grupos organizados: cuando inician las actividades de los grupos, los integrantes desconocen el lugar de reunión, por ello trae malestar a los participantes por la falta de información de cada reunión.
- Los líderes de la iglesia no disponen de un historial de los participantes y herramientas o mecanismos para dar seguimiento al desarrollo de actividades en los grupos organizados.
- Cruce de actividades en los planes de trabajo en los grupos organizados.
- Información no estructurada para reportar avance de actividades de los grupos organizados.
- Poca planificación del programa para cada evento, generando imprevistos en el momento del desarrollo de actividades del programa.

Para la solución de la problemática antes mencionada, es importante describir algunos proyectos que se hicieron para resolver problemas comunes. El Departamento de Informática y Sistemas de la Universidad de Las Palmas en Gran Canaria, España, implementó un sistema de información llamando “El Gestor de Coevaluación Orientado a Grupos, una herramienta de apoyo a la participación del alumno en el proceso de evaluación”. Este trabajo es una herramienta de software que permite la evaluación entre iguales inscritos en grupos. Las características más destacadas de este módulo son: permitir editar un cuestionario de Coevaluación, visualizar el cuestionario adaptado a cada grupo, responder al cuestionario y publicar los resultados de cada estudiante en el libro de calificaciones. Esta herramienta permitió ahorrar tiempo en la creación de cada actividad y en el cálculo de resultados (Rodríguez & Díaz, 2012). Además, mejoró el proceso de evaluación a cada integrante (alumno) y al mismo instante el evaluado puede acceder a los resultados en tiempo real porque la información está centralizada.

Así también, en el artículo “Indicador de estilo del Aprendizaje Preferencial Complementario”, fue automatizada vía web un Text Indicador del estilo de aprendizaje. Esto permitió identificar el estilo de aprendizaje de una persona, optimizando el tiempo en la evaluación del cuestionario tradicional (Díaz, Gil, & Santana, 2014). La implementación vía web permitió obtener los datos en tiempo real, así mismo mejorar el tiempo en la publicación de los resultados. Esta solución está orientada hacia las personas que miden su estilo de aprendizaje.

En inevitable dejar de mencionar a Facebook, siendo una de redes sociales más populares en el mundo, que permite la comunicación entre personas y la difusión de diferentes tipos de contenidos. Thompson (2012) dijo: “Facebook tiene un gran tipo, activa y leal audiencia que tiene que ser adoptado como un canal”. Facebook provee la funcionalidad de creación y gestión de grupos, empezando por la inscripción básica de miembros al grupo. Según Facebook (2016), los administradores del grupo pueden subir o cambiar las fotos de portada. Los miembros del grupo pueden compartir comentarios, fotos y videos, crear eventos; además, se puede acceder a un chat del grupo, asimismo, eliminar o bloquear a miembros con conductas ofensivas. Facebook permite la creación de eventos donde se añaden una foto, día, hora, lugar, descripción y organizadores. La desventaja de Facebook es que no se pueden obtener toda la información de los integrantes para gestionar los grupos de manera optimizada por medio de indicadores; además, no permite el registro de asistencia por estado, la elaboración del programa de cada evento y la evaluación a los asistentes.

La solución planteada en la investigación pudo resolver la problemática existente. Como producto de la investigación se desarrolló un sistema de información que aporta beneficios detallados a continuación.

En ciencia y tecnología: aportó desde las técnicas tales como el cuestionario para recopilación de la información, las entrevistas y el análisis de la información; además, se generaron nuevos conocimientos fortaleciendo así como las creencias en la administración de base de datos por medio del modelado los procesos de los grupos organizados; así mismo, la solución tecnológica reforzó las propuestas de la ingeniería de software porque se aplicaron patrones de diseño como MVC y Singleton. Desde la arista de procesos, el aporte fue la documentación de los procedimientos que se ejecutaron en las soluciones descritas en los casos de uso del negocio.

Entre los beneficiados directos de este sistema de información están: los líderes de los grupos, porque les permitió administrar a cada grupo organizado por medio de la inscripción de los integrantes, registro de cargos, el registro de los eventos (reunión de grupo, escuela sabática, entre otros); así como el registro de asistencia y la programación de cada evento. Asimismo, los capellanes de facultad monitorearon a cada grupo organizado accediendo a los reportes relacionados con la asistencia y la participación de los miembros; además, poseen la información de las parejas misioneras y los estudiantes de la Biblia de cada facultad. El pastor de iglesia accede a toda la información de la asistencia y las participaciones de sus feligreses, y monitorea la dinámica de los diferentes tipos de grupos. Entre los centros de aplicación se encuentran el área de Bienestar Universitario y la Iglesia Universitaria Villa Unión, tales tienen indicadores fiables para medir la calidad de servicio que brindan. Además, lograron tener un mejor control de cada evento; se ahorró tiempo en el proceso de evaluación; la calidad de resultados para la toma de decisiones mejoró. También mejoró la comunicación con los líderes de la institución y se logró reconocer con facilidad a los estudiantes con problemas espirituales accediendo a una ficha histórica por cada miembro del grupo.

Entre la comunidad beneficiada se encuentran, los consejeros de los grupos porque tienen la información automatizada para monitorear a sus aconsejados, pueden identificar a sus miembros con mayores faltas y con necesidades, esto les permite trabajar para ayudarlos a tener un mejor desempeño. El tiempo destinado a la evaluación se optimizó y teniendo así la información en tiempo real. Lograron visualizar las participaciones de cada evento que se llevaba a cabo. El área de Finanzas Alumnos brinda la beca de feligresía a los miembros (alumnos) que realmente son merecedores por su participación y desempeño en los grupos. Las facultades y escuelas académicas pudieron identificar a los alumnos con mejor desempeño en sus grupos organizados. La universidad mejoró en la programación de los eventos y reuniones espirituales.

En conclusión, el objetivo general de la investigación fue “Implementar un sistema de información web bajo la metodología de desarrollo ágil OpenUP y la NTP ISO/IEC 12207 para la gestión y evaluación de los grupos organizados de la Iglesia Universitaria Villa Unión de la Universidad Peruana Unión, filial Tarapoto” y los objetivos específicos fueron: realizar diagnóstico situacional de los grupos organizados, aplicar las interacciones de la metodología ágil OpenUP y validar e integrar la metodología ágil con el proceso de desarrollo de la NTP ISO/IEC 12207.

El estudio está estructurado en cinco capítulos, de la siguiente manera:

CAPÍTULO I. INTRODUCCIÓN: presenta la introducción al informe de investigación. Se describe la problemática, antecedentes, objetivos y justificación sobre la dinámica de los grupos

organizados. Se describe la situación de la Iglesia Universitaria Villa Unión con la dinámica de los grupos organizados.

CAPÍTULO II. MARCO TEÓRICO: presenta el sustento teórico de la investigación, presenta información sobre los grupos organizados, trabajo en equipo. Da a conocer las metodologías ágiles con sus características y se profundiza en la metodología OpenUP, así mismo, los lenguajes de programación, bases de datos y la Norma Técnica Peruana ISO/IEC12207 con sus respectivos procesos.

CAPÍTULO III. MATERIALES Y MÉTODOS: se describe brevemente cada fase de la metodología que se utilizó en el desarrollo de la investigación.

CAPÍTULO IV. CONSTRUCCIÓN DE LA PROPUESTA: presenta la implementación de la metodología de investigación, se presentan entregables obtenidos en cada fase y la aplicación de la metodología OpenUP para la implementación del proyecto, validado e integrado con la NTP ISO/IEC 12207.

CAPÍTULO V. RESULTADOS Y DISCUSIÓN: presenta los resultados obtenidos tras implementación de la solución tecnológica validándolo con los objetivos del proyecto. Además, se dan recomendaciones para trabajos futuros de nuevas investigaciones.

CAPÍTULO II

MARCO TEÓRICO

2.1 Introducción

El presente capítulo fundamenta los conceptos teóricos de la investigación. Empieza narrando conceptos de grupos organizados en la Iglesia Adventista, el trabajo en equipo donde se detallan los objetivos fundamentales, luego la metodología OpenUP aplicada para la gestión del proyecto comparada con otras metodologías de desarrollo ágil. Se presenta la Norma Técnica Peruana 12207; asimismo, los conceptos fundamentales de la ingeniería de software y de base de datos, lenguajes de programación y el desarrollo del sistemas de información.

2.2 Grupo organizado y liderazgo

Un grupo organizado (GPO) relacionado a la Iglesia Adventista está conformado en promedio de 8 a 15 personas. Santana et al., (2010) dijo: “Los grupos pequeños deben ser el estilo de vida de la iglesia y funcionar como la base para la relación de la comunidad, el crecimiento espiritual y el cumplimiento integral de la misión de acuerdo con los dones espirituales”. Por lo tanto, los grupos son la base de la iglesia, el centro para el crecimiento y la multiplicación de discípulos. En los inicios de la era cristiana, la iglesia se reunía por medio de pequeños grupos, en las casas de los conversos al cristianismo. Compartían la nueva fe que habían aceptado, comían juntos, oraban, abrían las escrituras y se animaban frente a la persecución existente, esto está registrado en libro de Hechos de los Apóstoles y se registran en los siguientes textos bíblicos: “Y perseverando unánimes cada día en el templo, y partiendo el pan en las casas, comían juntos con alegría y sencillez de corazón, alabando a Dios, y teniendo favor con todo el pueblo. Y el Señor añadía cada día a la iglesia los que habían de ser salvos” (Hechos 2:46-47, Reina-Valera 1960). El texto destaca

que compartían los alimentos en sus casas. Además, dice la Escritura, “y todos los días, en el templo y por las casas, no cesaban de enseñar y predicar a Jesucristo” (Hechos 5:42, Reina-Valera 1960). Se enseñaba de Jesucristo tanto en el templo y en las casas. También los siguientes textos los respaldan: Hechos 12:12, Romanos 16:5, 1 Corintios 16:19. Asimismo, Jesús, como maestro por excelencia tenía un grupo organizado de 12 integrantes de diferentes caracteres e ideologías, los cuales dejaron sus diferentes ocupaciones para juntos lograr el mismo objetivo: cumplir la misión de la predicación del evangelio a costa de todo, creando nuevos grupos y multiplicándose.

Entre las actividades principales del grupo están las reuniones; cada reunión del grupo tiene una estructura, todos participan, pero la dirección está asignada a una o varias personas, con un tiempo determinado. En cada reunión las personas cuentan sus experiencias vividas cada semana por medio de un diálogo abierto y directo; piensan en forma reflexiva, se estimulan unos a otros en una mayor relación con Jesucristo, oran unos por otros, reflexionan abriendo las Sagradas Escrituras; existe una participación con sus talentos y dones que posee cada miembro. Entre una de las actividades de la reunión está la confraternización, que tiene como objetivo que cada asistente se sienta como parte de una familia unida.

La escritora Elena G. de White alienta a los jóvenes con estas palabras: “Jóvenes y señoritas, ¿no podéis formar grupos y, como soldados de Cristo, alistaros en la labor, poniendo todo vuestro tacto, y capacidad y talento al servicio del Maestro, para que podáis salvar almas de la ruina? Organícense grupos en todas las iglesias para hacer esta obra. ¿Se organizarán los jóvenes y señoritas que realmente aman a Jesús como obreros, no solo para trabajar en favor de los que profesan ser observadores del sábado, sino también de aquellos que no son de nuestra fe?” (White, 1983). Además, argumenta: “si hay muchos miembros en la iglesia, organícense en pequeños

grupos para trabajar no solo por los miembros de la iglesia, sino en favor de los incrédulos. Si en algún lugar hay solamente dos tres que conocen la verdad, organícense en un grupo de obreros. Mantengan íntegro su vínculo de unión, cerrando sus filas por el amor y la unidad, estimulándose unos a otros para progresar y adquiriendo cada uno valor, fortaleza y ayuda de los demás” (White, 1983). White promueve la formación de grupos con el objetivo de cumplir la misión de la iglesia. La unidad en los pequeños grupos juega un papel importante dentro del movimiento de los adventistas. En resumen la formación de pequeños grupos promueve la misión, la cooperación y el compañerismo mutuo, es por ello que necesita mucha habilidad.

Hasta el momento se ha descrito algunos fundamentos existentes para la formación de grupos, pero también se destaca el liderazgo que debe poseer cada líder del grupo, y depende en gran medida para alcanzar los objetivos principales.

2.2.1 Liderazgo

Se define al liderazgo como “el proceso de dirigir e influir en las actividades laborales de los miembros de un grupo y se reconoce como componente fundamental que influye en la efectividad laboral de una organización” (Rozo & Abaunza, 2010). Entonces, un líder, es una persona que dirige e influye, siendo una capacidad innata o adquirida. Es muy importante destacar la capacitación a los líderes, porque un líder capacitado es un líder preparado. Entre las características que deben poseer los líderes están de ser dinámicos y espirituales, y que generen confianza hacia sus liderados, además innovadores, trabajadores y soñadores. Un líder es aquel a quien se puede imitar por el ejemplo. Entonces, el éxito del grupo depende en buena medida de un buen liderazgo, dirigiendo así al grupo al cumplimiento de sus metas.

Entre los tipos de líderes según la literatura son: los transformacionales que según Sebastián, Vilar, & Azzollini (2013), citando a Bass & Avolio (1987), poseen tres factores determinantes: “carisma, estimulación intelectual y consideración individualizada de las necesidades de los seguidores”. También Sebastián, Vilar, & Azzollini (2013), citando a Kouses & Posner (1993), mencionaron las siguientes características que poseen este tipo de líderes: a) son innovadores, se atreven a experimentar, b) son intuitivos, imaginan el futuro y describen sus ideas de manera vívida, involucrando a otros en ellas haciendo alusión a sus valores, intereses, esperanzas y sueños, c) despiertan confianza, comparten el poder, promueven el trabajo en equipo y la colaboración, lo cual ayuda a las personas a sentirse capaces y apoyadas, d) son buenos ejemplos a seguir, en ellos no existe diferencia entre lo que predicán y su actuación, e) apoyan a sus seguidores, animándolos, reconociendo y celebrando sus logros.

Una de las características mencionadas en el párrafo anterior es “compartir el poder”, a esto se determina delegación de responsabilidades; hacer referencia a los cargos en los grupos, que está basado en funciones, donde cada cargo tiene sus funciones bien definidas. De esa forma el encargado conoce lo que le toca llevar acabo sin ocupar responsabilidades que no le competen. Un verdadero líder es un soñador e innovador, porque no hace solo lo que le piden, sino que sueña con mucho más. En conclusión, un líder es aquel que llega al corazón de la persona, es empático poniéndose en el lugar del integrante. Y no es fácil para un líder cumplir todo lo que la literatura dice, y muchas veces también necesita de otro líder con más experiencia para ser motivado; además, en la actualidad, los líderes son jóvenes, y como tales, en su mayoría tienen poca experiencia y también poseen muchas necesidades de orientación. Un líder es aquel que inspira al liderado, que además de dirigir, comparte sus conocimientos y prepara nuevos líderes innovadores, reflexivos y creativos.

2.2.2 Trabajo en equipo

Según Cardona & Wilkinson (2006) dicen que trabajar en equipo “consiste en colaborar organizadamente para obtener un objetivo común”. Cada miembro del equipo tiene una concreta personalidad y unas habilidades, conocimientos y experiencias específicas que aportan, que se diferencian de los demás del resto del equipo. Según estos investigadores argumentan que la escucha, la colaboración y el optimismo son las actitudes esenciales de un espíritu deportivo, “de equipo”.

Existen nueve (9) diferentes roles de equipo con sus fortalezas y debilidades, fue definido por el Dr. Meredith Belbin por la década del 70, esta categorización se convirtió en una metodología de recursos humanos permitiendo identificar, mediante una simple encuesta de autopercepción, las tendencias de comportamiento profesional de cada individuo dentro de un equipo de trabajo. A estas tendencias de comportamiento se les denomina roles de equipo. En la *Tabla 1* se muestra los roles de equipo.

Tabla 1

Roles de equipo

Rol	Fortalezas	Debilidades
Cerebro	Creativo, imaginativo, poco ortodoxo. Resuelve problemas difíciles.	Ignora los incidentes. Demasiado absorto en sus propios pensamientos como para comunicarse eficazmente.
Investigador de recursos	Extrovertido, entusiasta, comunicativo. Busca nuevas oportunidades. Desarrolla contactos.	Demasiado optimista. Pierde el interés una vez que el entusiasmo inicial ha desaparecido.
Coordinador	Maduro, seguro de sí mismo. Aclara las metas a alcanzar. Promueve el debate entre los miembros del equipo.	Se le puede percibir como manipulador. Se descarga de trabajo personal.

Impulsor	Retador, dinámico, trabaja bien bajo presión. Tiene iniciativa y coraje para superar obstáculos.	Propenso a provocar. Puede ofender los sentimientos de la gente.
Monitor evaluador	Serio, perspicaz y estratega. Percibe todas las opciones. Juzga con exactitud.	Puede carecer de iniciativa y de habilidad para inspirar a otros.
Cohesionador	Cooperador, apacible, perceptivo y diplomático. Escucha, construye e impide los enfrentamientos.	Indeciso en situaciones cruciales.
Implementador	Disciplinado, leal, conservador en sus hábitos. Transforma las ideas en acciones. Práctico.	Inflexible en cierta medida. Lento en responder a nuevas posibilidades.
Finalizador	Esmerado, concienzudo, ansioso. Busca los errores y las omisiones. Realiza las tareas en el plazo establecido.	Tiende a preocuparse excesivamente. Reacio a delegar.
Especialista	Sólo le interesa una cosa a un tiempo. Aporta cualidades y conocimientos específicos	Contribuye sólo cuando se trata de un tema que conoce bien. Se explaya en tecnicismos.

Fuente: Aritzeta & Ayestarán (2003) adaptado de Belbin

El trabajo en equipo siempre tiene que ver con personas, lo que se busca en los equipos es cumplir con los objetivos de la razón de ser, al mismo tiempo, donde todos los integrantes puedan crecer juntos, no debe existir ningún tipo de rivalidades o afines. Todos los miembros con sus características pueden aportar y aprender, si hay que cambiar algunos aspectos que no benefician al grupo y hacia el alcance de los objetivos se tendrán que mejorar. Se deben considerar a un equipo como una familia, porque en un mundo competitivo es mejor crecer juntos como equipo. Porque no se compiten con los integrantes del mismo equipo sino con otros equipos.

Para el éxito de un equipo (según la literatura existente) se necesitan factores como: tiempo, misión y visión, práctica, objetivos claros, toma de decisiones, comunicación, seguimiento, liderazgo, compromiso, roles definidos y entusiasmo. Un buen desempeño de estos factores traerá resultados exitosos para el equipo organizado.

2.3 Sistema de información (SI)

Fernández (2010) dijo: “un sistema es un conjunto de componentes que interactúan entre sí”. Es decir, un sistema de información está conformado por diferentes procesos para lograr objetivos específicos, donde cada parte del sistema debe cumplir sus tareas asignadas. En la *Figura 1*, se muestra el modelo de cinco componentes que vienen a ser: objetivos, mecanismos de control, entradas, transformación y salidas, siendo la estructura general de todo sistema de información.

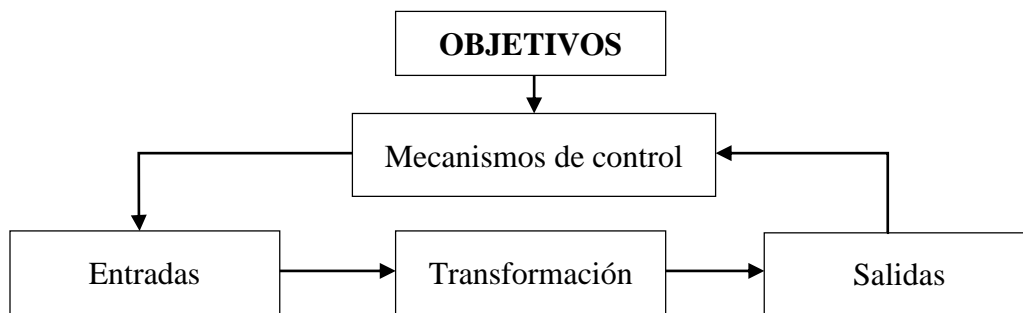


Figura 1. Modelo general de un sistema de información - Fernández (2010).

La información es uno de los activos más importantes que posee una organización, entre los objetivos principales de los sistemas de información es facilitar los procesos organizacionales, optimizando recursos con eficiencia y eficacia. Según Cotos & Taboada (2005) citando a Miguel y Piattini (1993) definen a un sistema de información como “un sistema capaz de recoger, almacenar y procesar datos para, después de un análisis obtener información útil y necesaria para la organización en la que está inmerso”. Entonces, los procesos principales de un sistema de información inicia desde la recogida de la información que es principalmente proveída por el usuario del sistema, de manera interna. Almacena la información en una base de datos y procesa dicha información para cada departamento de la organización.

Desde una perspectiva estratégica, Fernández (2010), citando a Andreu, Richar y Valor (1996), definieron a un sistema de información “como el conjunto formal de procesos, operando con un conjunto estructural de datos de acuerdo con la necesidad de una empresa, recopila, elabora y distribuye (parte de) la información necesaria para la operación de dicha empresa y para las actividades de control correspondientes, apoyando al menos en parte, la toma de decisiones necesaria para desempeñar las funciones y procesos de negocio de la empresa de acuerdo con su estrategia”. Entonces, un sistema de información forma parte vital en los procesos de la organización, porque sirven de soporte para un buen funcionamiento. Los avances tecnológicos en estos últimos años han posibilitado herramientas en diferentes plataformas para la construcción de sistemas de información aplicados a todo tipo de negocio o rubro a la que se dedican las organizaciones. Un sistema de información bien aplicado es capaz de optimizar procesos de forma increíble. El plus que proveen los sistemas de información es la ayuda en la toma de decisiones que está basada en la data histórica y el comportamiento de la organización.

2.3.1 Tipos de sistemas de información

Los sistemas de información están categorizados de acuerdo al eje de cada arista de las organizaciones. Existen los tipos de sistemas de información tales como: de procesamiento de transacciones, información ejecutiva, información gerencial, soporte a la toma de decisiones, sistema experto, planificación de recursos y automatización de oficinas.

2.3.1.1 Sistemas de información de procesamiento de transacciones

Los sistemas de procesamiento de transacciones (TSP, Transaction Processing Systems), “son creados para procesar grandes cantidades de datos relacionadas con transacciones rutinarias de

negocios, nóminas, inventarios” (Kendall, 2005). Estos sistemas son usados a diario en las organizaciones y su actividad es permanente, ante la falla de estos muchas organizaciones paralizan sus actividades; por eso, se necesitan que funcionen sin ningún tipo de interrupción. Entonces, este tipo de sistemas acompañan los procesos de negocio de la organización, produciendo bastantes registros de información, por lo cual se requiere que sean eficientes ante las peticiones de los usuarios.

El proyecto en mención está en el tipo de sistema transaccional porque procesa información de los datos de todos los grupos organizados y también como sistema de información ejecutiva por la emisión de reportes que ayuda a los ejecutivos de la iglesia e institución.

2.3.1.2 Sistemas de información ejecutiva

Los sistemas de información a ejecutivos (ESS, Executive Support Systems), permiten a los ejecutivos disponer de la información mediante herramientas gráficas y de comunicaciones. Por lo general están presentes en salas de reunión y oficinas corporativas personales. Estos sistemas dan la capacidad a los ejecutivos comprender sus entornos y amplían su visión para resolver problemas estratégicos. Además, mediante estas herramientas pueden gestionar las planificaciones.

2.3.1.3 Sistemas de información gerencial

Los sistemas de información gerencial (MIS, Management Information Systems) “dan apoyo a un espectro de tareas organizaciones mucho más amplio que los transaccionales, como el análisis y la toma de decisiones” (Kendall, 2005). Este tipo de sistemas se sirven de los sistemas transaccionales y otros recursos para brindar reportes de gestión a los que dirigen la empresa, logrando así pronósticos exactos del comportamiento futuro de la organización y tienen una base

de información fidedigna que ayudan a tomar decisiones para el bien de la empresa en sus diferentes aspectos.

2.3.1.4 *Sistemas de información de soporte a la toma decisiones*

Los sistemas de apoyo a la toma de decisiones (DSS, Decision Support Systems) se asemejan a los sistemas gerenciales ya que ambos usan la base de datos. Sin embargo, “pone énfasis en el apoyo a la toma de decisiones en todas sus fases” (Kendall, 2005). Se ajustan más al gusto de la persona o grupo que lo utiliza y en la actualidad está enfocada a los de negocios, con el objetivo de obtener ventajas económicas. Estos sistemas de información poseen técnicas de inteligencia de negocios usados actualmente, ya que en mundo en competitividad las decisiones organizaciones juegan un papel preponderante beneficiando a clientes, proveedores, personal y comunidad en conjunto.

2.3.1.5 *Sistemas expertos e inteligencia artificial*

Los sistemas de Inteligencia Artificial (AI, Artificial Intelligence), conocidos como sistemas expertos, son aquellos que tienen un comportamiento inteligente. Sus dos líneas de investigación son: la comprensión del lenguaje natural y el análisis de la capacidad para razonar un problema hasta su conclusión lógica. Ante un problema, el sistema experto presenta las soluciones más óptimas basadas en la información histórica generadas por los sistemas transaccionales. Es decir, son sistemas con un gran nivel de cálculo que hace uso de una rama muy importante de la ingeniería de sistemas. Si bien es cierto en la actualidad las empresas con un buen capital tiene el privilegio de contar con este tipo de sistemas, sería de gran ayuda a las empresas que recién están apareciendo en el mercado del mundo de los negocios, permitiendo el buen aprovechando de sus

recursos y fidelizando a sus clientes con una atención de calidad y por ende obteniendo más utilidades para nuevas inversiones.

2.4 Metodologías ágiles

2.4.1 El manifiesto ágil

Según Herrera & Valencía (2007) dijeron que “el manifiesto ágil es un documento que resume en cuatro valores y doce principios las mejores prácticas para el desarrollo de software, basados en la experiencia de 17 industrias del software, en procura del desarrollo más rápido y conservando su calidad”. Los cuatro (4) valores principales del manifiesto ágil son:

- **Los individuos e interacciones por encima de los procesos y las herramientas:** para la mejorar la productividad en los proyectos de software, las metodologías ponen mucho énfasis en el bienestar de los desarrolladores, valorando el recurso humano garantizando así el éxito del proyecto. El trabajo en equipo y de la mano con los clientes facilita la adaptación al entorno de trabajo y da mayor garantía que trabajar con procesos y herramientas rigurosas. En otras palabras, el objetivo es que el integrante se sienta muy bien y al mismo tiempo satisfecho al trabajar en el equipo. Las metodologías ágiles buscan tener un equipo bien comprometido en el proyecto y que disfruten del ambiente del trabajo. Las empresas grandes como Google tienen muchos ambientes recreativos para lograr estos objetivos del compromiso y la dedicación de sus trabajadores. Lo peor que le puede pasar a un equipo es sentirse presionado por quienes los dirigen y que no sean justas sus remuneraciones.

- **Software funcionando por encima de la documentación:** este es un aspecto muy importante en los resultados que prometen las metodologías ágiles. Los profesionales del software reconocen que es importante tener buena documentación, pero que requieren tiempo y costo la mantención de una documentación completa y actualizada. Es por ello que se busca tener solo documentación necesaria, clara, corta y fundamental, por medio de mecanismos más dinámicos y menos costosos como son la comunicación personal, el trabajo en equipo, la autodocumentación y los estándares.
- **La colaboración del cliente por encima de la negociación del contrato:** lo clásico de los contratos es que los usuarios finales contraten o solicitan software indicando que es lo que debe cumplir y esperan los resultados de acuerdo a las exigencias o expectativas en plazos establecidos. Sin embargo, en las metodologías ágiles los usuarios forman parte fundamental en el equipo. El cliente o usuario, forma parte directa y comprometida en el equipo de trabajo, llegando a ser un ingrediente en el éxito de un proyecto, porque tanto el cliente como el equipo buscan un beneficio común; por eso, es necesario no estar dividido. Es así que la participación del cliente debe ser constante, desde el inicio hasta el final del proyecto, teniendo una comunicación de calidad con el equipo. Porque después de todo, el cliente sabe lo que necesita o desea, es la persona indicada para corregir o hacer recomendaciones en cada fase del proyecto, con mucho énfasis en los requerimientos del software.
- **La respuesta al cambio por encima del seguimiento de un plan:** un proyecto de software se enfrenta a cambios en su proceso de ejecución por su naturaleza cambiante y una sociedad que se moderniza constantemente. Entonces, las metodologías ágiles tienen habilidades para responder a los cambios de forma eficiente, estos cambios van desde

configuraciones sencillas del software pasando por la aparición de nuevos productos en el mercado, comportamientos de los competidores, nuevas tendencias tecnológicas, etc. Eso quiere decir que la planificación en las metodologías ágiles no deben ser rigurosas o estrictas, sino que deben estar preparadas para la adopción a cambios que pueden surgir en el proceso de implementación.

El objetivo primordial de las metodologías ágiles es producir software de calidad en menor tiempo, que sea configurable y responda eficazmente a los cambios que se presenten. Steven Goldman dijo: “la agilidad es dinámica, específica en el contenido, acepta con entusiasmo el cambio y se orienta al crecimiento” (Pressman, 2010). Además, Pressman (2010) afirmó: “la agilidad es más que una respuesta efectiva al cambio. Fomenta una buena comunicación entre todas las partes involucradas. Se pone el énfasis en la entrega rápida de software funcional. Adopta al cliente como parte del equipo de desarrollo. Reconoce que los planes del proyecto tienen sus límites y que al mismo tiempo es flexible”. Según estos autores agilidad tiene mucho que ver con producir software funcional, que se adapte al cambio y es de valiosa importancia una buena comunicación en cada equipo de desarrollo.

2.4.1.1 Principios de agilidad

En la agilidad para el desarrollo de software se presentan principios que debe cumplir un equipo. De manera resumida tanto Herrera & Valencía (2007) y Pressman (2010) presentaron los siguientes principios de agilidad:

- La prioridad más alta es satisfacer al cliente a través de la entrega temprana y continúa de software: Se trata de hacer entregas de software funcional y útil en pocas semanas. Esto

permite eliminar con la incertidumbre, desconfianza, insatisfacción y desmotivación del cliente debido a largas esperas para ver resultados concretos. De esa manera la participación del cliente se hace más realidad porque el software está siendo probado, revisado y aprobado constantemente.

- Bienvenidos los cambios a los requerimientos, incluso los tardíos: los procesos ágiles aprovechan los cambios para la ventaja competitiva del cliente. Es decir, los cambios deben ser parte del proceso de maduración del software, siendo que el cliente describe una necesidad desde el punto de vista del usuario y que sus conocimientos técnicos lo pueden limitar para hacerte entender completamente. Esto induce al equipo de desarrollo a preferir los diseños flexibles, aumentando así la satisfacción del cliente y redundando finalmente en el beneficio del equipo.
- Liberar frecuentemente software funcionando, desde un par de semanas a un par de meses, con preferencia por los periodos más cortos: como el cliente espera ver el producto y eso debe ser entregado. Es muy poco conveniente después de varios meses entregar solo informes, modelos abstractos y planos. Se debe incluir las entregas software que se estén ya trabajando, esto es lo que motiva finalmente al cliente.
- Las personas del negocio y los desarrolladores deben trabajar juntos diariamente a lo largo del proyecto. Si bien es cierto los usuarios desconocen la parte técnica del proyecto tales como el lenguaje de programación, el diseño de base de datos, protocolos y otros aspectos; sin embargo, son ellos los que deben validar la funcionalidad y los resultados entregados por el software. La participación oportuna de los usuarios puede resultar determinante en el éxito de un proyecto reduciendo también el costo y el tiempo. Esto quiere decir que el usuario debe estar involucrado en todo el tiempo de la duración del proyecto.

- Construir proyectos en torno a individuos motivados. Darles el entorno y apoyo que necesiten, y confiar en ellos en que realizarán un buen trabajo. El ánimo, el sentido de pertenencia y la disposición del equipo de trabajo son fundamentales. La confianza que se les muestre forma parte de la motivación, respetando sus aportes y la comodidad al realizar su trabajo. Todo lo que permita mantener a un equipo animado y motivado debe hacerse constantemente.
- El método más efectivo y eficiente de compartir información, y dentro de un equipo de desarrollo, es la conversación cara a cara. Debe haber un buen sistema de comunicación tanto en los miembros del equipo y los usuarios. Se ha comprobado que la mejor forma de hacerlo es hablando personalmente. En lo posible, se evitan los intermediarios en el proceso de comunicación tales como: son el papel, el teléfono, el sistema de correo, y demás medios de comunicación.
- El software funcionando es la medida de progreso. Si bien hay diferentes maneras de medir el avance de un proyecto, es la cantidad de requerimientos implementados y funcionando la que más claridad y confiabilidad ofrecen para establecer una medida del avance del proyecto, y lo que da mayor confiabilidad son las funcionalidades probados y aprobadas por el usuario.
- Los procesos ágiles promueven el desarrollo sostenible. Los patrocinadores, desarrolladores y usuarios deberían ser capaces de mantener relaciones cordiales. Se debe trabajar de manera que lo urgente se imponga sobre lo más importante. Desde el inicio del proyecto se debe asignar responsabilidades y tareas de manera que siempre se puedan cumplir. Además, las buenas relaciones da seguridad a lo que se está desarrollando, y son como canales para el lograr un proyecto exitoso.

- La atención continua a la excelencia técnica y al buen diseño incrementan la agilidad. Además de satisfacer los requerimientos del usuario, los aspectos técnicos deben ser excelentes, independientemente de su cantidad y complejidad. La calidad debe ser vista desde dos perspectivas, la del usuario y la del equipo desarrollador. Para el personal técnico resulta evidente que cuanto más calidad tenga el software en cuanto a diseño y estándares de implementación, más rendimiento obtiene en las tareas de pruebas, mantenimiento, y mayor reusabilidad.
- La simplicidad, el arte de maximizar la cantidad de trabajo no hecho, es esencial. Se estima que el cliente nunca usará el 90% de la funcionalidad que se implementa sin que está haya sido solicitada. Se deben centrar los esfuerzos en lo que realmente importa, de manera simple, sin excederse en refinamientos y optimizaciones innecesarias. Si funciona así, déjelo así, si se va a perfeccionar u optimizar una rutina o programa se debe evaluar minuciosamente el costo beneficio. Esto permite la optimización de costo y tiempo.
- Las mejores arquitecturas, requerimientos y diseños emergen de los equipos auto-organizados. Los principios que rijan en equipo de trabajo deben surgir de su interior, los ajustes, estructuras administrativas deben formularse con la participación de todo el equipo teniendo siempre presente el bien colectivo, la responsabilidad es de todos. El compromiso y el cumplimiento a los mismos beneficiará al proyecto.
- En intervalos regulares, el equipo reflexiona sobre cómo volverse más efectivo, entonces afina y ajusta su comportamiento como corresponde. El equipo de trabajo está todo el tiempo dispuesto a cambiar lo que sea necesario para mejorar. En cada tarea siempre existe la posibilidad de hacerlo mejor la próxima vez. Esto mejora la calidad del producto.

2.4.1.2 *Comparación entre metodologías ágiles y tradicionales*

En la *Tabla 2* se detallan las principales diferencias de las metodologías ágiles con respecto a las tradicionales (“no ágiles”). Estas diferencias que afectan no sólo al proceso en sí, sino también al contexto del equipo, así como a su organización.

Tabla 2

Diferencias entre metodologías ágiles y tradicionales

Metodologías ágiles	Metodologías tradicionales.
Basadas en heurísticas provenientes de prácticas de producción de código.	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo.
Especialmente preparados para cambios durante el proyecto.	Cierta resistencia a los cambios.
Impuestas internamente (por el equipo).	Impuestas externamente.
Proceso menos controlado, con pocos principios.	Proceso mucho más controlado, con numerosas políticas/normas
No existe contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado.
El cliente es parte del equipo de desarrollo.	El cliente interactúa con el equipo de desarrollo mediante reuniones.
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio.	Grupos grandes y posiblemente distribuidos.
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menos énfasis en la arquitectura del software.	La arquitectura del software es esencial y se expresa mediante modelos

Fuente: Canós, Letelier, & Penadés (2010)

Mendoza, Durán, & Rivero (2010) dijeron que “se caracterizan por utilizar pocos artefactos, pocos roles y porque el cliente forma parte del equipo de desarrollo”. Fuentes (2015) mencionó que existen varias metodologías ágiles de desarrollo, entre las cuales se pueden mencionar: Programación Extrema (XP), Desarrollo adaptativo de software (DAS), Scrum, Método de desarrollo de sistemas dinámicos (MDSD), Crystal, Desarrollo impulsado por las características (DIC), Desarrollo esbelto de software (DES), Modelado Ágil (MA), Proceso Unificado Ágil (PUA), Proceso Unificado Abierto (OpenUP); y Herrera & Valencía (2007) añadió algunas más

como: SCRUM, Feature-Driven Development (FDD) y Lean Development (LD). A continuación se describen algunas de ellas, que incluyen sus valores y sus fases respectivas.

2.4.2 XP (Programación Extrema)

“La Programación Extrema es una forma social de desarrollo de software. Gran parte de su éxito se encuentra concentrado en las personas relacionadas con el producto que se pretende desarrollar” (Valderama, 2004). Uno de los principios fundamentales para su razón de ser, es una buena comunicación entre desarrolladores y clientes (usuarios finales). Valderama (2004), citando a Kent (2002) autor del libro “Extreme Programming Explained: Embrace the Change” menciona los siguientes valores fundamentales de la Programación Extrema:

1. **Comunicación:** promueve una buena comunicación entre sus miembros. El objetivo es potenciar la comunicación para lograr que las personas lleguen a respuestas creativas por ellos mismos. Sin una buena comunicación no hay Programación Extrema.
2. **Retroalimentación:** permite mantener al equipo informado de la situación actual en la que se encuentra, logrando así que clientes y desarrolladores avancen todo el tiempo en el camino del éxito.
3. **Simplicidad:** dos de los refranes de la Programación Extrema es “haga la cosa más simple que posiblemente funcione” y “usted no va a necesitarlo”. Con ello se busca hacer productos buenos y funcionales de una manera simple y no gastando esfuerzos con la adicción de funcionalidad innecesaria.

4. **Coraje:** permite la valentía para hacer frente a situaciones difíciles donde una decisión puede significar el éxito o el fracaso del proyecto.

2.4.2.1 Prácticas de la Programación Extrema

La Programación Extrema posee doce prácticas valiosas para la implementación de esta metodología.

- **Planeamiento del Juego (The Planning Game):** es donde clientes y desarrolladores elaboran un plan inicial para el desarrollo del proyecto y a la medida que este avanza, y se va refinando hasta su culminación. Los clientes y desarrolladores llegan a ser los jugadores y las fichas son pequeños escritos sobre tarjetas y una serie de movimientos asignando responsabilidades a los jugadores. Tiene el objetivo que el equipo de trabajo perfeccione su concepción acerca del sistema, y brindar un excelente control en el desarrollo.
- **Pruebas (Testing):** son divididas en dos grupos; el primero de ellos es el que comprende la escritura de pruebas unitarias “Test Units”; el segundo es la escritura de pruebas de aceptación “Acceptance Tests”. Las pruebas unitarias son implementadas por los desarrolladores a medida que se escribe el código. Y los clientes escribirán las pruebas de aceptación basados en los requisitos del sistema. El objetivo de las pruebas es validar los resultados obtenidos, asegurando una buena calidad del producto.
- **Programación en parejas (Pair Programming):** consiste en que dos desarrolladores producen código en un proyecto con un mismo ordenador, monitor y teclado. Es una de las prácticas más discutidas, pero según investigaciones han demostrado que esta práctica

genera un código de mejor calidad, ya que dos personas tienen más ideas de solución en una problemática específica.

- **Refactorización (Refactoring):** es el proceso de cambiar un sistema de software de modo que el comportamiento externo del código no sea alterado. Los miembros del equipo de desarrollo refactorizan el código del proyecto siempre y cuando sea necesario. El objetivo es lograr un código más simple y legible, para eliminar cualquier vestigio de duplicados, asimismo eliminar fuentes principales de errores y disminuirá posibles problemas a futuros desarrolladores guiándolos en la dirección correcta.
- **Diseño Simple (Simple Design):** establece que las tareas deben desarrollarse de la manera más simple. Esto establece que no debe ser agregado características funcionales que no formen parte de los requisitos en la planificación del juego. El objetivo es lograr un buen diseño para asegurar el éxito.
- **Propiedad colectiva del código (Collective Code Ownership):** establece que cualquier miembro del equipo de desarrollo tendrá la autoridad y la capacidad de realizar cambios en el código del proyecto para lograr su mejoramiento. Cada desarrollador logrará una completa capacitación en todo el producto.
- **Integración Continua (Continuous Integration):** estipula que deberán realizarse cada día varias integraciones del código del proyecto. Evita la mayor parte de los problemas que ocurren cuando un equipo integra el trabajo pasado largo tiempo, y comienzan los errores sin que nadie sepa dónde y por qué.
- **Cliente en el lugar (On-site Customer):** establece contar con el cliente en todo el proceso de planificación para equipar los requerimientos, asimismo aclarando dudas y tomar

decisiones críticas en las reglas de negocio del proyecto. La comunicación cara a cara del cliente con el desarrollador elimina cualquier tipo de malentendidos que genera cuellos de botella y atrasos en el proyecto, asimismo el cliente se siente importante para el logro de los objetivos.

- **Entregas pequeñas (Small Releases):** son los entregables de versiones estables y funcionales del código del proyecto después de culminar una iteración. Las entregas pequeñas, permite al cliente hacer un análisis más rápido y efectivo, evitando malentendidos con los requisitos del sistema.
- **40 Horas a la Semana (40 hours Week):** establece el tiempo promedio de trabajo del equipo de desarrollo del proyecto en 5 días hábiles. Se ha comprobado que una persona disminuye su rendimiento pasado las 8 horas diarias de trabajo. Entonces, una sobredosis en el tiempo de desarrollo no es la respuesta adecuada a los problemas en un proyecto. Se podría decir que el uso de los 8 remedios tales como: agua, descanso, ejercicio, luz solar, aire puro, buena nutrición, temperancia y esperanza en un Dios supremo, traerían muchos beneficios al equipo de desarrollo.
- **Estándares de código (Coding Standards):** logra un ambiente familiar en el código entre cada uno de los miembros del equipo de desarrollo. La aplicación de estándares permitirá la realización de refactorizaciones, construcción de pruebas unitarias y la comprensión por parte de otros programadores. El objetivo es que cada pieza de código sea clara y simple, sin importar la persona quien lo realice.
- **Metáfora (Metaphor):** es análoga o lo que la mayoría de las metodologías llaman arquitectura. Permite que el equipo de desarrollo tenga una visión común del funcionamiento del sistema través de una evocativa y simple descripción. Si no se

encuentra una variante para la construcción de la metáfora de un proyecto, el equipo de XP usará un sistema de nombres comunes para asegurarse de que cada miembro comprende el modo de funcionamiento general del sistema.

Programación Extrema se encuentra centrado en tres conceptos fundamentales: métodos, aprendizaje y dirección. El primero de ellos se refiere a seguir los principios y estrategias de la Programación Extrema. El segundo hace referencia a la necesidad de dejar a un lado los viejos conceptos y salir en busca de nuevos conocimientos que propicien el desarrollo; y el tercero resalta la necesidad de un líder en el equipo que vea el proceso como un todo y llame la atención de los miembros para impedir el desvío por caminos errados.

2.4.3 Enfoque de Reglas de Negocio

Para la construcción de sistemas basados en reglas de negocio tiene dos objetivos fundamentales: “reducir el tiempo de desarrollo y entregar un sistema preparado para enfrentar los cambios” (García & André, 2015). Para ello divide el dominio del problema en por lo menos tres aspectos separados, pero completamente relacionados tales como: datos, procesos y reglas.

“Una regla de negocio es una declaración que define, establece u obliga un cierto aspecto del negocio. Con ello se pretende imponer la estructura del negocio y de esa forma condicionar el comportamiento de este. Las reglas definen y controlan los procesos de negocio en una organización” (García & André, 2015). Para que un evento de negocio sea exitoso, es necesario cumplir las reglas establecidas. Las reglas de negocio se identifican durante el proceso de la captura y el análisis de requisitos.

2.4.3.1 Fases de la metodología con un Enfoque de Reglas de Negocio

- a) La primera fase de esta metodología se enfoca en conseguir un amplio dominio del negocio, conocer todos los detalles importantes, definir las reglas de negocio. A continuación se describe brevemente el objetivo de cada fase:
- **Definir el Alcance del Negocio:** el propósito de esta fase es la captura de los requisitos funcionales y no funcionales del sistema y establecer los objetivos del negocio.
 - **Descubrimiento:** en esta fase se permite a las reglas de negocio descubrir las decisiones importantes que están detrás de los eventos de negocio, la toma de decisiones es un punto de partida para ir descubriendo las principales reglas de negocio asociadas a cada decisión.
 - **Análisis:** en esta fase se hace un análisis de los datos, reglas y procesos. Se profundiza en el análisis de las reglas, para lograr este objetivo se propone una serie de pasos importantes para hacer un buen análisis de las reglas.
 - **Diseño:** esta fase permite diseñar el sistema basado en las reglas de negocio, integrando tres aspectos fundamentales: datos, reglas y procesos. Se introduce además, la tecnología o herramienta que permite aplicar el enfoque.
 - **Implementación y Prueba:** estas fases de desarrollo están relacionadas con la herramienta que se utiliza para manejar las reglas de negocio. Los desarrolladores solo necesitan documentarse sobre cómo trabajar con la tecnología y comenzar a implementar las reglas. A su vez, se puede ir implementando y probando las reglas definidas, ya que muchas de las tecnologías brindan la posibilidad de validar las reglas antes de ser desplegadas en la aplicación.

Las ventajas que permiten las reglas de negocio son las siguientes: programación declarativa, separación de los datos y la lógica, conocimiento centralizado, simplicidad, facilita el desarrollo de la aplicación, agilidad.

2.4.4 SCRUM

SCRUM es considerado como un marco de referencia para desarrollo ágil de productos software. Si bien es cierto es principalmente aplicable al desarrollo de software, hay un sin número de proyectos que no están relacionados con el software, han sido administrados con Scrum. Gimson (2012) mencionó que “ya en 2011 lo usaba el 75% de equipos ágiles alrededor del mundo”, y que está siendo “utilizado por grandes y pequeñas compañías, incluyendo Yahoo, Microsoft, Google, Lockheed Martin, Motorola, SAP, Cisco, GE Medical, Capital One y la Reserva Federal de los Estados Unidos”.

Según Palacio & Ruata (2011), Scrum “establece una reunión al inicio de cada sprint para determinar el trabajo que se va a realizar, otra al final para evaluar el resultado, y revisiones diarias que realiza el equipo en su auto-gestión”, como se muestra en la *Figura 2*.

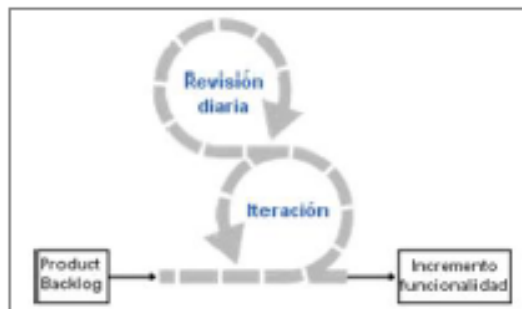


Figura 2. Diagrama del ciclo de Scrum (Palacio & Ruata, 2011).

Las reuniones diarias permiten al equipo una mayor comunicación, donde se despejan muchas dudas, y se consigue un equipo totalmente informado.

2.4.4.1 Principales elementos de SCRUM

Como toda metodología, según Gimson (2012), SCRUM presenta los siguientes elementos que se describen a continuación:

- **Product Backlog:** una lista priorizada de funcionalidades técnicas y de negocio. Estas funcionalidades son requisitos a muy alto nivel de lo que debe hacer la aplicación, donde se listan características, funciones, tecnología, mejoras, bugs, etc. que serán aplicadas al producto. El Product Backlog es el punto de inicio.
- **Sprint Backlog:** Lista de tareas de un sprint. El Sprint Backlog identifica y define el trabajo a ser alcanzado por el equipo de desarrollo durante un Sprint. A un nivel el Sprint Backlog identifica las características mientras que a otro nivel, identifica las tareas requeridas para implementar esas características.
- **Incremento:** parte de un sistema desarrollado en un Sprint. Este producto desarrollado es potencialmente entregable al final de cada Sprint, implica que todo está completamente terminado en cada Sprint; y se podría realmente empaquetar o desplegar inmediatamente después de la Revisión del Sprint con mínimas tareas, si bien a veces se necesitan ciertos trabajos de acabado tales como pruebas o documentación.

2.4.4.2 Roles

Scrum define varios roles, que están divididos en dos grupos. Por un lado, figuran los que están comprometidos con el proyecto y el proceso Scrum. Por otro lado, aquellos que en realidad no

forman parte del proceso Scrum, pero que deben tenerse en cuenta y cuya participación es muy importante.

2.4.4.2.1. Roles comprometidos con el proyecto

- **Product Owner:** representa la voz del cliente. Escribe historias de usuario, las prioriza, y las coloca en el Product Backlog.
- **Scrum Master:** Protege al equipo de distracciones y de otros elementos externos y lo mantiene enfocado. Elimina obstáculos que alejen al grupo de la consecución de objetivos del sprint. No es el líder del grupo, ya que el grupo se auto gestiona. Dirige los scrums diarios. Realiza el seguimiento del avance.
- **Equipo:** conformado por no más de 8 personas (si hay más se organizan varios equipos que trabajan sobre el mismo product backlog). Tiene la responsabilidad de entregar el producto. Son autónomos y auto-organizados. Deben entregar un conjunto de ítems del Backlog al final del Sprint.

2.4.4.2.2. Roles involucrados en el proceso

- **Usuarios:** son los destinatarios o beneficiarios finales del producto.
- **Stakeholders o interesados (Clientes, Proveedores):** Se refiere a la gente que hace posible el proyecto y para quienes el proyecto producirá el beneficio acordado que lo justifica.
- **Managers:** son aquellos que establecen el ambiente para el desarrollo del producto.

2.4.4.3 El proceso SCRUM

En marco de administración del proyecto donde las actividades de desarrollo (recolección de requerimientos, diseño, programación) tienen lugar. El período de desarrollo es de una iteración de 30 días llamada Sprint, si bien podría trabajarse con sprints de menor duración. El marco de trabajo de Scrum tiene tres componentes o cuatro según como se los mire, los cuales son:

- **Pre-sprint:** planificación del sprint.
- **Sprint:** ciclo de trabajo.
- **Post-sprint:** revisión y retrospectiva del sprint (pueden considerarse como dos componentes diferentes posteriores al sprint).

2.4.4.3.1. La pila Sprint

La pila del producto es la lista de funcionalidades que desea el cliente, ordenadas según la prioridad para él. La pila del sprint es la lista de tareas en las que se han descompuesto las funcionalidades de la pila del producto que se van a desarrollar en un sprint. Para cada tarea de la pila del sprint se indica la persona que la tiene asignada y el tiempo de trabajo previsto. Durante el desarrollo del sprint el equipo actualiza a diario en la pila del sprint los tiempos pendientes de cada tarea. Y el incremento es la parte del producto desarrollada en un sprint, y se debe encontrar completamente terminada y probada.

2.4.4.3.2. Las reuniones

Según Palacio & Ruata (2011), Scrum realiza tres reuniones con los objetivos de: planificación del sprint, seguimiento del sprint y revisión del sprint.

En la planificación del sprint, la duración máxima es 1 día, donde se determinan funcionalidades que se desarrollarán en el sprint. Cada funcionalidad es desglosada en tareas. Cada tarea se estima y se asigna a una persona del equipo. El resultado es la pila del sprint.

En el seguimiento del sprint, la reunión es diaria y breve en la que el equipo revisa la evolución del sprint. Cada uno expone la tarea en la que ha estado trabajando, en cuál va a trabajar y si necesita algo para poderla realizar. Cada miembro actualiza la estimación de tiempo pendiente de sus tareas.

En la revisión del sprint, la duración máxima es de cuatro (4) horas. Se muestra el incremento desarrollado a todas las personas implicadas en el proyecto. Esta no debe confundirse con una reunión retrospectiva para la mejora de las prácticas ágiles, el foco es el producto, no las prácticas ágiles de la organización.

2.4.5 Open Unified Process (OpenUP)

Según Eclipse (2016) Open Unified Process “OpenUP es un proceso unificado que aplica enfoques iterativos e incrementales dentro de un ciclo de vida estructurado”, como se ve en la *Figura 3*. OpenUP abraza una filosofía pragmática, ágil, que se centra en la naturaleza colaborativa de desarrollo de software. Se trata de un proceso de baja ceremonia herramientas-agnósticos que se puede ampliar para hacer frente a una amplia variedad de tipos de proyectos.

En la *Figura 3* se pueden ver los procesos de la metodología OpenUP, primero es la base, que es el proyecto completo dividido en sus respectivas fases: concepción, elaboración, construcción y transición. El proyecto es dividido en varias iteraciones, para cada iteración hay un plan

semanalmente, dicho plan o iteración es dividido en ítems o tareas diarias que llegarán a ser micro-incrementos en producto. En cada iteración resulta un demo o producto funcional. El esfuerzo personal en un proyecto OpenUP se organiza en micro-incrementos. Estos representan unidades cortas de trabajo que producen un ritmo constante, medibles del avance del proyecto (normalmente medido en horas o unos pocos días). “El proceso se aplica una intensa colaboración que el sistema está desarrollado de forma incremental por un equipo comprometido, auto-organizado. Estos micro-incrementos proporcionan un bucle de retroalimentación extremadamente corto que impulsa las decisiones de adaptación dentro de cada iteración” (Foundation Eclipse, 2016).

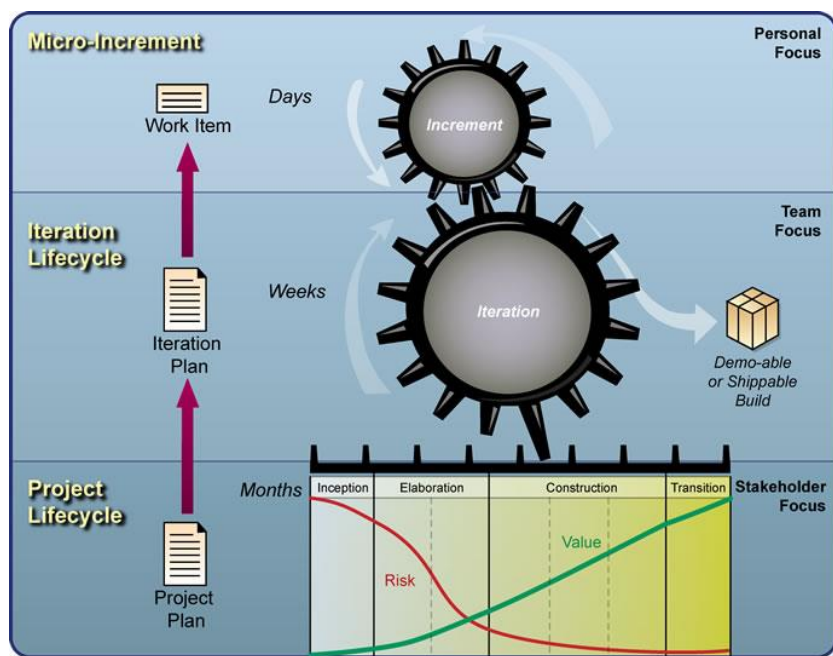


Figura 3. Capas de OpenUP: micro-incrementos, ciclo de vida y la iteración del ciclo de vida del proyecto. Fuente, Eclipse (2016)

OpenUP divide el proyecto en iteraciones: intervalos planificados, encajadas en tiempo típicamente se miden en semanas. Las iteraciones se centran que el equipo entregue valor incremental a las partes interesadas de una manera predecible. El plan de iteración define lo que

debe ser entregado dentro de la iteración, y el resultado es una acumulación de demostración capaz o entregable. En OpenUP, el equipo se auto-organiza en torno a cómo lograr los objetivos de iteración y comprometerse con la entrega de los resultados. Lo hacen mediante la definición y "tirando" las tareas de grano fino a partir de una lista de elementos de trabajo.

OpenUP estructura el ciclo de vida del proyecto en cuatro fases: inicio, elaboración, construcción y transición. El ciclo de vida del proyecto proporciona las partes interesadas y los miembros del equipo con la visibilidad y la toma puntos a lo largo del proyecto (Foundation Eclipse, 2016).

2.4.5.1 Roles de OpenUP

Según Eclipse (2016) define a OpenUP en seis roles principales:

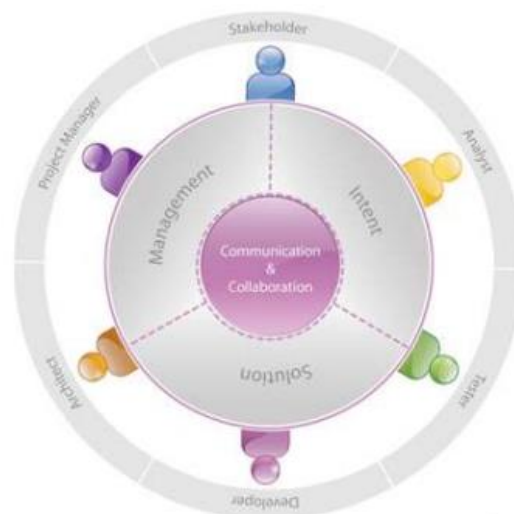


Figura 4. Principales roles en OpenUP y sus interacciones (Eclipse, 2016)

- **Analista:** este rol representa clientes y usuarios finales recopilando aportaciones y preocupaciones de los interesados para entender el problema a resolver y por la captura y fijación de prioridades para los requisitos.
- **Arquitecto:** es el responsable de definir la arquitectura de software, que incluye la toma de decisiones técnicas, clave que limitan el diseño general y la implementación del sistema. Esto incluye la identificación y documentación de los aspectos de gran importancia arquitectónica del sistema como vistas que describen los requisitos, el diseño, la implementación y el despliegue. De tal forma que asegura la escalabilidad del producto.
- **Desarrollador:** es responsable del desarrollo de una parte del sistema, incluye el diseño que se ajusta a la arquitectura, posiblemente prototipos de la interfaz de usuario, y luego implementa pruebas unitarias, y la integración de los componentes que forman parte de la solución.
- **Jefe de proyecto:** es el director de proyecto, quien lleva a cabo la planificación del proyecto, coordina las interacciones con las partes interesadas, y asegura que el equipo cumpla con los objetivos del proyecto.
- **Stakeholders:** este rol representa a los grupos de interés, cuyas necesidades deben ser satisfechas por el proyecto. Es un papel que puede ser jugado por cualquier persona que es (o potencialmente será) materialmente afectadas por el resultado del proyecto. Son conocidos como los clientes finales, y juegan un papel muy determinante ya que ellos conocen muy bien los procesos de negocio.
- **Tester o probador:** es responsable de las actividades básicas de la prueba de esfuerzo. Esas actividades incluyen la identificación, definición, implementación y realización de las

pruebas necesarias, así como el registro de los resultados de las pruebas y análisis de los resultados.

2.4.5.2 Disciplinas

OpenUP propone seis disciplinas para agrupar las tareas involucradas en el todo el proceso de la implementación de un proyecto. Gimson (2012) mencionó las siguientes disciplinas: requerimientos, arquitectura, desarrollo, prueba, administración de configuración y cambio y administración de proyecto.

- **Requerimientos (requirements):** explica cómo solicitar, analizar, validar y manejar los requerimientos para el sistema a desarrollar. Es importante entender la definición y alcance del problema que se intenta resolver. Se identifican a los stakeholders y se define el problema a resolver. Durante el ciclo de vida del proyecto se administran los cambios de los requerimientos.
- **Arquitectura (architecture):** el propósito es lograr elaborar una arquitectura robusta para el sistema. Explica cómo crear una arquitectura a partir de requerimientos arquitectónicamente (estructuralmente) importantes. Es en la disciplina de desarrollo donde se construye la arquitectura.
- **Desarrollo:** se explica cómo diseñar e implementar una solución técnica que esté conforme a la arquitectura y satisfaga los requerimientos de los clientes finales.
- **Prueba (Test):** explica cómo proveer retroalimentación sobre la madurez del sistema diseñando, implementando, ejecutando y evaluando pruebas. Es iterativa e incremental. Aplica la estrategia de “prueba lo antes posible y con la mayor frecuencia posible” para replegar los riesgos tan pronto como sea posible dentro del ciclo de vida del proyecto.

- **Administración de configuración y cambio (Configuration and change management):** explica cómo controlar los cambios de los artefactos, asegurando una evolución sincronizada del conjunto de productos (Work Products) que compone un sistema software. Esta disciplina se expande durante todo el ciclo de vida.
- **Administración del Proyecto (Project Management):** explica cómo entrenar, ayudar y apoyar al equipo, ayudándolo a manejar los riesgos y obstáculos que se encuentren al construir el software.

2.4.5.3 Tareas

Una tarea es una unidad de trabajo que se puede solicitar que lo realice un rol de trabajo. En OpenUP hay 19 tareas que los roles pueden realizar como actores principales (teniendo la responsabilidad de ejecutar esas tareas) o adicionales (ayudando y proveyendo información que se usa al ejecutar la tarea). La naturaleza colaborativa de OpenUP se ve manifiesta al tener los actores principales de las tareas interactuando con otros individuos al realizar las tareas. La *Tabla 3* presenta las listas de tareas a realizar por disciplina.

Tabla 3
Tareas de OpenUP

Disciplina	Tarea
Arquitectura	<ul style="list-style-type: none"> • Refinar la arquitectura • Visualizar (Proveer) la arquitectura
Desarrollo	<ul style="list-style-type: none"> • Implementar las pruebas de desarrollo • Implementar la solución • Correr las pruebas de desarrollo • Integrar y crear el desarrollo • Diseñar la solución
Administración de Proyecto	<ul style="list-style-type: none"> • Evaluar resultados

	<ul style="list-style-type: none"> • Administrar la iteración • Planear la iteración • Planear el proyecto • Solicitar cambios
Requerimientos	<ul style="list-style-type: none"> • Identificar y resaltar requerimientos • Detallar escenarios de caso de uso • Detallar requerimientos generales del sistema • Desarrollar una visión técnica
Prueba	<ul style="list-style-type: none"> • Crear Casos de Prueba • Implementar pruebas • Correr pruebas

Fuente: Gimson (2012)

2.4.5.4 Artefactos

Según Gimson (2012), “un artefacto (work product) es algo producido, modificado o utilizado por una tarea”. Los roles son responsables de crear y actualizar los artefactos. Los artefactos están sujetos a control de versión a través del ciclo de vida del proyecto. Los 17 artefactos de OpenUP se consideran esenciales y son los que debe utilizar un proyecto para capturar información relacionada con el producto y el proyecto, en detalle se presentan en la *Tabla 4*. Los proyectos pueden utilizar los artefactos de OpenUP o reemplazarlos con los propios elaborados.

Tabla 4

Artefactos de OpenUP

Disciplina	Artefacto	Tipo de Producto	Descripción General
Arquitectura	Architecture Notebook	Especificación	Describe el contexto para el desarrollo del software. Contiene las decisiones, razones, supuestos, explicaciones e implicancias de armar la arquitectura.
Desarrollo	Diseño	Solución	Describe la realización de la funcionalidad del sistema requerida en términos de componentes y sirve como una abstracción del código fuente.

Desarrollo	Construcción	Solución	Versión operacional de un sistema o parte del sistema que muestra un subconjunto de las funcionalidades que se proveerán en el producto final.
Desarrollo	Prueba	Solución	Las instrucciones que validan que los componentes individuales del software se comportan como se especificó.
Desarrollo	Implementación	Solución	Archivos de código software, archivos de datos y archivos secundarios como archivos de ayuda en línea que representan las partes gruesas de un sistema.
Administración de Proyecto	Plan de iteración	Plan	Un plan con gran nivel de detalle describiendo los objetivos, asignaciones de trabajo y criterios de valuación para la iteración
Administración de Proyecto	Plan de Proyecto	Plan	Reúne toda la información requerida para administrar el proyecto. Sus partes principales están compuestas por: un plan poco detallado que contiene las fases del proyecto y los hitos.
Administración de Proyecto	Lista de ítems de trabajo	Datos del proyecto	Contiene una lista de todo el trabajo calendarizado a realizarse dentro del proyecto, así como el trabajo propuesto que puede afectar al producto en este o futuros proyectos. Cada ítem de trabajo puede contener referencias a información relevante para llevar a cabo el trabajo descrito dentro del ítem de trabajo.
Administración de Proyecto	Lista de riesgo	Datos del proyecto	Lista abierta de riesgos conocidos, ordenados según importancia y asociados con acciones específicas de mitigación o contingencia.
Requerimientos	Especificación de requerimientos secundario	Especificación	Captura los requerimientos globales del sistema no capturados en escenarios o casos de uso, incluyendo requerimientos de calidad y requerimientos de funcionalidad global.
Requerimientos	Visión	Concepto	Contiene la definición de la vista que tienen los stakeholders sobre el producto a desarrollar, especificado en términos de necesidades y características claves de los stakeholders. Contiene una idea general del corazón de requerimientos imaginados para el sistema.
Requerimientos	Caso de Uso	Especificación	Captura la secuencia de acciones que un sistema puede realizar que produce un

			resultado observable de valor para los que interactúan con el sistema.
Requerimientos	Glosario	Elemento de modelo, Especificación	Define los términos importantes usados en el proyecto. Estos términos son la base para la colaboración efectiva con los stakeholders y otros miembros del equipo.
Requerimientos	Modelo de Caso de Uso	Modelo	Captura un modelo de las funcionalidades pretendidas en el sistema y su entorno, y sirve como un contrato entre el cliente y los desarrolladores.
Pruebas	Caso de Prueba	Especificación	Es la especificación de un conjunto de entradas de prueba, condiciones de ejecución y resultados esperados, identificados con el propósito de realizar la evaluación de algún aspecto particular de un escenario.
Pruebas	Log de Prueba	Dato de Proyecto	Recolecta salidas sin refinar capturadas durante una única ejecución de una o más pruebas para una sola corrida del ciclo de prueba.
Pruebas	Script de Prueba	Solución	Contiene las instrucciones paso a paso que realiza una prueba, permitiendo su ejecución. Estas pueden tomar forma ya sea de instrucciones textuales documentadas que se ejecutan manualmente o bien instrucciones que una computadora puede entender y permite la ejecución automática de la prueba.

Fuente: Gimson (2012)

2.4.5.5 Fases del OpenUP

2.4.5.5.1. Concepción

El objetivo de esta fase de comprender el alcance, y los objetivos del proyecto, se evalúa si es o no es viable. Hay cuatro objetivos para evaluar el alcance, los objetivos y la viabilidad siendo los siguientes:

- Entender qué construir. Determinar la visión, el alcance del sistema y sus límites.
Identificar quién está interesado en este sistema y por qué.

- Identificar la funcionalidad clave del sistema. Decidir qué requerimientos son los más críticos.
- Determinar al menos una posible solución. Identificar al menos una arquitectura candidata y su viabilidad respectiva.
- Entender el costo, el cronograma y los riesgos asociados al proyecto.

2.4.5.5.2. *Elaboración*

El objetivo de esta fase es establecer la línea base de la arquitectura del sistema y proporcionar una base estable para el gran esfuerzo de desarrollo de la siguiente fase. Hay objetivos para la fase de Elaboración que le ayudan a direccionar los riesgos asociados con los requisitos, la arquitectura, los costos y el cronograma que son los siguientes:

- Obtener un entendimiento más detallado de los requisitos. Tener un buen entendimiento de la mayoría de requisitos que le permitan crear un plan más detallado y obtener ganancia de los Stakeholders. Asegurarse de ganar profundidad en el entendimiento de los requisitos más críticos a ser validados por la arquitectura.
- Diseñar, implementar, validar y establecer la línea base para la arquitectura. Diseñar, implementar y probar un esqueleto estructural del sistema. Aunque, la funcionalidad no sea completa aún, muchas de las interfaces entre los bloques de construcción son implementadas y probadas. Esto se refiere a una arquitectura ejecutable.

- Mitigar los riesgos esenciales y producir un cronograma exacto y unos costos estimados. Muchos riesgos técnicos son dirigidos como un resultado de detallar los requisitos y de diseñar, implementar y probar la arquitectura. Redefinir y detallar el plan de proyecto de alto nivel.

2.4.5.5.3. *Construcción*

Esta fase se enfoca en diseño, implementación y prueba de las funcionalidades para desarrollar un sistema completo. El objetivo es desarrollar la aplicación basada en la arquitectura establecida. Los objetivos para la fase de Construcción que nos ayudan a tener un desarrollo con costo eficiente de un producto completo, una versión operativa del sistema que pueda ser entregada a la comunidad de usuarios son:

- Desarrollar iterativamente un producto completo que esté listo para hacer transición a la comunidad de usuarios. Describir los requisitos restantes, completar en detalles los diseños, completar la implementación y prueba del software. Liberar la primera versión operativa del software (versión beta) del sistema y determinar si los usuarios están listos para que la aplicación sea desplegada.
- Minimizar el costo de desarrollo y alcance algún grado de paralelismo. Optimizar los recursos y promover el paralelismo de desarrollo entre desarrolladores o equipos de desarrolladores.

2.4.5.5.4. *Transición*

El objetivo de esta fase es asegurarse que el software esté listo para entregar a los usuarios finales. Los objetivos para la fase de Transición que ayudan a afinar elegantemente la funcionalidad, el desempeño y la calidad total de la versión beta del producto desde el final de la fase previa son:

- La prueba beta valida que las expectativas del usuario sean satisfechas. Esto típicamente requiere algunas actividades de afinamiento, tales como depuración de errores y mejora del desempeño y la usabilidad.
- Lograr que los Stakeholders concuerden en que la implementación ha terminado. Esto puede implicar varios niveles de pruebas de aceptación del producto, incluyendo pruebas formales, informales y pruebas beta.
- Mejorar el desempeño en futuros proyectos a través de lecciones aprendidas. Documentar las lecciones aprendidas y mejorar el ambiente de los procesos y las herramientas para el proyecto.

2.4.5.6 *Subprocesos del ciclo de vida del proceso de desarrollo OpenUP*

Sommerville (2005) dijo: “un subproceso es un conjunto de actividades referenciadas con procedimientos específicos, que son desarrollados por personas con roles determinados”. En la *Tabla 5* se detallan los subprocesos de la metodología OpenUP y los objetivos que se deben cumplir en cada uno de ellos.

Tabla 5

Subprocesos del ciclo de vida del proceso de desarrollo OpenUP

Subprocesos	Objetivos
Concebir un nuevo proyecto	Concebir y definir un proyecto de software, con el fin de evaluar su viabilidad técnica, tecnológica, organizacional, ambiental y financiera.
Requerimientos y requisitos	Recolectar, analizar, aprobar y seguir la evolución de los requerimientos funcionales del cliente o interesado y los requisitos del software a través de la vida del producto y/o servicio.
Gestión del proyecto	Planear, ejecutar, controlar y socializar las actividades y resultados de un proyecto de software.
Gestión del riesgo	Identificación, valoración, relevancia, prevención, mitigación, control y respuesta a posibles riesgos que se generen en un proyecto de software.
Arquitectura	Transformar los requerimientos y requisitos significativos en una arquitectura que describa su estructura e identifique los componentes del software.
Diseño	Proporcionar un diseño que implemente el software y pueda ser verificado contra los requerimientos y los requisitos definidos.
Desarrollo	Implementar una solución técnica que cumpla con la Arquitectura definida y soporte los requerimientos de los grupos interesados.
Gestión de Pruebas	Diseñar, implementar, ejecutar y evaluar pruebas en cada uno de los componentes desarrollados.
Gestión de Cambios	Registrar, revisar y llevar a cabo solicitudes de cambios generadas en un proceso de desarrollo de software.
Implantación	Planificar y llevar a cabo la producción de una solución de software mediante el alineamiento de las necesidades de capacitación de los usuarios y el desarrollo de pruebas de funcionamiento.
Gestión Documental	Planificar, elaborar, aprobar, organizar y controlar la documentación requerida en un proceso de desarrollo de software.
Gestión de Auditorías	Planificar y ejecutar auditorías internas a proyectos de software con el fin de encaminarlos a una mejora continua.
Gestión de Plataforma Tecnológica	
Proceso de Desarrollo	Explorar, adaptar, acoger, evaluar y mejorar de manera incremental e iterativa los lineamientos y directrices propios del proceso de desarrollo unificado Open UP.

Fuente: Sommerville (2005)

2.4.5.7 Entregables

Según Ríos, Hinojosa, & Delago (2013) los entregables más importantes del OpenUP se presenta en la *Tabla 6*.

Tabla 6

Entregables del proyecto en OpenUP

Fase	Actividad	Documento
Inicio	Iniciación del proyecto Planeación del proyecto Identificación de requerimientos	Visión del sistema Plan de desarrollo de software Especificación de requisitos de software
Elaboración	Desarrollo de la arquitectura	Documento de arquitectura del sistema
Construcción	Definición de pruebas de la solución Construcción de la solución: - Traducción de pruebas a código. - Desarrollo del sistema	Plan de pruebas de software - Creación de archivos con los escenarios de prueba. - Codificación de la solución basándose en el comportamiento definido en las pruebas.
Transición	Despliegue de la solución	Manual de instalación Manual de usuario

Fuente: Ríos et al. (2013)

OpenUP es apropiado para proyectos a corto plazo y de bajo recursos, esto permite disminuir las probabilidades de fracaso e incrementa las probabilidades de éxito. Permite detectar errores tempranos en cada ciclo iterativo. Existe muy poca documentación, lo que permite dar el mayor tiempo al avance del proyecto, su enfoque está centrado hacia el cliente y las iteraciones son cortas.

2.4.6 Comparación de metodologías ágiles

Si bien es cierto hay muchas metodologías, las cuales difieren en algunos aspectos unas de otras, siendo el objetivo principal de todas producir software de calidad, en menor tiempo, sin embargo, es necesario hacer una comparación sobre la forma de trabajo de cada una de ellas. Se

consideraron las metodologías: Programación Extrema (XP), SCRUM, Enfoque de Reglas de Negocio y OpenUP.

	OpenUP	XP	SCRUM	Enfoque de Reglas de Negocio
Procesos (Fases)	<ul style="list-style-type: none"> • Concepción • Elaboración • Construcción • Transición 	<ul style="list-style-type: none"> • Exploración • Planificación • Iteraciones • Producción • Mantenimiento • Cierre del proyecto 	<ul style="list-style-type: none"> • Pre-sprint • Sprint • Post-sprint 	<ul style="list-style-type: none"> • Definir el Alcance del Negocio • Descubrimiento • Análisis • Diseño • Implementación y prueba
Roles	<ul style="list-style-type: none"> • Analista • Arquitecto • Desarrollador • Jefe de proyecto • Stakeholders • Tester o probador 	<ul style="list-style-type: none"> • Cliente • Programador • Probador • Rastreador • Coach o tutor • Consultor • Gestor o Manager 	<ul style="list-style-type: none"> • Product Owner • Scrum Master • Equipo • Usuarios • Stakeholders o interesados • Managers 	
Prácticas / disciplinas	<ul style="list-style-type: none"> • Requerimientos • Arquitectura • Desarrollo • Prueba • Administración de configuración y cambio • Administración del Proyecto 	<ul style="list-style-type: none"> • Planeamiento del Juego • Historias de usuario • Pruebas • Programación en parejas • Refactorización • Diseño simple • Propiedad colectiva del código • Integración continua • Cliente en el lugar • Entregas pequeñas • 40 horas a la semana 	<ul style="list-style-type: none"> • Pila de Producto • Planificación de Sprint • Reuniones • Revisiones • Retrospectiva 	<ul style="list-style-type: none"> • Programación declarativa • Separación de los datos y la lógica. • Conocimiento centralizado • Simplicidad • Facilita el desarrollo de la aplicación • Agilidad

Valores / principios	<ul style="list-style-type: none"> • Colaborar para alinear los intereses y compartir el conocimiento • Centrarse inicialmente en la arquitectura para minimizar los riesgos y organizar el desarrollo. • Balance de conflicto de prioridades para maximizar los valores de los interesados • Involucrar a las partes interesadas para la continua retroalimentación desarrollo 	<ul style="list-style-type: none"> • Estándares de código • Metáfora • Comunicación • Retroalimentación • Simplicidad • Coraje 	<ul style="list-style-type: none"> • Flexibilidad a cambios • Reducción del Time to Market • Mayor calidad del software • Mayor productividad • Maximiza el retorno de la inversión (ROI) • Predicciones de tiempos • Reducción de riesgos
----------------------	---	--	---

Fuente: elaboración propia

2.5 Lenguajes de programación

Un lenguaje de programación es un sistema estructurado y diseñado principalmente para que las máquinas y computadoras se entiendan entre sí y con nosotros. Contiene un conjunto de acciones consecutivas que el ordenador debe ejecutar. Los desarrolladores elaboran aplicaciones basados en estos. Según Pérez, Carretero, García, & Pérez (2006), las aplicaciones según el lenguaje utilizado hacen uso de compiladores, editores y bibliotecas.

Según Joyanes (2003), existen tres tipos de lenguajes usados en la actualidad: lenguaje máquina, lenguaje de bajo nivel (ensamblador) y lenguajes de alto nivel.

2.5.1 Lenguajes de alto nivel

Los lenguajes de alto nivel son los más utilizados por los programadores. Porque están diseñados para que las personas escriban y entiendan los programas de un modo mucho más fácil que los lenguajes máquina y ensambladores. Los lenguajes de programación que existen hoy son numerosos aunque la práctica demuestra que en su mayoría se usan: C, C++, COBOL, Visual BASIC, Java, C#, Python, Ruby, AngularJS. Y hoy en día, el mundo de Internet consume gran cantidad de recursos en forma de lenguajes de programación, tales como Java, HTML, HTML5, XML, JavaScript, Visual J, C#, AngularJS, PHP (Joyanes, 2003).

Según Pérez et al., (2006), los lenguajes de alto nivel permiten a los programadores expresar estructuras de datos y de control de forma más sencilla y más parecida a la lógica de las aplicaciones que a las necesidades arquitectónicas del computador. Entre los lenguajes de programación de alto nivel tenemos: COBOL, PL-I, Pascal, C, C++, Ada, Java, Python, Ruby, Perl, etc.

2.5.1.1 Java

“Java se ha convertido en un lenguaje estándar de la industria para el desarrollo de aplicaciones web en internet y es un lenguaje de programación que tiene características excelentes para el aprendizaje. Java fue desarrollado por Sun Microsystems en 1995, es un magnífico lenguaje orientado a objetos diseñado para distribuir contenidos a través de una red” (Joyanes, 2003).

Según Groussard (2014), java es, a la vez, un lenguaje y una plataforma de desarrollo. Es un lenguaje sencillo, orientado a objetos, distribuido, interpretado, robusto, securizado, independientes de las arquitecturas, portable, eficaz, multihilo y dinámico. El lenguaje de desarrollo Java es muy adecuado a los que inician en el desarrollo por tener una sintaxis didáctica. Posee características para convertirse en un lenguaje estándar. Una de las ventajas es la existencia de bastante documentación y un sinnúmero de librerías que se pueden usar para integrar en los diferentes proyectos.

Java es un descendiente de C++ que a su vez es descendiente directo de C. El diseño original fue concebido por James Gosling, Patrick Naughton, Chris Warth, Ed Frank y Mike Sheridan, ingenieros y desarrolladores de Sun Microsystems y tardaron dieciocho meses en terminar la primera versión de trabajo. Inicialmente, se llamaba “Oak” y se cambió por el nombre de Java en la primavera de 1995.

2.5.1.1.1. Tipos de datos básicos y variables

También son llamados primitivos, como son los números reales, enteros, los caracteres, los valores lógicos. En la *Tabla 7* se detallan el tipo, descripción, tamaño y clase equivalente de los tipos de datos de Java.

Tabla 7

Tipo de datos básicos y variables de Java

Tipo	Descripción	Tamaño	Clase equivalente
boolean	Valor lógico	1 bit	Boolean
Char	Carácter	16 bit	Character

Byte	Entero muy pequeño	8 bit	Byte
short	Entero pequeño	16 bit	Short
Int	Entero normal	32 bit	Integer
Long	Entero grande	64 bit	Long
Float	Número real de precisión simple	32 bit	Float
double	Número real de doble precisión	64 bit	Double
Void	Tipo vacío		

Fuente: Groussard (2014)

2.5.1.2 C# (C Sharp)

C# es un lenguaje de programación derivado de C y C++, simple, moderno, orientado a objetos, y confiable. C# se pronuncia “C Sharp”. La finalidad de C# es combinar la productividad de Visual Basic y la potencia de C++ en un solo lenguaje. C# es un lenguaje hecho con propósito general diseñado por Microsoft para su plataforma .NET. Sus principales creadores son Scott Wiltamuth y Anders Hejlsberg, éste último también diseñó el lenguaje Turbo Pascal y la herramienta RAD Delphi (Crovetto, 2004).

Es uno de los lenguajes más jóvenes y representa en la primera década del siglo XXI a un lenguaje de programación con la mayoría de las grandes propiedades que todo lenguaje debe cumplir (Joyanes, 2003).

2.5.1.3 PHP

Según Heurtel (2014), el lenguaje PHP (Personal Home Page) fue diseñado en 1994. Es un lenguaje de script que se ejecuta del lado del servidor, el código PHP se incluye en una página HTML normal. Por lo tanto, se puede comparar a otros lenguajes de script que se ejecutan según

le mismo principio: ASP (Active Server Pages), JSP (Java Server Pages) o PL/SQL Server Pages (PSP).

PHP es el lenguaje de programación más popular para el desarrollo web. Según las estadísticas, en la actualidad se usa en más 20 millones de páginas web y en más de un tercio de servidores Web en el mundo. Su origen se puede rastrear hasta 1994, cuando un desarrollador de nombre Rasmus Lerdorf creó por primera vez un conjunto de scripts CGI para monitorear visitas a la página electrónica donde mantenía su currículum. Sus características únicas son: buen rendimiento, portabilidad, fácil de usar, código libre, soporte comunitario, soporte a aplicaciones a terceros (Vikram, 2009).

Una de las ventajas de PHP es la existencia de muchos servidores gratuitos en la nube para la prueba de aplicaciones.

2.5.1.4 Ruby

Según Arias (2014), Ruby es un lenguaje de script interpretado para programación orientada a objetos con una filosofía y sintaxis muy limpia, lo que hace que programar sea elegante y divertido. Fue creado a principios de la década de los noventa en Japón y rápidamente ganó popularidad en el todo el mundo por estar enfocado hacia la personas.

Su creador, Yukihiro “Matz” Matsumoto, “mezcló partes de sus lenguajes favoritos (Perl, Smalltalk, Eiffel, Ada y Lisp) para formar un nuevo lenguaje que incorporará tanto la programación funcional como la imperativa” (Matsumoto, 2015). Su autor Matsumoto dijo “Ruby es potente y dinámico de código abierto, el lenguaje orientado a objetos que comencé a desarrollar

en 1993”. Rubí se ejecuta en muchas plataformas, incluyendo Linux y muchos sabores de Unix, MS-DOS, Windows, BeOS, y MacOS X. “Ruby es un lenguaje de programación dinámico un poco con una gramática compleja pero expresiva y una biblioteca central con clase y potente API” (Flanagan & Matsumoto, 2008).

2.5.1.5 Python

“Es un lenguaje de programación de alto nivel interpretado y multipropósito, en los últimos años su uso ha ido constantemente creciendo, convirtiéndose en uno de los lenguajes más usados en el desarrollo del software” (Fernández, 2012).

Según Chazallet (2015), “Python es un lenguaje de programación interpretado, multiparadigma, de alto nivel, con un tipado dinámico fuerte, dotado de una gestión automática de los recursos, de un alto grado de introspección y de un sistema de gestión de excepciones”. Python es libre y gratuito, funciona sobre todas las plataformas. Su sintaxis es minimalista, explícita, clara, sencilla y lo suficientemente cercana al lenguaje natural. Python fue diseñado para ser simple. Python con su framework Django es bien usado actualmente porque permite construir aplicaciones de manera ágil y eficiente. También, existen los frameworks: Pyramid, Bottle y Flask.

2.5.1.6 JavaScript

Según Chmiel (2013), JavaScript es uno de los lenguajes de programación más populares para aplicaciones en la web y ha sido elemento básico de la tecnología web desde finales de 1990. Los archivos de JavaScript pueden ser identificados por su extensión de archivo .js. El lenguaje permite a los usuarios realizar muchas de las tareas que ahora consideramos como sinónimo de Internet. A partir de las aplicaciones de comunicación y productividad asíncronas al intercambio de fotos y

juegos sociales. JavaScript permite a los usuarios interactuar directamente con el contenido web. Posee características tales como el reconocimiento de las pulsaciones de teclado y acelerar la capacidad de respuesta en virtud de alojamiento local van más allá de las capacidades de HTML.

JavaScript fue desarrollado por Brendan Eich a mediados de la década de 1990 para el navegador web Netscape Navigator con la esperanza de que podría funcionar como una especie de sistema operativo distribuido que pudiera competir con Microsoft. Fue llamado Mocha, y más tarde LiveScript durante su desarrollo inicial. Netscape acordó una licencia con Sun Microsystems, el creador del lenguaje de programación Java, para llamar finalmente JavaScript. De acuerdo con Eich, citado en una entrevista para InfoWorld, “la idea era hacer una scripting complementaria para ir con Java” (Chmiel, 2013). Las personas que no están familiarizados con la programación web a veces confunden JavaScript con Java. Mientras JavaScript toma algunas de sus convenciones de nomenclatura de Java. A diferencia de Java, que está destinado a programadores informáticos profesionales, JavaScript fue dirigido a diseñadores web y otros que no son programadores.

2.5.1.7 HTML5

Según Lei, Hu, & Zhou (2016), HyperText Markup Language 5 (HTML5) estándar que se finalizó en 2014, aprobado por la mayoría de navegadores convencionales para su entorno con todas la funciones. Con el uso de HTML5 las etiquetas son capaces de funcionar bien, sin uso de plugins que necesitan instalaciones rigurosas. HTML5 abarca nuevas etiquetas semánticas, multimedia sin plugins (audio, video y animaciones de acceso universal), formularios más usables y que validan automáticamente.

2.5.1.8 CSS3

Es el lenguaje de diseño llamado hoja de estilo en cascada (en inglés cascading style sheets), permitió trabajar la parte de la vista para los archivos HTML. Según Beati (2016), CSS3 posee el uso de cualquier tipografía con @font-face, nuevos modelos de color que incluyen translucidez y degradados sin imágenes, efectos visuales como sombras en textos y cajas, bordes redondeados y con imágenes, múltiples fondos para un mismo elemento, técnicas de manipulación de presentación y el movimiento de elementos (transformaciones, transiciones y animaciones).

2.5.2 Comparación entre lenguajes de programación

La compañía TIOBE (2016), especializada en la evaluación y seguimiento de la calidad del software, mide el índice de popularidad de los lenguajes de programación. El índice se actualiza una vez al mes. Las calificaciones se basan en el número de ingenieros calificados mundialmente, cursos y proveedores de terceros. Se utilizan los motores de búsqueda más populares como Google, Bing, Yahoo!, Wikipedia, Amazon, YouTube y Baidu para calcular las puntuaciones. Es importante tener en cuenta que el índice TIOBE no trata de encontrar el mejor lenguaje de programación por medio de la mayoría de las líneas de código que se han escrito.

“El índice TIOBE se puede utilizar para comprobar si sus habilidades de programación siguen siendo hasta la fecha o para tomar una decisión estratégica sobre qué lenguaje de programación debe adoptarse cuando se inicia la construcción de un nuevo sistema de software” (TIOBE, 2016). En la *Figura 5* se muestran a los diez primeros lenguajes de programación con más popularidad en el mundo.

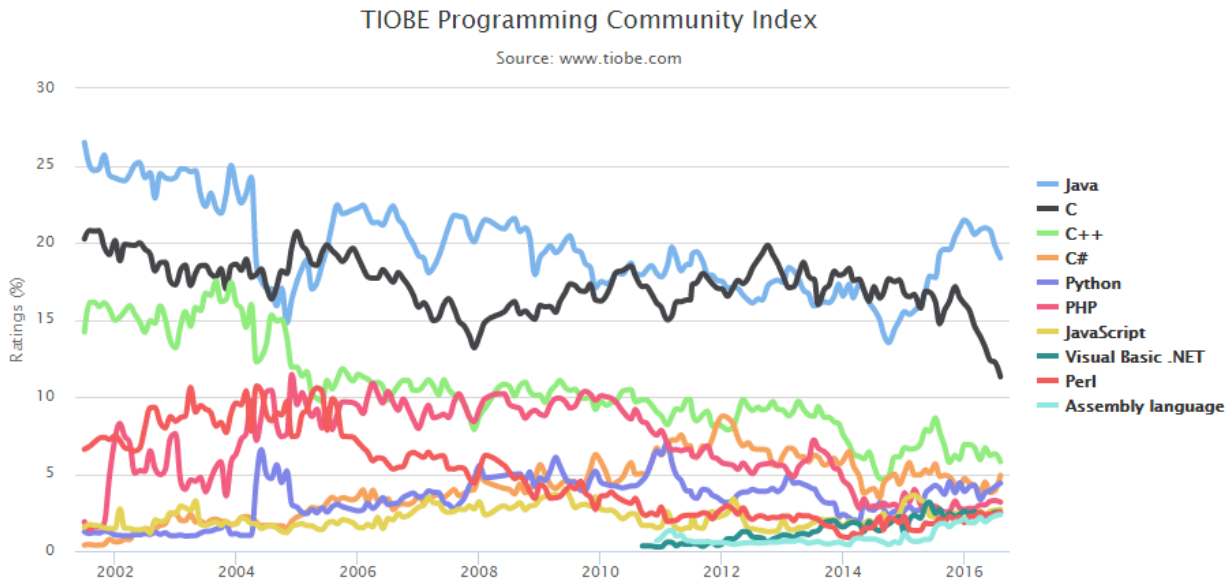


Figura 5. Ranking de lenguajes de programación según TIOBE Index for August 2016

Según TIOBE el lenguaje de programación Java lidera la popularidad a nivel mundial, bajó la popularidad en los últimos años, pero actualmente subió. Es probable que la programación de aplicaciones móviles en Android es el factor para su crecimiento. Java es seguido por los lenguajes más antiguos como C y C++. También se puede observar a los lenguajes Python, PHP y Javascript que están dando mucho que hablar en la actualidad por el uso de una diversidad de frameworks que enriquecen a los lenguajes y la industria del software. En la *Tabla 8* se pueden ver con más detalle los veinte (20) primeros lenguajes de programación con más popularidad, haciendo una comparación entre el año 2015 y el 2016.

Tabla 8

Nivel de popularidad de los lenguajes de programación

Agosto 2016	Agosto 2015	Lenguaje de programación	Calificación	Cambio
1	1	Java	19.01%	-0.26%
2	2	C	11.30%	-3.43%
3	3	C++	5.80%	-1.94%

4	4	C#	4.91%	0.07%
5	5	Python	4.40%	0.34%
6	7	PHP	3.17%	0.44%
7	9	JavaScript	2.71%	0.54%
8	8	Visual Basic .NET	2.52%	-0.19%
9	10	Perl	2.51%	0.39%
10	12	Assembly language	2.36%	0.60%
11	14	Delphi/Object Pascal	2.28%	0.87%
12	13	Ruby	2.28%	0.86%
13	11	Visual Basic	2.05%	0.26%
14	17	Swift	1.98%	0.80%
15	6	Objective-C	1.88%	-1.31%
16	37	Groovy	1.64%	1.27%
17	20	R	1.61%	0.60%
18	15	MATLAB	1.54%	0.31%
19	19	PL/SQL	1.35%	0.21%
20	95	Go	1.27%	1.19%

Fuente: TIOBE (2016)

2.6 Bases de datos

Una base de datos es un conjunto de elementos de datos interrelacionados, administrados como unidad. Sabana (2006) dice que “una base de datos es una colección de datos estructurados según un modelo que refleje las relaciones y restricciones en el mundo real”. Asimismo, Oppel dijo “los procesos que sufran éstos datos tendrán que conservar la integridad y seguridad” (Oppel, 2010). En fin el objetivo de las bases de datos es almacenar la información de los procesos de la organizaciones en tablas, también se usa objetos programados en lenguaje SQL para dar una mejor respuesta a las peticiones de los sistemas.

2.6.1 Sistemas de gestión de Bases de Datos (SGDB / DBMS)

Son un tipo de software muy específico, dedicado a servir de interfaz entre las bases de datos y las aplicaciones que la utilizan, consiguiendo, que el acceso a los datos se realice de una forma más eficiente, más fácil de implementar, y sobre todo, más segura (Sabana, 2006).

Por otro lado, Oppel (2010) dijo “el sistema de administración de bases de datos (DBMS) es el software proporcionado por el vendedor de la base de datos. Productos de software como Microsoft Acces, Oracle, PostgreSQL, Microsoft SQL Server, Sybase ASE, DB2, Ingres, MySQL, SQLite son DBMS”.

2.6.1.1 El lenguaje de Consulta Estructurado (SQL)

Según Sabana (2006), el *Structured Query Language* o Lenguaje de Consulta estructurado es el que permite la comunicación con el Sistema de Gestor de Base de Datos. Es una herramienta para organizar, gestionar y recuperar datos almacenados en una base de datos informática. Las sentencias SQL pertenecen a dos categorías principales: Lenguaje de Definición de Datos (DDL) y Lenguaje de Manipulación de Datos (DML). El DDL permite definir los objetos de la Base de Datos y el DML se utiliza para manejar los datos correspondientes.

Según Godoc (2014), el lenguaje SQL fue creado por IBM a principios de los años 70. Una start-up llamada Relacional Software produjo la primera versión comercial en 1979. Esta start-up se convirtió posteriormente en Oracle Corp. El lenguaje se divide en varios subconjuntos como se ve en la *Tabla 9*:

- El **DDL** (Data Definition Language), que agrupa a todos los comandos utilizados para crear, modificar y eliminar las estructuras de la base de datos (tablas, índices, vistas, etc). Se trata principalmente de los comandos CREATE, ALTER Y DROP.
- El **DML** (Data Manipulation Language), que agrupa los comandos utilizados para manipular los contenidos de la base de datos. Se trata principalmente de los comandos SELECT, INSERT, DELETE y UPDATE.

- El **DCL** (Data Control Lenguaje), que agrupa los comandos utilizados para administrar la seguridad de acceso a los datos, se trata principalmente de los comandos GRANT y REVOKE.
- El **TCL** (Transaction Control Lenguaje), que agrupa los comandos utilizados para administrar la confirmación o no de actualizaciones realizadas sobre la base de datos, se trata principalmente de los comandos COMMIT y ROLLBACK.

Tabla 9

Tipos de sentencias SQL

Sentencia		Descripción
DML (Lenguaje de Manipulación de Datos)	Manipulación de datos	
	SELECT	Recupera datos de la Base de Datos
	INSERT	Añade nuevas filas
	DELETE	Suprime filas
	UPDATE	Modifica datos
DDL (Lenguaje de Definición de Datos)	Definición de datos	
	CREATE TABLE *	Añade una nueva tabla
	DROP TABLE *	Suprime una tabla
	ALTER TABLE *	Modifica la estructura de una tabla
	CREATE VIEW *	Añade una nueva vista
	DROP VIEW *	Suprime una vista
	CREATE INDEX *	Construye un índice
	DROP INDEX *	Suprime el índice
	CREATE SYNONYM *	Define un alias para un nombre de la tabla
DROP SYNONYM *	Suprime un alias	
DCL (Lenguaje de Control de acceso)	Control de acceso	
	GRANT	Consede privilegios de acceso a usuarios
	REVOKE	

	Suprime privilegios de acceso a usuarios
Control de transacciones	
COMMIT	Finaliza la transacción actual
ROLLBACK	Aborta la transacción actual

Fuente: Ramos & Ramos (2007)

Para la obtimización en los procesos de base de datos se han diseñado diferefentes mecanismos que son: secuencias, funciones, triggers, procedimientos almacenados, cursores, entre otros. A continuación se describen algunos de ellos.

2.6.1.1.1. *Triggers*

Según Arias (2016), “trigger es el nombre de un conjunto de comnados que se ejecutan antes o después de que se realiza un cambio en una tablas de base de datos con las operaciones (INSERT, UPDATE y DELETE)”. Los Triggers, conocido como disparadores, son objetos que se asocian con tablas y se almacenan en la base de datos. Su nombre se deriva por el comportamiento que presentan en su funcionamiento.

Según Coronel (2011), “los disparadores automatizan procesos de negocio y mantienen en forma automática la integridad y consistencia de los datos, son de importancia decisiva en un ambiente de negocio moderno”. Por lo tanto, los disparadores ayudan en la automatización de los procesos de negocio, permiten avanzar en el tiempo de desarrollo de proyectos por ser sencillos y prácticas al momento de diseñarlos.

2.6.1.1.2. *Procedimientos almacenados*

Es un programa creado y almacenado físicamente en una base de datos. Sus ventajas están en la optimización de tareas que puede realizar al ser ejecutado, por lo que es hecho en el servidor de la misma base de datos. Como tal, posee acceso directo a los datos que necesita manipular y solo necesita enviar sus resultados de regreso al usuario (sistema), deshaciéndose de la sobrecarga resultante de comunicar grandes cantidades de datos salientes y entrantes.

2.6.1.1.3. Cursores

Es una variable que almacena filas de datos de una consulta. Un cursor puede verse como un iterador sobre la colección de filas que habrá en el set de resultados. Una vez almacenado el cursor es ejecutado en bucle para cumplir con las tareas asignadas; la principal ventaja es que se realiza en la base de datos y, por lo tanto, es rápido y eficiente.

Según Coronel (2011), un cursor “es un constructo especial empleado en los SQL procedimentales para retener renglones de datos devueltos, por una consulta de SQL”. Se puede considerar a los cursores como áreas de memorias reservadas donde la salida de la consulta se guarda, como un arreglo de columnas y filas. Existen dos tipos de cursores: implícitos y explícitos.

2.6.1.1.4. Secuencias

Es una función en la base de datos generadora de números, maneja número mínimo, máximo y contador; es usado normalmente para los valores de llaves primarias porque resulta muy eficiente y se les puede dar un variedad de usos.

2.6.1.1.5. Funciones

Es un objeto que permite el ingreso de valores para relizar procesos programados y te devuelve un solo valor como resultado, el tipo de valor del programador de base de datos. Según Coronel (2011), una función es básicamente un grupo de enunciados procedimental y de SQL, con nombre, que entrega a un valor (indicado por un enunciado RETURN en su código de programa).

2.6.1.1.6. Jobs

Según Benítez & Arias (2015), un job “es una tarea administrativa, compuesta de uno o más pasos, la cual es programada para ejecutarse automáticamente, en las fechas y horarios determinados. Automatizan las tareas rutinarias que el administrador de base de datos debe hacer, tales como backups, mantenimientos de índices y cualquier otra tarea que sea necesaria a automatizar”.

2.6.1.2 Gestores de base de datos

Según Cotos & Taboada (2005), “un gestor de base de datos es un programa o conjunto de programas cuyo propósito consiste en facilitar un acceso seguro, eficiente y transparente a la base de datos por parte de uno o más usuarios”. El gestor facilita la definición de los contenidos de la base de datos y la inserción, modificación y eliminación de nueva información. Entre las ventajas que ofrece un gestor de base de datos son los siguientes: control centralizado, independencia de los datos, fácil implementación, acceso directo por parte del usuario, control de redundancia, creación de perfiles de usuario, verificación de la integridad de la información.

Entre las bases de datos relacionales más conocidas podemos encontrar: MySQL, Oracle, Microsoft SQL Server, Microsoft Access, Visual FoxPro, PostgreSQL, Apache Dervy, BD2, DB2

Express-C, SQLite, IBM Informix, FileMaker y MongoDB. A continuación se describe algunos de ellos.

2.6.1.2.1. *PostgreSQL*

Según Sabana (2006), “es un sistema avanzado de administración de base de datos objeto relacionales (ORDBMS) de código abierto (gratuito y código fuente disponible). Posee interoperabilidad con otros DBMS: SQL. El almacenamiento es confiable, consistente y robusto. Manipulación potente, flexible y eficiente”. Una de las principales ventajas de este motor de base de datos es la sencillez tanto en la instalación y la administración. Por ser gratuito, se puede disponer de este para cualquier organización que se quiera implementar sistemas de información.

Para Chazallet (2015), PostgreSQL es una base de datos relacional libre muy robusta, fiable y con un buen rendimiento que ofrece gran cantidad de funcionalidades que permiten responder a necesidades muy avanzadas de manera particularmente eficaz.

Además, López, Núñez, & Gonzáles (2011) argumentaron que PostgreSQL representa una solución con la calidad requerida para satisfacer las demandas de los usuarios y es equiparable a cualquier sistema gestor propietario. Actualmente, es considerado como el sistema gestor de código abierto más avanzado a escala mundial por las características que posee, atribuible, por lo general, a los productos comerciales de alto calibre. Dispone de una serie de funcionalidades inherentes a las bases de datos con altas prestaciones que lo hacen apto para la mayoría de las aplicaciones, ya que fue diseñado y creado para tener un mantenimiento y ajuste mucho menor que otros productos. Además, ofrece ventajas significativas como elevada concurrencia, amplia

variedad de tipos de datos nativos, gran adaptabilidad, escalabilidad, entre otras, que justifican su preferencia.

2.6.1.2.2. *Oracle*

Oracle es un DBMS poderoso y robusto que funciona en muchos sistemas operativos diferentes, es el más popular en el mundo y tiene larga historia de desarrollo y uso. (Kroenke, 2003). “En 1977 fue presentada la primera versión comercial SQL por Oracle Corporation” (Ramos & Ramos, 2007).

2.6.1.2.3. *MySQL*

Es un gestor de base de datos sencillo e increíblemente rápido. Es uno de los motores más usados en internet. Entre las ventajas que poseen son las siguientes:

- Es gratis bajo ciertas circunstancias.
- Para usos comerciales, es una solución más barata que otros gestores de base de datos.
- Es muy rápido para recuperar información.
- Es una versión perfectamente estable en Windows.

2.6.1.2.4. *SQL Server*

Según Microsoft (2014), SQL Server, la base de la amplia plataforma de datos de Microsoft, ofrece un rendimiento fiable gracias a la integración de tecnologías en memoria, una rápida obtención de información útil a partir de cualquier tipo de datos, con herramientas que todos conocemos, como Excel, y una plataforma para compilar, implementar y administrar soluciones tanto locales como en la nube.

2.6.1.2.5. *SQLite*

Según Chazallet (2015), “SQLite es un motor de base de datos relacional escrito en C que, a diferencia de MySQL y PostgreSQL, no funciona según el modelo de cliente-servidor, sino que se ha diseñado para estar embebido en un programa”. Resulta útil, particularmente cuando se desea distribuir una aplicación que deba manipular datos que no dependan de un servidor de base de datos relacional, la desventaja es la disminución de rendimiento a mayor cantidad de datos.

2.6.1.3 *Ventajas y desventajas de los gestores de base de datos*

En la *Tabla 10* se detallan las principales ventajas y desventajas de tres de los gestores de bases de datos. En el proyecto se usa PostgreSQL por la afinidad del investigador por ser un gestor libre, intuitivo y eficiente.

Tabla 10

Ventajas y desventajas de los principales gestores de bases de datos

Base de datos	Ventajas	Desventajas
PostgreSQL	<ul style="list-style-type: none">- Relacional orientado a objetos y libre, publicado bajo la licencia BSD.- Posee el sistema denominado MVCC (Acceso concurrente multi-versión) conocido como alta concurrencia.- Soporta diferentes tipos de datos.- Soporta replicación de base de datos asíncrona, proporcionando tolerancia a fallos.- Posee un buen sistema de seguridad mediante la gestión de usuarios, grupos	<ul style="list-style-type: none">- Consume más recursos que MySQL, por lo que se necesita mayores características de hardware.- La sintaxis de algunos de sus comandos no es nada intuitiva, aunque posee amplia documentación de ellos.

	<ul style="list-style-type: none"> de usuarios, permisos y contraseñas. - Posee gran capacidad de almacenamiento. - Tiene algunas herramientas gráficas para administrar el servidor y sus bases de datos. 	
Oracle	<ul style="list-style-type: none"> - Base de datos con más orientación hacia internet. - Soporta todas las funciones que se posee un servidor serio. <ul style="list-style-type: none"> - Permite el uso de particiones para la mejora de la eficiencia, de replicación, hasta incluso base de datos distribuidas. - El software del servidor puede ejecutarse en una variedad de sistemas operativos. 	<ul style="list-style-type: none"> - Es un gestor propietario y su precio es elevado. - La instalación es trabajosa y cuando está mal configurado puede ser relativamente lento. <ul style="list-style-type: none"> - Es elevado el costo de formación en las herramientas de este software.
MySQL	<ul style="list-style-type: none"> - Es software libre, gestor de base de datos relacional, licenciado bajo la GPL de la GNU. <ul style="list-style-type: none"> - Es un sistema de administración de base de datos (DBMS) para tipo relacionales. - Su diseño multi-hilo le permite soportar una gran carga de forma muy eficiente. - Se adapta a varios entornos de desarrollo e interactúa con los lenguajes más usados como PHP, Perl, Java, Ruby y se integra con varios sistemas operativos. - Servidor de base de datos muy rápido, fiable y fácil de usar. - Posee un buen control de acceso de usuarios y seguridad de datos. 	<ul style="list-style-type: none"> - Soporte para disparadores es básico, por lo que tiene ciertas limitaciones. - Los privilegios para una tabla no se eliminan automáticamente. Debe usarse explícitamente un comando REVOKE para quitar los privilegios de una tabla. - Se tiene que usar las tablas InnoDB, que se respete las llaves secundarias al momento de eliminar los datos.

- Tiene una integración perfecta con PHP.
 - Soporte completo para cláusulas, funciones, tipos de datos y comandos estándar SQL.
-

Fuente: recopilación propia

2.6.2 Clasificación de motores de base de datos

DB-Engines es una iniciativa que clasifica a los sistemas de gestión de base de datos en función a su popularidad, donde participan 315 motores de base de datos. La clasificación se actualiza mensualmente. Se mide la popularidad de un sistema mediante el uso de los siguientes parámetros:

- **Número de menciones del sistema en los sitios web:** medido como el número de resultados en los motores de búsqueda. Se usa Google, Bing y Yandex para esta medición. Con el fin de contar con resultados sólo pertinentes, se busca con el <nombre del sistema> junto con el término base de datos, por ejemplo, "Oracle" y "base de datos".
- **Interés general en el sistema.** Para esta medición, se utiliza la frecuencia de las búsquedas en Google Trends que vienen a ser las tendencias mundiales.
- **Frecuencia de las discusiones técnicas sobre el sistema:** utilizan el número de preguntas relacionadas y el número de usuarios interesados en los conocidos relacionados con las TI de preguntas y respuestas en los sitios Stack Overflow y DBA Stack Exchange.
- **Número de ofertas de trabajo, en los que se menciona el sistema:** utilizan el número de ofertas en los principales motores de búsqueda de Indeed y Simply Hired.
- **Número de perfiles en redes profesionales, en los que se menciona el sistema.** Utilizan las más populares a nivel internacional que son LinkedIn y Upwork.

- **Relevancia en las redes sociales.** Cuentan con el número de tweets de Twitter, en la que se menciona el sistema.

En la *Figura 6* se muestra el nivel de popularidad a nivel mundial de los gestores de base de datos.

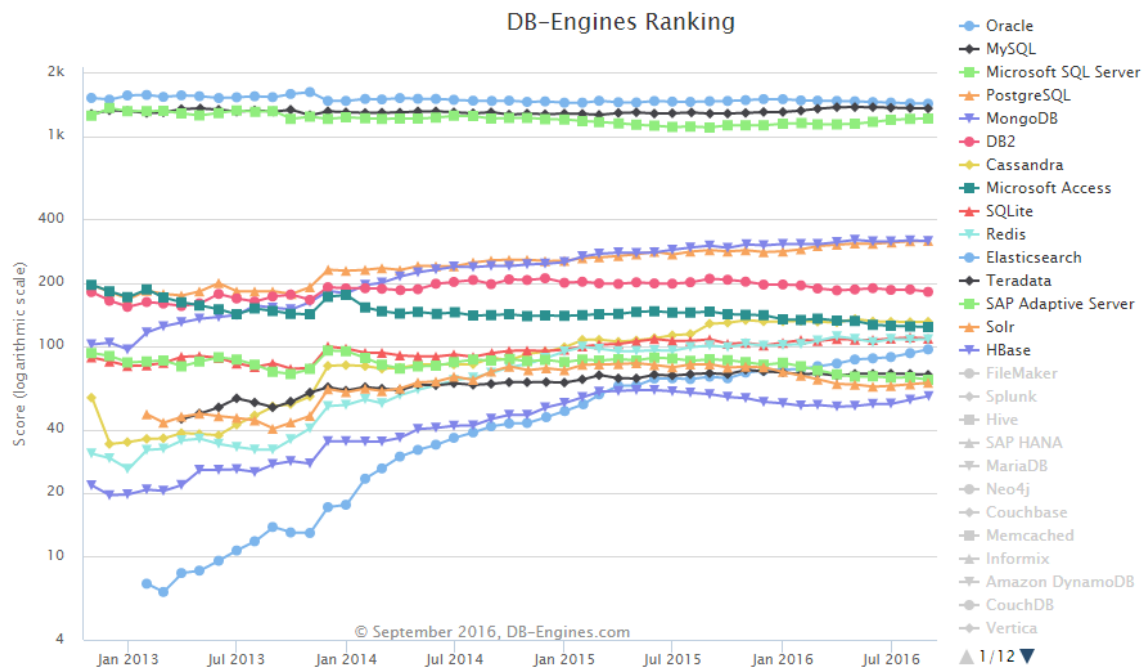


Figura 6. Niveles de popularidad de los gestores de Base de Datos - DB-Engines (agosto 2016)

Como se ve en la *Figura 6*, los motores con mayor popularidad en el mundo son Oracle, MySQL, Microsoft SQL Server y PostgreSQL. PostgreSQL está subiendo su nivel de popularidad, tal vez una de las ventajas que provee PostgreSQL es su licencia libre para el uso. En la *Tabla 11* se presenta la lista de los treinta (30) motores de base de datos con sus respectivos puestos de popularidad, entre septiembre de 2015 y agosto de 2016.

Tabla 11

Nivel de popularidad de Base de Datos

Rango			DBMS	Modelo de base de datos	Puntuación		
Setie 2016	Ago 2016	Setie 2015			Setie 2016	Agost 2016	Setiem 2015
1	1	1	Oracle	Relational DBMS	1425.56	-2.16	-37.81
2	2	2	MySQL	Relational DBMS	1354.03	-3.01	76.28
3	3	3	Microsoft SQL Server	Relational DBMS	1211.55	6.51	113.72
4	5	5	PostgreSQL	Relational DBMS	316.35	1.1	30.18
5	4	4	MongoDB	Document store	316	-2.49	15.43
6	6	6	DB2	Relational DBMS	181.19	-4.7	-27.95
7	7	8	Cassandra	Wide column store	130.49	0.26	2.89
8	8	7	Microsoft Access	Relational DBMS	123.31	-0.74	-22.68
9	9	9	SQLite	Relational DBMS	108.62	-1.24	0.97
10	10	10	Redis	Key-value store	107.79	0.47	7.14
11	11	14	Elasticsearch	Search engine	96.48	3.99	24.93
12	12	13	Teradata	Relational DBMS	73.06	-0.57	-1.2
13	13	11	SAP Adaptive Server	Relational DBMS	69.16	-1.88	-17.36
14	14	12	Solr	Search engine	66.96	1.19	-14.98
15	15	15	HBase	Wide column store	57.81	2.3	-1.22
16	16	17	FileMaker	Relational DBMS	55.35	0.34	4.35
17	17	18	Splunk	Search engine	51.29	2.38	9.06
18	18	16	Hive	Relational DBMS	48.82	1.01	-4.71
19	19	19	SAP HANA	Relational DBMS	43.42	0.68	5.22
20	20	25	MariaDB	Relational DBMS	38.53	1.65	14.31
21	21	21	Neo4j	Graph DBMS	36.37	0.8	2.83
22	24	24	Couchbase	Document store	28.54	1.14	2.28
23	23	22	Memcached	Key-value store	28.43	0.74	-3.99
24	22	20	Informix	Relational DBMS	28.19	-0.86	-9.76
25	25	28	Amazon DynamoDB	Document store	27.42	0.82	7.43
26	26	23	CouchDB	Document store	21.48	0.42	-5.12
27	27	30	Vertica	Relational DBMS	21.06	0.58	3.31
28	29	27	Netezza	Relational DBMS	19.81	0.34	-1.24
29	28	29	Microsoft Azure SQL Database	Relational DBMS	19.42	-0.11	0.75
30	30	26	Firebird	Relational DBMS	15.58	-0.32	-6.63

Fuente: DB-Engines (2016)

2.7 Servidor web

Según Ramos (2011), un servidor web es programa diseñado por aceptar peticiones HTTP del navegador y servir las páginas web que tiene alojadas. Cada sitio web tiene una dirección única conocida como URL (Uniform Resource Locator – Localizador Uniforme de Recursos). En una URL se pueden distinguir tres partes: el protocolo, el nombre del dominio del ordenador servidor y la ruta en el servidor. Existen muchos servidores web tales como: Apache Tomcat, Microsoft IIS, Sun Java System Web Server, Nginx, Lighttpd, Oracle HTTP Server, Xampp, Baby Web Server.

2.7.1 Apache Tomcat

Apache Tomcat es una implementación de código abierto de software de las tecnologías Java Servlet y Java Server Pages. Las especificaciones Java Servlet y Java Server Pages son desarrollados bajo la Java Community Process. Apache Tomcat se desarrolla en un entorno abierto y participativo y publicado bajo la licencia Apache versión 2. Apache Tomcat está destinada a ser una colaboración de los desarrolladores mejor de su clase en todo el mundo” (Tomcat, 2014).

2.8 Organización Internacional de Normalización

Según Miranda, Chamorro, & Rubio (2007), la Organización Internacional de Normalización (ISO), es un organismo con sede en Ginebra. Es una federación mundial de organismos nacionales de normalización que cuenta en la actualidad con más de 149 estados miembros. Es el encargado de la coordinación y unificación de las normas industriales, abarcando todos los campos de fabricación, comercio, y comunicaciones para todas las áreas industriales. Las ventajas que le trae a las organizaciones aparte de ser competitivos son por ejemplo incremento de las prestaciones de

sus servicios y productos, mayor satisfacción del cliente, mejor opinión del cliente, aumento de la productividad y eficiencia, reducción de gastos, mejora del nivel de comunicación y de satisfacción de los trabajadores, mayor competitividad y aumento de las oportunidades de venta.

La ISO cuenta con más 21000 Normas Internacionales publicadas. Según la ISO, una norma es un documento que proporciona los requisitos, especificaciones, directrices o características que se pueden utilizar constantemente para asegurar que los materiales, productos, procesos y servicios son adecuados para su propósito.

Según Yunkella & Bryden (2010) dijeron que “las normas pueden servir de base técnica para el comercio en productos finales y servicios entre compradores y vendedores, o como un medio para facilitar la conformidad con las reglamentaciones técnicas. Son desarrolladas a través de un proceso transparente, abierto y de consenso que involucra a las partes interesadas”.

2.8.1 NTP ISO/IEC 12207

NTP ISO/IEC 12207 es la unión de Norma Técnica Peruana (NTP) e International Electrotechnical Commission (IEC). Según Indecopi (2006), esta norma pertenece al área de tecnologías de información. Abarca todos los procesos del ciclo de vida del software. Fue elaborada por el Comité Técnico de Normalización de Ingeniería de Software, utilizando como antecedente a la Norma ISO/IEC 12207:1995, 2002, 2005 Information technology. Software life cycle processes.

“El objetivo de esta norma es establecer un marco de referencia común para los procesos del ciclo de vida de software, con una terminología bien definida a lo que puede hacer referencia la

industria del software. La norma es aplicable a la adquisición de sistemas, productos y servicios software y a la parte del software del firmware, independiente sea hecho interna o externamente a la organización. Contiene un conjunto de procesos, actividades y tareas diseñadas a ser adoptadas en los proyectos software” (Indecopi, 2006) .

Su limitación consiste que esta NTP describe la arquitectura de los procesos del ciclo de vida del software, pero no especifica los detalles de cómo implementar o llevar a cabo las actividades y tareas incluidas en el proceso.

Según Indecopi (2006), los procesos principales del ciclo de vida que define esta NTP son las siguientes:

- Procesos de adquisición
- Proceso de suministro
- Procesos de desarrollo
- Proceso de operación
- Proceso de mantenimiento

Además se encuentran los procesos que apoyo del ciclo de vida y los procesos organizativos del ciclo de vida que se detallan en la *Figura 7*.



Figura 7. Estructura de la Norma Técnica Peruana 12207. Fuente: NTP ISO/IEC 12207

2.8.1.1 *Proceso de desarrollo*

Según Indecopi (2006), el proceso de desarrollo, es uno de los procesos del ciclo del software. Contiene las actividades y tareas del desarrollador. El proceso contiene las actividades para el análisis de los requerimientos, diseño, codificación, integración, pruebas e instalación y aceptación del producto, se llevan a cabo basadas en el contrato.

Este proceso consta de las siguientes actividades:

- a) Implementación del proceso

- b) Análisis de los requerimientos del sistema
- c) Diseño de la arquitectura del sistema
- d) Análisis de los requerimientos del software
- e) Diseño de la arquitectura del software
- f) Diseño detallado del software
- g) Codificación y pruebas del software
- h) Integración del software
- i) Pruebas de calificación del software
- j) Integración del sistema
- k) Pruebas de calificación del sistema
- l) Instalación del software
- m) Apoyo de aceptación del software

2.9 Integración de la NTP ISO/IEC 12207 y la metodología ágil OpenUP

Se tomó el ciclo de desarrollo de la NTP ISO/IEC 12207 con el objetivo de poder integrar con la metodología ágil, validando el cumplimiento en parte, el estándar de calidad. Se elaboró la *Tabla 12* para poder visualizar mejor esta integración. En la primera columna se muestra la norma técnica con sus principales actividades y en la siguiente columna con la metodología ágil con sus principales fases.

Tabla 12

Integración de NTP 12207 y la metodología OpenUP

NTP ISO/IEC 12207 (Proceso de desarrollo)	OpenUP
<ul style="list-style-type: none"> • Análisis de los requerimientos del sistema 	<p>Concepción</p> <ul style="list-style-type: none"> - Entender qué construir. - Identificar la funcionalidad clave del sistema. - Determinar al menos una posible solución. - Entender el costo, el cronograma y los riesgos asociados al proyecto.
<ul style="list-style-type: none"> • Diseño de la arquitectura del sistema • Análisis de los requerimientos del software • Diseño de la arquitectura del software • Diseño detallado del software 	<p>Elaboración</p> <ul style="list-style-type: none"> - Obtener un entendimiento más detallado de los requisitos. - Diseñar, implementar, validar y establecer la línea base para la arquitectura. - Mitigar los riesgos esenciales y producir un cronograma exacto y unos costos estimados.
<ul style="list-style-type: none"> • Codificación y pruebas del software • Integración del software • Pruebas de calificación del software • Integración del sistema • Pruebas de calificación del sistema 	<p>Construcción</p> <ul style="list-style-type: none"> - Desarrollar iterativamente un producto completo que esté listo para hacer transición a la comunidad de usuarios. - Minimizar el costo de desarrollo y alcance algún grado de paralelismo.
<ul style="list-style-type: none"> • Instalación del software • Apoyo de aceptación del software 	<p>Transición</p> <ul style="list-style-type: none"> - La prueba beta valida que las expectativas del usuario sean satisfechas.

- Lograr que los Stakeholders concuerden en que la implementación ha terminado.
- Mejorar el desempeño en futuros proyectos a través de lecciones aprendidas.

Fuente: elaboración propia

En la *Figura 8* se muestra claramente la metodología ágil OpenUP y la Norma Técnica Peruana, donde se aplicó toda la metodología con el proceso de desarrollo de la NTP.

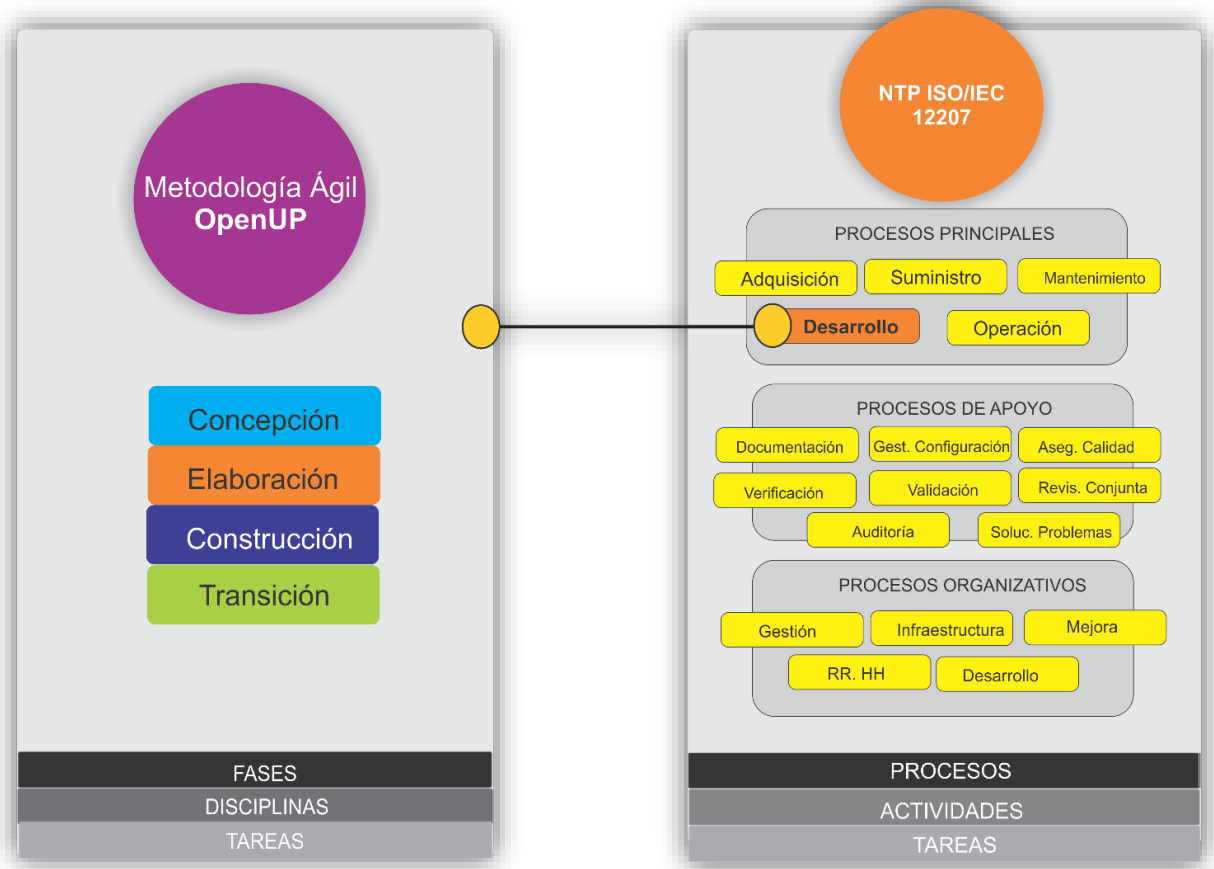


Figura 8. Metodología ágil OpenUP y NTP 12207

CAPÍTULO III

MATERIALES Y MÉTODOS.

3.1 Introducción

Este capítulo describe brevemente las partes de la metodología empleada. Empieza mostrando gráficamente el diseño de la investigación y continúa con el detalle de cada una de las fases.

3.2 Diseño de la investigación

El diseño de la investigación es el proceso metodológico de la investigación que involucra las fases del proyecto. Se divide en cuatro (4) fases principales, y cada una de ellas contiene sub fases mostrados en la *Figura 9*.



Figura 9. Diseño de la investigación – elaboración propia

3.2.1 Análisis situacional de los grupos

3.2.1.1 Información de los Grupos organizados

En esta fase, se investigó la dinámica de los grupos organizados, la información relacionada con la iglesia y las facultades de la universidad, se conoció el motivo de la existencia del grupo organizado, su misión y su visión como grupo dentro de la institución.

3.2.2 Identificación de requerimientos

3.2.2.1 Consulta a clientes directos

Esta etapa permitió obtener los requerimientos básicos que llegaron a ser las principales características del software construido. Para ello se aplicaron métodos de recolección de requisitos tales como encuestas, entrevistas y reuniones con los interesados del proyecto, esto en una primera fase y se consolidó en la siguiente fase.

3.2.3 Aplicación de la metodología y la NTP ISO/IEC 12207

OpenUP como metodología de desarrollo ágil presenta cuatro fases concretas en el proceso de desarrollo de proyectos de software. Estas fases son: concepción, elaboración, construcción y transición. Cada fase presenta objetivos, actividades y entregables.

3.2.3.1 Concepción

Esta fase permitió definir claramente lo que se tuvo que construir y llegando a la conclusión que el proyecto es viable. Se elaboró el documento de requerimientos del sistema, incluyendo los límites. Además, se especificó los requisitos de diseño y técnicos del software. Se elaboró el

documento visión (disponible en el *Apéndice A* de este documento) donde se definió los objetivos y entregables a cumplir.

3.2.3.2 *Elaboración*

Esta fase constó de varias actividades de preparación para el desarrollo. Se elaboraron varios documentos, tales como del diseño del sistema, diseño de base de datos, estándares de programación, diseño de la interfaz gráfica, el plan de pruebas y culminó con las cartas de aceptación del proyecto por parte de la iglesia.

3.2.3.3 *Construcción*

En esta fase se desarrolló el sistema por medio de iteraciones programadas por periodos de tiempo según niveles de complejidad. Por cada iteración, se obtuvo una versión del software y su documentación respectiva y el manual de uso para el usuario. Además, se aplicó las actividades que especifica la NTP ISO/IEC 12207 en el proceso de desarrollo, el objetivo fue obtener un software de calidad que cumpla con los requerimientos de los usuarios finales.

3.2.3.4 *Transición*

Esta etapa permitió preparar cada versión del software y de base de datos, para la puesta en producción. Hay algunos objetivos que forman parte de la transición que han sido puestas en las siguientes fases.

3.2.4 *Implantación de la solución y seguimiento*

3.2.4.1 *Implantación*

Esta etapa permitió la instalación del producto software de tal forma esté lista para los usuarios finales y su uso con los procesos correspondientes a los grupos organizados de la iglesia universitaria.

3.2.4.2 *Capacitación a los usuarios y pruebas*

En este proceso se capacitó a los usuarios con el uso del sistema. También abarcó las pruebas de software permitiendo la gestión de cambios y mejoras necesarias del producto. Esto en cada versión del producto, dando cumplimientos a las actividades que promueve la Norma 12207.

3.2.4.3 *Pruebas*

Este proceso permitió hacer pruebas de usuario, de tal forma que los beneficiarios finales puedan trabajar de manera eficiente, el objetivo fue eliminar cualquier indicio de error tras el proceso de desarrollo.

3.2.4.4 *Validación de resultados*

Este proceso permitió validar la propuesta con el sistema en producción, por medio de encuestas de satisfacción aplicado a los usuarios finales, el objetivo fue comprobar que el producto sea de calidad satisfaciendo las necesidades de los usuarios.

Una vez concluido todo el proyecto, se capacitó a los usuarios y finalmente se hizo la entrega final del producto. Se cerró el proyecto, con todos los documentos necesarios. Los cambios que puedan surgir más adelante lo gestionará la misma iglesia universitaria por medio del departamento de Sistemas de la Universidad.

3.3 Tipo de investigación

La investigación empírica es de tipo cuantitativa, porque contrasta las teorías de la metodología ágil OpenUP con la Norma Técnica Peruana en el desarrollo del sistema de información.

Según el propósito de esta investigación es descriptiva y aplicada:

- **Descriptiva:** porque se analizó, recolectó y validó los datos (información) de los procesos relacionados con los grupos organizados, con el objetivo de la obtención de requerimientos que llegaron a ser parte de las características funcionales de la solución propuesta.
- **Aplicada:** porque la investigación está encaminada a la solución de problemas prácticos, definidos y específicos. También porque se aplica el conocimiento para resolver problemas de cuya situación depende el beneficio de individuos (pastores, capellanes, líderes) y la comunidad (integrantes de los grupos e iglesia en general), mediante el uso de herramientas tecnológicas.

CAPÍTULO IV

CONSTRUCCIÓN DE LA PROPUESTA

4.1 Introducción

En este capítulo se detalla la construcción del sistema basado en el modelo de investigación empezando desde la primera fase que es “análisis de la situación actual de los grupos organizados” y se culmina con la “validación de los resultados”. En cada una las fases se presentan gráficos y tablas que resultan del desarrollo.

4.2 Análisis situacional de los grupos

4.2.1 Información de los Grupos organizados

En entrevista a Jimenez et al. (22 de abril de 2015) mencionaron que la Iglesia Universitaria Villa Unión de la Universidad Peruana Unión, filial Tarapoto en el semestre académico 2015-1 constó de 49 grupos pequeños, 17 escuelas sabáticas, 1 orquesta de cámara, 1 coro universitario, 1 club de líderes, 1 Culto de Alva, 1 clase de cuna, 2 clases de infantes y 1 clase de juveniles. En la *Figura 10* se representa la forma de organización de los grupos organizados.

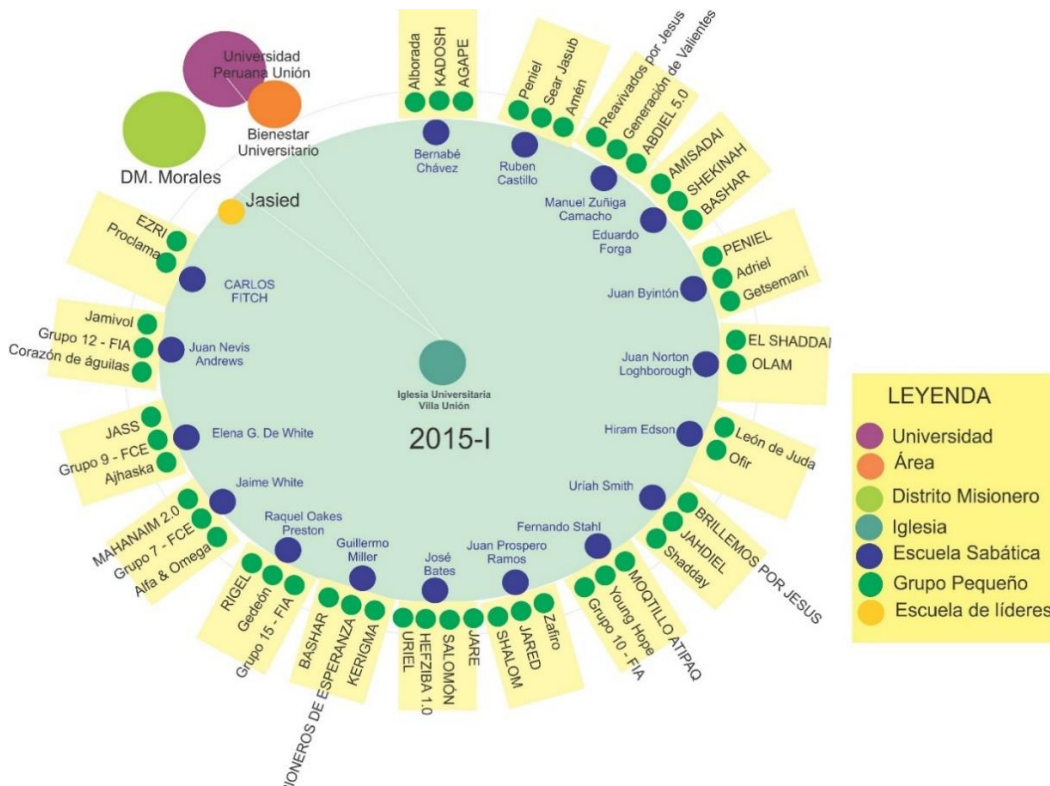


Figura 10. Organización de los grupos en la Iglesia Universitaria Villa Unión

Fuente: elaboración propia

La Iglesia Universitaria Villa Unión (de aquí en adelante iglesia) es la que lidera las actividades de cada grupo organizado. La iglesia por ser universitaria llega a alcanzar un poco más de 750 de miembros entre alumnos, docentes, personal y visitas. Es liderada por un pastor el cual coordina el desarrollo de las actividades con los diferentes departamentos que posee la iglesia. Siendo los más importantes el departamento de Escuela Sabática y Grupos Pequeños en este proyecto de investigación.

La iglesia está relacionada con el área institucional de Bienestar Universitario de la Universidad Peruana Unión, filial Tarapoto (de aquí en adelante *universidad*) siendo representado en cada facultad por un capellán, especialista en la organización de grupos organizados, al mismo tiempo es capacitador y orientador para los líderes de los grupos que pertenecen a cada facultad.

La organización es de la siguiente manera: la iglesia pertenece académicamente a la universidad y organizacionalmente a un distrito misionero (en este caso a Morales), el distrito misionero a una misión o asociación, así sucesivamente. La iglesia está organizada en escuelas sabáticas. A cada escuela sabática, pertenecen de dos a más grupos pequeños. Cada grupo pequeño académicamente pertenece a una facultad. La escuela sabática es representada por un director y demás cargos, y en cada grupo pequeño hay un líder, un co-líder (líder asociado) y un consejero (persona experimentada para apoyar y capacitar al líder y co-líder).

En cada grupo se organizan sub-grupos. Por ejemplo, en grupos pequeños se organizan parejas misioneras, cada pareja misionera inscribe a sus estudiantes de la Biblia y controla los estudios bíblicos de cada estudiante.

Cada grupo organizado tiene diferentes tipos de reuniones donde se desarrolla un programa con anticipación, se registra la asistencia de cada persona y se evalúa según los criterios establecidos en cada tipo de grupo y tipo de reunión.

En la *Figura 11* se muestra la organización de manera general con participación del pastor de iglesia, capellanes, cargos de la iglesia, escuelas sabáticas y grupos pequeños. También se visualiza los procesos que realizan y algunas funciones de los actores.

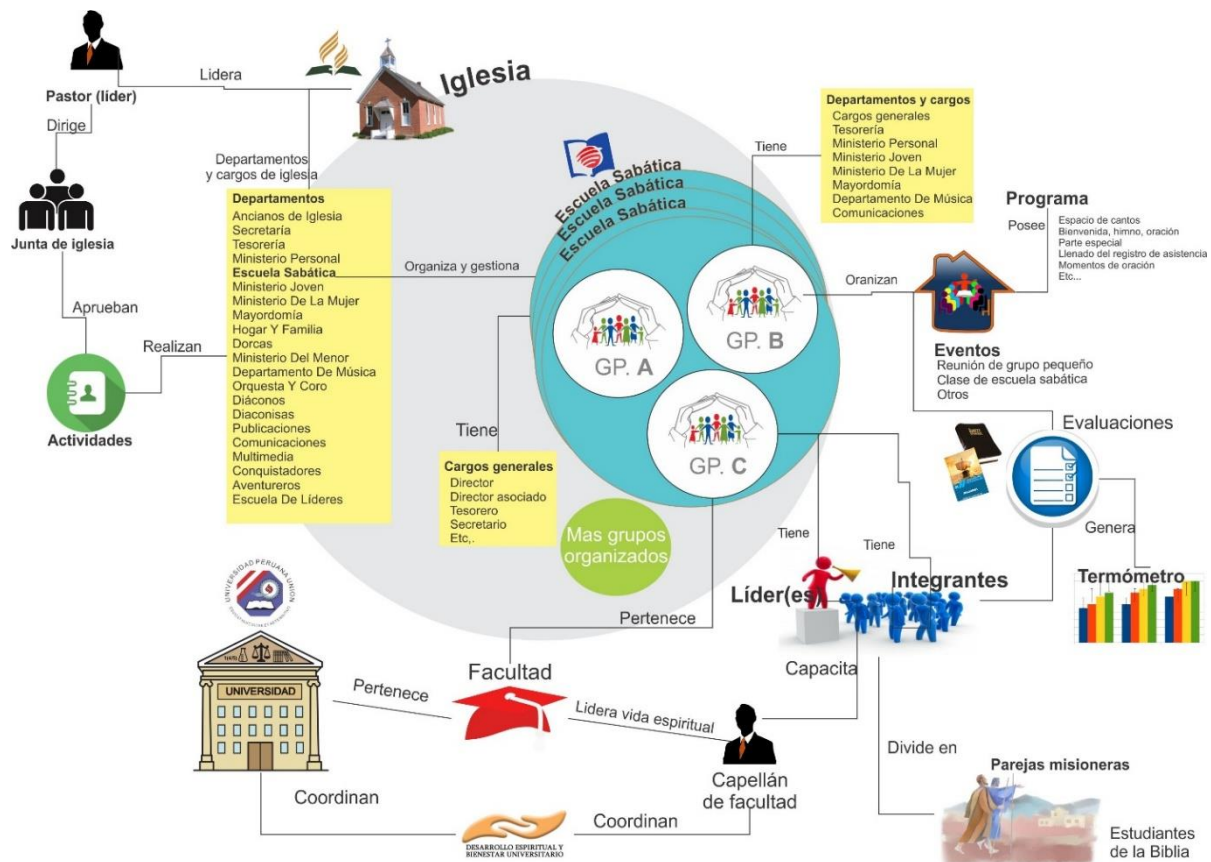


Figura 11. Organización de la iglesia, escuelas sabáticas y grupos pequeños en 2015-1. Fuente: elaboración propia

En conclusión, se ha investigado los procesos que se dan en cada tipo de grupo y se llegó a la conclusión que no poseen un sistema de información automatizado para sus diferentes procesos. Se identificaron a los actores principales de los procesos, que en la siguiente etapa del proyecto se aplicaron encuestas y entrevistas para lo obtención de requerimientos tanto funcionales como no funcionalidades.

4.2.2 Información de los Grupos organizados

4.2.2.1 Consulta a clientes directos

Esta etapa permitió la aplicación de una serie de entrevistas a especialistas de la organización de grupos, tales como: pastor de iglesia, capellanes de facultad, director de ministerio personal (MiPES), director de escuela sabática, director de Bienestar Universitario. Esta etapa fue muy importante porque permitió entablar un diálogo directo con las personas interesadas, que vienen a ser los usuarios principales del sistema, porque son los responsables de los procesos.

4.3 Concepción

Esta fase permitió cumplir los siguientes objetivos en el proyecto, que son la identificación de los requerimientos funcionales y no funcionales del sistema. Los requerimientos han sido obtenidos en la consulta a clientes directos y haciendo un análisis de una variedad de formatos que usan para la gestión y evaluación de los grupos organizados. En la *Tabla 13* se presenta los requerimientos funcionales que se obtuvieron, además se describe el nivel de prioridad y nivel de dificultad que generó construir cada funcionalidad.

4.3.1 Visión del producto

La visión del producto quedó plasmada en el documento de la Visión de la aplicación (Disponible completo en el Apéndice A). A continuación se detallan los elementos más relevantes de la visión del proyecto.

4.3.1.1 *Requerimientos funcionales*

Los requerimientos funcionales son las características que debe cumplir el sistema. Esto depende de los procesos que contiene el negocio. La *Tabla 13* presenta la lista de requerimientos obtenidos.

Tabla 13

Requerimientos funcionales del sistema

Módulo Seguridad			
N°	Descripción	Dific.	Prior.
1	El sistema permitirá el mantenimiento de los usuarios. Todas las personas inscritas tienen la posibilidad de disponer de un usuario para el acceso al sistema.	2	2
2	El sistema permite el mantenimiento de perfiles de usuario y accesos a los módulos del sistema. El perfil o rol de usuario especifica los accesos que tienen a las funcionalidades del sistema.	2	2
3	El sistema permite la asignación del perfil de usuario a uno o varios usuarios. Un usuario puede disponer uno o varios perfiles, disponiendo de un solo perfil por grupo en un ciclo o periodo.	2	2
4	El sistema permitirá al usuario el cambio de contraseña Al momento de cambiar la clave, el sistema lo encriptará para mayor seguridad y lo guardará en base de datos.	2	2
5	El sistema permitirá la personalización de accesos para los perfiles de usuario. Permitirá el cambio de accesos otorgados a un perfil de usuario para aumentar o disminuir accesos de manera automática.	2	2
6	El sistema permitirá la creación de usuario de forma independiente por cada persona. Cualquier persona registrada podrá crear un usuario solo ingresando a la página de autenticación. Solo tendrá que ingresar su DNI o código universitario, validará su usuario tentativo y confirmará. El sistema enviará la contraseña a su correo electrónico.	3	1
Módulo Administrar			
N°	Descripción	Dific.	Prior.
1	El sistema permitirá el registro de integrante al grupo. Bajo esta funcionalidad el sistema permitirá a los líderes y consejeros del grupo inscribir a sus integrantes al grupo. Una vez registrado los integrantes podrán visualizar los datos del grupo y la información de los demás integrantes para estar en constante comunicación.	3	3
2	El sistema permitirá el mantenimiento de los integrantes de un grupo. Permitirá actualizar sus datos personales, datos académicos y más información importante para el grupo.	3	3
3	El sistema permitirá el mantenimiento de los datos del grupo. El líder del grupo podrá actualizar los datos que pertenecen a su grupo, así mismo podrán subir su foto oficial, foto del lugar de reunión y su canto en audio MP3.	2	2
4	El sistema permitirá dar mantenimiento los cargos en el grupo.	2	2

	El líder (pastor) de iglesia dará mantenimiento a la configuración de los cargos por cada tipo de grupo en un período, cada líder grupo podrá asignar cargos a sus integrantes.		
5	El sistema permitirá registrar eventos del grupo. Cada líder registrará la información de cada evento a realizarse que viene a ser las reuniones del grupo.	3	3
6	El sistema permitirá elaborar el programa a desarrollarse en cada evento (reuniones, etc.). En el programa se añadirá cada actividad a desarrollarse, la duración y los responsables.	3	3
7	El sistema permitirá el envío de mensajes personalizados a cada correo de cada participante en un evento. Con un clic del líder, el sistema armará y enviará los mensaje de participación a cada persona, indicando el lugar del evento, la hora de inicio, duración de su actividad, entre otras más informaciones.	2	2
8	El sistema permitirá el registro de asistencia de cada persona al evento. El líder o encargado de la asistencia registrará la asistencia de los participantes del evento, teniendo en cuenta los estados: puntual, tardanza, falta y falta justificada.	3	3
9	El sistema permitirá el registro de asistencia por medio de código de barras. Previa configuración los asistentes podrán pasar con solo su documento de identidad o código de universidad para registrar su asistencia.	2	2
10	El sistema permitirá subir fotos de cada evento.	1	1
11	El sistema permitirá la creación de sub-grupos. El líder del grupo podrá crear varios sub-grupos en el grupo, tales como para grupo pequeño se crearán parejas misioneras.	2	3
11	El sistema permitirá el mantenimiento de estudiantes de la Biblia. El líder del grupo o el integrante del sub-grupo podrán dar mantenimiento a la lista de estudiantes de la Biblia y el curso bíblico de estudio.	2	2
12	El sistema permitirá el control de estudios bíblicos. Cada integrante de un sub-grupo podrá registrar cada estudio bíblico desarrollado con sus estudiantes de la Biblia.	2	2
13	El sistema permitirá el mantenimiento de indicadores por cada grupo. Cada líder del grupo tendrá la posibilidad de registrar sus metas al alcanzar por cada indicador previamente configurado de acuerdo al tipo de grupo.	2	3
14	El sistema permitirá la entrega de materiales. Cada líder podrá registrar la entrega de materiales, así mismo permitirá controlar el stock de cada material o producto.	3	2
15	El sistema permitirá el mantenimiento de incidencias.	3	2

Cada líder y consejero registrar las incidencias provocadas por los integrantes y más personas. Se registrarán las incidencias positivas y negativas.

Módulo Registros			
N°	Descripción	Dific.	Prior.
1	El sistema permitirá el mantenimiento de personas. Los usuarios administradores podrán registrar y actualizar la información de cada persona.	3	3
2	El sistema permitirá el mantenimiento de grupos. Los usuarios administradores podrán inscribir y reinscribir grupos en un periodo con sus respectivas dependencias.	3	3
3	El sistema permitirá el mantenimiento materiales y productos. El líder de grupo y líder pastor podrán registrar sus productos para luego ser entregados a las personas.	2	2
3	El sistema permitirá el registro de miembros de iglesia. El pastor de iglesia o secretaria de iglesia podrán registrar a los nuevos miembros de iglesia por bautismo o re bautismo y se verá en el termómetro (tabla comparativa) de cada grupo.	3	3

Módulo Configuraciones			
N°	Descripción	Dific.	Prior.
1	El sistema permitirá dar acceso a tipo de eventos por tipo de grupo. El pastor de iglesia o secretaria de iglesia podrán dar acceso a tipo de eventos por tipo de grupo de acuerdo a las actividades que realizan.	2	2
2	El sistema permitirá la configuración de cargos. El pastor de iglesia y administrador del sistema podrán configurar cargos por tipo de grupo organizado, organizándolo por departamento y cargos respectivos.	3	2
3	El sistema permitirá la configuración de programas sugeridos. Para la optimización en el registro de programa en cada evento. El pastor de iglesia o administrador podrá registrar el programa sugerido por cada tipo de grupo señalando la actividad y la duración en minutos.	2	2
4	El sistema permitirá la configuración por tipos de asistentes. Como cada tipo de eventos tiene sus tipos de asistentes, el sistema permitirá la configuración necesaria.	2	1
5	El sistema permitirá la configuración de la evaluación. Posibilitará la configuración de los rubros de evaluación por cada tipo de evento.	2	3

Módulo Reportes			
N°	Descripción	Dific.	Prior.
1	El sistema emitirá el reporte de asistencia. Todos los usuarios miembros podrán acceder al reporte de asistencia de su grupo por cada tipo de evento.	2	2

2	El sistema emitirá el reporte de escuelas sabáticas y grupos pequeños. Todos los usuarios miembros podrán acceder al reporte de grupos pequeños agrupados por escuelas sabáticas y la información de sus líderes y datos del grupo.	2	2
4	El sistema emitirá el reporte de cumpleaños de los integrantes del grupo. Todos los miembros podrán ver reporte de cumpleaños de los integrantes de su grupo, agrupados por mes.	1	1
5	El sistema emitirá el reporte de documentos personales. Todos los usuarios podrán acceder al reporte de su información tales como: ficha personal, ficha resumida, asistencias, participaciones, cargos, grupos, parentesco, asimismo, podrán actualizar sus datos personales.	2	2
6	El sistema emitirá el reporte para el monitoreo del grupo Todos los usuarios miembros podrán acceder al reporte de la información del grupo tales como: datos del grupo, lista de integrantes, cargos, lista de eventos, programa de cada evento, asistencia a los eventos, análisis de asistencia (gráfico) y el álbum de grupo (fotos de cada evento).	3	3
7	El sistema emitirá el reporte del termómetro de escuela sabática. Todos los usuarios miembros podrán acceder al reporte del termómetro de escuela sabática, el cual hace un cálculo de todos los ítems evaluados comparados con las metas (indicadores) establecidos por cada grupo.	2	3
8	El sistema emitirá el reporte de los grupos e integrantes registrados. El pastor y administradores del sistema podrán ver los reportes de los grupos registrados por tipo de grupo y periodo. Podrá ver el total de integrantes y la lista en un formato para la impresión.	2	3
9	El sistema emitirá el reporte de asistencia a los grupos. Se podrán visualizar por fecha de reunión los líderes que registraron asistencia por entidad académica. La lista de personas que asistieron puntual, con tardanzas, faltas y faltan justificadas.	2	3
10	El sistema generará el reporte de estadísticas generales. En este reporte se accederá a la siguiente información: totales de personas inscritas por entidad académica (universidades, facultades, escuelas), totales por tipos de personas, totales por religión y por género.	3	3
11	El sistema generará el reporte del estado por persona. Visualizará por medio de gráficos el estado de cada persona que incluye asistencias, evaluaciones, incidencias y participaciones en cada evento.	2	3
12	El sistema generará el reporte de análisis de asistencia.	2	2

	Se visualizará el un gráfico de análisis de asistencia por periodo, tipo de grupo, entidad académica y tipo de evento.		
13	El sistema emitirá reportes para el departamento del Ministerio Personal (MiPES). Los reportes para el ministerio personal serán los siguientes: lista de sub-grupos (parejas misioneras), el avance de estudios bíblicos, asistencia a la escuela misionera y el termómetro de la escuela sabática por cada grupo.	2	3
14	El sistema emitirá el reporte para el monitoreo general. Se podrá monitorear a los grupos registrados basados en su información de los datos del grupo, lista de integrantes, eventos, programas, asistencias, álbum del grupo y análisis de asistencia del grupo.	2	3
15	El sistema emitirá los reportes para gestión de los grupos. Se podrán acceder a los siguientes reportes de gestión: líderes con más eventos registrados, personas con más participaciones, personas con más asistencias, personas con más faltas y faltas justificadas, lista de instructores bíblicos, búsqueda de dones y talentos, personas con mejor evaluación, personas baja evaluación, evaluación por entidad académica, % de asistencia por entidad académica.	3	3
16	El sistema emitirá el reporte de entregas de materiales. Se podrá visualizar las entregas por personas y el resumen de entregas.	1	1
17	El sistema emitirá el reporte de personas no registradas por tipo de grupo.	1	2
Módulo Generalidades			
N°	Descripción	Dific.	Prior.
1	El sistema incorporará un chat del grupo. Este mecanismo permitirá los anuncios y la comunicación del líder con sus integrantes.	2	1
2	El sistema incorporará un centro de publicaciones de comentarios e imágenes.	3	1
3	El sistema posibilitará en envió de mensajes por cumpleaños a correos electrónicos de las personas.	2	1
4	El sistema mostrará una agenda de las actividades a nivel iglesia.	2	1
5	El sistema permitirá reproducir videos de la iglesia y música en audio. Mostrará enlaces de descargas de materiales para los grupos.	2	1
6	El sistema mostrará la información personal por cada persona. Posibilitará la actualización de su información personal. Posibilitará el registro de parentesco. Lista de cargos y funciones a cumplir.	3	2

Fuente: elaboración propia

Tabla 14

Nivel de dificultad

Dific: Dificultad	
Valor	Descripción
1	Baja
2	Media
3	Alta

Tabla 15

Nivel de prioridad.

Prior: Prioridad	
Valor	Descripción
1	Baja
2	Media
3	Alta

4.3.1.2 *Requerimientos no funcionales*

Los requerimientos no funcionales están agrupados en arquitectura y tecnologías. La *Tabla 16* presenta la lista de los requerimientos no funcionales obtenidos.

Tabla 16

Requerimientos no funcionales del sistema

N°	Descripción	Dific.	Prior.
1	Los usuarios pueden ingresar al sistema desde cualquier dispositivo electrónico, solo tiene que tener instalado un navegador web.	1	2
2	El sistema será desarrollado bajo una interfaz gráfica basada en controles web, tendrá el framework Bosdtrad para la adaptación de las páginas a los tamaños de pantalla.	1	2
3	El sistema estará disponible vía internet y red interna de la universidad.	1	3
4	El sistema se ejecutará bajo un servidor web de aplicaciones instalado en un sistema operativo Windows Server 2008 en adelante.	1	2
5	El sistema web trabajará con el gestor de base de datos PostgreSQL	2	2
6	Para el envío de mensajes a correo usará el protocolo SMTP.	1	1

Fuente: elaboración propia

4.3.1.3 Consideraciones del sistema

El sistema cuenta con las siguientes restricciones:

- Validaciones: la información ingresada a cada formulario es verificada por JavaScript y HTML5.
- Seguridad: para acceder al sistema necesitan una cuenta de usuario y contraseña, y el control de acceso es manejado por perfiles según grupos organizados.
- Escalabilidad: la arquitectura del sistema posibilitará el incremento de nuevas funcionalidades y módulos de manera flexible sin causar problemas complejos para los desarrolladores.
- Usabilidad: para que el usuario interactúe mejor con el software tendrá una interfaz gráfica ligera e intuitiva. Además, una correcta emisión de aviso de errores y advertencias que permita una buena comunicación entre usuario y sistema.
- Performance: garantiza un tiempo de inicio de sesión abierta sin uso de treinta (30) minutos, pasado este tiempo el sistema cerrará sesión automáticamente, lo que indica que el usuario iniciará sesión nuevamente.

Tras el proceso de entrevistas con los usuarios directos y el análisis de requerimientos se identificaron los siguientes roles de usuarios que se describen a continuación:

- Líder de grupo: realiza el registro de integrantes, actualiza datos del grupo, registra cargos del grupo, registrar eventos, registra el programa y asistencia, organiza los sub-grupos (parejas misioneras, etc.), registra los estudiantes de la Biblia, define indicadores y entrega

materiales. Por el lado, de reportes accede a: asistencia, monitoreo del grupo y termómetro de escuela sabática.

- Consejero de grupo: realiza el monitoreo del grupo, registra integrantes del grupo, asigna cargos, verifica que los eventos y la información se registre adecuadamente por parte del líder.
- Integrante (miembro) del grupo: posee todos accesos genéricos (reportes básicos), tales como: asistencia personal, participaciones, lista de grupos, edición de datos personales, registrar parentesco, monitorear grupo, cumpleaños de integrantes, lista de integrantes del grupo, publicaciones, ficha personal, registro de estudios bíblicos, entre otros.
- Administrador del sistema: realiza las funciones de administración de módulos, perfiles, cuentas de usuario y accesos. Posee, además, los accesos de configuración.
- Pastor de iglesia: gestiona los cargos, miembros, eventos de la iglesia. Tiene todos los reportes de la organización de los grupos, visualiza las evaluaciones a sus integrantes de los grupos, monitorea a cada grupo organizado.
- Capellán de facultad: gestiona los cargos, miembros, eventos de la iglesia. Tiene todos los reportes de la organización de los grupos, visualiza las evaluaciones a sus integrantes de los grupos, monitorea a cada grupo organizado.
- Administrador universidad: tiene los reportes asociados a la gestión de los grupos organizados tales como: lista de grupo, lista de integrantes, monitoreo de grupos, reporte de evaluaciones por facultad, entre otros.
- Director de MiPES: posee los reportes correspondientes al departamento de escuela sabática de iglesia tales como: reporte de parejas misioneras, instructores bíblicos, avance

de estudios bíblicos, asistencia a la escuela misionera, termómetro de la escuela sabática por cada grupo.

En la *Figura 12* se muestran los roles de usuario del sistema por medio del diagrama de casos de uso.

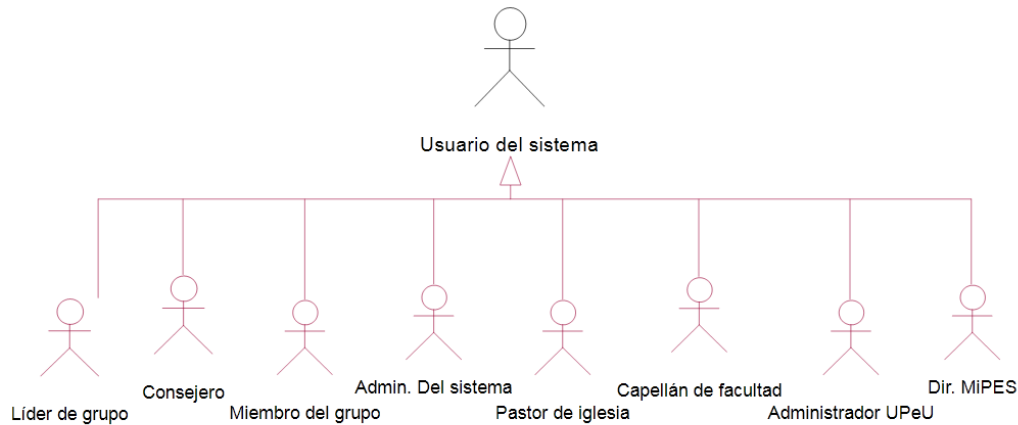


Figura 12. Roles de usuario del sistema

4.3.1.4 *Análisis de la solución*

La solución ha sido analizada de acuerdo a las necesidades de los clientes, considerando la viabilidad técnica y económica, además, el análisis costo - beneficio del proyecto, a esto se incrementó las restricciones en costo, tiempo y la definición del producto.

Las principales necesidades establecidas por los líderes de los grupos, consejeros, administrativos de la institución y capellanes de facultad en la organización de los grupos, se obtuvieron los siguientes alcances:

- Disponibilidad de la información de la inscripción de los miembros en los grupos organizados.
- Disponibilidad de la información en cuando a la organización de los grupos para las reuniones semanales.
- Evaluación de los integrantes de los grupos, definición de metas y evaluación de resultados.
- Disponibilidad de la información para el monitoreo basado en la asistencia, tardanzas, faltas y faltas justificadas a los eventos realizados.
- Acceso a la información de la programación de cada evento.
- Registro de incidencias de cada miembro del grupo.
- Mecanismo para el control de las actividades misioneras de cada pareja organizada en un grupo.
- Información disponible que ayude a la toma de decisiones a nivel de grupo, iglesia y distrito misionero para la aplicación de estrategias de mejora.

Las necesidades indicadas quedan satisfechas por los requerimientos del sistema dado la similitud de las expectativas que tienen los usuarios con la implementación del nuevo sistema de información.

Con la lista de requerimientos y habiendo analizado las necesidades de los usuarios es viable la construcción el diagrama de casos de uso y de actores.

4.3.1.4.1. Casos de uso del negocio

El diagrama de casos de uso viene a ser actividades que deberán realizarse para llevarse a cabo un proceso. Es una técnica para especificar el comportamiento de un sistema desde el punto de vista del usuario y describen la funcionalidad del negocio/sistema independientemente de la implementación. En el *Apéndice B* de este documento, se pueden visualizar los casos de uso que muestran los procesos para la gestión de los grupos.

4.3.1.4.2. Viabilidad técnica y económica

En la viabilidad técnica y económica se presentan las restricciones de hardware y de software para la construcción del sistema, así como su estado de disponibilidad. Esto incluye el software de ofimática para las labores documentarias. Las restricciones técnicas identificadas son las siguientes:

- a) Disponibilidad de un equipo de cómputo como servidor para alojar la base de datos.
- b) Disponibilidad de un equipo de cómputo para alojar el servidor web de aplicaciones.
- c) Disponibilidad de un equipo de cómputo para las fases de concepción, elaboración, construcción, transición, pruebas y validación de resultados.

- d) Herramienta IDE para la construcción de la interfaz gráfica y codificación de las funcionalidades bajo el lenguaje de programación Java.
- e) Herramienta Open Source para gestión gráfica, desarrollo y administración de PostgreSQL.
- f) Librerías JavaScript y Ajax para la optimización de las búsquedas.
- g) Librerías para la conexión del sistema con las bases de datos y envío de mensajes a correo electrónico.

Con una adecuada planificación de recursos y con vista a la maximización de las capacidades logísticas existentes, se adoptaron las siguientes medidas:

- Los requerimientos (a) y (b) quedaron cubiertos con el uso de una computadora con procesador Intel i5 y memoria RAM de 6GB, dadas las exigencias del servidor de base de datos. El requisito (c) quedó cubierto con el uso de la computadora del tesista, con procesador Intel® Core(TM)2 Duo y con 4GB de memoria RAM.
- El requisito (d) quedó cubierto con el uso de la herramienta IDE NetBeans 7.1.2 y el requisito (e) con la herramienta PgAdmin para la gestión de la base de datos PostgreSQL.
- El requisito (f) quedó cubierto con el uso de las librerías JQuery-1.2.1.pack.js, JQuery.js para uso de Ajax.

- El requisito (g) quedó cubierto con el uso de la librería PostgreSQL JDBC Driver para la conexión de base de datos y la librería Java-Mail.
- La elección del lenguaje de Java y del servidor web Apache Tomcat.

En cuanto a la viabilidad económica, tomando como base los ítems técnicos citados para la implementación, se establecieron los siguientes considerandos como parte del costo del proyecto:

- Los requisitos (a) y (b) a nivel de hardware fueron cumplidos gracias al departamento de la Dirección General de Tecnologías de Información de la UPeU Filial Tarapoto que posee estos recursos.
- En cuanto al requisito (d) quedó cubierto con el uso de la herramienta IDE NetBeans 7.1.2 que no requiere pago alguno y está disponible en su página web.
- En cuanto al requisito (e) referente al sistema administrador de base de datos, se trabajó con el manejador PostgreSQL 9.0, cuyo uso no requiere del pago por alguna licencia.

A continuación, se presenta el costo asumido por el concepto de personal según roles y funciones que desempeñaron durante la realización del proyecto.

Tabla 17

Costo de recursos humanos del proyecto

Rol	Abreviatura	Cantidad	Costo/Hora (S/.)
Jefe del proyecto	JP	1	16.00
Analista	AN	1	12.00
Arquitecto	AR	1	12.00
Desarrollador	DE	2	10.00
Tester o probador	TE	1	8.00

Fuente: elaboración propia

Tabla 18

Costo referencial del proyecto

Fase	Responsable	Horas estimadas.	Costo (S/.)	Total (S/.)
Concepción	DE	15	16.00	240.00
	AN	20	12.00	240.00
Elaboración	DE	18	12.00	216.00
	AR	30	12.00	360.00
	AN	30	12.00	360.00
Construcción	DE	500	10.00	5,000.00
	TE	70	8.00	560.00
Transición	DE	30	10.00	300.00
	AN	50	12.00	600.00
TOTAL				7, 876.00

Fuente: elaboración propia

4.3.1.4.3. *Análisis costo – beneficio*

El análisis permite presentar las razones y criterios como justificación para la implementación de este proyecto y la inversión económica, así como el grado de contribución en los procesos de gestión y evaluación de los grupos organizados tras su implementación.

En la *Tabla 19* se muestra los costos aproximados por cada proceso para la gestión y evaluación de los grupos organizados. Se considera un sueldo normal de S/. 1,200.00, con 6 días de trabajo semanal y 8 horas diarias. El trabajador recibe S/. 48 de por día y S/. 6 por hora. El costo es analizado en un determinado periodo académico.

Tabla 19

Análisis de costo aproximado

#	Detalle	Situación real			Con sistema		
		Cant.	Costo	Total	Cant.	Costo	Total
1	Proceso de organización de los grupos (incluye analizar a cada nuevo líder de grupo, obtención de los	53 grupos (1/2 h)	S/. 6	S/. 159	53 grupos (10 min)	S/. 6	S/. 53

	datos de cada líder, e información básica del integrante).						
2	Proceso de inscripción de los integrantes por parte de los líderes (redactar todos los datos de los datos de los integrantes y emitir el informe a sus capellanes)	53 grupos (4 h)	S/. 6	S/. 1272	53 grupos (1 h)	S/. 6	S/. 318
3	Proceso de registro de los ideales del grupo (informar a sus capellanes y pastor de iglesia)	53 grupos (1 h)	S/. 6	S/. 318	53 grupos (15 min)	S/. 6	S/. 79.5
4	Registro de reunión del grupo y elaboración del programa (incluye avisar a los participantes)	53 grupos (1.5 h por 20 reuniones)	S/. 6	S/. 9540	53 grupos (30 min por 20 reuniones)	S/. 6	S/. 3180
5	Registro de asistencia e informe a sus capellanes	53 grupos (20 min por 20 reuniones)	S/. 6	S/. 2120	53 grupos (5 min por 20 reuniones)	S/. 6	S/. 530
6	Elaboración del análisis de asistencias de los integrantes a los grupos organizados semanalmente.	53 grupos (6 min por 20 semanas)	S/. 6	S/. 636	53 grupos (1 min por 20 semanas)	S/. 6	S/. 106
7	Emisión de reportes por escuela, religión, de asistencias, participaciones, incidencias y muchas otras.	53 grupos (3 h)	S/. 6	S/. 954	53 grupos (10 min)	S/. 6	S/. 53
8	Gastos por impresión de registros, tablas comparativas, papeles	53 grupos	S/. 15	S/. 795	53 grupos	S/ 2	S/ 106
TOTALES				S/. 15,794			S/. 4,320

Fuente: elaboración propia

Con el uso del sistema ha reducido los gastos en un 73%. Hay muchos beneficios como la accesibilidad a la información, la motivación hacia los integrantes para asistir a su grupo por el control de asistencia y la organización, y elaboración de los programas por cada evento, los participantes se encuentran informados constantemente y al mismo tiempo son evaluados.

Una vez detallados el costo en recursos humanos y gastos a efectuarse en el proyecto, se deduce poca inversión en software gracias al uso de herramientas de código abierto como licencias gratuitas y el aprovechamiento de hardware de la institución (UPeU FT). Los costos están orientados más, al personal del proyecto (el tesista el único ejecutor haciendo el papel de todos los roles OpenUP).

Es preciso resumir las ventajas y beneficios que ofrece esta solución. Siendo el propósito la gestión y evaluación de grupos organizados. Permitió a los líderes tener la información de cada grupo constantemente actualizada y organizada sistemáticamente. Provee el mecanismo de control de asistencia en cada grupo, así como una mejor organización de cada evento programado. Permite medir el compromiso de cada persona con su grupo basado en sus asistencias, participaciones y evaluaciones semanales. El grupo es analizado de acuerdo a indicadores establecidos por los mismos integrantes y líderes promoviendo un mayor compromiso. Permite el registro de actividades misioneras por los estudios bíblicos. Los principales beneficios a nivel organizacional están en la calidad de reportes al que acceden, permitiendo tomar decisiones basados en ellos.

En este sentido, tras el análisis efectuado, la plataforma presenta más beneficios en lo futuro porque cada vez irá mejorando el control y la gestión de manera automatizada.

4.3.1.4.4. *Definición del sistema*

El sistema se ha definido a partir del diagrama de clases de análisis involucrando los procesos principales en el escenario del negocio. Dicho análisis permitió establecer y definir la arquitectura del sistema. La solución cubre los requerimientos a través de siete paquetes o módulos representados en el diagrama de paquetes como se muestra en la *Figura 13*.

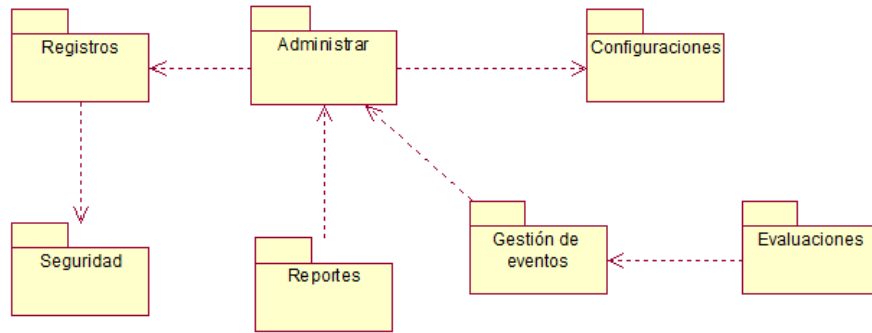


Figura 13. Diagrama de paquetes del sistema

4.3.1.4.4.1. Paquete Seguridad

Este paquete reúne las funcionalidades para el rol de administrador de sistema. Permite el mantenimiento de perfiles o roles del sistema, la configuración de módulos y opciones, la asignación de módulos para cada rol, la creación de usuarios y el reporte de los usuarios en cada grupo organizado. Es preciso mencionar que según el negocio modelado un rol es asignado a los usuarios por cada grupo organizado. Este paquete también permite la modificación de contraseñas de usuarios.

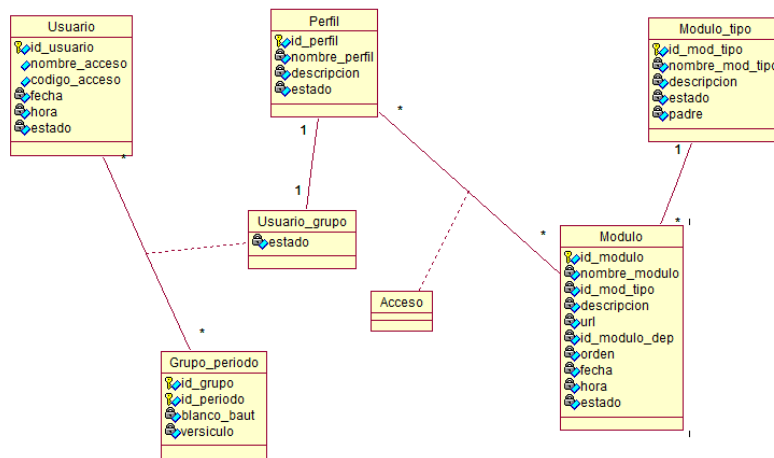


Figura 14. Diagrama de clase de análisis – Módulo Seguridad

4.3.1.4.4.2. Paquete Registros

Este paquete posee las funcionalidades de gestión de la información. Básicamente la información relacionada con el registro de los integrantes de cada grupo (Clase Persona), los datos cada grupo organizado, el registro de miembros de iglesia (bautismos) y toda la información relacionada como es mostrada en la *Figura 15*.

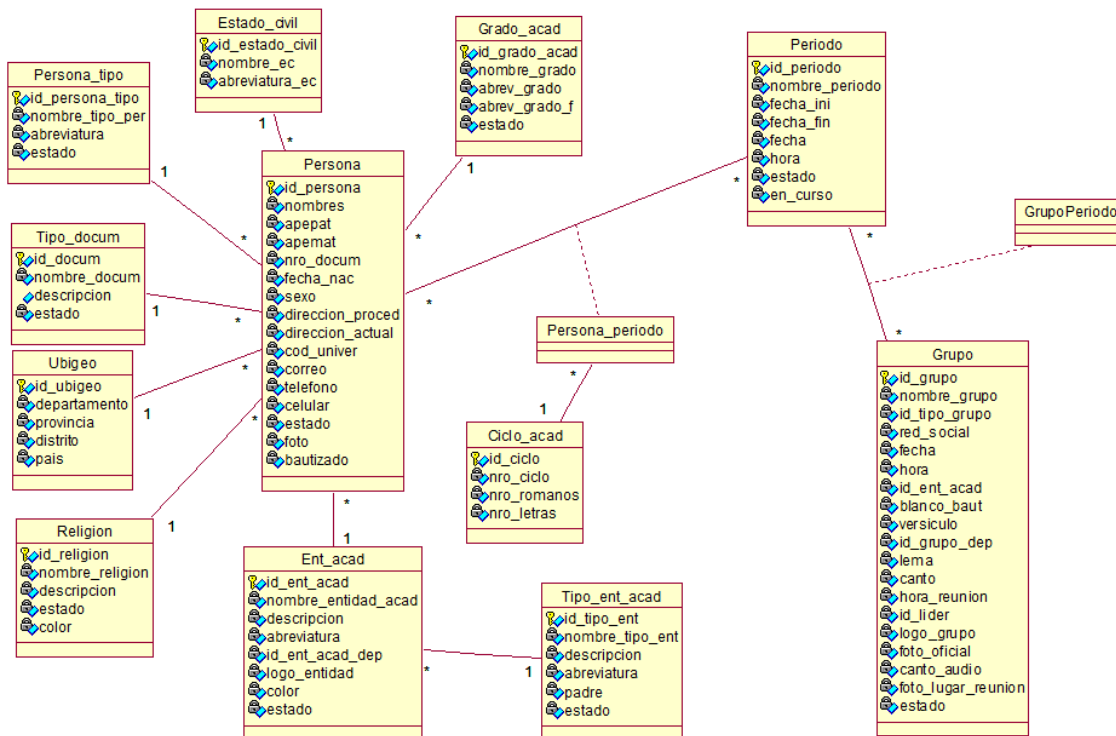


Figura 15. Diagrama de clases de análisis – Módulo Registros

4.3.1.4.4.3. Paquete administrar, gestión de eventos y evaluaciones

El paquete administrar posee las características de inscripción de integrantes al grupo, actualización de los datos de los integrantes por cada ciclo académico, la actualización de los datos del grupo, el registro de eventos (reunión del grupo, escuela sabática, culto Divino, culto joven,

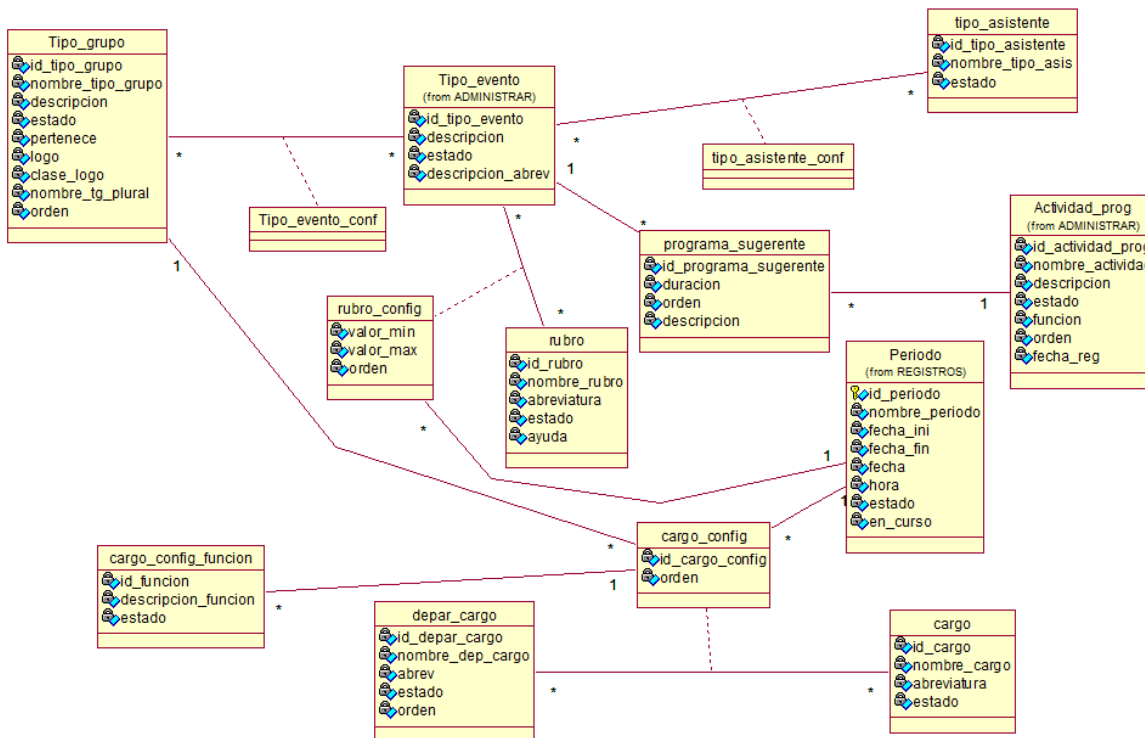


Figura 17. Diagrama de clases de análisis – Módulo configuraciones

4.3.1.4.4.5. Paquete reportes

Este paquete cumple con la emisión de informes de asistencia, programas, cargos, evaluaciones, análisis, gráficos y todos los reportes que servirán en la gestión de los grupos organizados.

Tras la fase de la concepción se planteó la solución al área de Bienestar Universitario e Iglesia Universitaria, por medio de la explicación de los beneficios que se obtendría con la implementación de este proyecto. El área administrativa de la institución decidió adquirir esta solución, haciendo un contrato de locación de servicios de software.

También se pidieron los permisos oficialmente al área de Bienestar Universitario y la Dirección General de Tecnologías de Información (DIGETI), que es la encargada de la gestión de las tecnologías de información en la institución.

4.4 Elaboración

4.4.1 Introducción

El desarrollo de esta fase permitió establecer una arquitectura consistente permitiendo regir el desarrollo del sistema de información en su totalidad. Tal arquitectura debe permitir realizar mejoras y al mismo tiempo la integración de nuevos módulos de funcionamiento. El catálogo de los requerimientos, los puntos críticos que aún quedaron pendientes, y se hace necesario representar la arquitectura en las vistas de los diferentes modelos del sistema, los cuales llegaron a ser la base para las fases de construcción y transición.

El objetivo principal de esta fase fue transformar los requerimientos y requisitos significativos en una arquitectura que describa su estructura e identifique los componentes demostrando así que el proyecto es viable.

4.4.2 Arquitectura de la solución

La arquitectura de la solución se presenta en alto nivel y los modelos arquitectónicos evaluados para posteriormente presentar la arquitectura final.

4.4.2.1 *Representación de la arquitectura*

La arquitectura está orientada a entornos web, con el diseño Cliente – Servidor. El servidor ejecuta las tareas evitando el trabajo a máquinas clientes de forma independiente. Los clientes acceden al servidor por medio de sus navegadores web.

Este diseño asegura la disponibilidad de la información automatizada a tiempo completo y desde un equipo estático o móvil desde cualquier lugar con conexión a internet o red local. Por ello, es importante que el diseño permita y garantice un óptimo aprovechando de las capacidades que tiene el sistema de información satisfaciendo los requerimientos funcionales y no funcionales del producto software. Para el cumplimiento de estas exigencias, la arquitectura posee las siguientes características:

- Está basada en el paradigma de programación orientado a objetos (POO). Permite la creación de clases y objetos relacionados entre sí por invocación de métodos desde las vistas, además garantiza una buena manipulación de datos y operaciones de manera encapsulada. También garantiza la gestión de cambios en menor tiempo y la reutilización de métodos.

4.4.2.2 *Arquitectura orientada hacia la presentación web*

Se hace del patrón de diseño MVC (Modelo – Vista - Controlador) que contiene tres componentes descritos a continuación:

- Modelo: gestiona la comunicación entre la capa de datos y el dominio de la aplicación.
- Vista: permite la visualización de la información, es el componente que interactúa con el usuario por medio de formularios, tablas y acciones a realizar.
- Controlador: interpreta los pedidos del usuario desde el componente Vista, trabajando de la mano con el Modelo.

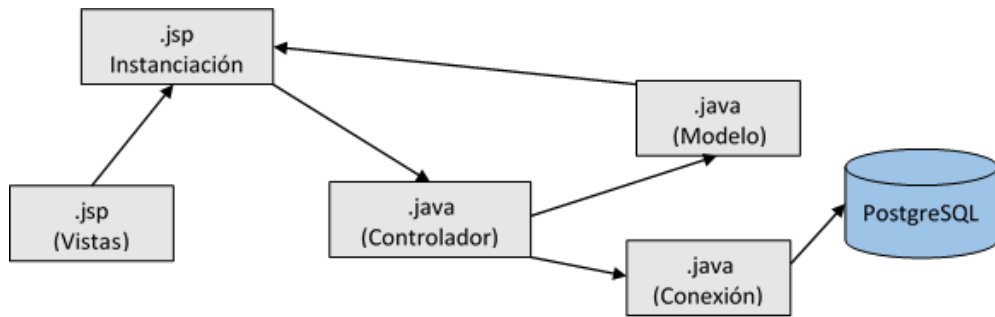


Figura 18. Patrón de diseño Modelo – Vista – Controlador

4.4.2.3 Vista de despliegue

En la *Figura 19* se describe mediante nodos la comunicación entre ellos, está compuesto por el nodo servidor de base de datos es responsable de enviar y recibir los datos pedidos, el nodo de servidor web y aplicaciones es el responsable de procesar el código fuente y trabajar de acuerdo a los pedidos del cliente mediante las funcionalidades del sistema.

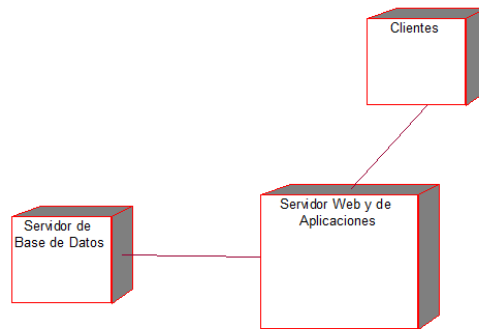


Figura 19. Diagrama despliegue

4.4.2.4 Diagrama de base de datos

El diagrama de base datos contiene todas las tablas de base datos relacionadas entre sí, permitiendo guardar la información de manera organizada. El sistema de información envía y recibe los datos hacia la base de datos respectiva. Las figuras: *Figura 20* y *Figura 21* muestran el diseño de base de datos del sistema.

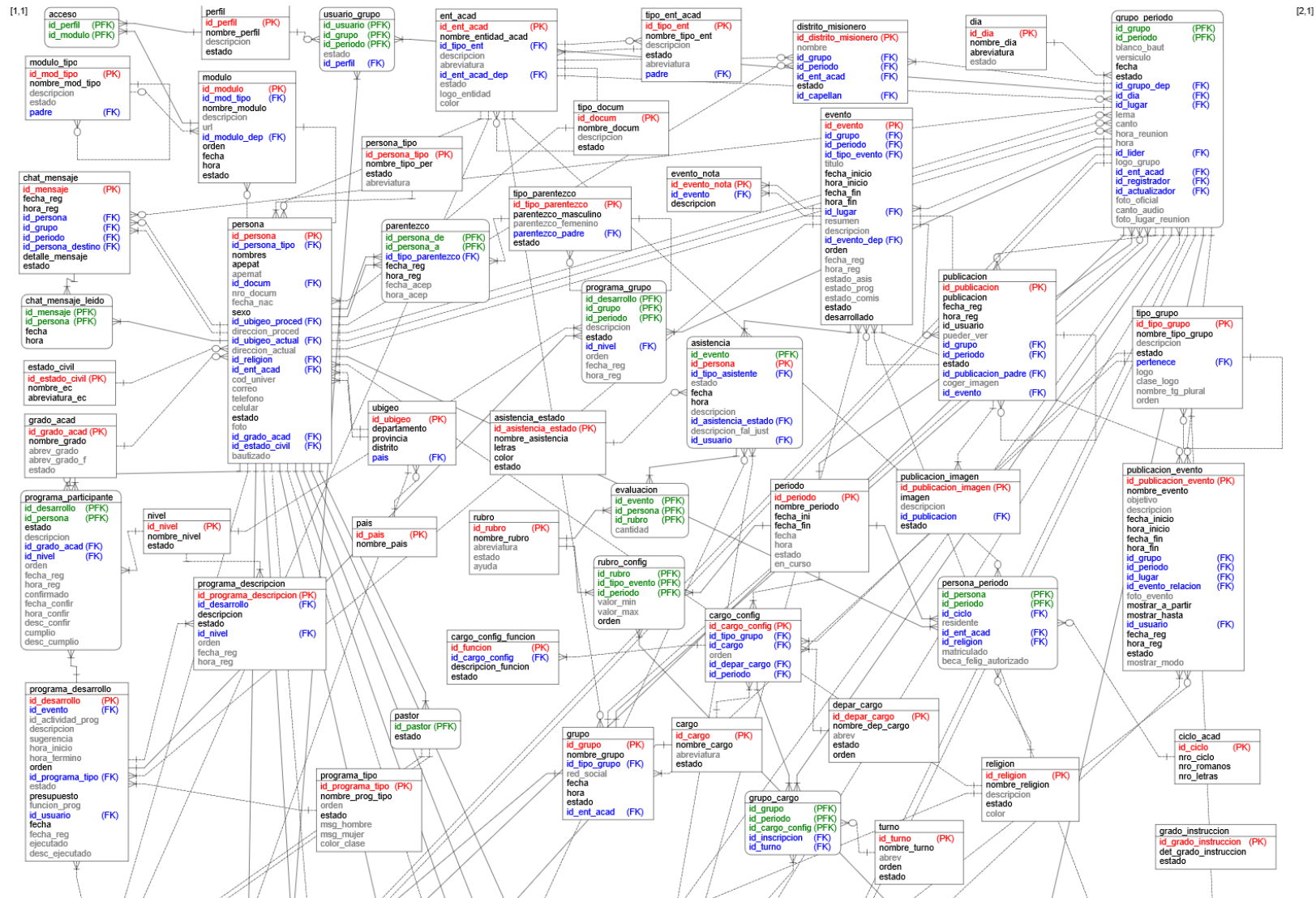


Figura 20. Diagrama de Base de Datos - Parte 1. Fuente: elaboración propia

4.5 Construcción

4.5.1 Introducción

Esta es una de las fases más importantes porque viene a ser el desarrollo de las fases anteriores. El éxito de los proyectos depende necesariamente de esta fase, porque es la construcción lo que genera el resultado de todo el esfuerzo. Al cliente final lo que le importa no es con qué arquitectura ha sido construido, sino que cumplan con sus expectativas, que solucione la problemática existente, optimizando sus procesos.

¿Qué es lo que podía salvar el proyecto si toda la planificación no se hizo adecuadamente? Sin lugar a dudar es la experiencia de los desarrolladores.

4.5.2 Lenguaje de programación

Java es un lenguaje orientado a objetos; robusto y flexible. El investigador tiene mucha afinidad con este lenguaje. El otro motivo de escoger este lenguaje de programación, es que los sistemas que posee la universidad en su mayoría están bajo este lenguaje, es por ello que no generaría problemas para futuros mantenimientos por el mismo departamento de la institución.

4.5.3 IDE Netbeans

Netbeans es un entorno de desarrollo integrado (IDE), tal vez uno de los más populares en la construcción de sistemas, integra una variedad de lenguajes de programación, y está siendo constantemente actualizado. Posee la opción de añadir nuevos lenguajes de programación a su paquete de librerías. Posee Licencia General Pública (GNU), tal vez una de las desventajas es que requiere que la computadora o laptop tenga un buen nivel de procesamiento y memoria, pero esto

en la actualidad ya no es problema porque contamos con equipos que tienen gran capacidad de procesamiento. Además, este IDE es muy interactivo e interesante porque ayuda de manera eficiente en el proceso de desarrollo.

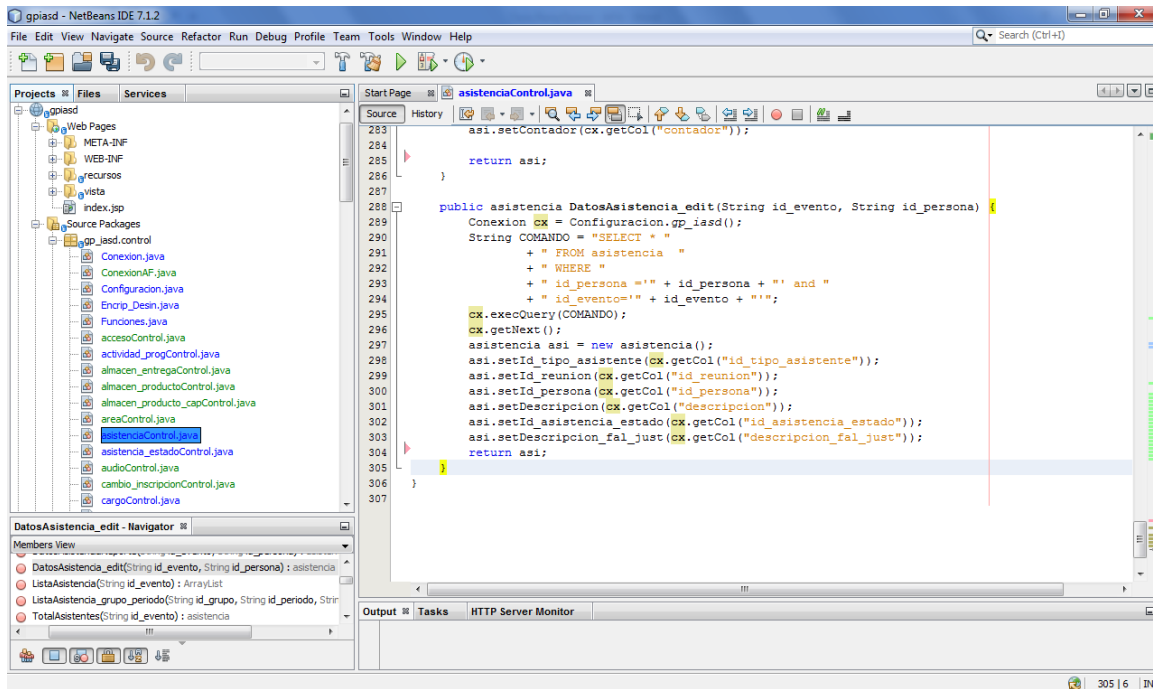


Figura 22. Vista del IDE Netbeans

4.5.4 Bootstrap

Bootstrap es un framework a nivel de interfaz de usuario permite la adaptación de las vista del sistema a diferentes tamaños de pantalla, permitiendo acceder al sistema que va desde un equipo completo de mesa hasta un dispositivo móvil. Bootstrap está constituido por librerías JavaScript y archivos CSS. Existen muchos framework Responsive Desing gratuitos y pagados, es una de las tecnologías actuales.



Figura 23. Página de inicio, vista en pantalla grande y pequeña con la tecnología Bootstrap

4.5.5 Librerías Java-Mail

Las librerías Java-Mail sirven como conexión para correos electrónicos, permite la autenticación y el envío de mensajes a cuentas de varios servidores de correo tales como Gmail, Hotmail, Yahoo. Se hace uso de estas librerías en el sistema en las funcionalidades tales como: creación de usuarios, recuperación de contraseñas, mensajes por cumpleaños, anuncios a los líderes relacionados con el control de asistencia, mensajes de anuncios a los participante en cada programación elaborado.

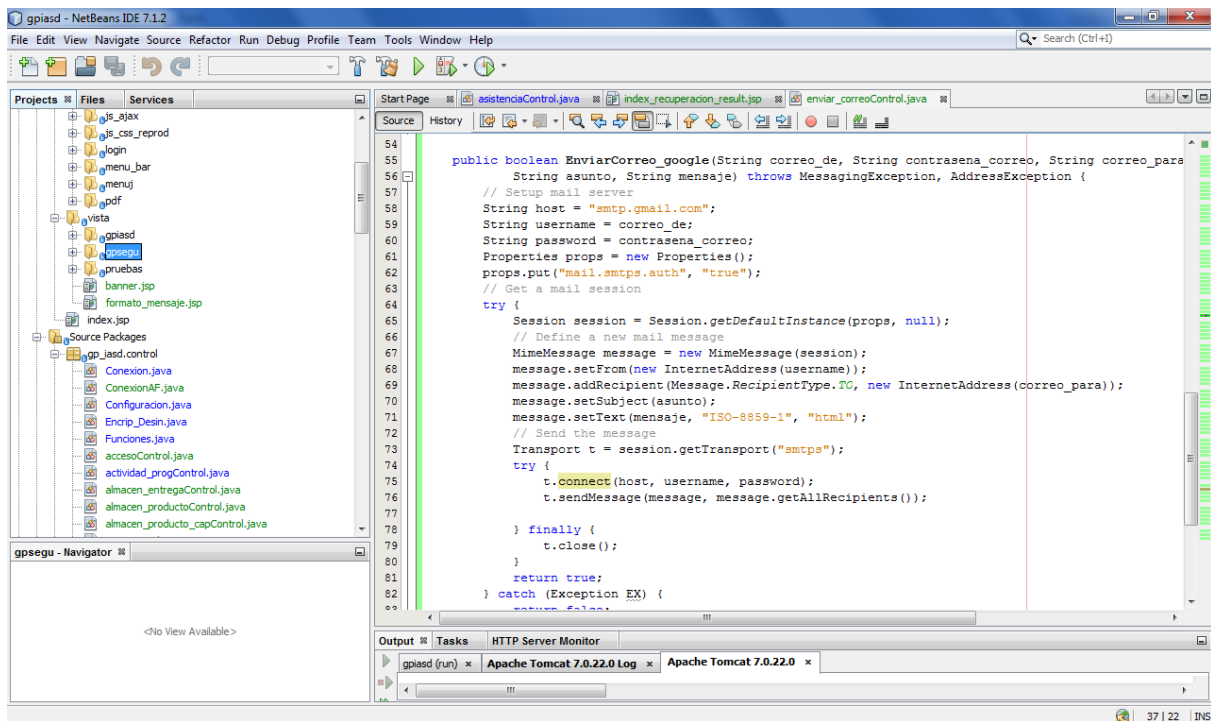


Figura 24. Método para envío de correos, haciendo uso de Java-mail

4.5.6 Gestor de base de PostgreSQL

PostgreSQL es un gestor de base de datos con licencia gratuita. Es potente y robusto en el procesamiento de datos, además, se instala sin ningún problema. Motor que permite una buena administración de su base de datos a nivel de usuario con pgAdmin. Se escogió este motor de base de datos porque la iglesia no cuenta con una base de datos propietaria, además, no se debe desperdiciar las principales ventajas que posee PostgreSQL interactuando con diferentes lenguajes de programación.

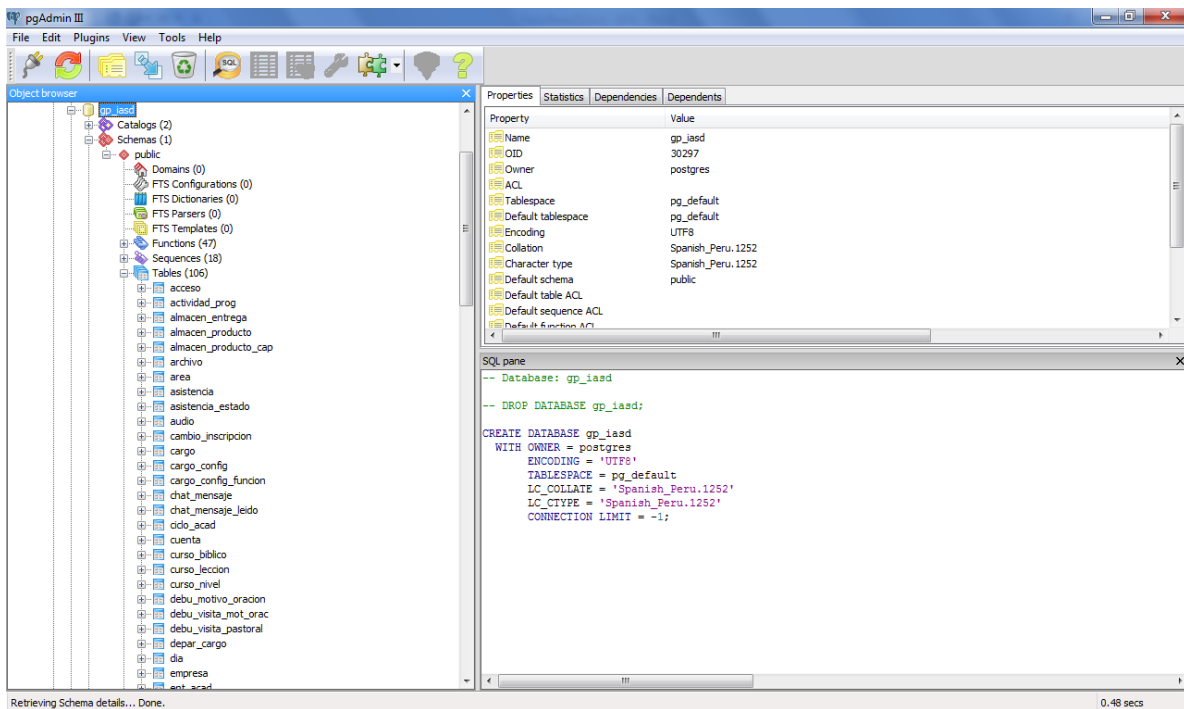


Figura 25. Panel de administración de base de datos PostgreSQL con pgAdmin

4.5.7 Conexión a base de datos

Para la conexión a la base de datos PostgreSQL se usó la librería JDBC versión 3, esta permite la conexión respectiva, y hay para diferentes motores de base de datos. Al principio en las capacitaciones a los usuarios finales se tuvo problemas con el acceso a la base de datos por el exceso de conexiones, para ello se implementó el patrón de diseño Singleton, que a continuación se detalla.

4.5.7.1 Singleton

Singleton es un patrón de diseño que permite crear una instancia estática de una clase, en donde los clientes acceden recurrentemente a un solo punto de acceso sin modificar su código. En este proyecto hace uso de este patrón en la conexión de base de datos para lograr las escalabilidad de la aplicación, en donde, los clientes al conectarse a la base de datos en primer lugar verifican si

hay una conexión abierta, si lo hay, usan la conexión existente, caso contrario crean una nueva instancia. Los resultados que ha generado este patrón es que el sistema es más rápido al presentar las peticiones del usuario, por lo que para cada pedido no se abre y cierra la conexión, sino que usa la misma conexión para todos. Lo podríamos comparar a este patrón como un puente que une la aplicación y el motor de base de datos.

Tabla 20

Implementación del patrón de diseño Singleton

```
package gp_iasd.control;

import java.sql.*;

public class Conexion {

    private static Connection cn = null;
    private static Statement st = null;
    private static ResultSet rs = null;
    private static String MError = new String();
    private static CallableStatement cst = null;
    private static Conexion INSTANCE = null;

    public Conexion() {
        mConexion("Postgresql", "localhost", "5432", "postgres", "gruposorganizados", "contrasena");
    }

    /**
     * Crea una instancia de la base de datos en caso de no estar creada.
     *
     */
    private synchronized static void createInstance() {
        if (INSTANCE == null) {
            INSTANCE = new Conexion();
        }
    }

    public static Conexion getInstance() {
        if (INSTANCE == null) {
            System.out.println("Se creó una nueva conexion");
            createInstance();
        }
        else{
            System.out.println("Se usó la conexión existente");
        }
        return INSTANCE;
    }
}
```

```

public static void deleteInstance() {
    INSTANCE = null;
    try {
        rs.close();
        st.close();
        cn.close();
        System.err.println("Conexión cerrada");
    } catch (Exception e) {
        MError = e.getMessage();
    }
}
//

private void mConexion(String gestorBD, String host, String puerto, String usuario, String nombreBD, String
clave) {

    try {
        if (gestorBD.equals("Mysql"))//compara el tipo de de bd
        {
            Class.forName("com.mysql.jdbc.Driver");
            cn = DriverManager.getConnection("jdbc:mysql://" + host + ":" + puerto + "/" + nombreBD + "",
usuario, clave);
            System.out.println("Coneccion Exitosa a MySQL");
        } else if (gestorBD.equals("Oracle")) {
            Class.forName("oracle.jdbc.driver.OracleDriver");//permite la conexion
            cn = DriverManager.getConnection("jdbc:oracle:thin:@" + host + ":" + puerto + ":" + usuario + "",
nombreBD, clave);
            System.out.println("Coneccion Exitosa a Oracle ");
        } else if (gestorBD.equals("Postgresql")) {
            Class.forName("org.postgresql.Driver");
            cn = DriverManager.getConnection("jdbc:postgresql://" + host + "/" + nombreBD + "/" + usuario,
clave);
            MError = "Exito conectando a postgresql";
        }
        cn.setAutoCommit(false);
        st = cn.createStatement();
        MError = "";
    } catch (Exception e) {
        MError = e.getMessage();
    }

}

public int execCommand(String com)//conexion por cada objeto de la base de datos
{
    int result = 0;
    try {
        result = st.executeUpdate(com);//actualiza inserta y elimina los datos
        MError = "";
    } catch (Exception e) {
        MError = e.getMessage();
        System.out.println("Error al insertar/Actualizar" + com);
    }

    return result;
}

```

```

public int execC(String com) throws SQLException//conexion por cada objeto de la base de datos
{
    int rss = st.executeUpdate(com);//actualiza inserta y elimina los datos
    return rss;
}

public void execQuery(String com) {
    try {
        rs = st.executeQuery(com);//ejecuta la consulta
        System.out.println(com);
        MError = "";
    } catch (Exception e) {
        Conexion.deleteInstance();
        MError = e.getMessage();
        System.out.println("No se puede hacer select " + MError);
    }
}

public boolean getNext() {
    boolean valor = false;
    try {
        if (rs.next()) {
            valor = true;
        } else {
            valor = false;
        }
        MError = "";
    } catch (Exception e) {
        MError = e.getMessage();
    }
    return valor;
}

public String getCol(String ncol) {
    String valor = new String();
    try {
        valor = rs.getString(ncol);
        if (valor == null) {
            valor = "";
        }
        MError = "";
    } catch (Exception e) {
        MError = e.getMessage();
    }
    return valor;
}

public void Commit();//guarda
{
    try {
        cn.commit();
    } catch (Exception e) {
        System.out.println(e.getMessage());
        RollBack();
    }
}

```

```

    }
}

public void RollBack()//desase la memoria temporal
{
    try {
        cn.rollback();
    } catch (Exception e) {
        System.out.println(e.getMessage());
    }
}

public String getValorOne(String com) {
    String valor = new String();
    try {
        System.out.println(com);
        rs = st.executeQuery(com);
        rs.next();
        valor = rs.getString(1);
        if (valor == null) {
            valor = "";
        }
        MError = "";
    } catch (Exception e) {
        MError = e.getMessage();
    }
    return valor;
}

public String getMError() {
    return MError;
}

///PARA EJECUTAR PROCEDIMIENTOS

public void procPreparar(String procNombre) {
    // int numParam = paramNom.length;
    try {
        cst = cn.prepareCall("{call " + procNombre + " }");

    } catch (Exception e) {
        MError = e.getMessage();
    }
}

public void procAsignaParam(int nomParam, String valParam) {
    try {
        cst.setString(nomParam, valParam);
    } catch (Exception e) {
        MError = e.getMessage();
    }
}

public void procAsignaParam(String nomParam, int valParam) {
    try {
        cst.setInt(nomParam, valParam);
    } catch (Exception e) {

```

```
MError = e.getMessage();
    }
}
}
```

Fuente: elaboración propia

4.5.8 Servidor web

Como servidor de pruebas y de alojamiento se tuvieron a Apache Tomcat versión 7, tal servidor está diseñado para compilar código Java.

4.5.9 Proceso de desarrollo de la Norma Técnica Peruana ISO/IEC 12007

El proceso de desarrollo de la NPT ISO/IEC 12207 define las actividades del desarrollar, lo cual han sido cumplidos en este proyecto tales actividades:

- Diseño de la arquitectura del software: se evaluó la arquitectura del software y de los diseños de las interfaces y de la base de datos teniendo en cuenta los criterios como la trazabilidad hacia los requerimientos, viabilidad del diseño, viabilidad de la operación y mantenimiento. Además se hizo las revisiones y validaciones conjuntas con los expertos en el negocio.
- Codificación y pruebas del software: se desarrolló y se probó cada componente del software de tal forma que satisfaga los requerimientos del negocio.

A continuación se muestran las actividades cumplidas según la NTP ISO/IEC 12207, no todas fueron llevadas a cabo por temas de tiempo y recursos, pero que es muy importante, ya que se aplican muy bien a proyectos de un equipo de varios desarrolladores. En la siguiente *Tabla 21* se listas las tareas realizadas por actividad.

Tabla 21

Actividades y tareas desarrolladas según NTP ISO/IEC 12207

N°	Etapas	Tarea
1	Implementación del proceso	Seleccionar, adaptar y usar aquellas normas, métodos, herramientas y lenguajes de programación (si están estipuladas en el contrato) que estén documentados.
2	Implementación del proceso.	Se debe preparar planes para realizar las actividades del desarrollo. Incluye normas específicas, métodos, herramientas, acciones y responsabilidades asociadas con el desarrollo y la calificación de todos de todos los requerimientos.
3	Análisis de los requerimientos del sistema	Analizar el uso específico previsto del sistema a ser desarrollado para especificar los requerimientos del sistema. La especificación de requerimientos deberá describir funciones y capacidades del sistema, requerimientos de negocio, organizativos y de usuario, requerimientos de seguridad física y de acceso, interfaces y requerimientos de operación y de mantenimiento, limitaciones de diseño y requerimientos de calificación.
4	Análisis de los requerimientos del sistema	Se deben evaluar los requerimientos teniendo en cuenta: <ul style="list-style-type: none"> a) Trazabilidad hacia las necesidades de la calificación b) Consistencia con las necesidades de la adquisición c) Capacidad para ser probados d) Viabilidad del diseño de la arquitectura del sistema e) Viabilidad de la operación y mantenimiento
5	Diseño de la arquitectura del sistema	Evaluar la arquitectura del sistema y los requerimientos para los elementos teniendo en cuenta los criterios: <ul style="list-style-type: none"> a) Trazabilidad hacia los requerimientos del sistema b) Consistencia con los requerimientos del sistema c) Adecuación de las normas y métodos de diseño usados. d) Viabilidad de los elementos software para cumplir con los requerimientos asignados e) Viabilidad de la operación y mantenimiento.
6	Análisis de los requerimientos software	Establecer: <ul style="list-style-type: none"> a) Especificaciones funcionales y de capacidad b) Definición de datos y requerimientos de la Base de datos. c) Requerimientos de instalación y aceptación del producto software. d) Documentación de usuario.
7	Diseño de la arquitectura del software.	Desarrollar y documentar un diseño a alto nivel para la base de datos.
8	Codificación y pruebas del software	Desarrollar cada unidad de software y de base de datos.

9	Codificación y pruebas del software	Probar cada unidad de software y de base de datos que satisfagan los requerimientos
	Codificación y pruebas del software	Evaluar el código software y los resultados de las pruebas teniendo en cuenta los requerimientos y el diseño de software.
10	Integración de software	Prepara un plan de integración para las unidades software y los componentes de software y probarlos a medida que se agrupan de acuerdo al plan de integración.
11	Integración de software	Realizar revisiones conjuntas
12	Instalación del software	Instalar el producto software de acuerdo al plan de instalación. Se deberá asegurar que el código software y las base de datos se inicialicen, ejecuten y terminan.
13	Instalación del software	Proporcionar apoyo en las revisiones y pruebas de aceptación llevadas a cabo por el adquirente del producto.
14	Apoyo a la aceptación del software	Completar y entregar el producto software tal como lo especifica el contrato.
15	Apoyo a la aceptación del software	Proporcionar formación inicial y continua y dar apoyo al adquirente tal como lo especifica el contrato.

Fuente: NTP/IEC 12207 y elaborado según tareas ejecutadas en el proyecto

4.5.1 Iteraciones de la Construcción

El sistema fue construido mediante iteraciones, que llega a ser el resultado de cada versión funcional. Para esto se construyeron modelos, controladores y vistas en el lenguaje de programación seleccionado, a continuación se presentan las iteraciones y los casos de uso desarrollados.

4.5.1.1 Iteración 1

La primera fase se enfocó en la implementación del proceso de autenticación y el registro de los grupos organizados, que viene a ser la base del panel de administración de cada grupo. En la *Tabla 22* se presenta la lista de casos de uso utilizados en esta primera fase.

Tabla 22

Iteración 1 - Autenticación y registro de Grupos

ID	Nombre de Caso de Uso	ID	Nombre de Caso de Uso
----	-----------------------	----	-----------------------

1	Iniciar Sesión	12	Modificar Lugar de reunión
2	Cerrar Sesión	13	Asignar lugar de reunión al grupo
3	Cambiar Contraseña	14	Asignar Grupo padre
4	Crear Usuario Personal	18	Crear Distrito misionero
5	Restaurar contraseña	19	Modificar Distrito misionero
6	Crear Grupo	20	Asignar Distrito misionero
7	Modificar Grupo	21	Ver lista de grupos
8	Eliminar Grupo	22	Crear Periodo
9	Crear Tipo de grupo	23	Modificar Periodo
10	Modificar Tipo de grupo	24	Eliminar Periodo
11	Crear Lugar de reunión	40	Ver Tipo de Grupo

4.5.1.2 Iteración 2

La segunda fase abarca el registro de personas y la inscripción de la persona al grupo organizado. Todo esto permite tener la data de cada integrante del grupo y la organización de cada grupo. La lista de los casos de uso implementados se muestra en la *Tabla 23*.

Tabla 23

Iteración 2 - Registro de personas e inscripción de integrantes al grupo

ID	Nombre de Caso de Uso	ID	Nombre de Caso de Uso
25	Crear Persona	16	Modificar Líder
26	Modificar Persona	17	Asignar líder
27	Buscar Ubigeo	51	Crear Integrante
28	Buscar Entidad Académica	52	Buscar Integrante por Nombres, código o Documento de Identidad
29	Buscar Ciclo	53	Añadir Integrante al Grupo
30	Buscar Periodo	54	Modificar Integrante
31	Buscar Religión	56	Eliminar integrante
32	Buscar Grado académico	57	Ver fotos de Integrantes
33	Buscar Tipo de Persona	58	Ver lista de Integrantes
15	Crear Líder		

4.5.1.3 Iteración 3

La tercera fase abarca la configuración de cargos, la actualización de datos del grupo, el registro de cargos en el grupo, la configuración de programas sugeridos y el registro de bautismos.

Tabla 24

Iteración 3 - Configuración de cargos, datos del grupo, entre otros

ID	Nombre de Caso de Uso	ID	Nombre de Caso de Uso
38	Crear Tipo de evento	47	Crear Tipo de asistente
39	Modificar Tipo de evento	48	Eliminar Tipo de asistente
40	Ver Tipo de Grupo	49	Buscar Tipo de Asistente
41	Configurar Cargos por Tipo de grupo	60	Buscar Responsable de Cargo
42	Ordenar cargos	61	Eliminar Responsable de Cargo
43	Crear Actividad	62	Ver Cargo
44	Modificar Actividad	35	Crear Bautismo
45	Eliminar Actividad	36	Modificar Bautismo
46	Asignar Actividad a Tipo de evento	37	Buscar Pastor oficiante

4.5.1.4 Iteración 4

La iteración 4 abarca la creación de eventos, elaboración del programa, control de asistencia y la evaluación a los asistentes a los eventos.

Tabla 25

Iteración 4 - Registros de eventos, elaboración del programa, asistencia y evaluación

ID	Nombre de Caso de Uso	ID	Nombre de Caso de Uso
63	Ver Evento	80	Ver resumen de Evento
64	Buscar Evento por Periodo y Mes	81	Crear Asistencia
65	Crear Evento	82	Modificar Asistencia
66	Buscar Tipo de evento por Tipo de Grupo	83	Eliminar Asistencia
67	Ver programa de evento	84	Buscar Asistente
68	Crear Actividad en Programa desarrollo	85	Buscar Tipo de Asistente
69	Buscar Actividad por nombre	86	Buscar Asistente por Nombres, código y Número de documento
70	Buscar Tipo de Programa	87	Ver Asistencia
71	Buscar Actividades por Tipo de evento	100	Subir Foto a Evento
72	Modificar Actividad	111	Eliminar Foto de Evento
73	Eliminar Actividad	112	Crear Publicación
74	Buscar responsable en Persona, en Integrantes y Grupo	113	Eliminar Publicación
75	Crear Nota en Programa desarrollo		
76	Modificar Nota en Programa desarrollo	114	Modificar Publicación
77	Eliminar Nota en Programa Desarrollo	115	Ver Publicación

78	Imprimir Programa desarrollo	116	Crear mensaje grupal
79	Crear Mensaje de participación (Correo)		

4.5.1.5 Iteración 5

Abarca la creación de sub grupos, registro de estudiantes, control de estudios bíblicos, registro de metas por indicadores y creación de parentescos.

Tabla 26

Iteración 5 - Sub grupos, estudiantes de la Biblia, estudios bíblicos e indicadores

ID	Nombre de Caso de Uso	ID	Nombre de Caso de Uso
88	Crear Sub grupo	103	Modificar Meta por Indicador del grupo
89	Modificar Sub Grupo	104	Eliminar Meta por Indicador del grupo
90	Eliminar Sub grupo	105	Crear Parentesco
91	Ver Sub grupo	106	Modificar Parentesco
92	Asignar Integrante a Sub grupo	107	Eliminar Parentesco
93	Eliminar Integrante de Sub grupo	108	Ver Parentesco
94	Buscar Estudiante de la Biblia	109	Buscar Tipo de relación por género
95	Buscar Curso Bíblico	117	Crear Incidencia
96	Modificar Estudiante de la Biblia	118	Modificar Incidencia
97	Eliminar Estudiante de la Biblia	119	Eliminar Incidencia
98	Crear Estudio Bíblico	120	Ver Incidencia
99	Buscar Lugar para Estudio Bíblico	121	Asignar Evidencia a Incidencia
100	Eliminar Estudio Bíblico de Estudiante de la Biblia	122	Eliminar Evidencia de Incidencia
101	Modificar Estudio Bíblico de estudiante de la Biblia	124	Ver Evidencias en Incidencia
102	Crear Meta por Indicador del grupo		

4.5.1.6 Iteración 6

Abarca el módulo de seguridad de la aplicación conformado por el registro de usuarios, perfiles, módulos y permisos a grupos.

Tabla 27

Iteración 6 - Usuarios, perfiles, módulos y permisos

ID	Nombre de Caso de Uso	ID	Nombre de Caso de Uso
124	Crear Perfil del sistema	132	Asignar Permiso a Perfil
125	Modificar Perfil	133	Eliminar Permiso de Perfil
126	Eliminar Perfil	134	Ver Permiso
127	Ver Perfil	135	Crear Usuario
128	Crear Módulo	136	Modificar Usuario
129	Modificar Módulo	137	Eliminar Usuario
130	Eliminar Módulo	138	Ver usuario
131	Ver Módulo		

4.5.1.7 Iteración 7

Esta iteración permite la elaboración de reportes para gestión de los grupos organización, ayudando a la toma de decisiones del grupo y de la iglesia. En la siguiente tabla se presenta la lista de reportes.

Tabla 28

Iteración 7 - Lista de reportes del sistema

N°	Nombre del reporte
1	Grupos registrados
2	Cumpleaños
4	Gráfico por entidad académica
5	Gráfico por religión
6	Grupos e integrantes
7	Asistencia a los grupos.
8	Estadísticas generales por: entidad académica, tipo de persona, por religión, por género.
9	Ficha por persona
10	Análisis de asistencias
11	Para MiPES: parejas misioneras, estudios bíblicos, Asistencia a Escuela Misionera, Termómetro de Escuela Sabática.
12	Monitoreo general
13	Monitoreo mi grupo
14	Reportes de gestión: Líderes con más eventos registrados, Personas con más participaciones, Personas con más asistencias, Personas con más faltas, Personas con más faltas justificadas, Instructores bíblicos, Buscar dones y talentos, Personas con mejor evaluación, Personas con baja evaluación, Evaluación por entidad académica, % de Asistencia por entidad académica, Lista de personas por número de asistencias.

4.6 Transición

Por cada iteración se hizo entrega una versión del sistema funcional. Una vez implementado se hizo entrega de la versión final del sistema, esto incluye el código completo, el código compilado, la base de datos. La entrega del producto final permite asegurar que el proyecto cumple con las expectativas de los usuarios finales. Los tres objetivos de esta fase es refinar la funcionalidad, performance, y calidad global del producto beta completado en la fase anterior:

- Prueba beta para validar que se cumplen las expectativas del usuario.
- Lograr constancia por parte de los stakeholders para asegurar que el despliegue esté completo. Hay varios niveles de pruebas para la aceptación del producto.
- Mejorar el rendimiento de proyecto futuro a través de lo aprendido.

Se logró obtener aprobación de haber implementado satisfactoriamente el proyecto, en los anexos: *Anexo 3*, *Anexo 4* y *Anexo 5* se muestra los reconocimientos emitidos por la iglesia y los capellanes de facultad.

CAPÍTULO V

RESULTADOS Y DISCUSIÓN

1.1. Introducción

El propósito de este capítulo es mostrar los resultados de la investigación realizada, basándose en los criterios de los objetivos generales. Se presentan los resultados en gráficos.

5.1 Resultados

La solución tecnológica ha sido desarrollada de acuerdo a los criterios de planificación del proyecto, se han dado solución a los cambios o actualizaciones solicitadas a través de las reuniones sostenidas con el personal involucrado, se capacitó con respecto al uso del sistema en los laboratorios de sistemas de la institución. A continuación, presentamos los resultados.

5.1.1 Inicio de sesión

Al ingresar al sistema se visualiza un formulario en el cual el usuario pueda ingresar su nombre de usuario y contraseña, una vez validado puede realizar las operaciones de acuerdo al rol que posee. Al mismo tiempo la ventana de inicio de sesión permite consultar a que grupo pertenece un integrante, solo basta ingresar su código universitario o número de documento de identidad, también permite crear un usuario automáticamente, la condición es que la persona debe estar registrado en el sistema, al crear su usuario el sistema envía automáticamente su contraseña a su correo electrónico y también permite restaurar su contraseña en caso de haberse olvidado. Como información adicional le presenta los eventos planificados como iglesia, los tipos de grupos registrados y el versículo de memoria de la lección de la escuela de la semana en curso (Ver Figura 26).

Universidad Peruana Unión Ingresar ¿Estoy inscrito? Crear usuario Radios en vivo Publicaciones

Bienvenido al Sistema de Grupos Organizados





Sistema de Grupos Organizados

[¿Olvidaste tu contraseña?](#)

Agenda de eventos

 **Semana de Oración Joven**
 Lunes, 09 de Noviembre - 07:30 pm

"Jóvenes y señoritas, ¿no podéis formar grupos y, como soldados de Cristo, alistaros en la labor, poniendo todo vuestro tacto, y capacidad y talento al servicio del Maestro, para que podáis salvar almas de la ruina? Organicéense grupos en todas las iglesias para hacer esta obra... ¿Se organizarán los jóvenes y señoritas que realmente aman a Jesús como obreros, no solo para trabajar en favor de los que profesan ser observadores del sábado, sino también de aquellos que no son de nuestra fe?"
Servicio Cristiano, p. 44.

Grupos organizados

- > Universidades (1)
- > Iglesias (1)
- > Escuelas Sabáticas (18)
- > Grupos Pequeños (53)
- > Escuelas de Líderes (4)

6 METAS

Estudio de la Biblia (LES)

Estudiar la Biblia con la ayuda de la guía de estudio de la Escuela Sabática es la estrategia establecida por Dios, a través

Versículo de memoria

> **LA CRISIS CONTINÚA**

> "Mas alábese en esto el que se hubiere de alabar: en entenderme y conocerme, que yo soy Jehová, que hago misericordia, juicio y justicia en la tierra; porque estas cosas quiero, dice Jehová" (Jer. 9:24).

Figura 26. Página inicio de sesión

5.1.2 Menú principal

Una vez ingresado al sistema, se presenta en la parte superior un menú de opciones organizados por sus módulos correspondientes, en la parte central se presenta los datos del grupos tales como el nombre, nombre de los líderes y los ideales del grupo, en el lado izquierdo se pueden ver otras opciones como el chat del grupo, solicitudes, lista de cumpleaños del día (Ver Figura 27).

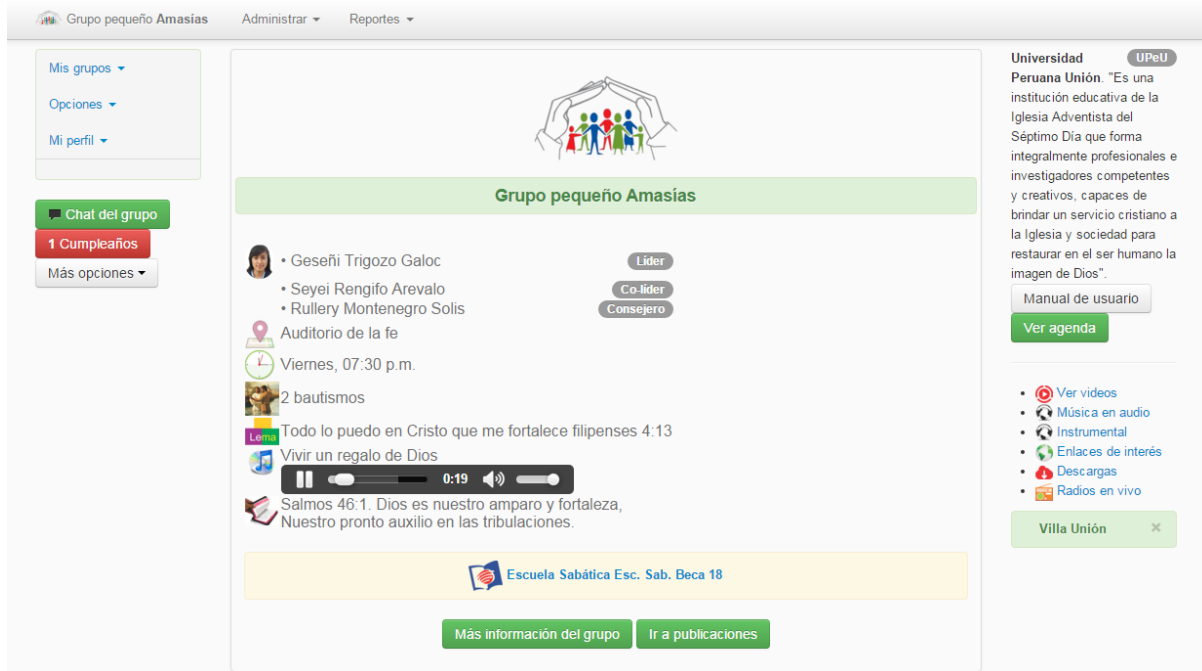


Figura 27. Menú principal al ingresar al sistema (rol: líder de grupo)

5.1.3 Registro de integrantes

La opción registro de integrantes se encuentra en el módulo Administrar. Permite visualizar la lista de integrantes del grupo organizado con sus datos, se hace la inscripción de nuevos integrantes, la actualización de sus datos y la eliminación respectivamente. Esta es una de las funcionalidades muy importantes ya que gracias ello se puede conseguir organizar bien al grupo y poder gestionarlo.

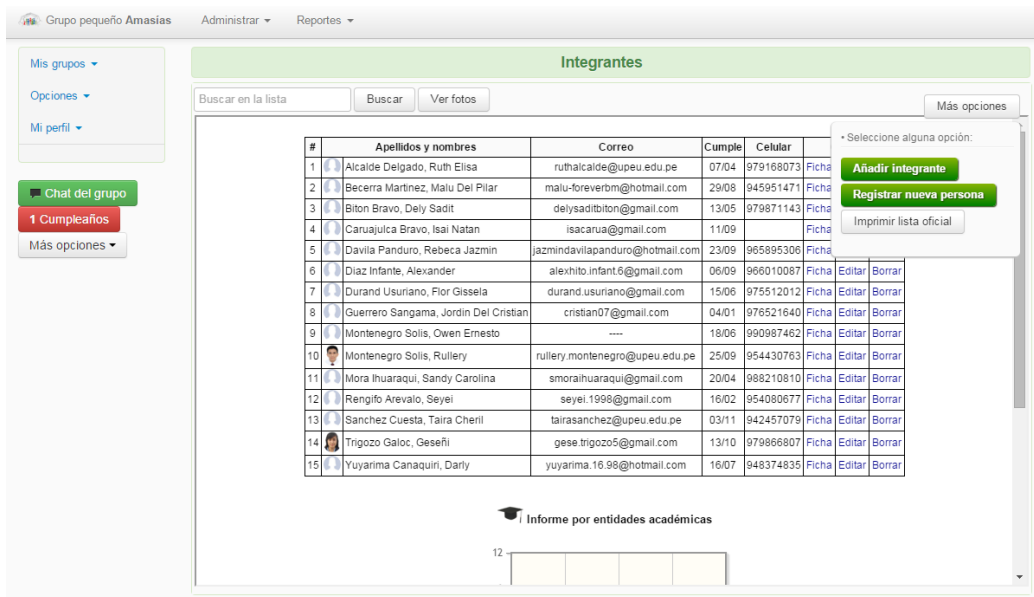


Figura 28. Gestión de integrantes del grupo

5.1.3.1 Información del integrante

Permite ver la información básica del integrante, la carrera que estudia el ciclo en curso, sus datos de contacto, la religión entre otra información importante para el grupo.

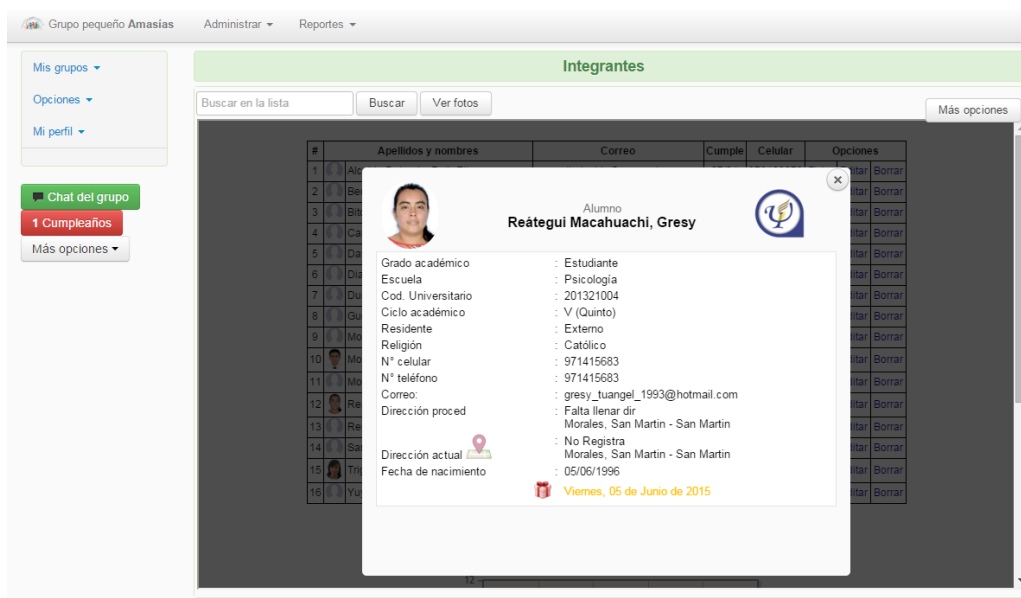


Figura 29. Visualización de los datos del integrante

5.1.3.2 Actualización de datos

Permite actualizar los datos de cada integrante de grupo. Esta opción también lo posee cada usuario, es decir, puede actualizar su propia información, sin depender necesariamente de su líder.

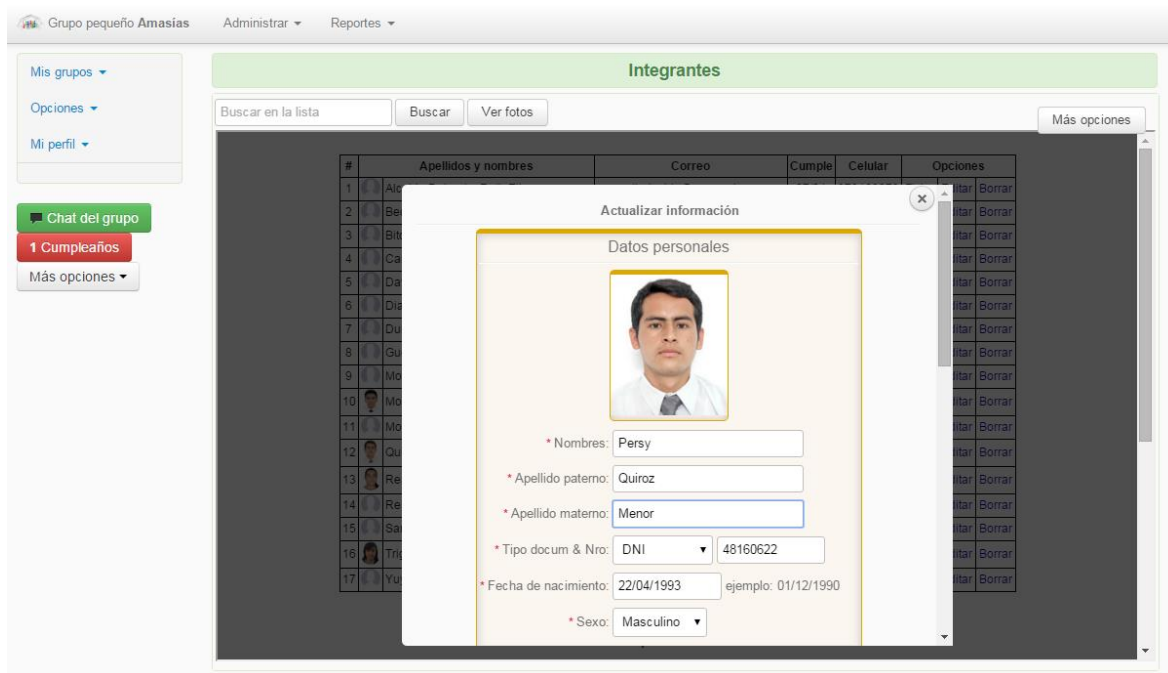


Figura 30. Actualización de datos del integrante

5.1.3.3 Inscripción de integrantes

Se hace la búsqueda de personas registradas en el sistema para inscribirlo al grupo, la búsqueda de la persona es por nombres, apellido paterno, apellido materno, código universitario o número de documento de identidad.

Grupo pequeño Amasias Administrar Reportes

Mis grupos Opciones Mi perfil

Chat del grupo 1 Cumpleaños Más opciones

Integrantes

Búsqueda de personas

Volver a lista Nombres rob

Búsqueda con: rob

#	Apellidos y nombres	Código universitario	Tipo de persona	Añadir
1	González Torrez, Esleiter Robinson	201220691	Alumno	Añadir
2	Saldaña Dávila, Martin Justo Roberto		Personal general	Añadir
3	Chuquihuanga Cruz, Robert	200710464	Alumno	Añadir
4	Rengifo Cárdenas, Robert Randolf	201321914	Alumno	Ya está en un grupo
5	Cueva Vargas, Roberto Carlos	201210608	Alumno	Añadir
6	Moncada Lopez, Roberto Carlos	9710018	Docente	Ya está en un grupo
7	Navarro Portocarrero, Roby	201222175	Alumno	Ya está en un grupo

Figura 31. Búsqueda e inscripción de personas al grupo

5.1.3.4 Reporte e impresión de lista de integrantes

Para uso diversos se hace la impresión de la lista de integrantes, donde se puede visualizar la información de cada uno tales como: nombres y apellidos completos, código universitario, entidad académica en la que estudia o pertenece, ciclo académico (si es alumno), dirección, fecha de nacimiento, número de celular y correo electrónico (Ver Figura 32).

#	Apellidos y nombres	Cod. Univer	Ent. Acad.	Ciclo	Dirección	Fecha Nac.	Celular	Correo
1	Alcalde Delgado, Ruth Elisa	201522257	Ing. Amb.	II	Jr. Los Martires # 214 Santa Lucia	07/04/1996	979168073	ruthalcalde@upeu.edu.pe
2	Becerra Martínez, Malu Del Pilar	201522228	Ing. Amb.	I	Jr. Los M?rtires N? 254	29/08/1998	945951471	malu-foreverbm@hotmail.com
3	Biton Bravo, Dely Sadit	201522258	Ing. Amb.	I	Jr. Los Martires # 214 Santa Lucia	13/05/1998	979871143	delysaditbiton@gmail.com
4	Caruajulca Bravo, Isai Natan	201522492	Ing. Sist.	I	A. H. San Juan Mz. P Lt 216	11/09/1998		isacarua@gmail.com
5	Davila Panduro, Rebeca Jazmin	201520559	Cont. Trib.	I	Jr. Los Martires # 254 Santa Lucia	23/09/1998	965895306	jazmindavilapanduro@hotmail.com
6	Diaz Infante, Alexander	201510588	Ing. Amb.	I	UPeU - FT	06/09/1997	966010087	alexhito.infant6@gmail.com
7	Durand Usuriano, Flor Gissela	201522253	Ing. Amb.	I	Libre Por Ver / BIENESTAR	15/06/1998	975512012	durand.usuriano@gmail.com
8	Guerrero Sangama, Jordin Del Cristian	201522255	Ing. Amb.	I		04/01/1998	976521640	cristian07@gmail.com
9	Montenegro Solis, Owen Ernesto	201520992	Cont. Trib.	I	av Peru 353	18/06/1998	990987462	----
10	Montenegro Solis, Rullery	201121514	Cont. Finan.	IX	No Registra	25/09/1993	954430763	rullery.montenegro@upeu.edu.pe
11	Mora Ihuaquai, Sandy Carolina	201321117	Ing. Amb.	I	UPeU - FT	20/04/1995	988210810	smoraihuaquai@gmail.com
12	Quiroz Menor, Persy	201011895	Ing. Sist.		Jr. Integración 200	22/04/1993	941946943	persy.quiroz@upeu.edu.pe
13	Reátegui Macahuachi, Gresy	201321004	Psic.	V	No Registra	05/06/1996	971415683	gresy_tuangel_1993@hotmail.com
14	Rengifo Arevalo, Seyei	201522222	Ing. Amb.	I	upeu internado	16/02/1998	954080677	seyei.1998@gmail.com
15	Sanchez Cuesta, Taira Cheril	201522251	Ing. Amb.	I	Jr. Santa Ines #226	03/11/1998	942457079	tairasanchez@upeu.edu.pe
16	Trigozo Galoc, Geseñi	201322643	Ing. Sist.	V	Jr. Los Mártires Resd: Davila	13/10/1994	979866807	gese.trigozo5@gmail.com
17	Yuyarima Canaquiri, Darly	201522214	Ing. Amb.	I	Jr. Los M?rtires N? 254	16/07/1998	948374835	yuyarima.16.98@hotmail.com

Lema: *Todo lo puedo en Cristo que me fortalece Filipenses 4:13*- Canto: *Vivir un regalo de Dios- Blanco: 2*
Versículo: *Salmos 46:1. Dios es nuestro amparo y fortaleza, Nuestro pronto auxilio en las tribulaciones.*

Figura 32. Lista para impresión de integrantes del grupo

5.1.3.5 Ficha del integrante

La ficha del integrante es un reporte de las asistencias por cada integrante, lista de participaciones y las incidencias ya sean positivas o negativas.

Grupo pequeño Amasias Administrar Reportes

Mis grupos Opciones Mi perfil

Chat del grupo 1 Cumpleaños Más opciones

Integrantes

Volver a lista de integrantes

Alumno: **Geseñi Trigozo Galoc**
Código universitario: 201322643
Escuela: Ingeniería de Sistemas - Ciclo V (Quinto)
Residencia: Externo
Religión: Adventista

FICHA PERSONAL

ASISTENCIA

1.	Reunión de grupo pequeño	Integrante del grupo	28/08/2015	P
2.	Reunión de grupo pequeño	Integrante del grupo	04/09/2015	P
3.	Clase de Escuela Sabática	Integrante del grupo	05/09/2015	P
4.	Reunión de grupo pequeño	Integrante del grupo	11/09/2015	P
5.	Clase de Escuela Sabática	Integrante del grupo	12/09/2015	P
6.	Reunión de grupo pequeño	Integrante del grupo	18/09/2015	P
7.	Clase de Escuela Sabática	Integrante del grupo	19/09/2015	P
8.	Reunión de grupo pequeño	Integrante del grupo	25/09/2015	P
9.	Clase de Escuela Sabática	Integrante del grupo	26/09/2015	P
10.	Reunión de grupo pequeño	Integrante del grupo	02/10/2015	P
11.	Clase de Escuela Sabática	Integrante del grupo	03/10/2015	P
12.	Reunión de grupo pequeño	Integrante del grupo	09/10/2015	P
13.	Clase de Escuela Sabática	Integrante del grupo	10/10/2015	P

PARTICIPACIONES

#	Actividad	Fecha	Hora	Evento	Grupo
1.	Espacio de cantos	04/09/2015	07:30 pm	Reunión de grupo pequeño	Grupo pequeño Adel

Figura 33. Ficha del integrante (reporte de asistencias, participaciones e incidencias)

5.1.4 Administrar datos del grupo

Los datos del grupo está en el módulo administrar, permite actualizar los datos del grupo tales como lugar de reunión, día, hora, blanco de bautismos, lema, título del canto tema, versículo de memoria. Además, permite cargar la foto oficial del grupo, la foto del lugar de reunión y el canto tema en formato MP3 (Ver Figura 34).



Figura 34. Opción datos del grupo, módulo administrar

5.1.5 Cargos del grupo

Permite la gestión de los cargos que tienen algunos integrantes en el grupo. Los cargos son configurados previamente por tipo de grupo, solo los líderes pueden seleccionar la persona que lo desempeñará. Además se pueden visualizar las funciones que tiene que desempeñar.

Grupo pequeño Amasías Administrar Reportes

Mis grupos Opciones Mi perfil

Chat del grupo 1 Cumpleaños Más opciones

Cargos

Grupo pequeño Amasías
Lugar de reunión: Auditorio de la fe
Día: Viernes
Hora: 07:30 p.m.

Actualizar

Lista de cargos

#	Departamento	Cargo	Responsable	Borrar
1	Cargos generales	• Consejero	Rullery Montenegro Solís	Borrar
		• Co-Líder	Seyel Rengifo Arevalo	Borrar
		• Coordinador de actividades sociales	Alexander Diaz Infante	Borrar
		• Secretario	Ruth Elisa Alcalde Delgado	Borrar
2	Tesorería	• Director	Taira Chenil Sanchez Cuesta	Borrar
3	Ministerio Personal	• Coordinador	Isai Natan Carvajalca Bravo	Borrar
4	Ministerio Joven	• Coordinador	Sandy Carolina Mora Ihuaraqui	Borrar
5	Ministerio De La Mujer	• Coordinador	Malu Del Pilar Becerra Martinez	Borrar
6	Mayordomía	• Coordinador	Owen Ernesto Montenegro Solís	Borrar
7	Departamento De Música	• Coordinador	Deiy Sadit Biton Bravo	Borrar
		• Coordinador asociado	Alexander Diaz Infante	Borrar
8	Comunicaciones	• Coordinador	Jordin Del Cristian Guerrero Sangama	Borrar

Figura 35. Gestión de cargos en el grupo

5.1.6 Eventos

Permite el registro de los eventos que realizan los grupos organizados, se selecciona el tipo de evento, título, fecha y hora, lugar, resumen y descripción. Cada evento permite registrar el programa a desarrollar, control de asistencia y la evaluación de acuerdo al tipo de evento. (Figura 36).

Grupo pequeño Amasías Administrar Reportes

Mis grupos Opciones Mi perfil

Chat del grupo 1 Cumpleaños Más opciones

Eventos

Actualizar Nuevo evento

Periodo: 2015-2

Fecha del evento	Hora	Programa	Asistencia	Fotos	Informe	Opciones
Reunión de grupo pequeño						
Viernes, 13 de Noviembre de 2015	07:30 pm					
Viernes, 06 de Noviembre de 2015	07:30 pm					
Viernes, 30 de Octubre de 2015	07:30 pm					
Viernes, 23 de Octubre de 2015	07:30 pm					
Viernes, 16 de Octubre de 2015	07:30 pm					
Viernes, 09 de Octubre de 2015	06:30 pm					
Viernes, 02 de Octubre de 2015	07:30 pm					
Viernes, 25 de Septiembre de 2015	07:30 pm					
Viernes, 18 de Septiembre de 2015	07:30 pm					
Viernes, 11 de Septiembre de 2015	07:30 pm					
Viernes, 04 de Septiembre de 2015	07:30 pm					

Figura 36. Registro de eventos por grupo

5.1.6.1 *Elaboración del programa*

Se elabora la programación del evento, con actividad y la asignación de responsables. Posee las siguientes opciones:

- Añadir actividad: permite registrar actividades al programa con una hora de inicio y una hora final (tiempo) y se asignan responsables.

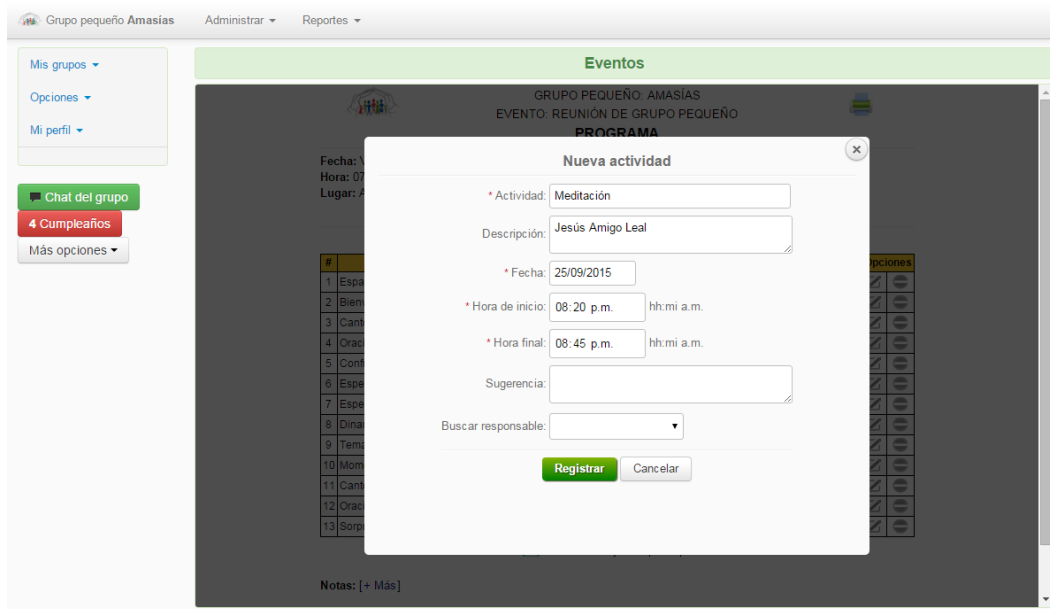


Figura 37. Añadir actividad – registro del programa

- Configuración: permite configurar más partes en el programa tales como por ejemplo pauteadores, maestros de ceremonia, ujieres, audio y video.
- Copiar formato: es una funcionalidad que permite ahorrar tiempo, es decir, se elabora un programa sugerido con actividades y duración para cada tipo de evento y aquí solo se copian automáticamente.

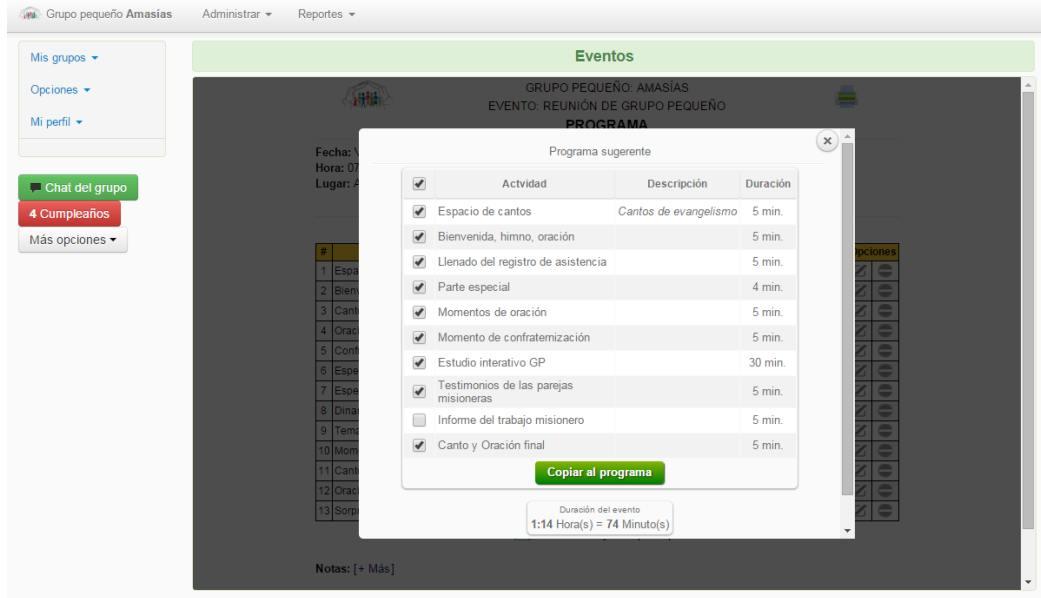


Figura 38. Copiar programa sugerente – registro del programa

- Envío de mensaje a correos: esta es una de las funcionales que usa los servicios gratuitos de correos electrónicos. El sistema envía un mensaje de participación en el programa al correo de cada participante, esto ayuda a que el programa sea anunciado a cada participante y así se evita cualquier olvido.

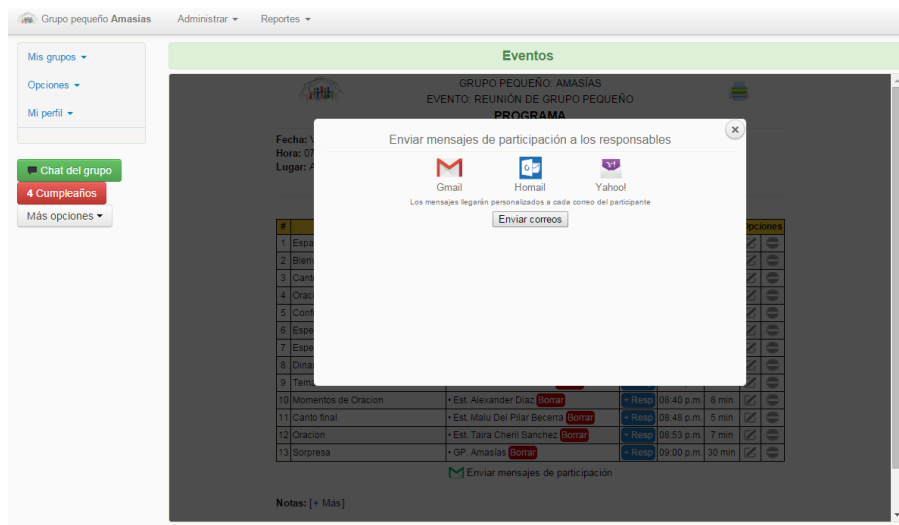


Figura 39. Envío de mensajes de participación

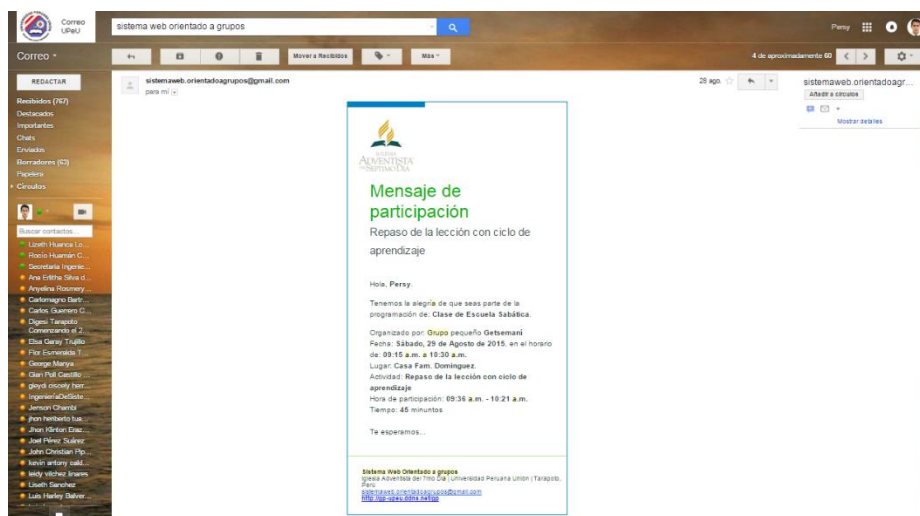


Figura 40. Mensaje de participación que le llega a cada participante

- Formato de impresión: permite imprimir el programa (si fuera necesario).



GRUPO PEQUEÑO AMA SÍAS
EVENTO: REUNIÓN DE GRUPO PEQUEÑO
PROGRAMA

Fecha: Viernes, 18 de Septiembre de 2015

Horario: 07:30 p.m. - 09:00 p.m.

Lugar: Auditorio Auditorio de la fe

N°	ACTIVIDAD	RESPONSABLE	HORA	TIEMPO
1	Espacio de cantos	Est. Alexander Diaz	07:30 p.m.	10 min
2	Bienvenida	Est. Sandy Carolina Mora	07:40 p.m.	3 min
3	Canto inicial	Est. Flor Gissela Durand	07:43 p.m.	5 min
4	Primera Oracion	Est. Malu Del Pilar Becerra	07:48 p.m.	7 min
5	Confraternización : <i>Preguntar a cada joven y a cada señorita, ¿Cómo lo paso la semana?</i>	Est. Rullery Montenegro	07:55 p.m.	20 min
6	Especial	Est. Dely Sadit Biton	08:15 p.m.	5 min
7	Dinamica	Est. Seyei Rengifo	08:20 p.m.	8 min
8	Tema Central	Est. Geseñi Trigozo	08:28 p.m.	17 min
9	Momentos de Oracion	Est. Ruth Elisa Alcalde	08:45 p.m.	7 min
10	Canto final	Est. Isai Natan Caruajulca	08:52 p.m.	5 min
11	Oracion	Est. Darly Yuyarima	08:57 p.m.	3 min

Figura 41. Formato de impresión del programa

5.1.6.2 Registro de asistencia y evaluación

Permite registrar la asistencia de los integrantes y otras personas al evento. Existen estados de asistencia tales como: puntual, tardanzas, faltas y faltas justificadas.

Grupo pequeño Amasias Administrar Reportes

Mis grupos Opciones Mi perfil

Chat del grupo 4 Cumpleaños Más opciones

Eventos

GRUPO PEQUEÑO: **AMA SÍAS**
EVENTO: CLASE DE ESCUELA SABÁTICA
ASISTENCIA

Fecha: Sábado, 31 de Octubre de 2015
Hora: 09:00 a.m. - 10:30 a.m.
Lugar: Aula c-305

← Eventos Actualizar + Añadir integrs (+) Asistentes Barras Finalizar

Lista de asistentes (x) eliminar

#	Apellidos y nombres	Tipo asistente	Asistencia	Est	Evaluar	Edit	Borr
1	Alcalde Delgado, Ruth Elisa	Integrante del grupo	P T F FJ	P			
2	Becerra Martínez, Malu Del Pilar	Integrante del grupo	P T F FJ	P			
3	Biton Bravo, Dely Sadit	Integrante del grupo	P T F FJ	P			
4	Caruajulca Bravo, Isai Natan	Integrante del grupo	P T F FJ	P			
5	Davilla Panduro, Rebeca Jazmin	Integrante del grupo	P T F FJ	T			
6	Diaz Infante, Alexander	Integrante del grupo	P T F FJ	P			
7	Durand Usuriano, Flor Gissela	Integrante del grupo	P T F FJ	F			
8	Guerrero Sangama, Jordin Del Cristian	Integrante del grupo	P T F FJ	P			
9	Montenegro Solis, Rullery	Integrante del grupo	P T F FJ	P			
10	Montenegro Solis, Owen Ernesto	Integrante del grupo	P T F FJ	P			
11	Mora Ihuaquai, Sandy Carolina	Integrante del grupo	P T F FJ	P			
12	Rengifo Arevalo, Seyei	Integrante del grupo	P T F FJ	P			
13	Sanchez Cuesta, Taira Cheril	Integrante del grupo	P T F FJ	P			
14	Trigozo Galoc, Geseñi	Integrante del grupo	P T F FJ	FJ			
15	Yuyarima Canaquiri, Darty	Integrante del grupo	P T F FJ	P			

Más información:

Figura 42. Registro de asistencia al evento

En el registro de asistencia se evalúan los rubros configurados por tipo de evento. Tomando por ejemplo se han evaluado “días de estudio de escuela sabática” y “días de estudio de reavivados por su Palabra” que automáticamente se mostrarán en la tabla comparativa (termómetro de escuela sabática).

5.1.7 Sub grupos

El sistema permite organizar a los integrantes del grupo de tipos de sub-grupos, con un número máximo. Por ejemplo los tipos de grupos pequeños se pueden organizar en “parejas misioneras”.

5.1.9 Definición de indicadores (metas)

Cada grupo organizado define las metas para los indicadores establecidos; esta es una funcionalidad que ayuda a gestionar la evaluación.

The screenshot displays a web application interface for defining indicators. The main content area is titled 'Definición de indicadores' and contains a form for the 'Grupo pequeño Getsemani'. The form is organized into three sections: 'Comunión', 'Relacionamiento', and 'Misión'. Each section lists specific indicators with corresponding target values (Metas) in input fields. A 'Guardar' button is located at the bottom right of the form. The interface also includes a sidebar with navigation options like 'Mis grupos', 'Opciones', and 'Mi perfil', and a top navigation bar with 'Administrar' and 'Reportes'.

Indicador	Meta
Comunión	
Promedio de personas que estudian diariamente Reavivados por su Palabra	15
Promedio de personas que estudian diariamente su lección	15
Relacionamiento	
Promedio de personas que asisten a la Reunión del Grupo Pequeño	15
Promedio de personas que asisten a la clase de Escuela Sabática	15
Misión	
N° de parejas misioneras que dan estudios Bíblicos	2
N° de estudiantes de la Biblia	4
N° de bautismos	2

Figura 45. Definición de indicadores

5.1.10 Reportes de gestión

Se presentan una lista de reportes que ayudan a la gestión y permiten la evaluación de los grupos organizados.

5.1.10.1 Grupos registrados

Se presenta una lista de grupos organizados por dependencias y por tipo de grupo.

Grupo pequeño Getsemani Administrar Reportes

Mis grupos Opciones Mi perfil

Chat del grupo 4 Cumpleaños Más opciones

Grupos registrados

Periodo: 2015-2 Tipo de grupo: Grupo pequeño (53) Cargo: Solo nombre del grupo Buscar

Grupo pequeño Getsemani
Reporte de Grupos Pequeños - 2015-2

#	Lugar de reunión	Escuela Sabática	Grupo pequeño	Líder del grupo	Entidad Acad.
1	Aula C-101	Bernabé Chavez	JARE	Noemi Mendoza Salas	FCS
			Reavivados por Jesus	Roby Navarro Portocarrero	FCS
			RIGEL	Juan Carlos Guerrero Rojas	FCS
2	Aula C-206	Ruben Castillo	Sear Jasub	Jhan Arly Sanchez Tarrillo	FIA
			Proclama	Juan Carlos Bustamante Sanchez	FIA
3	Aula B-104	Manuel Zuñiga Camacho	KADOSH	Heydi Lizeth Marlo Salazar	FIA
			"ORION"	Abel Rivera Cervantes	FIA
4	Aula B-105	Eduardo Forga	Gedeón	Jheimy Daniel Tuanama Tirado	FCE
			GEDEONES EN ACCION	James Segura Chávez	FCE
			HEFZIBA 1.0	Margarita Huancas Herrera	FCE
5	Iglesia San Alejandro	Juan Byintón	Ajhaska	Ana Iris Vásquez Ríos	FIA
			Adriel	Orfa Guerrero Cruz	FCS
			Getsemani	Ana Keila Apaestegui Huamán	FCE
6	Aula C-103	Juan Norton Loughborough	Jamivol	Merly Ramirez Irene	FCS
			JASS	Sara Elizabeth Orbeagozo Fasanando	FCS

Figura 46. Lista de grupos organizados

5.1.10.2 Reporte de asistencia por grupo

Se reporta la asistencia por tipo de evento realizado, este reporte ayuda a gestionar y evaluar el nivel de asistencia por cada persona, porque el objetivo está orientado a la mejora de las personas.

Grupo pequeño Getsemani Administrar Reportes

Mis grupos Opciones Mi perfil

Chat del grupo 4 Cumpleaños Más opciones

Asistencia

Periodo: 2015-2 Tipo: Reunión de grupo pequeño (14) Asistente: Todos

Grupo pequeño: Getsemani
Lugar de reunión: Casa Fam. Dominguez
Días: Viernes | Hora: 07:30 p.m.

Tipo de evento: Reunión de grupo pequeño
Reporte de asistencia 2015-2

#	Apellidos y nombres	Código	Agosto		Septiembre				Octubre			Noviembre			
			Viernes 07	Viernes 14	Viernes 21	Viernes 28	Viernes 04	Viernes 11	Viernes 18	Viernes 25	Viernes 02	Viernes 09	Viernes 16	Viernes 23	Viernes 30
1	Isuisa Amasifuen, Ledvin		P	P	P	P	P	P	P	P	P	P	P	P	P
2	Apaestegui Huamán, Ana Keila	201110823	P	P	P	P	P	P	P	P	P	P	P	P	P
3	Bustamante Sanchez, Mirian	201121374	P	P	P	P	F	P	P	F	F	F	F	F	F
4	Carbajal Ochavano, Ruby	201310481	P	P	P	P	P	P	P	P	P	P	P	P	P
5	Carlos Cruz, Edinson Raul	201310192	P	P	P	F	P	P	P	P	P	P	P	P	P
6	Castillo Melendres, Reyner	201510528	P	P	P	P	P	P	P	P	P	P	P	P	P
7	Chinchay Liacsahuache, Hulmer William	201220740	P	F	F	P	P	P	P	P	P	P	P	P	F
8	Chumacero Córdoba, Lizdey	201410827	P	P	P	P	P	P	P	P	P	P	P	P	P
9	Flores Villanueva, Clarita	201410806	P	P	P	P	P	P	P	P	P	P	P	F	F
10	Gil García, Isai	201122889	P	P	P	P	P	P	P	P	P	P	P	P	P
11	Gonzales Guzmán, Maribel		F	P	F	F	P	F	P	P	P	P	P	P	P
12	Herrera Farfán, Gleydi Ciscely	201121532	P	P	P	P	P	P	P	P	P	P	P	P	P
13	López Usin, Javier		P	P	P	F	P	P	P	P	F	F	P	F	F
14	Lozano Tapullima, Estelith	201121507	P	P	P	P	F	P	F	F	F	F	P	P	P
15	Quiroz Menor, Dany	201420624	P	P	P	P	P	P	P	P	P	F	P	P	P
16	Quiroz Menor, Persy	201011895	P	P	P	P	P	P	P	P	P	P	P	P	P
17	Ruise Camacho, María Elizabeth	201420617	P	P	P	P	P	P	P	P	P	P	P	P	P

Figura 47. Reporte de asistencia a cada grupo

5.1.10.3 *Reporte: documentos de la persona*

Cada persona que pertenece o haya participado a un grupo podrá visualizar su asistencia, participaciones, evaluaciones e incidencias por periodo (Ver *Figura 48*).

Grupo pequeño Getsemani Administrar Reportes

Mis documentos

Periodo: 2015-2 Documento: Ficha personal Aceptar

Docente: Persy Quiroz Menor
Código universitario: 201011895
Escuela: Ingeniería de Sistemas Residente: Externo
Religión: Adventista

FICHA PERSONAL

ASISTENCIA			
1.	Reunión de grupo pequeño	Integrante del grupo	07/08/2015 P
2.	Reunión de grupo pequeño	Integrante del grupo	14/08/2015 P
3.	Reunión de grupo pequeño	Integrante del grupo	21/08/2015 P
4.	Reunión de grupo pequeño	Integrante del grupo	28/08/2015 P
5.	Clase de Escuela Sabática	Integrante del grupo	29/08/2015 P
6.	Reunión de grupo pequeño	Integrante del grupo	04/09/2015 P
7.	Reunión de grupo pequeño	Integrante del grupo	11/09/2015 P
8.	Reunión de grupo pequeño	Integrante del grupo	18/09/2015 P
9.	Reunión de grupo pequeño	Integrante del grupo	25/09/2015 P
10.	Reunión de grupo pequeño	Integrante del grupo	02/10/2015 P
11.	Reunión de grupo pequeño	Integrante del grupo	09/10/2015 P
12.	Reunión de grupo pequeño	Integrante del grupo	16/10/2015 P
13.	Reunión de grupo pequeño	Integrante del grupo	23/10/2015 P
14.	Reunión de grupo pequeño	Integrante del grupo	30/10/2015 P
15.	Clase de Escuela Sabática	Integrante del grupo	31/10/2015 P
16.	Reunión de grupo pequeño	Integrante del grupo	06/11/2015 P
17.	Clase de Escuela Sabática	Integrante del grupo	07/11/2015 P

Figura 48. Reporte documentos de la persona

5.1.10.4 *Monitoreo de mi grupo*

Permite visualizar toda la información relacionada al grupos tales como: datos del grupo, integrantes, cargos, lista de eventos realizados, asistencia a los grupos, álbum de fotos y análisis de asistencias.

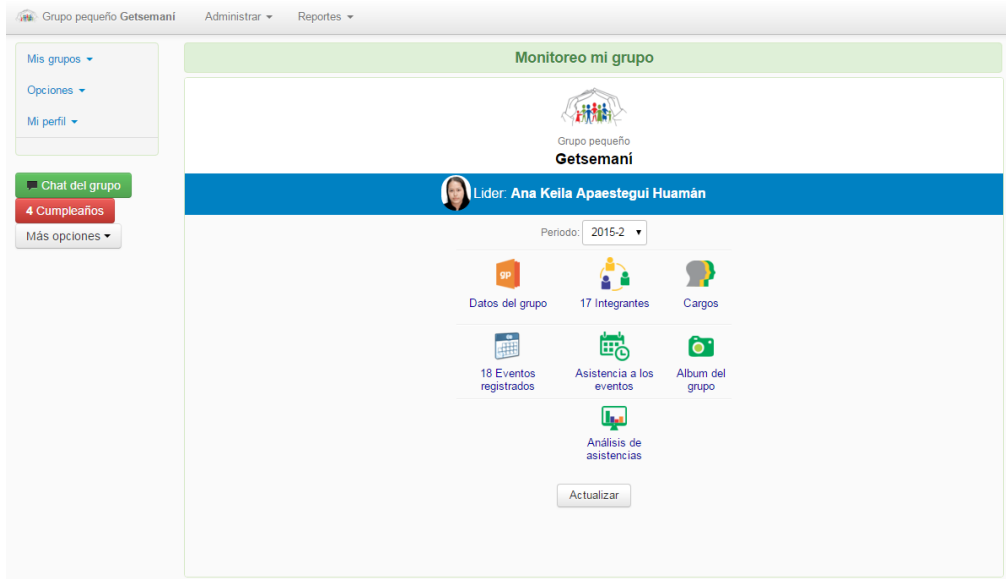


Figura 49. Monitoreo mi grupo

5.1.10.4.1. Análisis de asistencias

Por medio de un gráfico de barras se puede visualizar el nivel de asistencia por cada evento realizado, donde permite analizar si el grupo está mejorando o disminuyendo su asistencia.

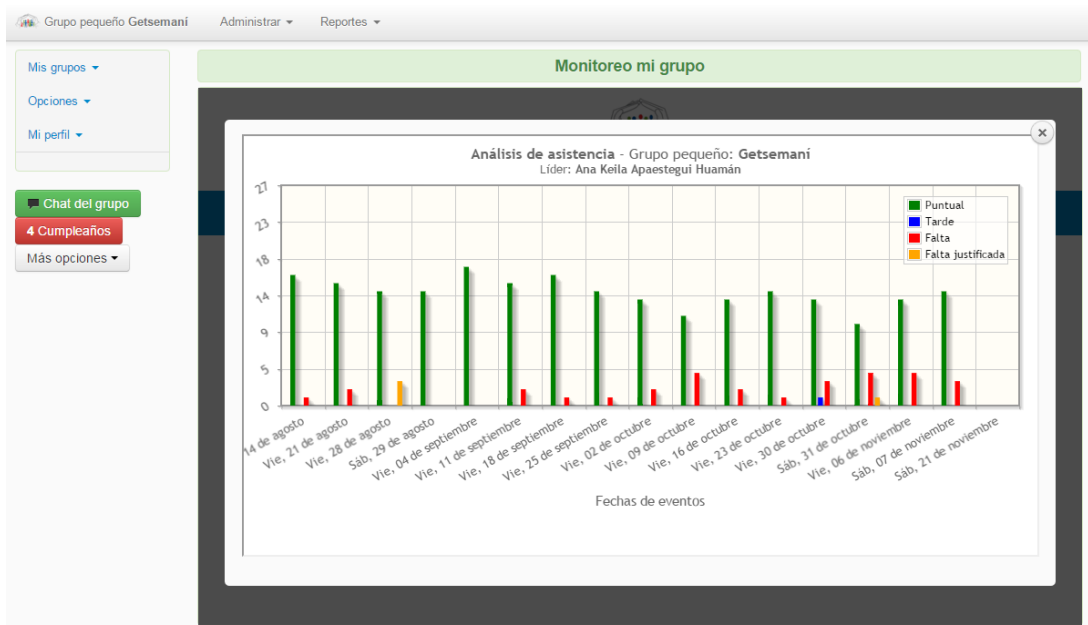


Figura 50. Análisis de asistencias del grupo

5.1.10.5 Termómetro de Escuela Sabática

La Iglesia Universitaria Villa Unión trabaja con una tabla comparativa ahora llamado termómetro, que permite evaluar a cada grupo. Es por ello, este sistema busca ese fin, se construyó el sistema para capturar esos indicadores también (Ver *Figura 51*).



Figura 51. Termómetro de la Escuela Sabática

5.1.10.6 Reporte de grupos e integrantes

Este informe permite ver el total de integrantes por tipo de grupo, la lista de grupos, los datos del grupo y el total de integrantes, todos ordenados por entidad académica y de mayor a menor la cantidad de integrantes.

Iglesia Universitaria Villa Unión Registros Administrar Reportes Configuraciones

Grupos e integrantes

Periodo: 2015-2 Tipo de grupo: Grupo pequeño (53) Aceptar

Iglesia Universitaria Villa Unión
Periodo: 2015-2
Reporte de Grupos Pequeños

#	Grupo pequeño	Lider	Co-lider	Consejero	Total Integ	Lugar de reunión	Día reunión	Hora reunión	Entidad Acad.	Info
1	Gedeón	Jherimy Daniel Tuanama Tirado			25	Salón UPeU	Viernes	07:30 am	FCE	
2	HEFZIBA 1.0	Margarita Huancas Herrera	Pier Alfredo Espejo Inga	José Antonio Manzanares Sipión	24	Aula Salón de música	Viernes	07:00 pm	FCE	
3	URIEL	Zacarías Otoniel Huancas Facundo	Elio Jefferson Huanuiri Goñas	Jezer Donald AYAY PINEDO	22	Aula C-304	Viernes	07:30 pm	FCE	
4	EZRI	Daniel Bermeo Huaman	Lucila Calderón Guerrero		20	Aula B-106	Viernes	07:30 pm	FCE	
5	Alborada	Doris Violeta Guevara Arteaga			18	Salón UPeU	Viernes	07:00 pm	FCE	
6	León de Juda	Cynthia Marvellit Pizarro Silva			18	Aula B-101	Viernes	07:30 pm	FCE	
7	Getsemani	Ana Keila Apaestegui Huamán	María Elizama Rojas Campos	Maribel Gonzales Guzmán	17	Casa Fam. Dominguez	Viernes	07:30 pm	FCE	
8	AGAPE	Gabriela Acsa Aranda Saboya			15	Salón UPeU	Viernes	07:30 pm	FCE	
9	PIONEROS DE ESPERANZA	Abner Daniel Chingo Maldonado	Nole Geraldo Rivera		15	Aula C-305	Viernes	07:30 pm	FCE	
10	SHEKINAH	Carlos Guerrero Cordova	Jeiner Moises Anticona Quepque		12	Aula AULA C- 306	Viernes	07:00 pm	FCE	
11	Generación de Valientes	Wesley Nilver Moreto Guerrero			11	Aula B-103	Viernes	07:30 pm	FCE	
12	BASHAR	Oscar Nelver Llamoctanta Maluquish			9	Salón UPeU	Viernes	07:30 pm	FCE	
13	Grupo 9 - FCE	Keny Macedo Hidalgo			8	Salón UPeU	Viernes	07:00 pm	FCE	
14	GEDEONES EN ACCIÓN	James Segura Chávez		Josue Edison Turpo Chazman	5	Aula C 206	Viernes	07:00 pm	FCE	

Figura 52. Reporte de grupos e integrantes

5.1.10.7 Reporte de asistencia a los grupos

Permite un reporte de asistencia a los grupos filtrado por entidad académica y seleccionando la fecha del evento. Se pueden visualizar los grupos que registraron la asistencia, las personas que asistieron, faltaron y motivos de falta. Este informe ayuda a los capellanes para ayudar a los jóvenes que están pasando por dificultades y así orar por ellos, porque para Dios nada es imposible. (Ver Figura 53).

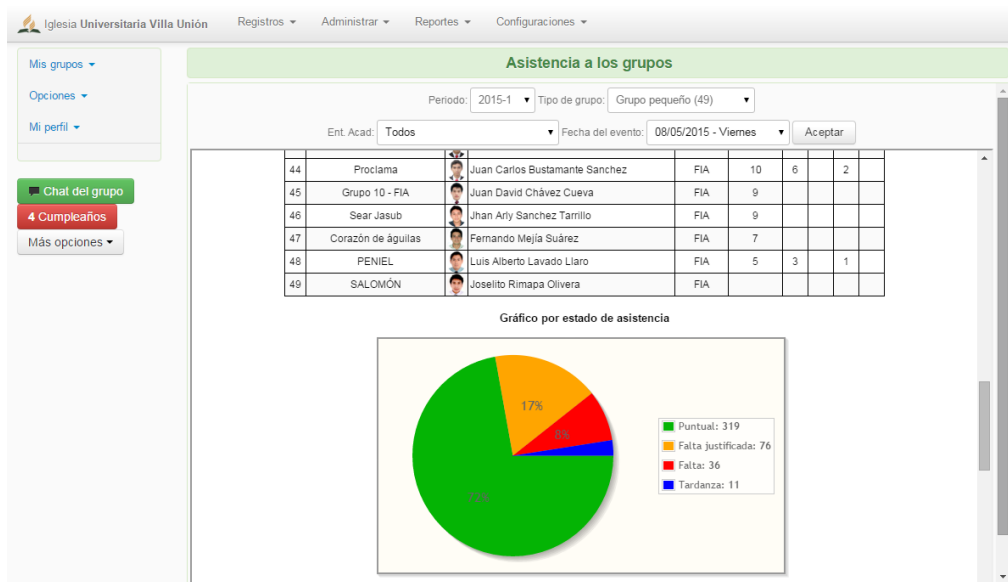


Figura 53. Asistencia a los grupos por fecha

5.1.10.8 Estadísticas generales

Estos reportes permiten visualizar el total de inscritos por entidad académica, tipo de persona, religión y género.

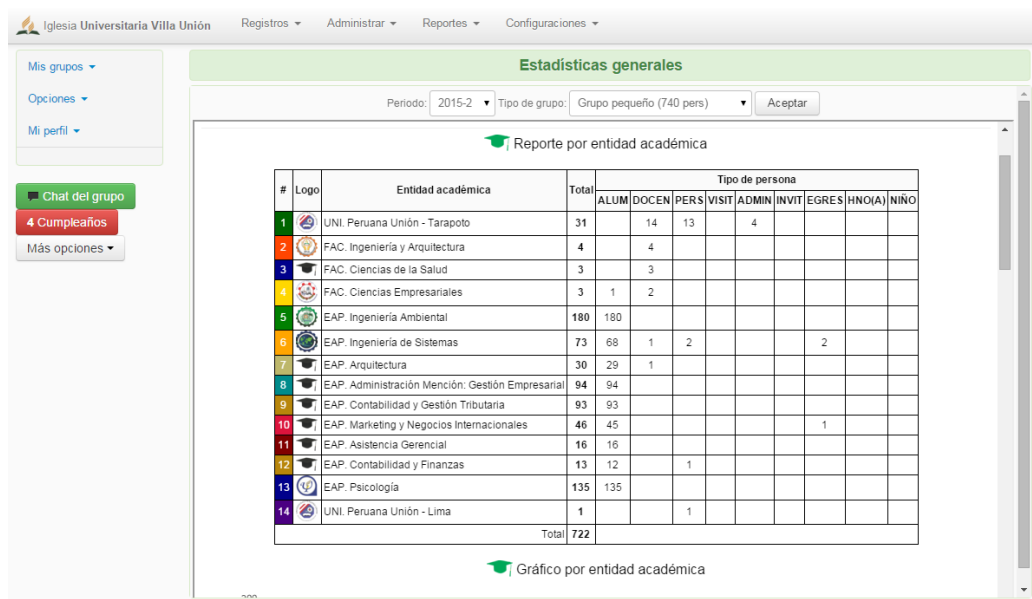


Figura 54. Estadísticas generales (reporte)

5.1.10.9 Lista de personas por número de asistencia

Este reporte permite gestionar el total de asistencias de cada persona, por medio de varios filtros tales como: tipos de persona, tipo de evento, religión, Bautizado (si o no), estado de asistencia y entidad académica (Ver *Figura 55*).

The screenshot shows a web application interface for 'Iglesia Universitaria Villa Unión'. The main content area is titled 'Reportes de gestión' and contains a section for 'Lista de personas por número de asistencias'. This section includes a grid of checkboxes for filtering data based on: Tipo de persona (Alumno, Docente, Personal general, Visitante, Administrativo, Invitado, Egresado, Hermano (a), Niño (a)), Tipo de evento (Grupo pequeño, Reunión de grupo pequeño, Evento especial, Culto Joven, Clase de Escuela Sabática), Religión (Adventista, Católico, Evangélico, Otro, Ninguno, Adventista Reforma), Bautizado (SI, NO), and Ent. Acad. (Adm, Arq., Asist. Geren., FCS, FCE, Cont. Finan., Cont. Trib., Ing. Amb., Ing. Sist., FIA, MKT, UPeU - L, UPeU - T, Psic.). Below the filters, there are buttons for 'Volver a inicio', a dropdown menu for 'Indique el número:' set to 'Mayor o igual a', a text input field with '10', and an 'Aceptar' button.

Below the filter section, there is a table titled 'Lista de personas por cantidad de asistencias' for the period '2015-2'. The table has the following data:

#	Apellidos y nombres	Cod. Univer	Ent. Acad.	Religión	Bautizado	Total
1	Arevalo Chujutalli Ingrith Fiorella	201222169	Administración Mención: Gestión Empresarial	Católico		14
2	Arevalo Uriarte Franklin	201310187	Administración Mención: Gestión Empresarial	Adventista	SI	10
3	Arevalo Diaz Sindy Cinthia	201320894	Administración Mención: Gestión Empresarial	Católico		14
4	Bernal Tikla Marleni	201221330	Administración Mención: Gestión Empresarial	Adventista	SI	18
5	Bustamante Sanchez Mirian	201121374	Administración Mención: Gestión Empresarial	Adventista	SI	17

Figura 55. Lista de persona por número de asistencias

5.1.10.10 Análisis de asistencias

Este reporte permite analizar la asistencia por tipo de grupo en un periodo establecido (Ver *Figura 56*).

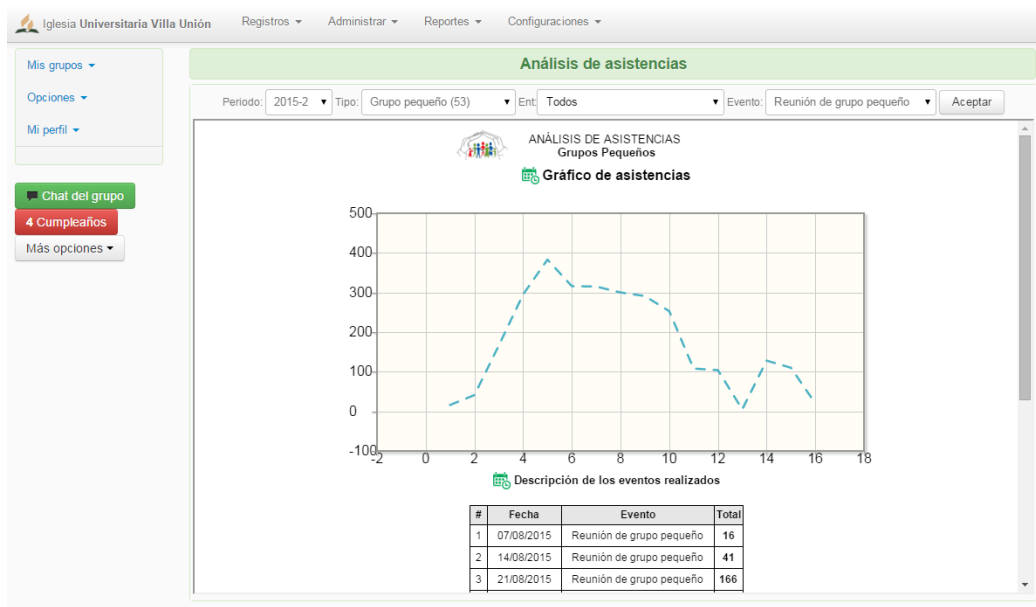


Figura 56. Análisis de asistencias

5.1.10.11 Otros reportes

Cabe mencionar a raíz de la información existentes se elaboran más reportes que es muy importante mencionarlos tales como:

- Lista de parejas misioneras: este reporte es una lista y el total de sub-grupos en cada grupo, se muestran los integrantes por cada uno de ellos y por periodo.
- Reporte de estudios Bíblicos: se muestran la lista de estudiantes de la Biblia, el curso Bíblico que se está desarrollando y la lección que están en curso, esto permite ver el avance y la frecuencia del trabajo de las parejas misioneras (sub grupos).
- Reporte de asistencia a la escuela misionera: se armó este reporte que ayude a la gestión al departamento de MiPES (Ministerio Personal y Escuela Sabática), mostrando la asistencia

de cada integrante de una pareja al tipo de evento “escuela misionera” que consta de capacitaciones a cada misionero.

- Termómetro de la Escuela Sabática por grupo: permite obtener la información de su situación de cada grupo organizado, la búsqueda es por periodo, tipo de grupo y entidad académica (Ver *Figura 57*).

#	Grupo pequeño	Líder	Ent. Acad.	Total Integ	Termóm
1	Gedeón	Jheimy Daniel Tuanama Tirado	FCE	25	Termómetro
2	HEFZIBA 1.0	Margarita Huancas Herrera	FCE	24	Termómetro
3	URIEL	Zacarías Otoniel Huancas Facundo	FCE	22	Termómetro
4	EZRI	Daniel Bermeo Huaman	FCE	20	Termómetro
5	León de Juda	Cyntia Marvelit Pizarro Silva	FCE	19	Termómetro
6	AGAPE	Gabriela Acsa Aranda Saboya	FCE	15	Termómetro
7	Getsemani	Ana Keila Apaestegui Huamán	FCE	15	Termómetro
8	PIONEROS DE ESPERANZA	Abner Daniel Chingo Maldonado	FCE	15	Termómetro
9	Alborada	Doris Violeta Guevara Arteaga	FCE	14	Termómetro
10	Generación de Valientes	Wesley Nilver Moreto Guerrero	FCE	12	Termómetro

Figura 57. Termómetro de la Escuela Sabtica – Gestión

- Monitoreo general: permite acceder a toda la información del grupo organizado tales como datos del grupo, lista de integrantes, cargos, eventos, asistencia, álbum (fotos de los eventos) y un análisis de asistencia que permite ver la frecuencia (Ver *Figura 58*).

#	Grupo pequeño	Lider	Ent. Acad.	Total Integ
1	Gedeón	Jheimy Daniel Tuanama Tirado	FCE	25
2	HEFZIBA 1.0	Margarita Huancas Herrera	FCE	24
3	URIEL	Zacarías Otoniel Huancas Facundo	FCE	22
4	EZRI	Daniel Bermeo Huaman	FCE	20
5	León de Juda	Cynthia Marvelit Pizarro Silva	FCE	19
6	AGAPE	Gabriela Acsa Aranda Saboya	FCE	15
7	Getsemani	Ana Keila Apaestegui Huaman	FCE	15
8	PIONEROS DE ESPERANZA	Abner Daniel Chingo Maldonado	FCE	15
9	Alborada	Donis Violeta Guevara Arteaga	FCE	14
10	Generación de Valientes	Wesley Nilver Moreto Guerrero	FCE	12
11	SHEKINAH	Carlos Guerrero Cordova	FCE	12

Figura 58. Monitoreo general de los grupos

- Líderes con más eventos registrados (reporte de gestión): permite ver el cumplimiento del control de asistencia en cada grupo.
- Personas con más participaciones (reporte de gestión): brinda una lista de las personas que más participan en todos o en un determinado evento.
- Personas con más asistencias (reporte de gestión): permite ver una lista de personas con más asistencias a los grupos, la búsqueda es por tipo de grupo, tipo de evento y religión.
- Personas con más faltas y faltas justificadas (reporte de gestión): emite un reporte de las personas que tienen más faltas a los grupos, esta información ayuda a trabajar en la motivación por parte de cada capellán y pastor de iglesia, así como también al sub-área de Psicología del área de Bienestar Universitario.

- Reporte de instructores Bíblicos: informe de personas que tienen estudiantes de la Biblia, buscados por género.
- Búsqueda de dones y talentos (reporte de gestión): por medio de una actividad buscada de pueden obtener una lista de personas que con frecuencia participan, esto ayuda a la determinación de dones y talentos.
- Personas con mejor evaluación (reporte de gestión): permite una lista de persona con mejor promedio en los rubros evaluados.
- Personas con baja evaluación (reporte de gestión): permite una lista de persona con baja promedio en los rubros evaluados, esto permite trabajar identificando necesidades en bien de cada integrante del grupo.
- Evaluación por entidad académica: permite identificar a las carreras que tienen una mejor evaluación en los rubros, ayuda, ya que permite trabajar en mejorar el desempeño de cada alumno.
- Porcentaje de asistencia por entidad académica (reporte de gestión): permite ver el reporte de los estados de asistencia a los eventos por entidad académica por tipo de grupo y tipo de evento (Ver *Figura 59*).

Iglesia Universitaria Villa Unión Registros Administrar Reportes Configuraciones

Mis grupos Opciones Mi perfil

Chat del grupo
3 Cumpleaños
Más opciones

Reportes de gestión

Tipo de grupo: Grupo pequeño (53) Evento: Reunión de grupo pequeño Aceptar Volver a inicio

Porcentaje de asistencia por entidad académica

#	Entidad académica	Total personas	Total registros	% de asistencia			
				P	T	F	FJ
1	EAP. Ingeniería Ambiental	180	426	76.29%	2.82%	15.73%	5.16%
2	EAP. Administración Mención: Gestión Empresarial	98	397	83.88%	0.00%	12.59%	3.53%
3	EAP. Psicología	135	326	87.42%	1.84%	7.98%	2.76%
4	EAP. Contabilidad y Gestión Tributaria	92	325	78.15%	0.31%	12.62%	8.92%
5	EAP. Ingeniería de Sistemas	73	299	85.95%	1.67%	10.70%	1.67%
6	EAP. Marketing y Negocios Internacionales	46	182	83.52%	0.55%	9.34%	6.59%
7	UNI. Peruana Unión - Tarapoto	29	105	80.95%	3.81%	5.71%	9.52%
8	EAP. Asistencia Gerencial	16	50	78.00%	2.00%	12.00%	8.00%
9	EAP. Contabilidad y Finanzas	11	42	83.33%	2.38%	9.52%	4.76%
10	EAP. Arquitectura	20	24	83.33%	0.00%	8.33%	8.33%
11	FAC. Ingeniería y Arquitectura	4	13	84.62%	0.00%	0.00%	15.38%
12	FAC. Ciencias Empresariales	3	12	58.33%	0.00%	33.33%	8.33%
13	FAC. Ciencias de la Salud	5	6	100.00%	0.00%	0.00%	0.00%

Figura 59. Porcentaje de estados de asistencia por entidad académica

5.1.11 Configuraciones

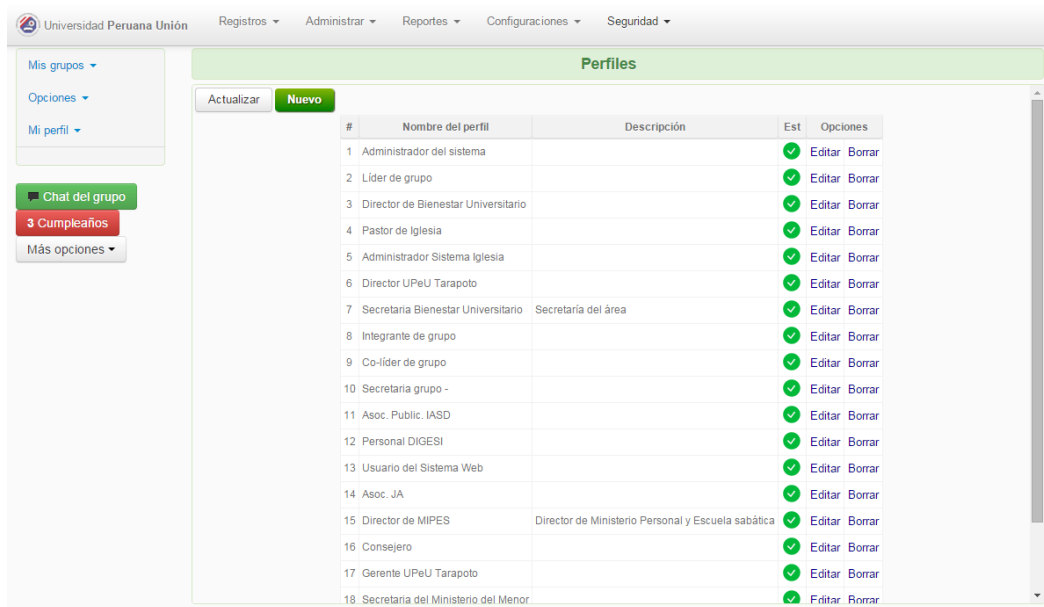
El módulo de configuraciones sirve de soporte de información tales como:

- Acceso a tipo de eventos: permite asignar a cada tipo de grupo ciertos tipos de eventos.
- Configuración de cargos: permite configurar un patrón de cargos para cada tipo de grupo, de esta manera el líder de grupo solo puede asignar los cargos.
- Programas sugeridos: el pastor de iglesia o especialista de programas elabora una lista de actividades con tiempo para cada tipo evento, de esta manera cada líder copia este formato facilitando el tiempo en digitación.
- Tipos de eventos asistentes: se asigna ciertos tipos de asistentes por cada tipo de evento.

5.1.12 Seguridad

Este módulo permite las configuraciones de accesibilidad al sistema de información contando con las siguientes funcionalidades:

- Perfiles o roles: se crean perfiles o roles y se les da ciertos accesos de acuerdo a las actividades que desempeña, respetando los criterios de confidencialidad de la información, para luego asignarlos a los usuarios por cada grupo organizado (Ver *Figura 60*).



#	Nombre del perfil	Descripción	Est	Opciones
1	Administrador del sistema		✓	Editar Borrar
2	Líder de grupo		✓	Editar Borrar
3	Director de Bienestar Universitario		✓	Editar Borrar
4	Pastor de Iglesia		✓	Editar Borrar
5	Administrador Sistema Iglesia		✓	Editar Borrar
6	Director UPeU Tarapoto		✓	Editar Borrar
7	Secretaria Bienestar Universitario	Secretaría del área	✓	Editar Borrar
8	Integrante de grupo		✓	Editar Borrar
9	Co-líder de grupo		✓	Editar Borrar
10	Secretaria grupo -		✓	Editar Borrar
11	Asoc. Public. IASD		✓	Editar Borrar
12	Personal DIGESI		✓	Editar Borrar
13	Usuario del Sistema Web		✓	Editar Borrar
14	Asoc. JA		✓	Editar Borrar
15	Director de MIPES	Director de Ministerio Personal y Escuela sabática	✓	Editar Borrar
16	Consejero		✓	Editar Borrar
17	Gerente UPeU Tarapoto		✓	Editar Borrar
18	Secretaria del Ministerio del Menor		✓	Editar Borrar

Figura 60. Perfiles o roles del sistema - Seguridad

- Módulos: son la lista de funcionalidades que posee el sistema a las cuales el usuario tendrá acceso de acuerdo a las opciones que posee el perfil.
- Perfil – accesos: permite asignar accesos a cada perfil del sistema (Ver *Figura 61*).

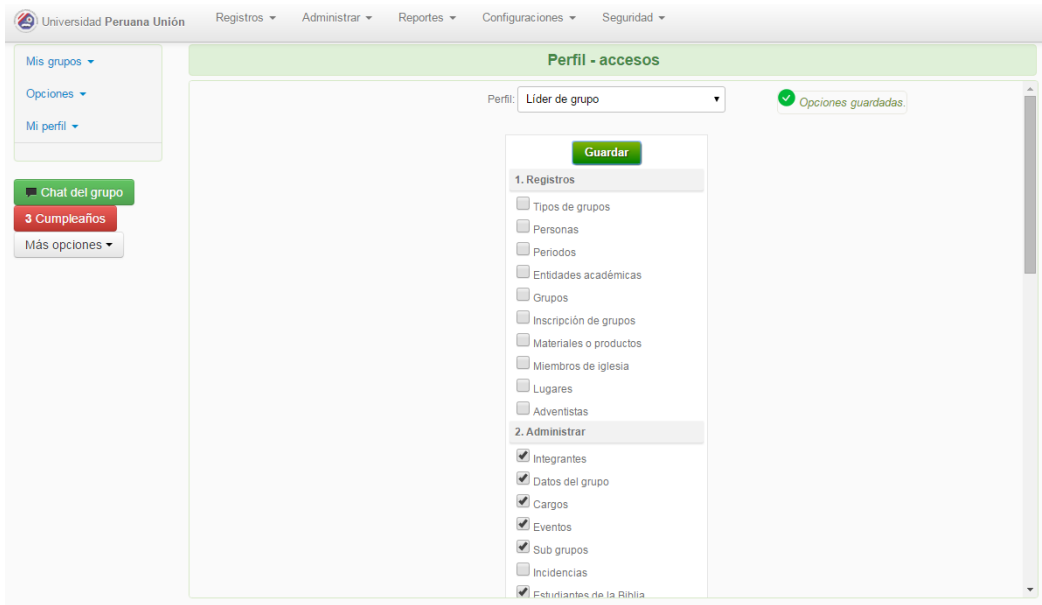


Figura 61. Asignación de accesos a los perfiles

- **Usuarios:** esta funcionalidad permite la creación de usuario del sistema y la asignación de roles. Cabe mencionar que se elaboró una solución el problema de creación de usuarios y olvido de contraseñas, es decir, las persona pueden crear su usuario (sino tienen) al momento de acceder al sistema, el requisito principal es estar registrado en el sistema y tener su correo actualizado porque a este será enviado su contraseña nueva creado por el sistema. Y ante el olvido de contraseña, también el usuario lo puede recuperar con tan solo ingresar su número de documento o código universitario y su contraseña será enviada a su correo electrónico.

#	Foto	Usuario	Nombre completo	N° grupos	Estado	Opciones
1		daniel.chingo	Abner Daniel Chingo Maldonado	(3)	✓	Editar, Borrar
2		adesmiro-ze	Adesmiro Zelada Escobedo	(1)	✓	Editar, Borrar
3		Cholan	Alejandro Cholan Prado	(2)	✓	Editar, Borrar
4		alexander	Alexander Goycochea Flores	(2)	✓	Editar, Borrar
5		AndreaNogueiraVargas	Andrea Nogueira Vargas	(0)	✓	Editar, Borrar
6		aunero	Auner Obed Llamo Delgado	(2)	✓	Editar, Borrar
7		betsabeth.padilla	Betsabeth Teresa Padilla Macedo	(1)	✓	Editar, Borrar
8		candy.gonza	Candy Emili Gonza Chunga	(0)	✓	Editar, Borrar
9		claudia_95	Claudia Yuleisy Santos Díaz	(3)	✓	Editar, Borrar
10		daniel-bh	Daniel Bermeo Huaman	(4)	✓	Editar, Borrar
11		danielhanco	Daniel William Hanco Pantoja	(2)	✓	Editar, Borrar
12		darv-om	Darv Quiroz Menor	(0)	✓	Editar, Borrar

Figura 62. Usuarios del sistema (seguridad).

5.1.13 Más opciones

Se implementaron otras opciones tales como:

Publicaciones (1), esta funcionalidad permite la publicación de fotos o mensajes de tal manera todos puedan ver, funciona tipo una red social, permitiendo conocer cómo están desarrollando sus eventos cada grupo.

Chat del grupo (2), permite la conversación entre los integrantes del grupo sobre temas diversos, en la actualidad sirve como un buzón de mensajes y anuncios.

Documentos del grupo (3), permite subir la foto oficial del grupo, la foto del lugar de reunión y su canto tema en formato MP3.

Ubicación de los lugares de reunión (4), haciendo uso del API de Google Maps, se ubican los puntos de los lugares de reunión de cada grupo organizado (Ver *Figura 63*).

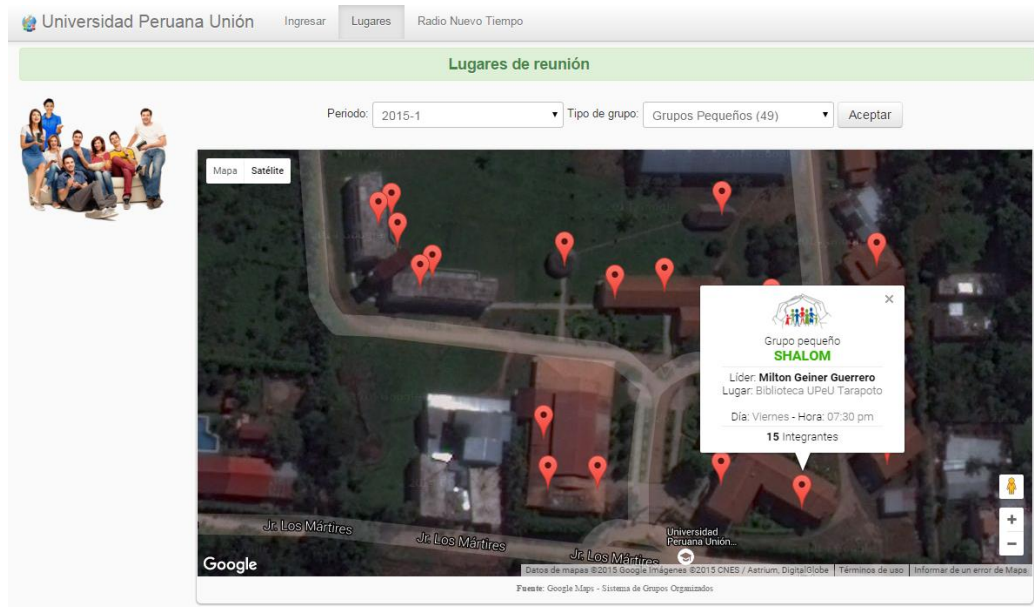


Figura 63. Lugares de reunión con puntos de referencia de Google Maps

5.1.14 Encuesta a líderes de grupos pequeños

Al final del semestre académico, se realizó una encuesta para conocer las mejoras obtenidas tras la implementación de este proyecto que aparecen en los anexos: *Anexo 1* y *Anexo 2*. Se seleccionaron dos grupos de personas compuestos por 12 líderes que no usaron el sistema y 14 líderes que si usaron. El primero por aquellos que registraron constantemente en el sistema y el otro por aquellos que no lo hicieron por temas de accesibilidad o porque consideraron que no es de mayor importancia. La encuesta tuvo el objetivo de conocer las mejoras obtenidas tras la implementación del sistema según la perspectiva de los líderes. En los siguientes gráficos se muestran 10 preguntas aplicadas, se hace las comparaciones entre estos dos tipos de grupos y se analizan las mejoras obtenidas.

Pregunta 1:

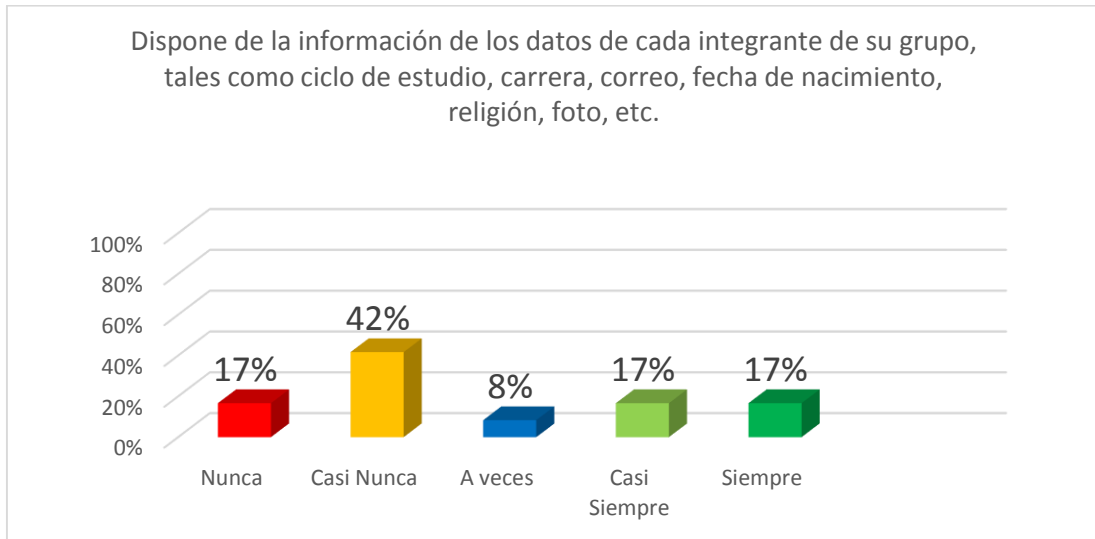


Gráfico 1. Preguntar 1, líderes que no usan el sistema

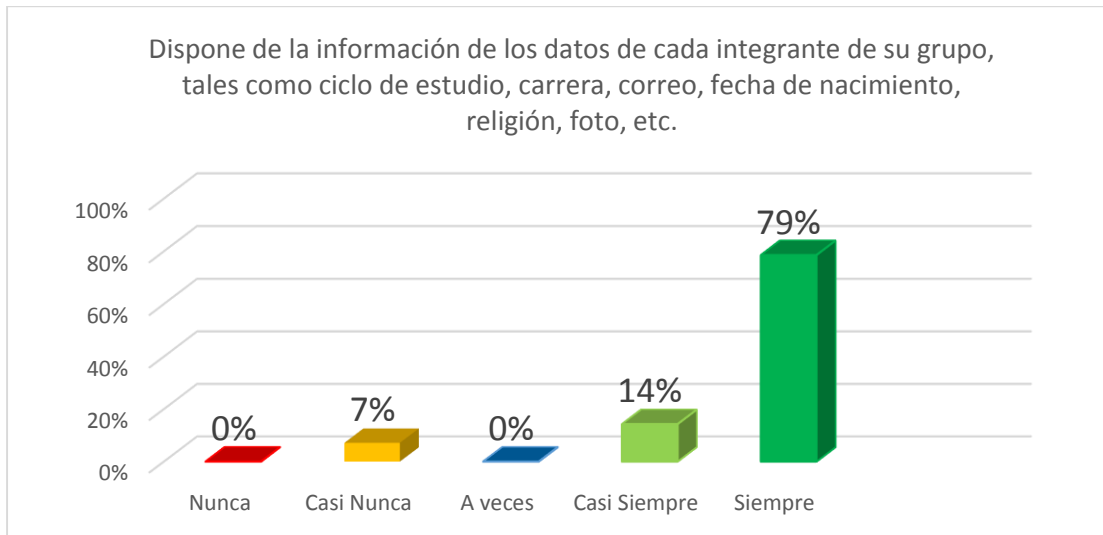


Gráfico 2. Preguntar 1, líderes que usan el sistema

Análisis: El 79% de los líderes que usan el sistema sienten disponer siempre la información de cada uno de los integrantes, mientras que solo el 17% de los líderes que no usan el sistema. En conclusión: el sistema permitió que los líderes dispongan de la información de datos de cada integrante de grupo, se ve una mejora.

Pregunta 2:

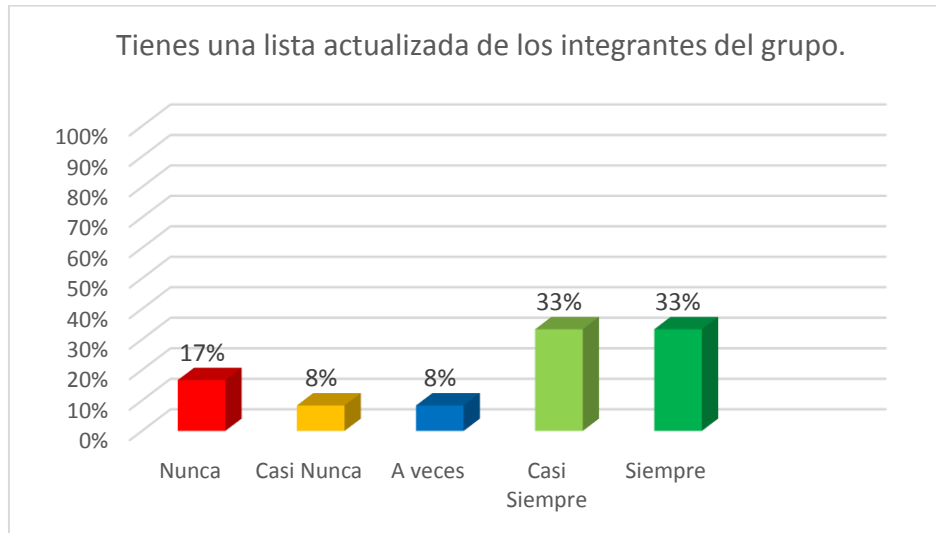


Gráfico 3. Pregunta 2, líderes que no usaron el sistema

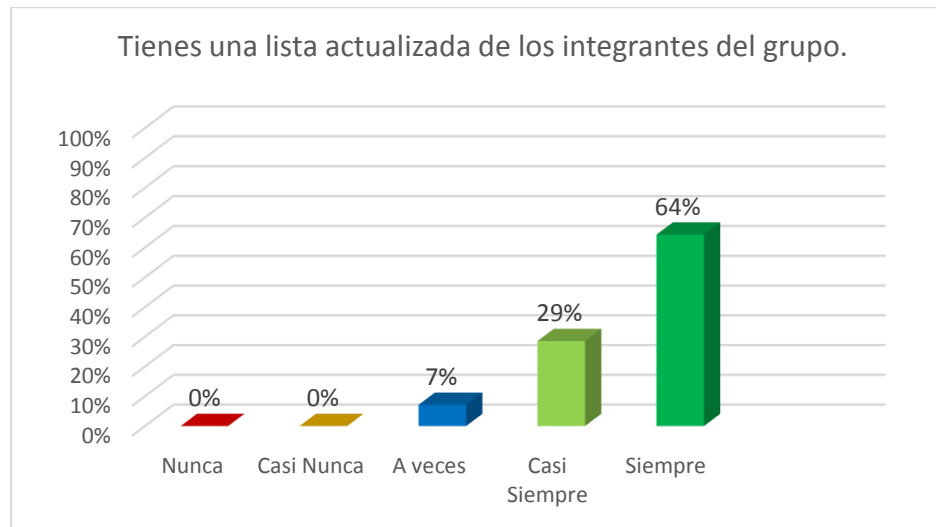


Gráfico 4. Pregunta 2, líderes que usan el sistema

Análisis: En los gráficos anteriores (*Gráfico 3* y *Gráfico 4*), 33% de los líderes que no usan el sistema tienen una lista actualizada, mientras con la implementación del sistema el 64% mantiene su lista actualizada de los integrantes. Según la perspectiva de los líderes ha mejorado la gestión de la información relacionado con cada integrante.

Pregunta 3:

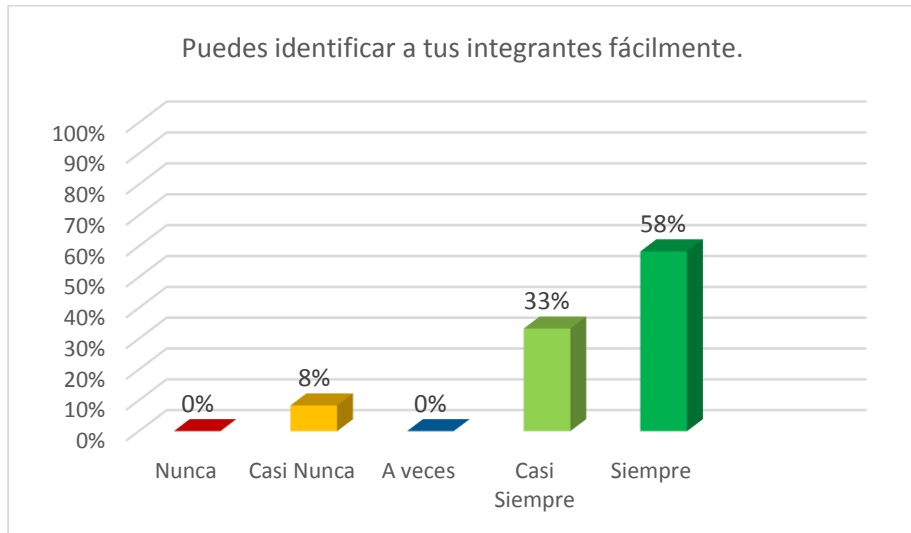


Gráfico 5. Pregunta 3, líderes que no usan el sistema

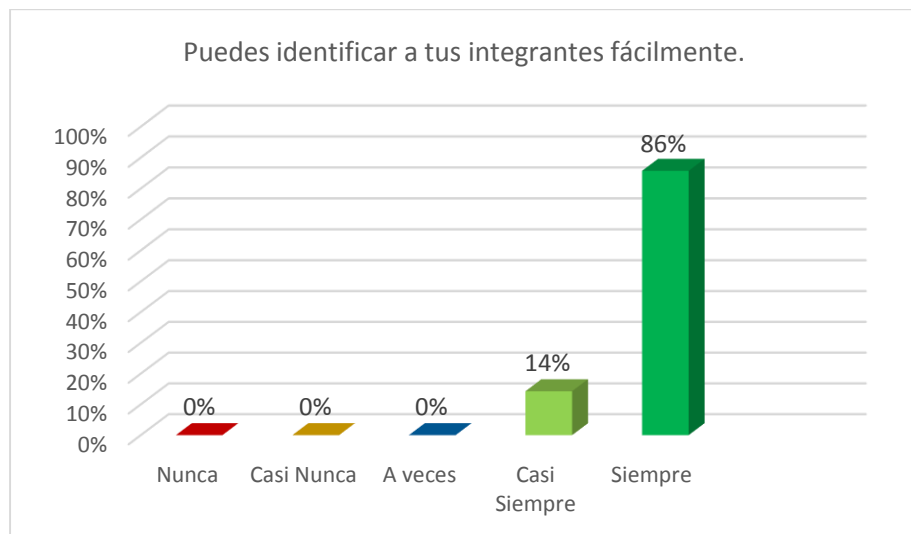


Gráfico 6. Pregunta 3., líderes que usan el sistema

Análisis: Esta pregunta tiene que ver con la personalidad del líder y la empatía que tenga con sus liderados; pero según la encuesta, el sistema ayudó a identificar mejor a sus integrantes hacia un 86% de 58% de los que no usan el sistema.

Pregunta 4:

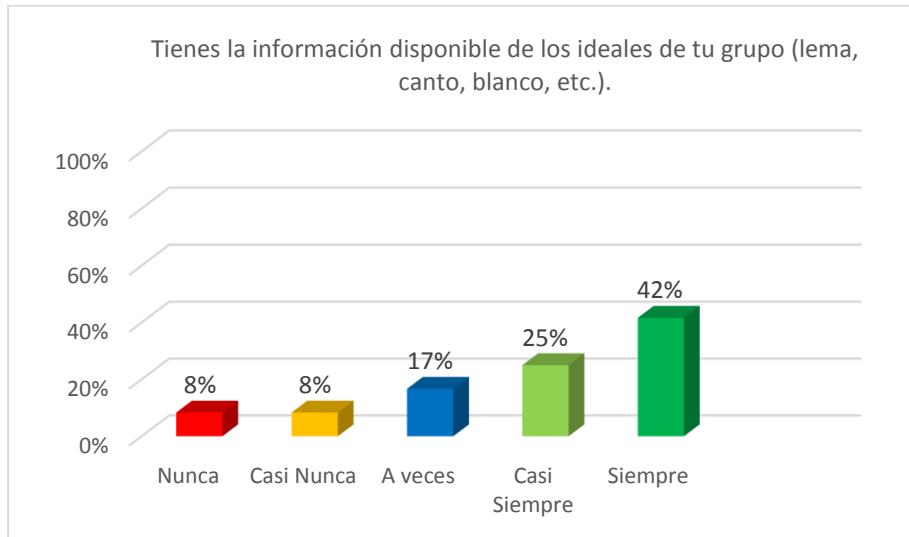


Gráfico 7. Pregunta 4, líderes que usan el sistema

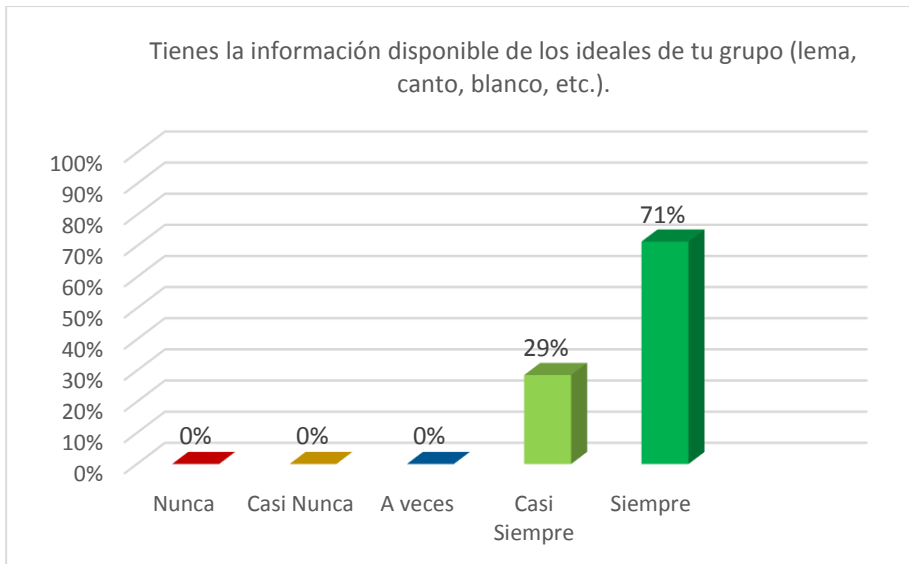


Gráfico 8. Pregunta 4, líderes que no usan el sistema

Análisis: Según los resultados los líderes que usan el sistema se sienten tener la información de los ideales de grupo más de los que no usan el sistema. Ha mejorado en un 29%, porque los líderes tienen la información disponible.

Pregunta 5:

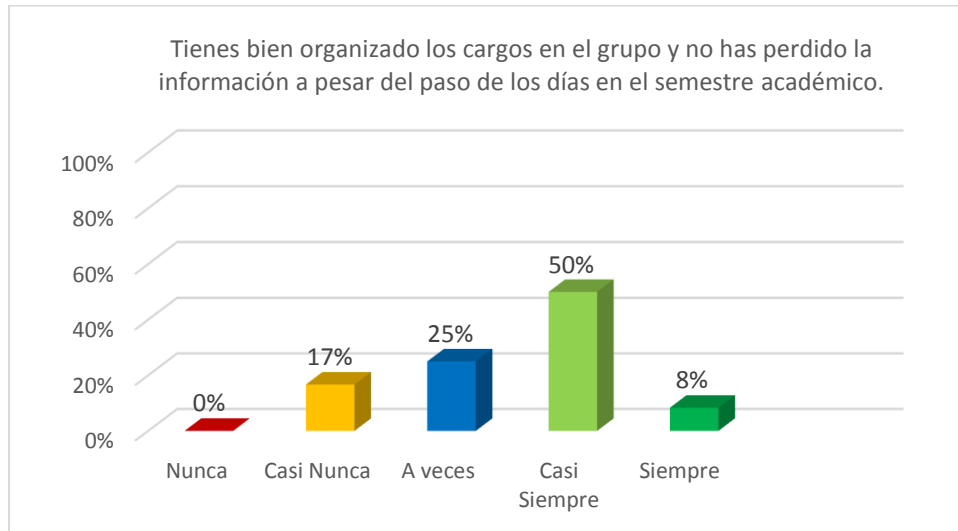


Gráfico 9. Pregunta 5, líderes que no usan el sistema

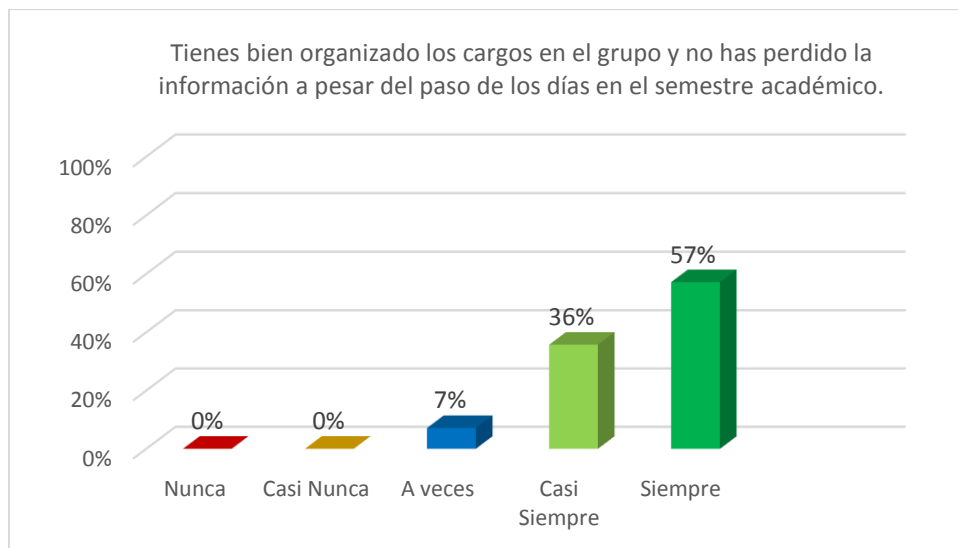


Gráfico 10. Pregunta 5, líderes que usan el sistema

Análisis: Con la implementación del sistema, la organización de los cargos en el grupo ha mejorado, se tiene la información disponible de cada responsabilidad. Un 58% (casi siempre y siempre) de los líderes que no usan el sistema tiene bien organizado su grupo, mientras que los que usan el sistema son un 93% (casi siempre y siempre).

Pregunta 6:

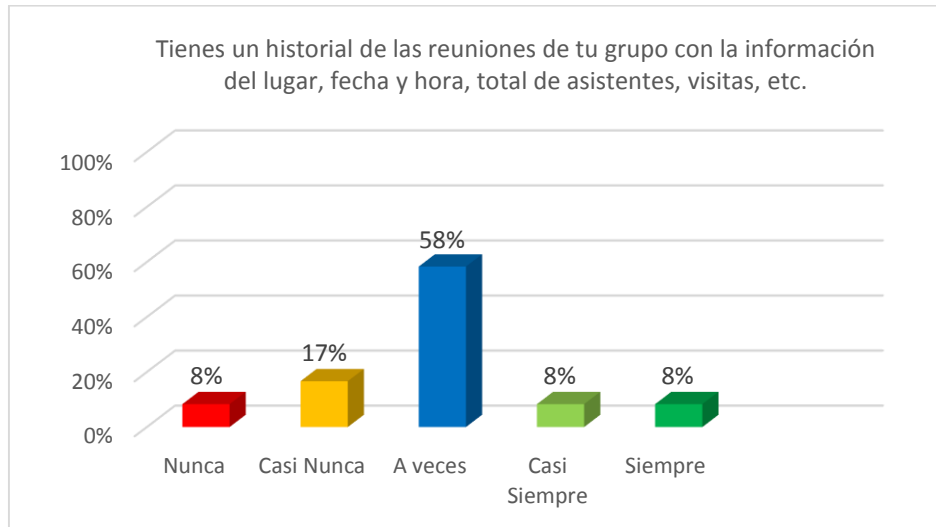


Gráfico 11. Pregunta 6, líderes que no usan el sistema

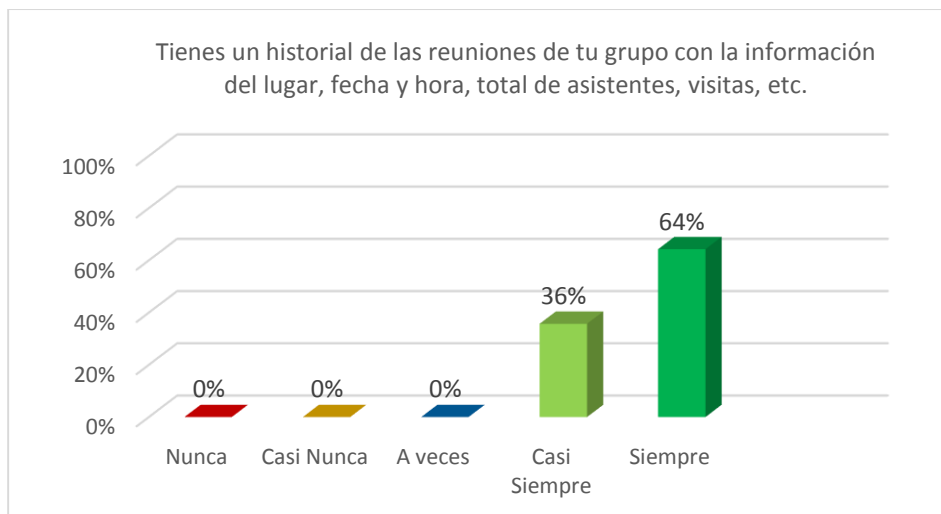


Gráfico 12. Pregunta 6, líderes que usan el sistema

Análisis: De los líderes que no usan el sistema solo el 16% tiene un historial de cada reunión organizada, y los líderes que usan el sistema han llegado a un 100% en su historial. El sistema cumple con sus propósitos de generar un historial de información.

Pregunta 7:

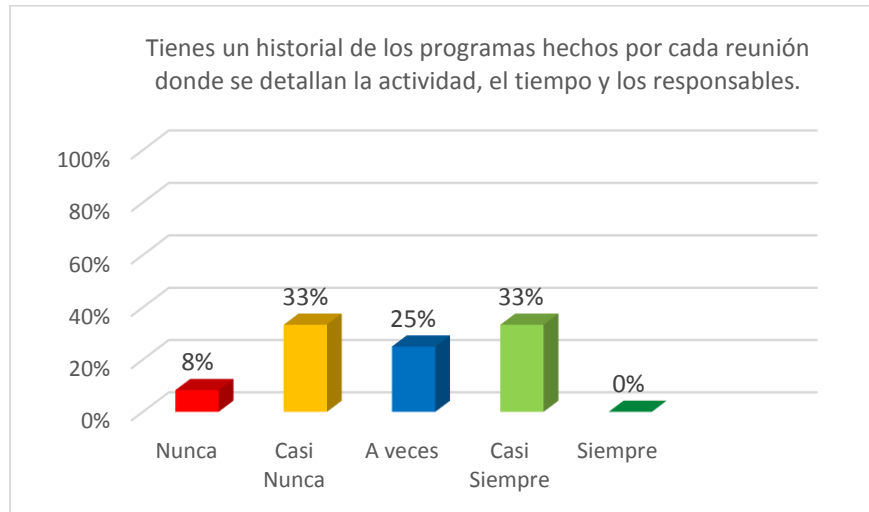


Gráfico 13. Pregunta 7, líderes que no usan el sistema

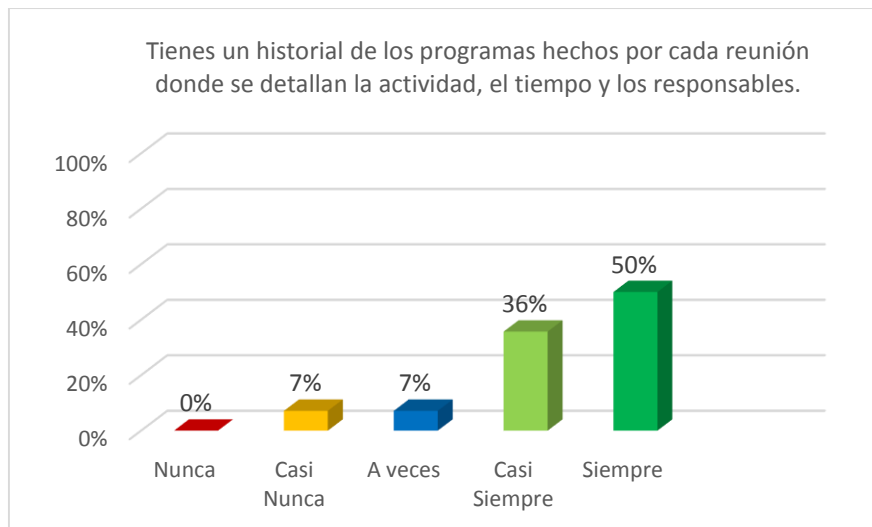


Gráfico 14. Pregunta 7, líderes que usan el sistema

Análisis: Según la perspectiva de los líderes, se tiene un historial del programa para cada reunión, y con el proceso automatizado ha mejorado en un 50%.

Pregunta 8:

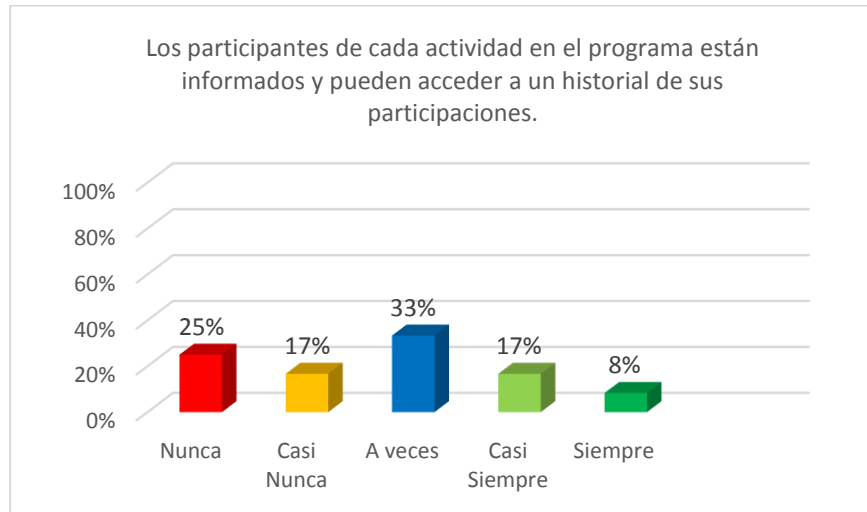


Gráfico 15. Pregunta 8, líderes que no usan el sistema

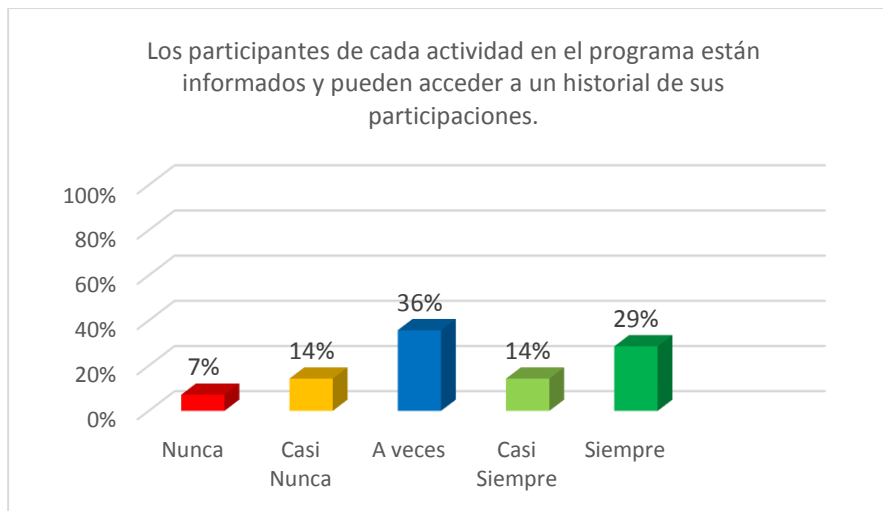


Gráfico 16. Pregunta 8, líderes que usan el sistema

Análisis: Ha mejorado en una mínima cantidad (20%) de participantes que están informados sobre sus participaciones en los programas que realizan los grupos.

Pregunta 9:

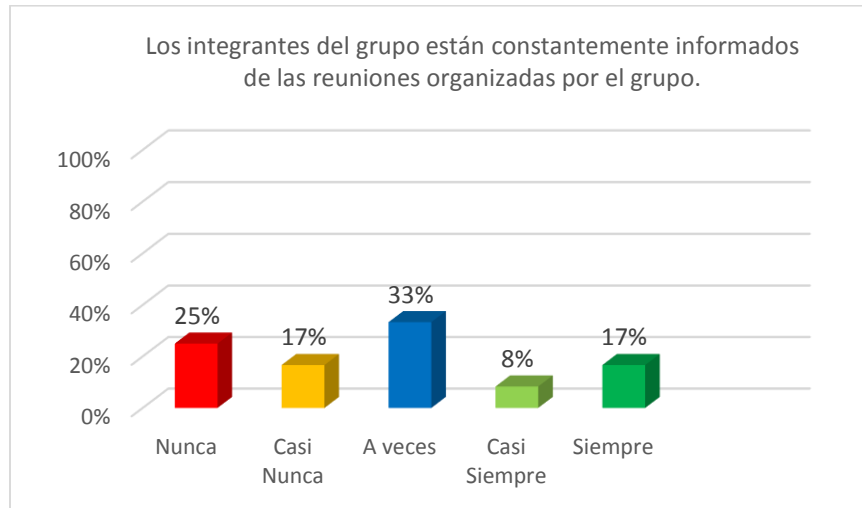


Gráfico 17. Pregunta 9, líderes que no usan el sistema

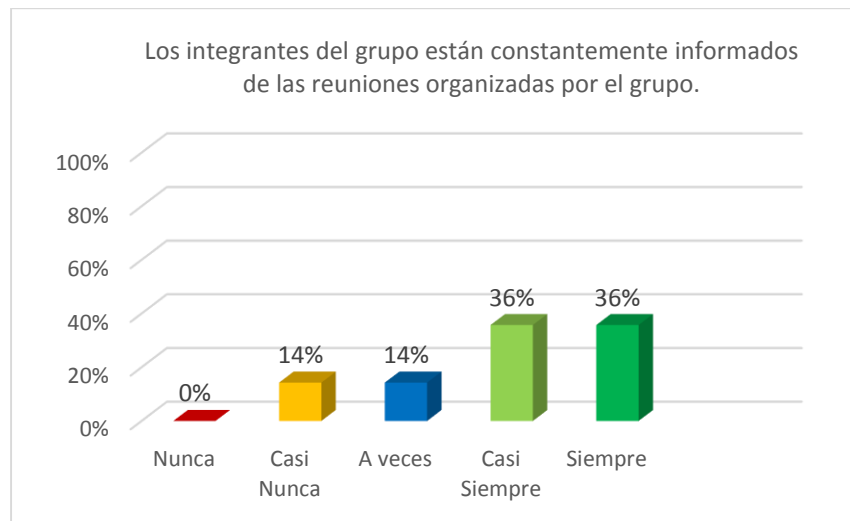


Gráfico 18. Pregunta 9, líderes que usan el sistema

Análisis: Con la implementación del sistema, ha mejorado 20% en los integrantes de cada grupo estén mejor informados que cada reunión organizada por cada grupo.

Pregunta 10:

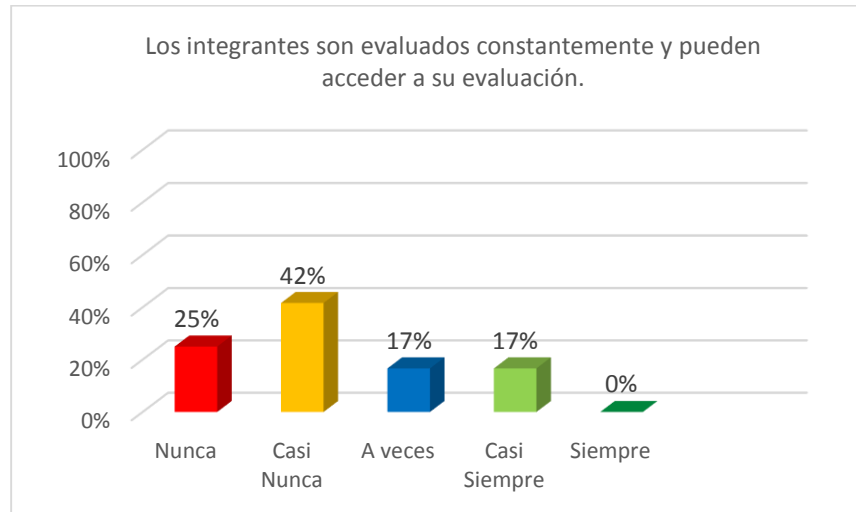


Gráfico 19. Pregunta 10, líderes que no usan el sistema

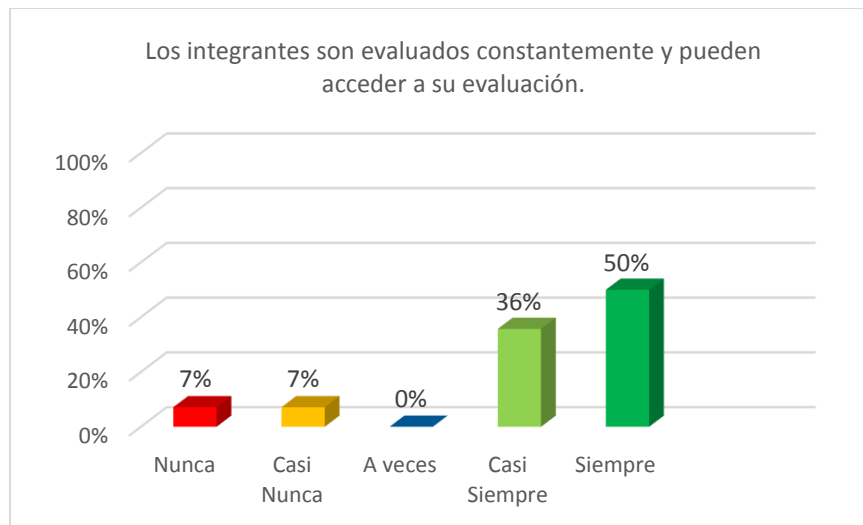


Gráfico 20. Pregunta 10, líderes que usan el sistema

Análisis: El proceso de evaluación ha mejorado en un 50% con respecto a lo anterior y los integrantes pueden ver sus evaluaciones y tomar decisiones de mejora.

5.2 Conclusiones

- En las primeras fases de este proyecto, se logró conocer la problemática existente resultado de la carencia de la información de los integrantes de los grupos categorizados por: alumnos, docentes, personal general, hermanos (as), visitantes, invitados, egresados, niños (as) que permite la elaboración de informes de trabajo, la gestión de los grupos organizados y el proceso de evaluación. El diagnóstico situacional de los grupos organizados permitió conocer la dinámica organizacional, que a su vez trajo como resultado una lista de requerimientos que llegaron a formar parte de las características del sistema de información. El análisis de los requerimientos dio como resultado un buen diseño de base de datos, que es la base para construir un sistema de calidad que satisfaga eficazmente las necesidades de los clientes.
- La aplicación de metodologías ágiles, como OpenUP para el desarrollo de proyectos de software, permite entregar el producto funcional en menor tiempo y de calidad, medido por la satisfacción de los clientes. La implementación por medio de fases y actividades traen como resultados documentos como: visión, arquitectura del sistema, de base de datos, manuales de usuario y manual de instalación son claves para obtener un producto de calidad. Además, las fases están bien definidas para poder aplicarlas, y la construcción de una forma interactiva permite en cada versión entregar software funcional previamente probado. El uso de estándares de calidad como la NTP ISO/IEC 12207 en su ciclo de desarrollo permitió validar y apoyar las actividades de metodología ágil.
- Las características que posee el sistema de información dio solución a la problemática identificada, porque satisface los requerimientos de los usuarios finales, permitiendo un buen desarrollo de los procesos de gestión y evaluación de los grupos organizados, con las

principales funcionalidades tales como: el registro de grupos, inscripción de integrantes, cargos, eventos, programas, asistencia, sub grupos, estudiantes de la Biblia y evaluaciones a los integrantes; además, brinda mecanismos de análisis y un sinfín de reportes que ayuda a la toma de decisiones a nivel de integrante, grupo, iglesia y universidad.

5.3 Recomendaciones y trabajos futuros

- Trabajar con los expertos del tema, para la definición de indicadores fiables que midan el desempeño de cada integrante de grupo y del grupo mismo. Esto ayudaría a cada persona trabajar en su mejoramiento personal, tales como puntualidad, participación, comportamiento y espiritualidad. En tal sentido, como trabajo futuro, se tendría que implementar un cuadro de mando para la toma de decisiones en tiempo real.
- Mejorar el diseño para dispositivos móviles. Adaptar los procesos de registro de integrantes y control de asistencia a reuniones, o hacer uso del nuevo framework de diseño web Materialize implementado por Google, ya que este se adapta muy bien en dispositivos móviles. Actualizar el framework “Bootstrap” a “Materialize” implementado por Google, para dar una mejor interoperabilidad en el uso del sistema.
- Implementar la funcionalidad para evaluar cada actividad de los participantes, midiendo así el grado de responsabilidad. También, se pueden añadir puntajes para que cada participante se sienta motivado hacia una mejora continua. Además, elaborar más reportes que permitan la toma de decisiones para la mejora de cada integrante. Implementar un chat interactivo, para mantenerse comunicados constantemente y, un mecanismo para compartir materiales basados en archivos entre líderes de la iglesia y los grupos organizados.

REFERENCIAS

- Arias, A. (2016). *Fundamentos de Programación y Bases de Datos*. (I. C. Academy, Ed.) (2da ed.). Retrieved from <https://books.google.com.pe/books?id=AyxpCwAAQBAJ>
- Arias, Á. (2014). *Aprende a programar con Ruby on Rails*. Retrieved from <https://books.google.com.pe/books?id=iU7mCgAAQBAJ>
- Aritzeta, A., & Ayestarán, S. (2003). Aplicabilidad Teoría Roles Belbin: Estudio Longitudinal. *Revista de Psicología General Y Aplicada*.
- Beati, H. (2016). *HTML5 y CSS3 - Para diseñadores*. (A. G. Editor, Ed.). México. Retrieved from <https://books.google.com.pe/books?id=kaywDAAAQBAJ>
- Benítez, M. A., & Arias, Á. (2015). *Curso de Introducción a la Administración de Bases de Datos*. (IT Campus Academy, Ed.). Retrieved from <https://books.google.com.pe/books?id=NUSiCgAAQBAJ>
- Canós, J., Letelier, P., & Penadés, C. (2010). Metodologías Ágiles en el Desarrollo de Software, 1–8.
- Cardona, P., & Wilkinson, H. (2006). Trabajo En Equipo. *IESE Business School*, 3. Retrieved from <http://www.iese.edu/research/pdfs/OP-07-10.pdf>
- Chazallet, S. (2015a). *Python 3: Los fundamentos del lenguaje*. Retrieved from <https://books.google.com.pe/books?id=JGjSrXIPPscC>
- Chazallet, S. (2015b). *Python 3: Los fundamentos del lenguaje*.
- Chmiel, M. (2013). JavaScript. In *Salem Press Encyclopedia*.
- Coronel, C. (2011). *Bases de Datos, Diseño, Implementación y Administración*. (C. L. E. L. Editores, Ed.).
- Cotos, J. M., & Taboada, J. A. (2005). *Sistemas de información medioambiental*. (Netbiblo, Ed.).
- Crovetto, C. (2004). *Programación en C# (Sharp)* (primera ed). Lima - Perú.
- DB-Engines, I. (2016). DB-Engines Ranking. Retrieved September 2, 2016, from <http://db-engines.com/en/ranking>
- Díaz, M., Gil, F. J., & Santana, J. P. (2014). Indicador de estilo del Aprendizaje Preferencial, 83–88.
- Eclipse, F. (2016). Open UP. Retrieved from <http://epf.eclipse.org/wikis/openup/index.htm>
- Eclipse, F. (2016). Open UP. Retrieved May 26, 2014, from https://www.eclipse.org/epf/openup_component/openup_vision.php
- Facebook. (2016). Conceptos básicos sobre los grupos. Retrieved from <https://es-es.facebook.com/help/162866443847527/>
- Fernández, A. (2012). *PYTHON 3 al descubierto*. (R. Libros, Ed.). Retrieved from <https://books.google.com.pe/books?id=PgXVfsUbtsQC>

- Fernández, V. (2010). *Desarrollo de sistemas de información: una metodología basada en el modelado*. (U. P. de Catalunya, Ed.). Barcelona, España.
- Flanagan, D., & Matsumoto, Y. (2008). *The Ruby programming Language*. Reilly Media.
- Fuentes, J. R. . (2015). *Desarrollo de Software Ágil: Extremme Programming y Scrum*. Retrieved from <https://books.google.com.pe/books?id=TxRpCwAAQBAJ>
- García, J., & André, M. (2015). Implementación del enfoque de reglas de negocio utilizando motores de reglas en el desarrollo de aplicaciones Java. *Ciencias de La Información*, 46(1), 41–46.
- Gimson, L. (2012). Metodologías ágiles y desarrollo basado en conocimiento Trabajo final integrador.
- Godoc, E. (2014). *SQL: Los fundamentos del lenguaje*. (Ediciones ENI, Ed.). Retrieved from <https://books.google.com.pe/books?id=thwzkdfIB-EC>
- Groussard, T. (2014a). *JAVA 8: Los fundamentos del lenguaje Java (con ejercicios prácticos corregidos)*. Retrieved from <https://books.google.com.pe/books?id=9zWQOpRm0AoC>
- Groussard, T. (2014b). *JAVA 8: Los fundamentos del lenguaje Java (con ejercicios prácticos corregidos)*.
- Herrera, E., & Valencica, L. E. (2007). Del manifiesto ágil sus valores y principios. *Scientia et Technica*, (34), 381–385.
- Heurtel, O. (2014). *PHP 5.5: Desarrollar un sitio Web dinámico e interactivo*. Retrieved from <https://books.google.com.pe/books?id=wYozPzHGnsUC>
- Indecopi, C. D. R. T. Y. C. (2006). Ntp-Iso / Iec 12207 Tecnología de La Información. *Norma Técnica Peruana*, (Lima 41).
- Jimenez, A., Fernández, I., & Hanco, D. (2015). *Entrevista Personal*. Tarapoto.
- Joyanes, L. (2003). *Fundamentos de programación*. (C. Fernández Madrid & A. Nieva, Eds.) (Tercera ed). Aravaca - Madrid (España).
- Kendall, J. (2005). *Análisis y diseño de sistemas*. Retrieved from <https://books.google.com.pe/books?id=5-rZA0FggusC>
- Kroenke, D. (2003). *Procesamiento de bases de datos: fundamentos, diseño e implementación* (Pearson Ed). Retrieved from http://books.google.com.pe/books?id=7ORUWItwcNEC&dq=historia+oracle+base+de+datos&hl=es&source=gbs_navlinks_s
- Lei, Z., Hu, W., & Zhou, H. (2016). Deployment of a Web-based Control Laboratory Using HTML5. *International Journal of Online Engineering*, 12(7), 18–24. Retrieved from <http://online-journals.org/index.php/i-joe/article/view/5819>
- López, G., Núñez, L., & Gonzáles, Y. (2011). Propuesta de Guía para los procesos de migración de datos hacia PostgreSQL mediante tecnologías libres. *Tono: Revista Técnica De La Empresa De Telecomunicaciones De Cuba, S.A*, 8(1), 30–33.

- Matsumoto, Y. (2015). Acerca de Ruby. Retrieved from <https://www.ruby-lang.org/es/about/>
- Mendoza, M., Durán, L., & Rivero, N. (2010). MBD 1.0 - Metodología de Desarrollo de Bodegas de Datos Para Micro, Pequeñas y Medianas Empresas. *UIS Ingenierías*, 9(1), 85–101.
- Microsoft. (2014). SQL Server. Retrieved from <http://www.microsoft.com/es-es/server-cloud/products/sql-server/#fbid=hSO-pZKV1he>
- Miranda, F., Chamorro, A., & Rubio, S. (2007). *Introducción a la gestión de la calidad*. Retrieved from <https://books.google.com.pe/books?id=KYSMQQyQAbYC>
- Oppel, A. (2010). *Fundamentos de Bases de Datos*. (F. Castellanos Rodriguez, M. Á. Luna Ponce, & Z. García García, Eds.) (Primera ed). Mexico, D.F.
- Palacio, J., & Ruata, C. (2011a). *Scrum Manager Gestión de proyectos*. Retrieved from <http://www.scrummanager.net>
- Palacio, J., & Ruata, C. (2011b). *Scrum Manager Gestión de proyectos*.
- Pérez, J., Carretero, J., García, F., & Pérez, J. M. (2006). *Problemas resueltos de programación en lenguaje C++*. Retrieved from <https://books.google.com.pe/books?id=hERLHWu5vGIC>
- Pressman, R. S. (2010). *Ingeniería de software un enfoque práctico*. (M. Á. Toledo Castellanos, P. Roig Vázquez, M. I. Rocha Martínez, M. T. Zapata Terrazas, & Z. García García, Eds.) (Septima ed). Mexico.
- Ramos, A. (2011). *APLICACIONES WEB (NOVEDAD 2011)*. (Editorial Paraninfo, Ed.). Retrieved from <https://books.google.com.pe/books?id=LXs3YIMoeNgC>
- Ramos, A., & Ramos, M. J. (2007). *Operaciones con bases de datos ofimáticas y corporativas*. (E. Paraninfo, Ed.) (Primera Ed). Retrieved from <http://books.google.com.pe/books?id=hwNWRHc79PMC>
- Ríos, S., Hinojosa, C., & Delgado, R. (2013). APLICACIÓN DE LA METODOLOGIA OPENUP EN EL DESARROLLO DEL SISTEMA DE DIFUSIÓN DE GESTIÓN DEL CONOCIMIENTO DE LA ESPE, 10. Retrieved from <http://repositorio.espe.edu.ec/bitstream/21000/6316/1/AC-SISTEMAS-ESPE-047042.pdf>
- Rodríguez, J. C., & Díaz, M. (2012). El Gestor de Coevaluación Orientado a Grupos. *Iberian Conference on Information Systems & Technologies*, 4. Retrieved from <http://web.a.ebscohost.com/ehost/pdfviewer/pdfviewer?sid=3590ad70-7e6a-49f8-971e-a9ba9939cb35@sessionmgr4001&vid=1&hid=4114>
- Rozo, S., & Abaunza, M. (2010). Liderazgo transaccional y transformacional. *Avances En Enfermería*, 28(2), 62–72. Retrieved from <http://search.ebscohost.com/login.aspx?direct=true&db=lth&AN=65553984&lang=es&site=ehost-live>
- Sabana, M. (2006). *Modelamiento e Implementación de Base de Datos*. (E. Aburto Correa, M. Sabana Mendoza, & G. Apolinario García, Eds.) (Primera ed). Lima, Perú.
- Santana, H., Chaves, J., Ferreira, C., De Oliveira, A., Mesquita, W., Moacir, M., ... Mendes, S.

- (2010). *Grupos pequeños, teoría y práctica* (1ra ed.). Buenos Aires, Argentina.
- Sebastián, P., Vilar, P., & Azzollini, S. (2013). Liderazgo , equipos y grupos de trabajo – su relación con la satisfacción laboral. *Revista de Psicología (PUCP)*, 31(1), 18. Retrieved from http://www.scielo.org.pe/scielo.php?script=sci_arttext&pid=S0254-92472013000100006&lang=es
- Sommerville, I. (2005). *INGENIERÍA DEL SOFTWARE*. (M. M. Romo, M. Caicoya, J. A. Clares, & J. A. Hernán, Eds.) (Septima Ed). Madrid (España): PEARSON EDUCACIÓN S.A.
- Thompson, M. (2012). Facebook. *EContent*, 35(6), 8–10. Retrieved from <http://web.a.ebscohost.com/ehost/pdfviewer/pdfviewer?sid=b56d5e5d-7a9c-4326-968a-5dd9bf5beed7%40sessionmgr4004&vid=1&hid=4101>
- TIOBE. (2016). TIOBE Index for August 2016. Retrieved September 2, 2016, from <http://www.tiobe.com/tiobe-index/>
- Tomcat, A. (2014). Apache Tomcat. Retrieved from <http://tomcat.apache.org/>
- Valderama, S. (2004). Programación extrema en pocos minutos: planificando la transición. *Revista Técnica de La Empresa de Telecomunicaciones de Cuba S.A.*, 41–45.
- Vikram, V. (2009). *Fundamentos de PHP*. (F. Castellanos Rodriguez, M. Á. Luna Ponce, & S. García García, Eds.) (Primera ed). Mexico.
- White, E. (1983). *Servicio Cristiano*. Buenos Aires, Argentina.
- Yunkella, K., & Bryden, A. (2010). *Organismos Nacionales de Normalización en Países en Desarrollo. Progresar Rapidamente*.

ANEXOS

ENCUESTA PARA MEDIR EL NIVEL DE EFECTIVIDAD EN LOS PROCESOS DE GESTIÓN Y EVALUACIÓN DE LOS GRUPOS ORGANIZADOS EN LA SITUACIÓN ACTUAL

Hola mi nombre es Persy Quiroz Menor, egresado de la carrera de Ingeniería de Sistemas de la Universidad Peruana Unión Filial Tarapoto, lo cual permitirá conocer el nivel de efectividad en la gestión de los grupos organizados en la situación actual que ayudará a mi proyecto de investigación. Está dirigida a los líderes de grupos pequeños de la Iglesia Universitaria Villa Unión.

Indicaciones: Marque con (x) de una respuesta por cada pregunta presentada a continuación.

#	ÍTEM DE EVALUACIÓN (INDICADOR)	NUNCA	CASI NUNCA	A VECES	CASI SIEMPRE	SIEMPRE
1	Dispone de la información de los datos de cada integrante de su grupo, tales como ciclo de estudio, carrera, correo, edad, etc.					
2	Tienes una lista actualizada de tus integrantes del grupo.					
3	Puedes identificar a tus integrantes fácilmente.					
4	Tienes la información disponible de los ideales de tu grupo (lema, canto, blanco, etc.).					
5	Tienes bien organizado los cargos en el grupo y no has perdido la información a pesar del paso de los días en el semestre académico.					
6	Tienes un historial de las reuniones de tu grupo con la información del lugar, fecha y hora, total de asistentes, visitas, etc.					
7	Tienes un historial de los programas hechos por cada reunión donde se detallan la actividad, el tiempo y los responsables.					
8	Los participantes de cada actividad en el programa están informados y pueden hacer a su historial de participaciones					
9	Los integrantes del grupo se están constantemente informados de las reuniones de los grupos.					
10	Tienes un historial de estados de asistencia de tus integrantes a las reuniones de grupo y ellos pueden acceder a dichos informes.					
11	Los integrantes son evaluados constantemente y pueden acceder a su evaluación.					
12	Tienes un álbum histórico de fotos de las reuniones en donde todos pueden acceder y recordar a pesar del paso de los semestres.					
13	Dispones de la información de las parejas misioneras en tu grupo y cada integrante conoce la organización.					
14	Tienes un control de los estudiantes de la Biblia que estudian con las parejas misioneras					
15	Tienes registrado las metas del grupo en la Escuela Sabática.					
16	Tus integrantes conocen las metas y procesos de evaluación del termómetro de escuela sabática (tabla comparativa).					
17	Tienes un control de tus informes de asistencia y metas como grupo					
18	Tienes un historial de cada integrante que se detallan sus estados de asistencia, cargos que desempeñan y participaciones por cada ciclo académico.					
19	Sientes que los líderes de la iglesia, pastores y administradores de la universidad están comprometidos en el proceso de evaluación para la mejora de tu grupo.					
20	Tienes mecanismos de análisis de asistencias a cada reunión de tu grupo					

Anexo 1. Instrumento (encuesta) aplicado a líderes, para conocer la situación actual

ENCUESTA PARA MEDIR EL NIVEL DE EFECTIVIDAD EN LOS PROCESOS DE GESTIÓN Y EVALUACIÓN DE LOS GRUPOS ORGANIZADOS TRAS LA IMPLEMENTACIÓN DEL SISTEMA DE INFORMACIÓN

Hola mi nombre es Persy Quiroz Menor, egresado de la carrera de Ingeniería de Sistemas de la Universidad Peruana Unión Filial Tarapoto, lo cual permitirá conocer el nivel de efectividad en la gestión de los grupos organizados tras la implementación del sistema de información. Está dirigida a los líderes de grupos pequeños de la Iglesia Universitaria Villa Unión, que usan el sistema de información para gestionar su grupo.

Indicaciones: Marque con (x) de una respuesta por cada pregunta presentada a continuación.

#	ÍTEM DE EVALUACIÓN (INDICADOR)	NUNCA	CASI NUNCA	A VECES	CASI SIEMPRE	SIEMPRE
1	Dispone de la información de los datos de cada integrante de su grupo, tales como ciclo de estudio, carrera, correo, edad, etc.					
2	Tienes una lista actualizada de tus integrantes del grupo.					
3	Puedes identificar a tus integrantes fácilmente.					
4	Tienes la información disponible de los ideales de tu grupo (lema, canto, blanco, etc.).					
5	Tienes bien organizado los cargos en el grupo y no has perdido la información a pesar del paso de los días en el semestre académico.					
6	Tienes un historial de las reuniones de tu grupo con la información del lugar, fecha y hora, total de asistentes, visitas, etc.					
7	Tienes un historial de los programas hechos por cada reunión donde se detallan la actividad, el tiempo y los responsables.					
8	Los participantes de cada actividad en el programa están informados y pueden hacer a su historial de participaciones					
9	Los integrantes del grupo se están constantemente informados de las reuniones de los grupos.					
10	Tienes un historial de estados de asistencia de tus integrantes a las reuniones de grupo y ellos pueden acceder a dichos informes.					
11	Los integrantes son evaluados constantemente y pueden acceder a su evaluación.					
12	Tienes un álbum histórico de fotos de las reuniones en donde todos pueden acceder y recordar a pesar del paso de los semestres.					
13	Dispones de la información de las parejas misioneras en tu grupo y cada integrante conoce la organización.					
14	Tienes un control de los estudiantes de la Biblia que estudian con las parejas misioneras					
15	Tienes registrado las metas del grupo en la Escuela Sabática.					
16	Tus integrantes conocen las metas y procesos de evaluación del termómetro de escuela sabática (tabla comparativa).					
17	Tienes un control de tus informes de asistencia y metas como grupo					
18	Tienes un historial de cada integrante que se detallan sus estados de asistencia, cargos que desempeñan y participaciones por cada ciclo académico.					
19	Sientes que los líderes de la iglesia, pastores y administradores de la universidad están comprometidos en el proceso de evaluación para la mejora de tu grupo.					
20	Tienes mecanismos de análisis de asistencias a cada reunión de tu grupo					

Anexo 2. Instrumento (encuesta) aplicado a líderes, para conocer em mejoramiento de la gestión, tras la implemnetación del sistema de información

RECONOCIMIENTO

El Pastor de la IGLESIA UNIVERSITARIA VILLA UNIÓN – UPeU
TARAPOTO, quien suscribe;

RECONOCE:

Que, **Persy Quiroz Menor**, bachiller de la carrera de Ingeniería de Sistemas, por su aporte con su trabajo de investigación denominado *"Implementación de un sistema de información bajo la metodología OpenUP para la mejora de la gestión y evaluación de los grupos organizados de la iglesia Universitaria Villa Unión de la Universidad Peruana Unión Filial Tarapoto"*.

Dicho trabajo ha cumplido con los objetivos del proyecto en mención que es de gran ayuda para nuestra Iglesia Universitaria. Integra los módulos de inscripción de grupos pequeños, inscripción de integrantes, programación de reuniones, elaboración de programas, control de asistencias, evaluación a los asistentes, entre otros. Se rescata la emisión de reportes que brinda el sistema, ayudando a la toma de decisiones a nuestra iglesia.

Este documento es válido para los fines que se requiera.

Morales, 10 de diciembre 2015



Melbin Mamani Quispe
Pastor Iglesia Universitaria Villa Unión
UPeU/FT



Jr. Los Mártires 218 – Sta Lucía
- Morales.
Telef. 521414 / 525537
Fax: 529450
Tarapoto, Perú

RECONOCIMIENTO

El Capellán de la **FACULTAD DE INGENIERÍA Y ARQUITECTURA – UPeU TARAPOTO**, quien suscribe;

RECONOCE:

Que, **Persy Quiroz Menor**, bachiller de la carrera de Ingeniería de Sistemas, por su aporte con su trabajo de investigación denominado *“Implementación de un sistema de información bajo la metodología OpenUP para la mejora de la gestión y evaluación de los grupos organizados de la iglesia Universitaria Villa Unión de la Universidad Peruana Unión Filial Tarapoto”*.

Dicho trabajo ha cumplido con los objetivos del proyecto en mención que es de gran ayuda para nuestra facultad. Integra los módulos de inscripción de grupos pequeños, inscripción de integrantes, programación de reuniones, elaboración de programas, control de asistencias, evaluación a los asistentes, entre otros. Se rescata la emisión de reportes que brinda el sistema, ayudando a la toma de decisiones a nuestra facultad.

Este documento es válido para los fines que se requiera.

Morales, 10 de diciembre 2015



Cesar Augusto Valladolid Suarez
Capellán de la Facultad de Ingeniería y Arquitectura

Anexo 4. Reconocimiento de capellanía de la Facultad de Ingeniería y Arquitectura



Jr. Los Mártires 218 – Sta Lucía
- Morales
Telef. 521414 / 528537
Fax: 529450
Tarapoto, Perú

RECONOCIMIENTO

El Capellán de la **FACULTAD DE CIENCIAS DE LA SALUD – UPeU TARAPOTO**, quien suscribe;

RECONOCE:

Que, **Persy Quiroz Menor**, bachiller de la carrera de Ingeniería de Sistemas, por su aporte con su trabajo de investigación denominado *"Implementación de un sistema de información bajo la metodología OpenUP para la mejora de la gestión y evaluación de los grupos organizados de la iglesia Universitaria Villa Unión de la Universidad Peruana Unión Filial Tarapoto"*.

Dicho trabajo ha cumplido con los objetivos del proyecto en mención que es de gran ayuda para nuestra facultad. Integra los módulos de inscripción de grupos pequeños, inscripción de integrantes, programación de reuniones, elaboración de programas, control de asistencias, evaluación a los asistentes, entre otros. Se rescata la emisión de reportes que brinda el sistema, ayudando a la toma de decisiones a nuestra facultad.

Este documento es válido para los fines que se requiera.

Morales, 10 de diciembre 2015

Lic. Daniel William Hancoco Pantoja
Capellán de la Facultad de Ciencias de la Salud

Anexo 5. Reconocimiento de capellanía de la Facultad de Ciencias de la Salud

APÉNDICES

Apéndice A

SISTEMA DE GRUPOS ORGANIZADOS | SGO
Documento Visión de la aplicación

VERSIÓN 1.0

1. Historia de Revisión.

Fecha	Versión	Descripción	Autor (es)
15-Abr-2015	1.0	Creación del documento de visión	Persy Quiroz Menor
27-Abr-2015	1.1	Actualización del documento	Persy Quiroz Menor

- **Propósito:** Este documento visión tiene como propósito recolectar, analizar y describir las necesidades para la construcción del Sistema de Grupos Organizados todo descrito en alto nivel. Se especifican a los stakeholders y los usuarios principales de la aplicación, detallando las necesidades de cada uno de ellos encontrando las razones que justifican el proyecto. Se mencionan además las características de la aplicación, que incluyen restricciones y criterios de aceptación aplicables al caso.
- **Nombre de la aplicación:** Sistema de Grupos Organizados (SGO)
- **Posicionamiento**

El problema de	Muchos personas que asisten a los grupos y que no se tiene un buen registro de <i>inscritos</i> por grupo organizado. La carga manual de la información de cada integrante tiene un impacto significativo en tiempo.
Afecta a	<ol style="list-style-type: none"> 1. Organizadores de los grupos 2. Líderes de los grupos organizados. 3. Integrantes de los grupos 4. Pastor de iglesia 5. Capellanes de facultad 6. Consejeros de los grupos 7. Administradores de la institución.
Cuyo impacto es	<ol style="list-style-type: none"> a) No llevar un registro automatizado de los inscritos en cada grupo organizado, esto lleva no a una buena organización. b) No se pueden obtener reportes exactos de los inscritos por facultad, carrera, sexo, religión , etc.
Una solución exitosa	Inscribir a las personas en cada grupo de manera automatizada, con todos los datos requeridos para la emisión de los reportes que ayude a cada grupo y cada integrante en pro de una mejora continua.

El problema de	Gran cantidad de asistentes a los eventos de los grupos que no se tiene un buen control de asistencia.
Afecta a	<ol style="list-style-type: none"> 1. Organizadores de los eventos (Grupos pequeños, Escuela Sabática, entre otros). 2. Líderes de los grupos organizados. 3. Integrantes de los grupos 4. Pastor de iglesia 5. Capellanes de facultad 6. Consejeros de los grupos
Cuyo impacto es	<ol style="list-style-type: none"> a) No llevar un registro automatizado de la asistencia registrado por estados (Puntual, Tardanza, Falta, Falta Justificada). b) No se pueden obtener reportes automatizados y un análisis de asistencias.

Una solución exitosa	Registrar la asistencia de integrantes y visitas a los eventos programados por cada grupo organizado. Con esta información se obtendrán reportes en tiempo real, y ayudaría motivando la asistencia de cada integrante.
-----------------------------	---

El problema de	Muchas participaciones de las personas en cada evento del grupo, que no tiene un historial y una programación anticipada.
Afecta a	<ol style="list-style-type: none"> 1. Participantes de cada evento. 2. Pauteadores de cada evento (reunión, etc.). 3. Líderes de los grupos organizados. 4. Capellanes de facultad 5. Consejeros de los grupos 6. Encargados de multimedia del grupo
Cuyo impacto es	<ol style="list-style-type: none"> a) No tener una programación anticipada por cada reunión que describa la actividad, el tiempo y los responsables. b) Genera muchos imprevistos en el evento, sin tener una preparación de cada participante.
Una solución exitosa	Registrar la programación de cada evento de manera anticipada, así mismo publicar la programación y avisar a cada participante por medio de un mensaje al celular o correo personal. Tener un historial de participaciones de manera automatizada.

El problema de	La carga manual del proceso de evaluación de indicadores a grupos organizados tiene un costo en tiempo muy significativo.
Afecta a	<ol style="list-style-type: none"> 1. Evaluadores de indicadores a los grupos organizados. 2. Líderes de la Escuela Sabática y otros. 3. Capellanes de facultad
Cuyo impacto es	a) El no tener un proceso de evaluación automatizado significa depender netamente de los evaluadores, en tal caso que no lo hagan se desconocería. Además no hay disponibilidad de manera general a todo el resultado de las evaluaciones.
Una solución exitosa	Tener un registro automatizado de las evaluaciones de los indicadores a cada grupo organizado y que cada grupo lo haga, sin depender de los evaluadores. Tales evaluadores solo se dedicarán a analizar los resultados y tomar planes de mejora.

El problema de	No se tienen evidencias de los eventos organizados por medio de fotos.
Afecta a	<ol style="list-style-type: none"> 1. Pastor de iglesia 2. Capellanes de facultad 3. Líderes de iglesia 4. Administradores de la institución.
Cuyo impacto es	a) No poder monitorear a todos los grupos organizados, sin conocer las actividades de manera general de cada uno de ellos.

Una solución exitosa	Tener un registro automatizado de fotos por cada evento realizado, la asistencia, el programa hecho y las evaluaciones de indicadores.
-----------------------------	--

El problema de	Pérdida de información de cada grupo y la lista de cargos, para las coordinaciones por cada departamento de la iglesia.
Afecta a	<ol style="list-style-type: none"> 1. Pastor de iglesia 2. Capellanes de facultad 3. Líderes de iglesia 4. Líderes de la iglesia. 5. Integrantes del grupo
Cuyo impacto es	a) Desconocimiento de la organización de cada grupo, por ende trae como consigo el incumplimiento de las responsabilidades de cada responsable.
Una solución exitosa	Tener un registro automatizado de los datos del grupo y la organización con sus cargos respectivos organizados por departamento.

El problema de	Falta de información de las parejas misioneras en cada grupo, lista de estudiantes y control de estudios Bíblicos
Afecta a	<ol style="list-style-type: none"> 1. Pastor de iglesia 2. Capellanes de facultad 3. Director de ministerio personal de la iglesia 4. Estudiantes de la Biblia 5. Parejas misioneras
Cuyo impacto es	a) Información inexacta de la actividad misionera de cada grupo organizado.
Una solución exitosa	Poseer un registro automatizado de las parejas misioneras de los grupos organizados, lista de estudiantes de la Biblia y el control de los estudios bíblicos.

El problema de	No tener una tabla comparativa (termómetro) por grupo y un análisis de por cada persona.
Afecta a	<ol style="list-style-type: none"> 1. Pastor de iglesia 2. Capellanes de facultad 3. Directores de las Escuelas Profesionales 4. Director de Escuela Sabática
Cuyo impacto es	a) Información inexacta de la actividad misionera de cada grupo organizado.
Una solución exitosa	Tener una tabla comparativa de los indicadores de manera automática que permita obtener resultados de la situación de cada grupo organizado, y el análisis de la situación de cada persona inscrita en un grupo.

El problema de	No tener reportes importantes que ayude a la toma de decisiones basado en los grupos organizados y los integrantes de los mismos.
Afecta a	1. Pastor de iglesia 2. Capellanes de facultad 3. Administradores de la institución.
Cuyo impacto es	a) No se pueden emitir informes confiables de la organización de los grupos organizados y todos los procesos que realizan.
Una solución exitosa	Poseer en el sistema una lista de opciones con diferentes reportes tales como organización, asistencia, programación, participaciones, evidencias, evaluaciones, de cada grupo e integrante.

Para	Líderes de grupos organizados, pastor de iglesia, capellanes de facultad, coordinadores de grupos organizados, administrativos, integrantes de los grupos, entre otros.
Quienes	Requieren mecanismos que automaticen la gestión de los grupos organizados desde la inscripción hasta los procesos de evaluación.
SGO	Es la Aplicación Web (Sistema de Grupos Organizados)
Qué	<ul style="list-style-type: none"> - Registra los datos de cada persona que participa de los grupos organizados. - Permite realizar el proceso de inscripción de los integrantes al grupo. - Provee una lista de completa con los datos de los inscritos, lo que incluye datos personas, de contacto y académicos. - Permite el registro de los datos del grupo y los cargos. - Permite registra los eventos de cada grupo, la programación del evento, asistencia y la evaluación a los asistentes. - Permite el control de parejas misioneras, estudiantes de la Biblia y estudios bíblicos. - Permite estadísticas de asistencia por periodo, facultad, tipo de grupo y evento. - Genera historial y reportes con formato para la impresión, entre muchas opciones.
Se diferencia de	El proceso actual (registro de algunos procesos manualmente).
Nuestra Aplicación	<ul style="list-style-type: none"> - Ofrece una infraestructura web con capacidad de soportar operaciones desde cualquier navegador con acceso a internet. - Ha sido construido con tecnologías modernas capaz de adaptarse a dispositivos móviles. - Se adapta a las necesidades de los usuarios finales con capacidad de integrar nuevos módulos.

Descripción de los Stakeholders y Usuarios

Representatividad	Persy Quiroz Menor
Categoría	Desarrollador
Tipo	Desarrollador de software
Responsabilidad	Analizar, diseñar, construir y probar el sistema
Criterios de éxito	Sistema que cumple todos los requisitos y cubre las necesidades del cliente definidas en el proyecto.
Nivel de participación	Implementa el software y realiza el mantenimiento en la aparición de cambios.
Entregables	Software (Código fuente y compilado) y documentación.
Comentarios / otros aspectos	

Representatividad	Mebin Mamani Quispe, Israel Fernández, Daniel Hanco y Cesar Valladolid
Categoría	Pastor de iglesia y capellanes de facultad
Tipo	Organizador de grupos
Responsabilidad	Establecen algunos requisitos de la aplicación y determina el cumplimiento de los mismos.
Criterios de éxito	El sistema registrar los datos de las personas, registrar los grupos y permitir la inscripción de los integrantes.
Nivel de participación	Revisan las funcionalidades de inscripción de datos de las personas, de los grupos y la inscripción. Registan a los grupos organizados por facultad.
Entregables	Formatos de inscripción de los grupos e integrantes y formatos diversos.
Comentarios / otros aspectos	

Representatividad	Abraham Jiménez Milla
Categoría	Administrador
Tipo	Administrador principal de la gestión de grupos
Responsabilidad	Definir requisitos del sistema y aprobar las versionar presentadas
Criterios de éxito	El sistema debe emitir reportes que permita mejorar la gestión de los grupos. Eliminando problemas por falta de información.

Nivel de participación	Valida las versiones del sistema y aprueba el ingreso de la versión a producción.
Entregables	Cartas de aceptación.
Comentarios / otros aspectos	

Representatividad	Saulo Salinas Arias
Categoría	Usuario
Tipo	Director de MiPES
Responsabilidad	Proveer información de la dinámica de las actividades misioneras de la iglesia en grupo organizados.
Criterios de éxito	El sistema debe registrar parejas misioneras, gestionar la lista de estudiantes de la Biblia y el control de estudios bíblicos.
Nivel de participación	Valida el módulo de MiPES y recomienda nuevos cambios.
Entregables	Formatos de parejas misioneras, estudiantes de la Biblia y control de estudios bíblicos
Comentarios / otros aspectos	

Representatividad	Margarita Huancas, Jhan Arly Sánchez, y los 51 líderes de los grupos organizados.
Categoría	Usuarios
Tipo	Líderes de Grupos Pequeños
Responsabilidad	Probar el sistema relacionado con la inscripción, asistencia y programación del evento
Criterios de éxito	El sistema debe cumplir con las necesidades de los líderes permitiendo una buena gestión de sus grupos.
Nivel de participación	Valida el módulo de gestión de los grupos.
Entregables	No aplica
Comentarios / otros aspectos	El líder es la pieza fundamental porque es un tipo de usuario que provee de la mayor información para la gestión de los grupos.

- **Tabla de necesidades**

Nota: La prioridad del negocio va desde de 1 al 5 de mayor a menor.

Necesidad	Prioridad del Negocio	Problema que origina la necesidad	Solución actual	Soluciones propuestas
Se requieren registrar los datos de todos los grupos organizados	1	Datos desactualizados de los grupos con sus líderes y no está disponible para todos los interesados	El sistema que se usa es manualmente en hojas de Excel.	La aplicación ofrece el módulo de inscripción de grupos con todos los datos requeridos y genera la lista para impresiones
Se requiere inscribir a los integrantes de los grupos organizados con todos sus datos que permita a partir de ellos emitir reportes.	2	Proceso de inscripción lento, data desactualizada de cada integrante	La inscripción se hace manualmente por parte de los líderes. Los capellanes reciben la lista.	Inscripción de integrantes de manera automatizada, se registran los todos datos necesarios de cada integrante.
Se requiere obtener la lista de cargos en cada grupo	3	Falta de información de la organización interna de cada grupo, y se desconoce la información de cada responsable	Se registran mininamente solo algunos cargos básicos manualmente.	Registro automatizado de los cargos por cada grupo realizado por el líder. El pastor de iglesia configuración la estructura de los cargos y los líderes solo seleccionan los responsables.
Se requiere registrar cada evento realizado por grupo organizado	2	Falta de información de los eventos por cada grupo, esto trae consigo desconocimiento de la situación de cada uno de ellos	Solo algunos líderes registrar en algún lugar su reunión y la publicación es muy parcial	Se registra de manera automatizada las reuniones de cada grupo, indicando el título, lugar, hora, resumen y descripción.

Necesidad	Prioridad del Negocio	Problema que origina la necesidad	Solución actual	Soluciones propuestas
Se requiere tener formatos de las partes de la programación para cada evento, el tiempo y los responsables	2	Poco conocimiento de la programación de cada evento, trae consigo improvisaciones y falta de preparación	Pocos líderes publican las programación de cada evento	El sistema permitirá la elaboración de los programas cada tipo de evento con todas sus partes, se asigna cada actividad, descripción, sugerencias, tiempo y responsables.
Se requiere registrar la asistencia de cada integrante y visitas a cada evento realizado, la asistencia lleva los siguientes estados: puntual, tardanza, falta y falta justificada	2	Falta de información de la asistencia de las personas a los eventos organizados por cada grupo	Solo algunas reuniones se registran y se hacen manualmente	Por medio del sistema de llevará un control eficiente de la asistencia de cada persona, permitiendo los estados de puntual, tardanza, falta y falta justificada. Se selecciona el tipo de asistencia y se agrega alguna descripción. Además permite el control de asistencia con solo su código.
Se requiere evaluar cada persona por tipo de evento realizado	3	Carencia de indicadores fiables que puedan medir el buen desempeño de cada persona en los grupos organizados	Actualmente se hace manualmente y se usa una tabla comparativa para medir cada indicador, pero esto siempre depende de los organizados, sino se da énfasis fracasa	El sistema permitirá configurar los rubros de evaluación por tipo de evento, cada persona es evaluada al momento mismo del control de asistencia. Con dicha evaluación el sistema provee reportes de la misma, incluyendo asistencia, participaciones e incidencias.

Necesidad	Prioridad del Negocio	Problema que origina la necesidad	Solución actual	Soluciones propuestas
Se requiere llevar el control de la actividad misionera en cada grupo y a nivel de iglesia. Permitted registrar parejas misioneras, estudiantes de la Biblia y control de los estudios bíblicos.	3	No se tiene un buen control de los estudiantes de la Biblia, frialdad en la actividad misionera de cada grupo. El desconocimiento del mismo trae consigo un monitoreo inestable.	Se distribuyen formato del control de los estudios y estudiantes de la Biblia, pero no se monitorea fielmente.	El sistema permitirá registrar a todas las parejas misioneras, ellas mismas cargarán sus estudiantes de la Biblia y los estudios Bíblicos. El sistema emitirá un reporte del consolidado de los avances de los estudios bíblicos en tiempo real.
Se requiere tener reportes de asistencia, participación y evaluación de cada integrante del grupo y a nivel de grupo	3	Falta de una ficha histórica por cada integrante de grupo. No se pueden distinguir con facilidad a nuevos líderes basados en su desempeño dentro del grupo.	No se cuenta con un histórico por cada integrante del grupo, lo que hace manualmente es difícil obtenerlo. No se tiene mecanismos de monitoreo para cada grupo.	El sistema obtiene ese histórico de asistencias, participaciones, evaluaciones e incidencias por cada persona, además por medio de gráficos se puede notas el desempeño de cada persona en el grupo. También permite el monitoreo a cada grupo dando un análisis de asistencia por medio de gráficos.
Se requieren reportes fiables para los informes que emite la iglesia sobre los grupos organizados.	3	Falta de información fiable para la emisión de informes	Los mecanismos actualmente son manuales y no son fiables.	El sistema provee un sinfín de reportes de la dinámica de los grupos basados en los procesos que realizan.

- **Descripción de la Aplicación:** La aplicación es un sistema de información con plataforma web, que va a ser utilizado desde cualquier sistema operativo con navegador web. Provee los mecanismos para la gestión de los grupos organizados tales como registro de grupos, inscripción de integrantes, registro de eventos, elaboración de programas de cada evento, control de asistencia, evaluación de asistentes, control de parejas misioneras, entre otras opciones adicionales. Permite la generación de reportes para su publicación, que se pueden hacer uso en informes, y sobre todo ayuda en la toma de decisiones importantes de la iglesia y la institución cristiana.

- **Restricciones:**
 - El sistema es capaz de trabajar con el servidor de aplicaciones Apache Tomcat y usa el motor de base de datos PostgreSQL.
 - Necesita el uso de internet o intranet de la institución para el acceso a la aplicación.

- **Requerimientos de documentación:** Se deberá desarrollar un manual de usuario para el entrenamiento de nuevos usuarios, donde se describe la interacción con cada módulo del sistema.

- **Glosario de la Aplicación**

Término	Definición
SGO	Sistema de Grupos Organizados
IASD Universitaria Villa Unión	Iglesia Adventista del Séptimo Día – Universitaria Villa Unión, iglesia donde se aplicó la investigación. Pertenece a la Universidad Peruana Unión – Filial Tarapoto.
UPeU FT	Universidad Peruana Unión Filial Tarapoto, institución de la Iglesia Adventista.
Capellán de facultad	Líder del área espiritual por cada facultad. Existen actualmente tres capellanes, uno para cada facultad. Las facultades son las siguientes: Facultad de Ingeniería y Arquitectura, Facultad de Ciencias Empresariales y Facultad de Ciencias de la Salud. El capellán es la persona de organizar los grupos y registrar en el sistema. Y monitorea a cada grupo de su facultad.
Líder de grupo	Persona que representa a cada grupo organizado. Es la que registra a cada integrante, registra los eventos y controla la asistencia.
Pastor de iglesia	Coordinan la organización de los grupos juntamente con los capellanes de facultad. Monitoreo a todos grupos de manera genérica.
Coordinador de grupos pequeños	Persona encargada de coordinar las reuniones de los grupos organizados, trabaja de antemano con los capellanes y el pastor de iglesia
Director de Escuela Sabática	Persona encargada que organizar y distribuir las escuelas sabáticas en la iglesia, trabaja juntamente con los capellanes y pastor de iglesia.
Director de MiPES	Persona encargada de la organización de la parejas misioneras y el monitoreo de la cada actividad misionera de la iglesia.
Director de Bienestar Universitario	Persona encargada de monitorear la situación actual de cada miembro del grupo organizado que estudia en la universidad. Juntamente con los capellanes aprueban las becas por fe ligresía adventista, tras una evaluación de los criterios establecidos.

Apéndice B

SISTEMA DE GRUPOS ORGANIZADOS | SGO
Especificación de Requerimientos de la Aplicación

VERSIÓN 1.1

2. Historia de Revisión.

Fecha	Versión	Descripción	Autor (es)
17-Abr-2015	1.0	Creación del documento de visión	Persy Quiroz Menor
22-Abr-2015	1.1	Actualización del documento	Persy Quiroz Menor

Propósito: En este documento se detallan los requerimientos del software para gestión de los grupos organizados (SGO), organizado en tres aspectos fundamentales para su desarrollo: las Especificaciones Funcionales, modelo de Casos de Uso, tanto en diagrama como en descripción, y las Especificaciones suplementarias. Todo esto establece los lineamientos y las restricciones que debe considerar el equipo de desarrollo del proyecto en la construcción de la aplicación software.

Nombre de la aplicación

- Sistema de Grupos Organizados (SGO)

Casos de Uso (CU)

3. Resumen de los casos de uso y los actores

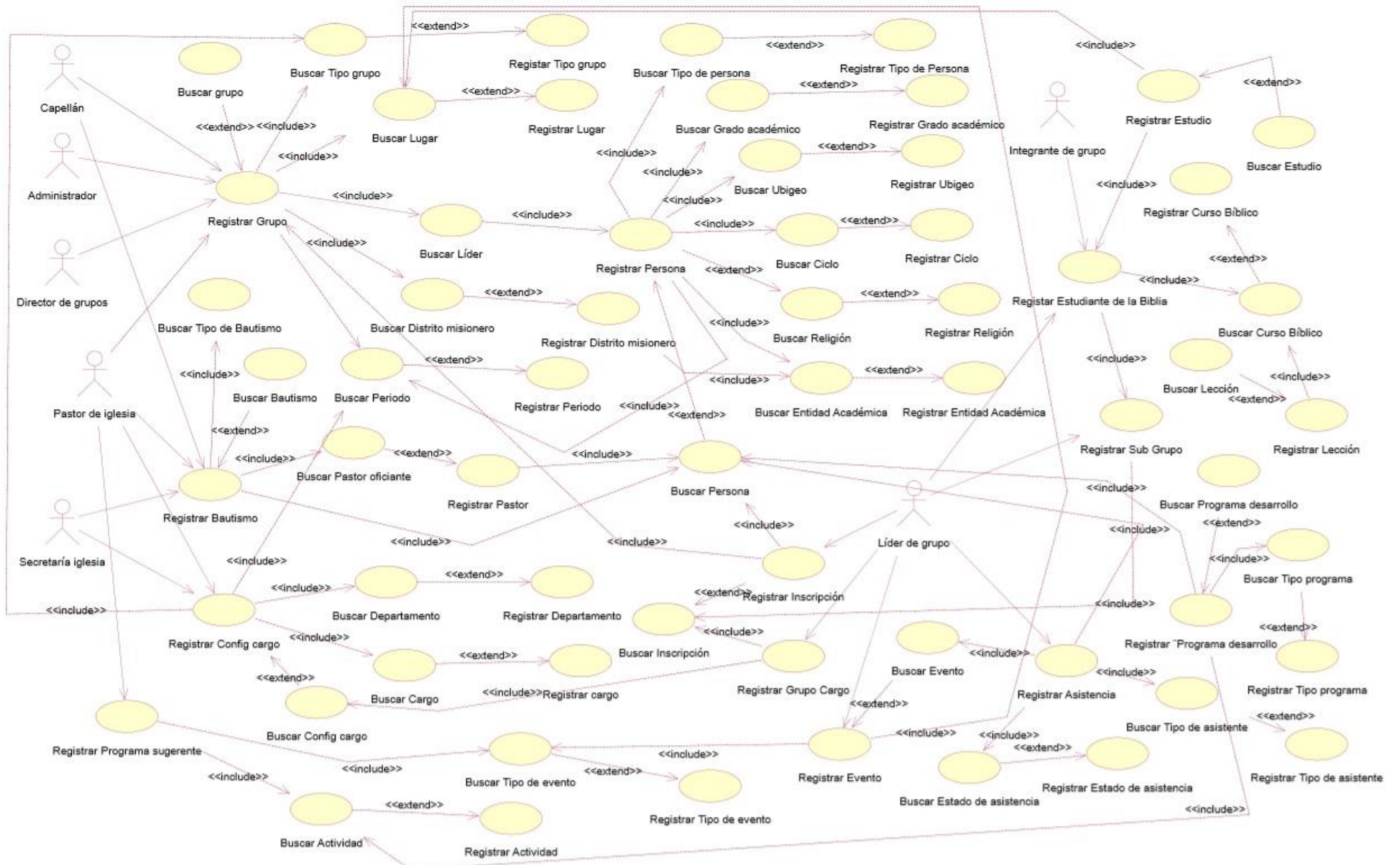
ID	Nombre del Caso de Uso	Actor(es)
1	Iniciar Sesión	Todos
2	Cerrar Sesión	Todos
3	Cambiar Contraseña	Todos
4	Crear Usuario Personal	Todos
5	Restaurar contraseña	Todos
6	Crear Grupo	Capellán, Director de grupos, Administrador
7	Modificar Grupo	Capellán, Director de grupos, Administrador
8	Eliminar Grupo	Capellán, Director de grupos, Administrador
9	Crear Tipo de grupo	Administrador
10	Modificar Tipo de grupo	Capellán, Director de grupos, Administrador
11	Crear Lugar de reunión	Capellán, Director de grupos, Administrador
12	Modificar Lugar de reunión	Capellán, Director de grupos, Administrador
13	Asignar lugar de reunión al grupo	Capellán, Director de grupos, Administrador
14	Asignar Grupo padre	Capellán, Director de grupos
15	Crear Líder	Capellán, Director de grupos, Administrador
16	Modificar Líder	Capellán, Director de grupos, Administrador
17	Asignar líder	Capellán, Director de grupos
18	Crear Distrito misionero	Administrador
19	Modificar Distrito misionero	Administrador
20	Asignar Distrito misionero	Capellán, Director de grupos, Administrador
21	Ver lista de grupos	Todos
22	Crear Periodo	Administrador
23	Modificar Periodo	Administrador
24	Eliminar Periodo	Administrador
25	Crear Persona	Capellán, Director de grupos, Administrador, Líder de grupo
26	Modificar Persona	Capellán, Director de grupos
27	Buscar Ubigeo	Capellán, Director de grupos, Administrador
28	Buscar Entidad Académica	Capellán, Director de grupos, Administrador
29	Buscar Ciclo	Capellán, Director de grupos, Administrador
30	Buscar Periodo	Capellán, Director de grupos, Administrador
31	Buscar Religión	Capellán, Director de grupos, Administrador
32	Buscar Grado académico	Capellán, Director de grupos, Administrador
33	Buscar Tipo de Persona	Capellán, Director de grupos, Administrador
34	Ver Persona	Capellán, Director de grupos, Administrador
35	Crear Bautismo	Capellán, Pastor de iglesia, Secretaría iglesia
36	Modificar Bautismo	Capellán, Pastor de iglesia, Secretaría iglesia
37	Buscar Pastor oficiante	Capellán, Pastor de iglesia, Secretaría iglesia

ID	Nombre del Caso de Uso	Actor(es)
38	Crear Tipo de evento	Capellán, Pastor de iglesia
39	Modificar Tipo de evento	Capellán, Pastor de iglesia
40	Ver Tipo de Grupo	Capellán, Pastor de iglesia
41	Configurar Cargos por Tipo de grupo	Capellán, Pastor de iglesia
42	Ordenar cargos	Capellán, Pastor de iglesia
43	Crear Actividad	Capellán, Pastor de iglesia
44	Modificar Actividad	Capellán, Pastor de iglesia
45	Eliminar Actividad	Capellán, Pastor de iglesia
46	Asignar Actividad a Tipo de evento	Capellán, Pastor de iglesia
47	Ordenar Actividad en Tipo de evento	Capellán, Pastor de iglesia
48	Crear Tipo de asistente	Capellán, Pastor de iglesia
49	Eliminar Tipo de asistente	Capellán, Pastor de iglesia
50	Buscar Tipo de Asistente	Capellán, Pastor de iglesia
51	Crear Integrante	Líder de grupo
52	Buscar Integrante por Nombres, código o Documento de Identidad	Líder de grupo
53	Añadir Integrante al Grupo	Líder de grupo, Capellán, Pastor de iglesia
54	Modificar Integrante	Líder de grupo
55	Ver Ficha de Integrante	Líder de grupo, Capellán, Pastor de iglesia
56	Eliminar integrante	Líder de grupo
57	Ver fotos de Integrantes	Líder de grupo, Capellán, Pastor de iglesia
58	Ver lista de Integrantes	Líder de grupo, Capellán, Pastor de iglesia
59	Modificar Grupo	Líder de grupo
60	Buscar Responsable de Cargo	Líder de grupo
61	Eliminar Responsable de Cargo	Líder de grupo
62	Ver Cargo	Líder de grupo, Capellán, Pastor de Iglesia
63	Ver Evento	Líder de grupo, Capellán, Pastor de Iglesia
64	Buscar Evento por Periodo y Mes	Líder de grupo
65	Crear Evento	Líder de grupo
66	Buscar Tipo de evento por Tipo de Grupo	Líder de grupo
67	Ver programa de evento	Todos
68	Crear Actividad en Programa desarrollo	Líder de grupo
69	Buscar Actividad por nombre	Líder de grupo
70	Buscar Tipo de Programa	Líder de grupo
71	Buscar Actividades por Tipo de evento	Líder de grupo
72	Modificar Actividad	Líder de grupo
73	Eliminar Actividad	Líder de grupo
74	Buscar responsable en Persona, en Integrantes y Grupo	Líder de grupo
75	Crear Nota en Programa desarrollo	Líder de grupo
76	Modificar Nota en Programa desarrollo	Líder de grupo
77	Eliminar Nota en Programa Desarrollo	Líder de grupo
78	Imprimir Programa desarrollo	Líder de grupo
79	Crear Mensaje de participación (Correo)	Líder de grupo
80	Ver resumen de Evento	Líder de grupo
81	Crear Asistencia	Líder de grupo
82	Modificar Asistencia	Líder de grupo
83	Eliminar Asistencia	Líder de grupo
84	Buscar Asistente	Líder de grupo
85	Buscar Tipo de Asistente	Líder de grupo
86	Buscar Asistente por Nombres, código y Número de documento	Líder de grupo

ID	Nombre del Caso de Uso	Actor(es)
87	Ver Asistencia	Líder de grupo
88	Crear Sub grupo	Líder de grupo
89	Modificar Sub Grupo	Líder de grupo
90	Eliminar Sub grupo	Líder de grupo
91	Ver Sub grupo	Líder de grupo
92	Asignar Integrante a Sub grupo	Líder de grupo
93	Eliminar Integrante de Sub grupo	Líder de grupo
94	Buscar Estudiante de la Biblia	Líder de grupo
95	Buscar Curso Bíblico	Todos
96	Modificar Estudiante de la Biblia	Todos
97	Eliminar Estudiante de la Biblia	Todos
98	Crear Estudio Bíblico	Todos
99	Buscar Lugar para Estudio Bíblico	Todos
100	Eliminar Estudio Bíblico de Estudiante de la Biblia	Todos
101	Modificar Estudio Bíblico de estudiante de la Biblia	Todos
102	Crear Meta por Indicador del grupo	Líder de grupo
103	Modificar Meta por Indicador del grupo	Líder de grupo
104	Eliminar Meta por Indicador del grupo	Líder de grupo
105	Crear Parentesco	Todos
106	Modificar Parentesco	Todos
107	Eliminar Parentesco	Todos
108	Ver Parentesco	Todos
109	Buscar Tipo de relación por género	Todos
110	Subir Foto a Evento	Líder de grupo
111	Eliminar Foto de Evento	Líder de grupo
112	Crear Publicación	Todos
113	Eliminar Publicación	Todos
114	Modificar Publicación	Todos
115	Ver Publicación	Todos
116	Crear mensaje grupal	Todos
117	Crear Incidencia	Pastor de iglesia, Capellán
118	Modificar Incidencia	Pastor de iglesia, Capellán
119	Eliminar Incidencia	Pastor de iglesia, Capellán
120	Ver Incidencia	Pastor de iglesia, Capellán
121	Asignar Evidencia a Incidencia	Pastor de iglesia, Capellán
122	Eliminar Evidencia de Incidencia	Pastor de iglesia, Capellán
123	Ver Evidencias en Incidencia	Pastor de iglesia, Capellán
124	Crear Perfil del sistema	Administrador
125	Modificar Perfil	Administrador
126	Eliminar Perfil	Administrador
127	Ver Perfil	Administrador
128	Crear Módulo	Administrador
129	Modificar Módulo	Administrador
130	Eliminar Módulo	Administrador
131	Ver Módulo	Administrador
132	Asignar Permiso a Perfil	Administrador
133	Eliminar Permiso de Perfil	Administrador
134	Ver Permiso	Administrador
135	Crear Usuario	Administrador, Capellán
136	Modificar Usuario	Administrador, Capellán

ID	Nombre del Caso de Uso	Actor(es)
137	Eliminar Usuario	Administrador, Capellán
138	Ver usuario	Administrador, Capellán

4. Diagrama de Casos de Uso



5. Especificaciones de Casos de Uso.

A continuación se especifica cada caso de uso. Se describe el mismo, la pre-condición, flujo del proceso y la post-condición.

ID del caso de Uso	Nombre
1	Iniciar Sesión
Descripción	
El actor inicia sesión en el sistema	
Pre-condición	
El actor no se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa la dirección web del sistema. 2. El actor ingresa su usuario y contraseña y presiona el botón “Ingresar” 3. El sistema valida usuario y contraseña provistas 4. A. Si la validación es exitosa, el sistema obtiene el perfil, grupo y periodo en curso, y el actor es transferido al menú principal. B. Si la validación no es exitosa, el actor es devuelto al login del sistema. 	
Post-condición	
El actor se encuentra logeado en el sistema, con perfil, grupo asignado y el periodo en curso.	

ID del caso de Uso	Nombre
2	Cerrar Sesión
Descripción	
El actor cierra su sesión en el sistema	
Pre-condición	
El actor tiene su sesión abierta en el sistema	
Flujo	
<ol style="list-style-type: none"> 1. El actor presionó el link “Cerrar Sesión” dentro de la opción “Mi perfil” 2. El sistema destruye la información de la Sesión del actor. 3. El actor es transferido a la página del login del sistema. 	
Post-condición	
El actor no se encuentra logeado en el sistema.	

ID del caso de Uso	Nombre
3	Cambiar Contraseña
Descripción	
El actor desea cambiar su contraseña de acceso	
Pre-condición	
El actor se encuentra logeado en el sistema	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona el link “Cambiar Contraseña”. 2. El actor es transferido al formulario para cambio de contraseña. 3. El actor introduce su actual y nueva contraseña y presiona el botón “Validar”. 4. A. Si la validación es exitosa. El sistema emite un mensaje correcto. 	

B. Si la validación es falsa. El sistema emite un mensaje de error.
Post-condición
El actor ingresa al sistema utilizando su nueva contraseña.

ID del caso de Uso	Nombre
4	Crear Usuario Personal
Descripción	
El actor desea crear su usuario de ingreso al sistema	
Pre-condición	
El actor no tiene creado un usuario de acceso al sistema. El actor está registrado en el sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor accede a la dirección web del sistema. 2. El presiona la opción “Crear mi usuario”. 3. El actor selecciona el Tipo de Documento e ingresa su número de documento y le da clic en el botón “Buscar” 4. A. Si la búsqueda es correcta. El sistema emite los datos del actor. <ol style="list-style-type: none"> A.1. El actor ingresa un usuario tentativo a crear, y le da clic en “Validar” A.2. El sistema valida usuario tentativo. <ol style="list-style-type: none"> A.2.1. Si es correcto, el actor da clic en “Crear usuario” <ol style="list-style-type: none"> A.2.1.1. El sistema crea usuario y genera una contraseña y le envía a correo electrónico. A.2.1.2. El actor recibe correo electrónico. A.2.2. Si no es correcto, el actor ingresa otro usuario tentativo. B. Si la búsqueda es incorrecta, muestra el mensaje de no encontrado. El actor ingresa otra búsqueda 	
Post-condición	
El actor tiene creado su usuario de acceso al sistema.	

ID del caso de Uso	Nombre
5	Restaurar Contraseña
Descripción	
El desea restaurar su contraseña por olvido.	
Pre-condición	
El actor olvidó su contraseña.	
Flujo	
<ol style="list-style-type: none"> 1. El actor accede a la dirección web del sistema. 2. El presiona la opción “Olvidé mi contraseña”. 3. El actor selecciona el Tipo de Documento e ingresa su número de documento y le da clic en el botón “Buscar” 4. A. Si la búsqueda es correcta. El sistema emite los datos del actor. <ol style="list-style-type: none"> A.1. El actor da clic en “Cambiar contraseña”. A.2. El sistema genera una nueva contraseña y envía al correo electrónico del actor. 	

B. Si la búsqueda es incorrecta, muestra el mensaje de no encontrado. El actor ingresa otra búsqueda.
Post-condición
El actor tiene una nueva contraseña para acceder al sistema.

ID del caso de Uso	Nombre
6	Crear Grupo
Descripción	
El actor desea agregar un nuevo grupo al sistema.	
Pre-condición	
El actor se encuentra logeado al sistema y tiene acceso a esta opción	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona la opción “Inscripción de grupos” en la sección de “Registros”. 2. El actor presiona el ícono más (+) y da clic en la opción “Registrar Grupo”. 3. El actor ingresa: nombre del grupo, líder, Grupo al que pertenece, lugar reunión, día, hora, entidad académica (distrito misionero interno) y da clic en el botón “Registrar”. 4. El sistema crear el registro del grupo en la Base de Datos. 5. El actor es transferido a la vista de lista de grupos. 	
Post-condición	
El nuevo Grupo existe en el sistema.	

ID del caso de Uso	Nombre
7	Modificar Grupo
Descripción	
El actor desea modificar un Grupo	
Pre-condición	
El actor se encuentra logeado en el sistema El actor está en la vista de lista de Grupos	
Flujo	
<ol style="list-style-type: none"> 1. El actor seleccionar el Grupo y da clic en “Editar datos” 2. El actor ingresa los datos a reemplazar y presiona el botón “Actualizar”. 3. El sistema actualiza el registro del Grupo en la Base de Datos. 4. El actor es transferido a la vista de lista de Grupos. 	
Post-condición	
El Grupo tiene su nueva información en el sistema.	

ID del caso de Uso	Nombre
8	Eliminar Grupo
Descripción	
El actor desea eliminar un Grupo	
Pre-condición	
El actor se encuentra logeado en el sistema El actor está en la vista de lista de Grupos	

Flujo	
<ol style="list-style-type: none"> 1. El actor seleccionar el Grupo y da clic en “Eliminar Grupo” 2. El sistema elimina de la base de datos al Grupo, y si tiene registros asociados, simplemente lo cambia a estado desactivo. 3. El actor es transferido a la lista de Grupos. 	
Post-condición	
El Grupo no está en la lista de Grupos registrados.	

ID del caso de Uso	Nombre
9	Crear Tipo de Grupo
Descripción	
El actor desea agregar un Tipo de grupo	
Pre-condición	
El actor se encuentra logeado en el sistema El actor está en la vista de lista de Tipos de Grupos	
Flujo	
<ol style="list-style-type: none"> 1. El actor da clic en el botón “Nuevo”. 2. El actor ingresa los datos del Tipo de grupo: nombre, nombre plural, descripción, Tipo grupo padre, logo y da clic en el botón “Registrar”. 3. El sistema agrega al Tipo de grupo en la base de datos. 4. El actor es transferido a la lista de Tipos de Grupos. 	
Post-condición	
El Tipo de Grupo aparece en la lista de Tipos de grupos en el sistema.	

ID del caso de Uso	Nombre
10	Modificar Tipo de Grupo
Descripción	
El actor desea modificar un Tipo de grupo	
Pre-condición	
El actor se encuentra logeado en el sistema El actor está en la vista de lista de Tipos de Grupos	
Flujo	
<ol style="list-style-type: none"> 1. El actor da clic en el botón “Editar”. 2. El actor ingresa los datos y da clic en “Actualizar”. 3. El sistema actualiza el Tipo de Grupo en la base de Datos 4. El actor es transferido a la lista de Tipos de Grupos 	
Post-condición	
El Tipo de Grupo aparece con su información actualizada en el sistema.	

ID del caso de Uso	Nombre
11	Crear Lugar de reunión
Descripción	
El actor desea agregar un Lugar de reunión	
Pre-condición	
El actor se encuentra logeado en el sistema	

El actor está en la vista de lista de Lugares de reunión
Flujo
<ol style="list-style-type: none"> 1. El actor da clic en el botón “Nuevo”. 2. El actor introduce la información del lugar y presiona el botón “Registrar”. 3. El sistema agrega al Nuevo lugar en la base de datos. 4. El actor es transferido a la lista de Lugares.
Post-condición
El lugar se aparece en la lista de lugares registrados.

ID del caso de Uso	Nombre
12	Modificar Lugar de reunión
Descripción	
El actor modificar la información del Lugar de reunión.	
Pre-condición	
El actor se encuentra logeado en el sistema El actor está en la vista de lista de Lugares de reunión	
Flujo	
<ol style="list-style-type: none"> 1. El actor busca, selecciona el lugar y da clic en el link “Editar” 2. El actor modificar los datos del lugar y presiona el botón “Actualizar”. 3. El sistema modifica la información del Lugar en la Base de Datos. 4. El actor es transferido a la lista de Lugares. 	
Post-condición	
El lugar se aparece en la lista de lugares registrados con su información actualizada.	

ID del caso de Uso	Nombre
13	Asignar lugar de reunión al grupo
Descripción	
El actor desea añadir un lugar de reunión para el grupo	
Pre-condición	
El actor se encuentra logeado en el sistema El actor está en el formulario del registro del Grupo	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa los valores de búsqueda del lugar. 2. El sistema busca en la base de datos los lugares de reunión por nombre o dirección. 3. Si el actor encuentra el lugar selecciona, caso contrario el sistema da la opción de “Registra Nuevo Lugar” 	
Post-condición	
El Lugar está asociado al Grupo.	

ID del caso de Uso	Nombre
14	Asignar Grupo Padre
Descripción	
El actor desea asignar un grupo padre a un grupo	
Pre-condición	

El actor se encuentra logeado en el sistema El actor está en el formulario del registro del Grupo
Flujo
<ol style="list-style-type: none"> 1. Sistema lista los grupos registrados filtrado por tipo dependiente de acuerdo al Tipo de grupo seleccionado. 2. El actor seleccionar el Grupo.
Post-condición
El grupo padre está añadido en el formulario del Grupo.

ID del caso de Uso	Nombre
15	Crear Líder (Persona)
Descripción	
El actor desea registrar un líder para el grupo	
Pre-condición	
El actor se encuentra logeado en el sistema El actor está en el formulario del registro del Grupo	
Flujo	
<ol style="list-style-type: none"> 1. El actor da clic en el botón “Registrar nueva persona”. 2. El actor ingresa los datos obligatorios de la persona y presiona el botón “Registrar”. 3. El sistema registra los datos del líder en la Base de Datos. 	
Post-condición	
La persona está añadido al grupo como líder.	

ID del caso de Uso	Nombre
16	Modificar Líder
Descripción	
El actor desea actualizar los datos del líder de grupo	
Pre-condición	
El actor se encuentra logeado en el sistema El actor se encuentra en la lista de Grupos registrados.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el nombre del líder y da clic sobre el botón “Editar datos”. 2. El actor ingresa actualiza la información del líder tales como: matriculado, ciclo, tipo de residente, religión, celular, etc. Y presiona el botón “Actualizar”. 3. El sistema actualiza la información del líder en la Base de Datos. 4. El actor es transferido a la lista de grupos y líderes. 	
Post-condición	
El líder tiene su información actualizada en el sistema.	

ID del caso de Uso	Nombre
17	Asignar Líder
Descripción	
El actor desea asignar líder al grupo	

Pre-condición
El actor se encuentra logeado en el sistema El actor se encuentra en el formulario del grupo.
Flujo
<ol style="list-style-type: none"> 1. El actor buscar al líder por sus nombres y/o apellidos. 2. El sistema de manera asíncrona presenta va presentando las posible opciones del líder buscado. 3. El actor selecciona el líder.
Post-condición
El líder es asignado al grupo.

ID del caso de Uso	Nombre
18	Crear Distrito Misionero
Descripción	
El actor desea agregar un distrito misionero.	
Pre-condición	
El actor se encuentra logeado en el sistema El actor está en la lista de distritos misioneros.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el botón “Nuevo”. 2. El actor ingresa la información del Distrito Misionero. 3. El sistema crea un registro en la Base de Datos. 4. El actor es transferido a la lista de Distritos Misioneros. 	
Post-condición	
El distrito misionero está registrado en el sistema.	

ID del caso de Uso	Nombre
19	Modificar Distrito Misionero
Descripción	
El actor desea actualizar los datos del distrito misionero.	
Pre-condición	
El actor se encuentra logeado en el sistema El actor está en la lista de distritos misioneros.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el botón “Editar”. 2. El actor actualiza la información del Distrito Misionero. 3. El sistema actualizar la información del Distrito misionero en la Base de Datos. 4. El actor es transferido a la lista de Distritos Misioneros. 	
Post-condición	
El distrito misionero tiene su información actualizada en el sistema.	

ID del caso de Uso	Nombre
20	Asignar Distrito Misionero
Descripción	
El actor desea asignar un distrito misionero al grupo.	

Pre-condición
El actor se encuentra logeado en el sistema El actor está en el formulario del registro del Grupo.
Flujo
<ol style="list-style-type: none"> 1. El sistema lista los distritos misioneros. 2. El actor selecciona el distrito misionero.
Post-condición
El grupo tiene asignado el distrito misionero.

ID del caso de Uso	Nombre
21	Ver lista de Grupos
Descripción	
El actor desea ver los grupos registrados y su información relacionada	
Pre-condición	
El actor se encuentra logeado en el sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presión el link “Grupos registrados” de la sección Administrar. 2. El actor selecciona los criterios de búsqueda del grupo. 3. El sistema presenta la lista de grupos registrados según criterio de búsqueda. 	
Post-condición	

ID del caso de Uso	Nombre
22	Crear Periodo
Descripción	
El actor desea agregar un nuevo Periodo al sistema.	
Pre-condición	
El actor se encuentra logeado al sistema El actor se encuentra en la lista de Periodos.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona el botón “Nuevo periodo”. 2. EL actor introduce el nombre, fecha de inicio, fecha final, estado en curso y presiona el botón “Registrar”. 3. El sistema crea el registro del Periodo en la Base de Datos. 4. El actor es transferido a la Lista de Periodos. 	
Post-condición	
El nuevo Periodo existe en la lista de Periodos.	

ID del caso de Uso	Nombre
23	Modificar Periodo
Descripción	
El actor desea modificar un Periodo.	
Pre-condición	
El actor se encuentra logeado al sistema El actor se encuentra en la lista de Periodos.	

Flujo
<ol style="list-style-type: none"> 1. El actor presiona el link “Editar” 2. El actor ingresa la nueva información y presiona el botón “Actualizar”. 3. El actor es transferido a la Lista de Periodos.
Post-condición
El Periodo tiene su información actualizada en el sistema.

ID del caso de Uso	Nombre
24	Eliminar Periodo
Descripción	
El actor desea eliminar un Periodo.	
Pre-condición	
El actor se encuentra logeado al sistema El actor se encuentra en la lista de Periodos.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona el link “Eliminar” 2. El Sistema elimina el registro del Periodo de la Base de Datos 3. El actor es transferido a la lista de Periodos. 	
Post-condición	
El Periodo no se encuentra en la lista de Periodos.	

ID del caso de Uso	Nombre
25	Crear Persona
Descripción	
El actor desea registrar una nueva Persona en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema El actor se encuentra en la lista de Personas	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona el link “Nueva persona” 2. El actor ingresa nombres, apellido paterno, apellido materno, escoge el tipo de documento, ingresa en número de documento, fecha de nacimiento, género, busca el distrito de procedencia, ingresa la dirección de procedencia, busca el distrito actual, ingresa la dirección actual, escoge el tipo de persona, busca la entidad académica, ingresa el código universitario, selecciona el ciclo (si es alumno), el tipo de residente, selecciona el grado académico, la religión, estado de bautismo (si o no), ingresa el número de celular y teléfono, y el correo electrónico de la Persona. 3. El sistema crea el registro de la Persona en la Base de Datos. 4. El actor es transferido a la lista de Personas. 	
Post-condición	
La nueva Persona está registrada en el sistema.	

ID del caso de Uso	Nombre
26	Modificar Persona
Descripción	

El actor desea modificar la información de una Persona.
Pre-condición
El actor se encuentra logeado al sistema El actor se encuentra en la lista de Personas
Flujo
<ol style="list-style-type: none"> 1. El actor presiona el link “Editar” 2. El actor ingresa la nueva información de la Persona y presiona el botón “Actualizar” 3. EL sistema actualiza los datos de la Persona en la Base de Datos 4. El actor es transferido a la lista de Personas.
Post-condición
La Persona tiene su información actualizada en el sistema.

ID del caso de Uso	Nombre
27	Buscar Ubigeo
Descripción	
El actor desea buscar un Ubigeo (distrito) para añadirlo a la persona.	
Pre-condición	
El actor se encuentra logeado al sistema El actor se encuentra en el formulario de Persona	
Flujo	
<ol style="list-style-type: none"> 1. El actor introduce en el texto de búsqueda los valores 2. El sistema busca en la base de datos el Ubigeo según valores de búsqueda. 3. El actor selecciona el Ubigeo de encontrado. 	
Post-condición	

ID del caso de Uso	Nombre
28	Buscar Entidad Académica
Descripción	
El actor desea buscar una Entidad Académica para añadirlo a la persona.	
Pre-condición	
El actor se encuentra logeado al sistema El actor se encuentra en el formulario de Persona	
Flujo	
<ol style="list-style-type: none"> 1. El actor introduce en el texto de búsqueda. 2. El sistema busca en la base de datos por nombre de entidad académica. 3. El actor selecciona la Entidad Académica 	
Post-condición	

ID del caso de Uso	Nombre
29	Busca Ciclo
Descripción	
El actor desea seleccionar un Ciclo académico si la Persona es de Tipo “Alumno”	
Pre-condición	
El actor se encuentra logeado al sistema	

El actor se encuentra en el formulario de Persona
Flujo
<ol style="list-style-type: none"> 1. El actor seleccionar el Tipo de Persona “Alumno”. 2. El sistema busca la lista de ciclos académico. 3. El actor selecciona el ciclo académico
Post-condición

ID del caso de Uso	Nombre
30	Buscar Periodo
Descripción	
El actor desea añadir los registros de la Persona a un Periodo	
Pre-condición	
El actor se encuentra logeado al sistema El actor se encuentra en el formulario de Persona	
Flujo	
<ol style="list-style-type: none"> 1. El sistema captura el Periodo en Curso y lo asigna los datos al mismo. 	
Post-condición	
La información de la Persona es actualizada de en el Periodo en Curso.	

ID del caso de Uso	Nombre
31	Buscar Religión
Descripción	
El actor desea seleccionar una religión para la Persona	
Pre-condición	
El actor se encuentra logeado al sistema El actor se encuentra en el formulario de Persona	
Flujo	
<ol style="list-style-type: none"> 1. El sistema busca las Religiones registradas en la Base de Datos. 2. El actor selecciona la religión de la Persona. 	
Post-condición	
La Religión está añadida a la Persona.	

ID del caso de Uso	Nombre
32	Buscar Grado Académico
Descripción	
El actor desea seleccionar un Grado académico para la Persona	
Pre-condición	
El actor se encuentra logeado al sistema El actor se encuentra en el formulario de Persona	
Flujo	
<ol style="list-style-type: none"> 1. El sistema busca los Grados Académicos registradas en la Base de Datos. 2. El actor selecciona el grado académico de la Persona. 	
Post-condición	

El Grado académico está añadida a la Persona.

ID del caso de Uso	Nombre
33	Buscar Tipo de Persona
Descripción	
El actor desea seleccionar un Tipo para la Persona	
Pre-condición	
El actor se encuentra logeado al sistema El actor se encuentra en el formulario de Persona	
Flujo	
<ol style="list-style-type: none"> 1. El sistema busca los Tipos de Personas registradas en la Base de Datos. 2. El actor selecciona el tipo de la Persona. 	
Post-condición	

ID del caso de Uso	Nombre
34	Listar Persona
Descripción	
El actor desea ver la información de un Persona	
Pre-condición	
El actor se encuentra logeado al sistema	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona el link “Personas” en la sección “Registros”. 2. El actor es transferido a la lista de Personas. 3. El actor hace click en el nombre de la Persona que quiere ver. 4. El sistema presenta la Vista de la Persona en una ventana modal. 	
Post-condición	

ID del caso de Uso	Nombre
35	Crear Bautismo
Descripción	
El actor desea agregar un Bautismo en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona el link “Miembros de iglesia” en la sección “Registros”. 2. El actor presiona el botón “Nuevo miembro”. 3. El actor ingresa la información del Bautismo tales como: tipo de bautismo, persona, nivel educativo, estado civil, entidad académica, iglesia anterior, religión anterior, padre, madre, modo de conversión, fecha de bautismo, pastor oficiante, nombre del evento, grupo al que pertenece. Y presión el botón “Registrar”. 4. El sistema crea un registro en la base de datos del Bautismo. 	
Post-condición	
El nuevo Bautismo existe en el sistema.	

ID del caso de Uso	Nombre
36	Modificar Bautismo
Descripción	
El actor desea modificar un Bautismo.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de Bautismos.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el Tipo de bautismo y da click "Editar". 2. El actor ingresa los datos a actualizar y da click en "Actualizar". 3. El sistema modifica el registro del Bautismo en la Base de Datos. 	
Post-condición	
El Bautismo tiene su nueva información en el sistema.	

ID del caso de Uso	Nombre
37	Buscar Pastor Oficiante
Descripción	
El actor desea agregar un pastor oficiante al bautismo.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en formulario de Bautismo.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa el valor de búsqueda. 2. El sistema busca el pastor oficiante en la base de datos. 3. El actor selecciona el oficiante. 	
Post-condición	
El Pastor oficiante está añadido al Bautismo.	

ID del caso de Uso	Nombre
38	Crear Tipo de evento
Descripción	
El actor desea agregar un Tipo de evento al sistema.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona el link "Tipos de eventos" y da click en el botón "Nuevo". 2. El actor ingresa el nombre y estado del Tipo de evento y da click en "Registrar". 3. El sistema crear un registro de Tipo de evento en la Base de Datos. 4. El actor es transferido a la lista de Tipos de eventos. 	
Post-condición	
El Tipo de evento está registrado en el sistema.	

ID del caso de Uso	Nombre
39	Modificar Tipo de evento

Descripción	
El actor desea modificar un Tipo de evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de Tipos de eventos	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona el link “Editar”. 2. El actor modifica la información del Tipo de evento y presiona el botón “Actualizar”. 3. El sistema actualiza el registro del Tipo de evento en la Base de Datos. 4. El actor es transferido a la lista de tipos de eventos. 	
Post-condición	
El Tipo de evento tiene su información actualizada en el sistema.	

ID del caso de Uso	Nombre
40	Ver Tipo de Grupo
Descripción	
El actor desea ver la información de un Tipo de evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de Tipos de eventos	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el Tipo de evento. 2. El actor es transferido a la vista del Tipo de evento. 	
Post-condición	

ID del caso de Uso	Nombre
41	Configurar Cargos por Tipo de grupo
Descripción	
El actor desea añadir o eliminar un cargo por Tipo de Grupo	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de Cargos configurados.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona el botón “Nuevo” 2. El actor ingresa los valores de búsqueda del departamento y cargo, y presiona el botón “Registrar”. 3. El sistema crea un nuevo registro de la configuración del cargo en la Base de Datos. 4. El actor es transferido a la lista de cargos por tipo de grupo. 	
Post-condición	
El cargo está añadido a la configuración de cargos en el sistema.	

ID del caso de Uso	Nombre
42	Ordenar cargos

Descripción
El actor desea cambiar el orden de un cargo en una lista de cargos
Pre-condición
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de Cargos configurados.
Flujo
<ol style="list-style-type: none"> 1. El actor presiona sobre los íconos “subir” o “bajar” orden. 2. EL sistema actualiza el orden del cargo en la Base de Datos.
Post-condición
El cargo está ordenado según configuración en el sistema.

ID del caso de Uso	Nombre
43	Crear Actividad
Descripción	
El actor desea añadir una nueva Actividad para el programa sugerido.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de programa sugerido	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Nueva actividad” 2. El actor ingresa el nombre de la actividad, descripción, fecha, hora de inicio, hora de término, sugerencia y selecciona el responsable, y presiona el botón “Registrar”. 3. EL sistema crea un nuevo registro de la actividad en la Base de datos. 4. El actor es transferido a la vista del programa sugerido. 	
Post-condición	
La actividad aparece en el programa sugerido.	

ID del caso de Uso	Nombre
44	Modificar Actividad
Descripción	
El actor desea actualizar una Actividad.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de programas sugeridos.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “editar” de la lista de actividades. 2. El actor ingresa los nuevos datos a actualizar y da clic en el botón “Actualizar actividad”. 3. EL sistema actualiza el registro de la actividad en la Base de datos. 4. El actor es transferido a la vista del programa sugerido. 	
Post-condición	
La actividad aparece actualizada en el programa sugerido.	

ID del caso de Uso	Nombre
--------------------	--------

45	Eliminar Actividad
Descripción	
El actor desea eliminar una Actividad del programa sugerido.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de programa sugerido	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el botón “Eliminar”. 2. El sistema eliminar el registro de la actividad de la Base de datos. 3. El actor es transferido a la vista del programa sugerido. 	
Post-condición	
La actividad no aparece en el programa sugerido.	

ID del caso de Uso	Nombre
46	Asignar Actividad a Tipo de evento
Descripción	
El actor desea asignar una actividad al programa sugerido	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de programa sugerido.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Nueva actividad”. 2. El actor ingresa el nombre, descripción, duración y presiona el botón “Registrar”. 3. El sistema crea un registro del programa sugerido en la base de datos. 4. El actor es transferido a la vista del Programa sugerido. 	
Post-condición	
La actividad aparece en el programa sugerido.	

ID del caso de Uso	Nombre
47	Ordenar Actividades en Tipo de evento
Descripción	
El actor desea ordenar la lista de actividades por tipo de evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de programa sugerido.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Ordenar”. 2. El actor mueve las actividades de acuerdo al orden que desea y presiona el botón “Guardar cambios”. 3. El sistema actualiza el orden de las actividades en la base de datos. 4. El actor es transferido a la vista del Programa sugerido 	
Post-condición	
Las actividades aparecen ordenadas en la vista del programa sugerido.	

ID del caso de Uso	Nombre
48	Crear Tipo de asistente
Descripción	
El actor desea agregar un Tipo de asistente al sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista tipos de eventos – asistentes.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “(+) Tipo de asistente”. 2. El actor ingresa el tipo de asistente y presiona el botón “Registrar”. 3. El sistema crea un nuevo registro de Tipo de asistente en la Base de datos. 4. El actor es transferido a la vista tipos de eventos – asistentes. 	
Post-condición	
EL Tipo de asistente está registrado en el sistema.	

ID del caso de Uso	Nombre
49	Eliminar Tipo de asistente
Descripción	
El actor desea eliminar un Tipo de asistente del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista tipos de eventos – asistentes.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Borrar”. 2. El sistema eliminar el registro del Tipo de asistente de la Base de datos. 3. El actor es transferido a la vista tipos de eventos – asistentes. 	
Post-condición	
EL Tipo de asistente no existe en el sistema.	

ID del caso de Uso	Nombre
50	Buscar Tipo de Asistente
Descripción	
El actor desea buscar Tipo de asistente por nombre.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista tipos de eventos – asistentes.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona ingresa los valores de búsqueda del Tipo de asistente. 2. El actor selecciona el Tipo de asistente buscado. 	
Post-condición	

ID del caso de Uso	Nombre
51	Crear Integrante
Descripción	

El actor desea Registrar los datos de un integrante.
Pre-condición
El actor se encuentra logeado al sistema. El actor se encuentra en la vista lista de integrantes del grupo.
Flujo
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Más opciones” y presiona la opción “Nueva persona”. 2. El actor los datos de la nueva persona y presiona el botón “Registrar”. 3. El sistema crea un registro de persona en la base de datos. 4. EL actor es transferido a la lista de integrantes.
Post-condición
La nueva persona está registrada en el sistema.

ID del caso de Uso	Nombre
52	Buscar Integrante por Nombres, código o Documento de Identidad
Descripción	
El actor desea buscar una persona para añadirlo como integrante.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en el formulario de búsqueda de personas.	
Flujo	
<ol style="list-style-type: none"> 1. El actor selecciona un filtro de búsqueda tales como: nombres, apellido paterno, apellido materno, código y número de documento 2. El sistema busca en la base de datos. 	
Post-condición	
EL integrante buscado es encontrado.	

ID del caso de Uso	Nombre
53	Añadir Integrante al Grupo
Descripción	
El actor desea crear un integrante de grupo en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista lista de integrantes del grupo.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Más opciones” y presiona la opción “Añadir integrante”. 2. El actor ingresa los valores de búsqueda de la persona. 3. El sistema busca en la base de datos a las personas. 4. A). Si la persona no está en un grupo, el actor da clic en la “Añadir”. B). Si la persona ya está en un grupo, el actor da clic en “Ya está en un grupo”. B.1. El actor lee las condiciones de cambio de grupo para el integrante y da clic en “Inscribir persona”. 5. El actor es transferido a la lista de integrantes. 	
Post-condición	

El nuevo integrante está aparece en la lista.

ID del caso de Uso	Nombre
54	Modificar integrante
Descripción	
El actor desea actualizar los datos del integrante.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista lista de integrantes del grupo.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Editar”. 2. El actor ingresa los datos a actualizar y presiona el botón “Actualizar”. 3. El sistema actualiza el registro del integrante en la base de datos. 4. El actor es transferido a la lista de integrantes. 	
Post-condición	
El integrante tiene sus datos actualizados en el sistema.	

ID del caso de Uso	Nombre
55	Ver Ficha de Integrante
Descripción	
El desea ver ficha de un Integrante	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista lista de integrantes del grupo.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Ver ficha”. 2. El actor es transferido a la vista de Ficha de persona. 	
Post-condición	
El actor ve la ficha de un Integrante.	

ID del caso de Uso	Nombre
56	Eliminar Integrante
Descripción	
El actor desea eliminar un integrante de su lista.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista lista de integrantes del grupo.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Borrar”. 2. El sistema elimina la inscripción del integrante de la Base de datos y crea un registro histórico de la operación. 3. El actor es transferido a la lista de integrantes. 	
Post-condición	
El Integrante no aparece en la lista de integrantes.	

ID del caso de Uso	Nombre
57	Ver fotos de Integrantes
Descripción	
El actor desea ver las fotos de los integrantes del grupo organizado	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista lista de integrantes del grupo	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el botón “Ver fotos”. 2. El sistema presenta al actor la foto de cada integrante, nombres, carrera y número de celular. 	
Post-condición	

ID del caso de Uso	Nombre
58	Ver lista de Integrantes
Descripción	
El actor desea ver la lista de integrante registrados.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa a la opción “Integrantes” de la sección “Administrar”. 2. El sistema muestra la información de cada integrante del grupo. 	
Post-condición	

ID del caso de Uso	Nombre
59	Modificar Grupo
Descripción	
El actor desea actualizar los datos de su grupo organizado.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de “Datos del grupo”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el botón “Editar datos”. 2. El sistema presenta el formulario de actualización. 3. El actor ingresa los datos del grupo a actualizar y da click en el botón “Actualizar”. 4. El sistema actualiza el registro del grupo en la Base de Datos. 5. El actor es transferido a la vista “Datos del grupo”. 	
Post-condición	
El grupo tiene su información actualizada en el sistema.	

ID del caso de Uso	Nombre
60	Buscar Responsable de Cargo

Descripción	
El actor desea actualizar los datos de su grupo organizado.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Cargos” del grupo.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en la opción "Añadir responsable". 2. El actor busca el integrante para añadirlo como responsable del cargo y da click e el botón “Asignar cargo”. 3. El sistema registra el cargo del grupo en la Base de Datos. 4. El actor es transferido a la vista de lista de cargos. 	
Post-condición	
El responsable es asignado a un cargo.	

ID del caso de Uso	Nombre
61	Buscar Responsable de Cargo
Descripción	
Eliminar Responsable de Cargo	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Cargos” del grupo.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el link "Borrar". 2. El sistema presenta un cuadro de diálogo para confirmar la eliminación. 3. El sistema elimina el registro del cargo de la Base de datos. 	
Post-condición	
El responsable es eliminado del cargo asignado.	

ID del caso de Uso	Nombre
62	Ver cargo
Descripción	
El actor desea ver la información de un cargo.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en la opción “Cargos” en la sección “Administrar”. 2. El sistema presenta la lista de los cargos y los responsables. 	
Post-condición	

ID del caso de Uso	Nombre
63	Ver evento
Descripción	
El actor desea ver la información de un evento.	
Pre-condición	

El actor se encuentra logeado al sistema.
Flujo
<ol style="list-style-type: none"> 1. El actor da click en ver “Eventos” en la sección “Administrar”. 2. El sistema presenta la vista de lista de eventos.
Post-condición

ID del caso de Uso	Nombre
64	Buscar Evento por Periodo y Mes
Descripción	
El actor desea listar los eventos por periodo y mes.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de eventos.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en la opción “Cargos” en la sección “Administrar”. 2. El actor selecciona un periodo y un mes. 3. El sistema presenta los resultados de búsqueda. 	
Post-condición	

ID del caso de Uso	Nombre
65	Crear Evento
Descripción	
El actor desea registrar un evento en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Lista de eventos”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el botón “”Nuevo evento”. 2. El actor selecciona el tipo de evento, ingresa título, fecha de inicio, hora de inicio, hora final, hora final, resumen, descripción y estado. Y da click en el botón “Registrar”. 3. El sistema registra los datos del evento en la Base de datos. 4. El actor es transferido a la vista de lista de eventos. 	
Post-condición	
El nuevo evento está registrado en el sistema.	

ID del caso de Uso	Nombre
66	Buscar Tipo de evento por Tipo de Grupo
Descripción	
El actor desea seleccionar un Tipo de evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en el formulario de registro de evento.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el botón “”Nuevo evento”. 	

2. El sistema lista los tipos de eventos por Tipo de grupo	
Post-condición	
ID del caso de Uso	Nombre
67	Ver programa de evento
Descripción	
El actor desea ver el programa de un evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Lista de eventos”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el icono de programa de un evento. 2. El sistema presenta el programa de un evento, con sus partes, actividades, tiempo y responsables. 	
Post-condición	

ID del caso de Uso	Nombre
68	Crear Actividad en Programa desarrollo
Descripción	
El actor desea añadir una nueva Actividad para el programa del evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de “elaboración del programa”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Nueva actividad” 2. El actor ingresa el nombre de la actividad, descripción, fecha, hora de inicio, hora de término, sugerencia y selecciona el responsable, y presiona el botón “Registrar”. 3. EL sistema crea un nuevo registro de la actividad en la Base de datos. 4. El actor es transferido a la vista del programa. 	
Post-condición	
La actividad aparece en el programa.	

ID del caso de Uso	Nombre
69	Buscar Actividad por nombre
Descripción	
El actor busca una actividad para seleccionar y añadirlo al programa.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en el formulario de “Actividad”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa los valores de búsqueda de la actividad. 2. El sistema busca las actividades coincidentes registradas. 3. El actor selecciona la actividad buscada. 	
Post-condición	

ID del caso de Uso	Nombre
70	Buscar Tipo de Programa
Descripción	
El actor quiere seleccionar los tipos (partes) para el programa.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Programa” del evento.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en la opción “Configuración”. 2. El actor chequea los tipos de programa que añadirá al programa y da click en el botón “Guardar”. 3. El sistema guarda la configuración de los tipos de programa en la Base de Datos. 4. El actor es transferido a la vista del programa del evento. 	
Post-condición	
La configuración de los tipos aparece en el programa.	

ID del caso de Uso	Nombre
71	Buscar Actividades por Tipo de evento
Descripción	
El actor desea copiar actividades de un programa sugerido de acuerdo al Tipo de evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Programa” del evento.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en la opción “Copiar formato”. 2. El sistema lista las actividades del programa sugerido. 3. El actor selecciona las actividades que copiará a su programa y da click en el botón “copiar al programa”. 4. El sistema registra las actividades del programa en la base de datos. 5. El actor es transferido a la vista del programa del evento. 	
Post-condición	
Las actividades copiadas aparecen en la lista del programa.	

ID del caso de Uso	Nombre
72	Modificar Actividad en Programa desarrollo
Descripción	
El actor desea actualizar una Actividad en el programa del evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de “elaboración del programa”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el ícono “editar” de la lista de actividades. 	

<ol style="list-style-type: none"> 2. El actor ingresa los nuevos datos a actualizar y da clic en el botón “Actualizar actividad”. 3. EL sistema actualiza el registro de la actividad en la Base de datos. 4. El actor es transferido a la vista del programa.
Post-condición
La actividad aparece actualizada en el programa.

ID del caso de Uso	Nombre
73	Eliminar Actividad en Programa desarrollo
Descripción	
El actor desea eliminar una Actividad del programa.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de “elaboración del programa”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el botón “Eliminar”. 2. El sistema eliminar el registro de la actividad de la Base de datos. 3. El actor es transferido a la vista del Programa. 	
Post-condición	
La actividad no aparece en el programa.	

ID del caso de Uso	Nombre
74	Buscar responsable en Persona, en Integrantes y Grupo
Descripción	
El actor desea asignar un responsable de la actividad en el programa de un evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Programa” del evento.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en la opción “+ Resp”. 2. El actor selecciona el Tipo de responsable (Persona, integrante, Grupo y descripción) e ingresa los valores de búsqueda. Y da click en “Registrar”. 3. El sistema registra el responsable de la actividad en la Base de Datos. 4. El sistema actualiza la vista de programa del evento. 	
Post-condición	
El responsable aparece en la actividad asignada.	

ID del caso de Uso	Nombre
75	Crear Nota en Programa desarrollo
Descripción	
El actor desea registrar una nota en el programa del evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Programa” del evento.	

Flujo	
<ol style="list-style-type: none"> 1. El actor da click en la opción “+ Más” en la sección “Notas”. 2. El actor ingresa el detalle de la nota y da click en “Aceptar”. 3. El sistema registra la nota en la Base de Datos. 4. El sistema actualiza la vista de programa del evento. 	
Post-condición	
La nota aparece en el programa.	

ID del caso de Uso	Nombre
76	Modificar Nota en Programa desarrollo
Descripción	
El actor desea actualizar una nota en el programa del evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Programa” del evento.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el icono “Editar” de la lista de notas. 2. El actor ingresa los cambios en la nota y presiona el botón “Actualizar nota”. 3. El sistema modifica los datos de la nota en la Base de Datos. 4. El sistema actualiza la vista de programa del evento. 	
Post-condición	
La nota no aparece en la lista de notas.	

ID del caso de Uso	Nombre
77	Eliminar Nota en Programa Desarrollo
Descripción	
El actor desea eliminar una nota en el programa del evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Programa” del evento.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el icono “Borrar” en la lista de notas. 2. El sistema elimina el registro de la nota en la Base de Datos. 3. El sistema actualiza la vista de programa del evento. 	
Post-condición	
La nota no aparece en la lista de notas.	

ID del caso de Uso	Nombre
78	Imprimir Programa desarrollo
Descripción	
El actor desea imprimir el programa del evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Programa” del evento.	
Flujo	

<ol style="list-style-type: none"> 1. El actor da click en el icono “Imprimir”. 2. El sistema le presenta en una nueva ventana el formato de impresión del programa.
Post-condición

ID del caso de Uso	Nombre
79	Crear Mensaje de participación (Correo)
Descripción	
El actor desea enviar un mensaje de participación al correo electrónico del participante.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Programa” del evento.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el link “Enviar mensajes de participación” y en el botón “Enviar correos”. 2. El sistema se autentifica en la cuenta de Gmail con el correo del sistema y envía correos personalizados a los diferentes participantes en el programa. 3. El actor es transferido a la vista del “Programa” del evento. 	
Post-condición	
El mensaje de correo ha sido enviado al participante.	

ID del caso de Uso	Nombre
80	Ver resumen de Evento
Descripción	
El actor desea ver el resumen de un evento registrado.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de “Eventos” registrados.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el ícono “Informe” del evento. 2. El sistema muestra un resumen de los datos del evento, programa y asistencia. 	
Post-condición	

ID del caso de Uso	Nombre
81	Crear Asistencia
Descripción	
El actor desea registrar la asistencia a un evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Eventos” registrados.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el ícono “Asistencia” y el link “+ Añadir integts”. 2. El actor selecciona a los asistentes, el tipo de asistente y el estado de la asistencia, y da click en el botón “Guardar asistencia”. 3. El sistema guarda los registros de los asistentes en la base de Datos. 	

4. El sistema actualiza la vista de “Asistencia”.
Post-condición
Los asistentes aparecen en la lista de asistencia con sus respectivos estados de: Puntual, tarde, falta y falta justificada.

ID del caso de Uso	Nombre
82	Modificar Asistencia
Descripción	
El actor desea actualizar la asistencia de un asistente (persona).	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Asistencia”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el ícono “Editar” o en las letras de estados de asistencia: P=Puntual, T=Tardanza, F=Falta y FJ=Falta Justificada. 2. El sistema actualiza los datos de la asistencia en la Base de Datos. 3. El sistema actualiza la vista “Asistencia”. 	
Post-condición	
La asistencia aparece con su información actualizada.	

ID del caso de Uso	Nombre
83	Eliminar asistencia
Descripción	
El actor desea eliminar una o toda la asistencia.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Asistencia”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el ícono “Borrar” o en el link “(x) eliminar”. 2. El sistema actualiza los datos de la asistencia en la Base de Datos. 3. El sistema actualiza la vista “Asistencia”. 	
Post-condición	
La asistencia aparece con su información actualizada.	

ID del caso de Uso	Nombre
84	Buscar Asistente
Descripción	
El actor desea añadir asistentes que no necesariamente sean integrantes del grupo.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Asistencia”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el link “(+) Asistentes”. 	

<ol style="list-style-type: none"> El actor busca al asistente por nombres, código o número de documento, y da click en el link “Registrar”. El sistema registra la asistencia de la persona en la Base de Datos. El sistema actualiza la vista “Asistencia”.
Post-condición
La persona buscada aparece como asistente en la lista de asistencia.

ID del caso de Uso	Nombre
85	Buscar Tipo de Asistente
Descripción	
El actor desea seleccionar un tipo de asistente.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Asistencia”.	
Flujo	
<ol style="list-style-type: none"> El sistema lista los tipos de asistente de acuerdo al tipo de evento. El actor selecciona el tipo de asistente correspondiente. 	
Post-condición	

ID del caso de Uso	Nombre
86	Buscar Asistente por Nombres, código y Número de documento
Descripción	
El actor desea buscar al asistente usando código de barras	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Asistencia”.	
Flujo	
<ol style="list-style-type: none"> El actor da click en el link “Barras”. El actor personaliza la búsqueda, por tipo de asistente, estado y método de búsqueda y da click en “Aceptar”, e ingresa los valores de búsqueda. El sistema busca en la base de datos y registra los datos de la asistencia de manera automática. El sistema actualiza la vista “Búsqueda” en asistencia. 	
Post-condición	
La persona buscada se registra en la asistencia.	

ID del caso de Uso	Nombre
87	Ver Asistencia
Descripción	
El actor desea ver la asistencia en cada evento.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Eventos”.	
Flujo	

<ol style="list-style-type: none"> 1. El actor da click en el icono “Asistencia”. 2. El sistema presenta la lista de asistentes al evento con sus estados y tipos de asistentes. 	
Post-condición	
ID del caso de Uso	Nombre
88	Crear Sub grupo
Descripción	
El actor desea crear un sub grupo	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Sub grupos”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el Tipo de sub grupo a registrar y da click en el botón “+ Sub grupo” 2. El sistema registra un nuevo sub grupo en la base de datos y actualiza la lista de sub grupos del sistema. 	
Post-condición	
El sub grupo está registrado en el sistema.	

ID del caso de Uso	Nombre
89	Modificar Sub Grupo
Descripción	
El actor desea actualizar un sub grupo del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Sub grupos”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en sub grupo a actualizar y da click en el botón “Editar” 2. El actor actualiza la información del sub grupo y da click en “Actualizar”. 3. El sistema actualiza la información del sub grupo en la Base de Datos. 4. El sistema actualiza la vista de sub grupos registrados. 	
Post-condición	
El sub grupo tiene su información actualizada en el sistema.	

ID del caso de Uso	Nombre
90	Eliminar Sub grupo
Descripción	
El actor desea eliminar un sub grupo del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Sub grupos”.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el ícono “Eliminar”. 2. El sistema elimina el registro del sub grupo de la Base de Datos. 	

3. El sistema actualiza la lista de sub grupos registrados.
Post-condición
El sub grupo no aparece en la lista de los registrados.

ID del caso de Uso	Nombre
91	Ver Sub grupo
Descripción	
El actor desea ver la información de un sub grupo del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista "Sub grupos".	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre el nombre del sub grupo. 2. El sistema presenta la vista de información del sub grupo. 	
Post-condición	

ID del caso de Uso	Nombre
92	Asignar Integrante a Sub grupo
Descripción	
El actor desea añadir integrantes a un sub grupo.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista "Sub grupos".	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre un icono "check" de la lista. 2. El actor busca al integrante por sus nombres y da click en "Registrar". 3. El sistema registra el integrante del sub grupo en la Base de Datos. 4. El sistema actualiza la lista de integrantes del sub grupo. 	
Post-condición	
El nuevo integrante aparece en la lista del sub grupo.	

ID del caso de Uso	Nombre
93	Eliminar Integrante de Sub grupo
Descripción	
El actor desea eliminar un integrante de un sub grupo.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista "Sub grupos".	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre un icono "check" de la lista. 2. Da click en el icono "eliminar" integrante. 3. El sistema eliminar el integrante del sub grupo de la Base de Datos. 4. El sistema actualiza la lista de integrantes del sub grupo. 	
Post-condición	
El integrante no aparece en la lista de integrantes del sub grupo.	

ID del caso de Uso	Nombre
94	Buscar Estudiante de la Biblia
Descripción	
El actor desea añadir un estudiante de la Biblia a un sub grupo.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa a la opción “Estudiantes de la Biblia” de la sección “Administrar”. 2. El actor da click en el ícono “Estudiantes” 3. El sistema presenta la vista de estudiantes de la Biblia de un sub grupo. 4. El actor da click en el link “+ Añadir estudiante”. 5. El actor busca a la persona, selecciona el estado civil, el curso bíblico a estudiar y da click en el botón “Registrar”. 6. El sistema registra el estudiante de la Biblia en la base de datos. 7. El actor es transferidos a la vista “Lista de estudiantes de la Biblia”. 	
Post-condición	
El estudiante de la Biblia aparece en la lista del sistema.	

ID del caso de Uso	Nombre
95	Buscar Curso Bíblico
Descripción	
El actor desea añadir un estudiante de la Biblia a un sub grupo.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en el formulario de registro de un estudiante de la Biblia.	
Flujo	
<ol style="list-style-type: none"> 1. El sistema busca la lista de cursos disponibles. 2. El actor selecciona un curso Bíblico. 	
Post-condición	

ID del caso de Uso	Nombre
96	Modificar Estudiante de la Biblia
Descripción	
El actor desea actualizar un estudiante de la Biblia en un sub grupo.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Lista de estudiantes de la Biblia”	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre el ícono “Editar” de los estudiantes. 2. El actor ingresa los datos actualizar tales como: dirección, celular, teléfono y da click en el botón “Actualizar datos”. 3. El sistema actualiza la información del estudiante en la Base de Datos. 4. El actor es transferido a la lista de estudiantes. 	

Post-condición	
El estudiante tiene su información actualizada en el sistema.	
ID del caso de Uso	Nombre
97	Eliminar Estudiante de la Biblia
Descripción	
El actor desea eliminar un estudiante de la Biblia.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista “Lista de estudiantes de la Biblia”	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre el ícono “Borrar” estudiante de la Biblia. 2. El sistema eliminar el registro del integrante de la Base de Datos. 3. El actor es transferido a la lista de estudiantes. 	
Post-condición	
El estudiante no aparece en la lista del sistema.	

ID del caso de Uso	Nombre
98	Crear Estudio Bíblico
Descripción	
El actor desea registrar un estudio de la lección del curso bíblico.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa a la opción “Control de cursos Bíblicos”. 2. El actor selecciona el sub grupo al que pertenece. 3. El sistema muestra la lista de los estudiantes de la Biblia y el avance de los estudios de las lecciones de cada curso. 4. El actor da click en número de lección. 5. El actor ingresa la fecha del estudio, hora, lugar, nota, selecciona los instructores bíblicos y da click en el botón “Registrar”. 6. El sistema registra el estudio en la Base de Datos. 7. El sistema actualiza la vista “Control de estudios bíblicos” 	
Post-condición	
El estudio bíblico está registrado en el sistema.	

ID del caso de Uso	Nombre
99	Buscar Lugar para Estudio Bíblico
Descripción	
El actor desea buscar por nombre el lugar donde se realizó el estudio bíblico.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en el formulario de registro del estudio bíblico.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa el valor de búsqueda del lugar del estudio. 	

2. El sistema automáticamente busca en la base de datos.	
Post-condición	
ID del caso de Uso	Nombre
100	Eliminar Estudio Bíblico de Estudiante de la Biblia
Descripción	
El actor desea eliminar un estudio bíblico estudiado.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de Estudios bíblicos.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el check de la lección estudiada. 2. El actor da click en el link “(X) eliminar estudio”. 3. El sistema elimina el registro del estudio de la Base de Datos. 4. El actor es transferido a la lista de estudiantes de la Biblia y de cursos bíblicos. 	
Post-condición	
El estudio bíblico está eliminado del sistema.	

ID del caso de Uso	Nombre
101	Modificar Estudio Bíblico de estudiante de la Biblia
Descripción	
El actor desea modificar un estudio bíblico.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de Estudios bíblicos.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el check de la lección estudiada. 2. El actor da click en el link Editar estudio”. 3. El actor actualiza la información del estudio bíblico y da click en el botón “Actualizar”. 4. El sistema actualiza el registro del estudio de la Base de Datos. 5. El actor es transferido a la lista de estudiantes de la Biblia y de cursos bíblicos. 	
Post-condición	
El estudio bíblico está actualizado en el sistema.	

ID del caso de Uso	Nombre
102	Crear Meta por Indicador del grupo.
Descripción	
El actor desea ingresar las metas del grupo organizado de cada indicador previamente configurado de acuerdo al tipo de grupo.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa a la opción: “Definición de indicadores”. 2. El sistema presenta la vista de tabla de indicadores. 	

<ol style="list-style-type: none"> 3. El actor ingresa las metas por cada indicador y presiona el botón “Guardar”. 4. El sistema registra las metas de los indicadores en la Base de datos. 5. El actor es transferido de vuelta a la tabla de indicadores.
Post-condición
Las metas están definidas en el sistema.

ID del caso de Uso	Nombre
103	Modificar Meta por Indicador del grupo
Descripción	
El actor desea ingresar las metas del grupo organizado de cada indicador previamente configurado de acuerdo al tipo de grupo.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la tabla de indicadores.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa los valores de las metas a actualizar y presiona el botón “Guardar”. 2. El sistema actualiza las metas en la Base de Datos. 3. El actor es transferido de vuelta a la tabla de indicadores. 	
Post-condición	
Las metas están actualizadas en el sistema.	

ID del caso de Uso	Nombre
104	Eliminar Meta por Indicador del grupo
Descripción	
El actor quitar una meta de la tabla de indicadores.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la tabla de indicadores.	
Flujo	
<ol style="list-style-type: none"> 1. El actor quita el valor de la meta de cada indicador y presiona el botón “Guardar”. 2. El sistema elimina la meta en la Base de Datos. 3. El actor es transferido de vuelta a la tabla de indicadores. 	
Post-condición	
La meta está eliminada del sistema.	

ID del caso de Uso	Nombre
105	Crear Parentesco
Descripción	
El actor desea añadir una relación de parentesco con otra persona del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa a la opción “Mi información personal” de la sección “Perfil”. 	

<ol style="list-style-type: none"> 2. El actor da click en el ícono “Parentesco” y da click en el link “+ Añadir”. 3. El actor busca a la Persona y el sistema le lista las posibles relaciones de acuerdo al sexo de la persona buscada y da click en “Registrar”. 4. El sistema registra la relación en la base de datos. 5. El actor es transferido a la vista lista de parentescos.
Post-condición
La relación de parentesco está registrado en el sistema.

ID del caso de Uso	Nombre
106	Modificar Parentesco
Descripción	
El actor desea actualizar la relación de parentesco con otra persona del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de parentescos	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el link “Editar”. 2. El actor modifica el tipo de relación y da click en botón “Actualizar”. 3. El sistema actualiza la relación de parentesco en la Base de Datos. 4. El sistema actualiza la lista de parentescos. 	
Post-condición	
La relación de parentesco está actualizado en el sistema.	

ID del caso de Uso	Nombre
107	Eliminar Parentesco
Descripción	
El actor desea eliminar la relación de parentesco con otra persona del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de parentescos	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click en el link “Eliminar”. 2. El sistema elimina la relación de parentesco de la Base de Datos. 3. El sistema actualiza la lista de parentescos. 	
Post-condición	
La relación de parentesco ya no aparece en el sistema.	

ID del caso de Uso	Nombre
108	Ver Parentesco
Descripción	
El actor desea ver la relación de parentesco con otra persona del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de “Mi información personal”	
Flujo	

<ol style="list-style-type: none"> 1. El actor da click en el ícono "Parentesco". 2. El sistema presenta la vista de parentescos.
Post-condición

ID del caso de Uso	Nombre
109	Buscar Tipo de relación por género
Descripción	
El actor busca el tipo de relación según persona seleccionada.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de registro de Parentesco.	
Flujo	
<ol style="list-style-type: none"> 1. El actor selecciona una persona. 2. El sistema busca los tipos de relaciones por el género de la persona. 	
Post-condición	

ID del caso de Uso	Nombre
110	Subir Foto a Evento
Descripción	
El actor desea subir una foto del evento realizado.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista "Eventos".	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre el link "fotos" y da click sobre el botón "Nueva foto". 2. El actor selecciona una foto de su equipo y da click sobre el botón subir foto. 3. El sistema sube la foto al servidor y le presenta la vista de la foto. 4. El actor ingresa una descripción de la foto y le da click en "Publicar". 5. El sistema guarda la ruta de la foto y la descripción en la Base de datos. 6. El actor es trasferido a la vista de "Eventos" 	
Post-condición	
La foto aparece en la lista del sistema.	

ID del caso de Uso	Nombre
111	Eliminar Foto de Evento
Descripción	
El actor desea eliminar una foto del evento realizado.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista "Eventos".	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre el ícono "Borrar" foto. 2. El sistema elimina la foto y el registro de la Base de datos. 3. El actor es transferido a la lista de "Eventos". 	
Post-condición	

La foto no aparece en la lista de fotos del evento.

ID del caso de Uso	Nombre
112	Crear Publicación
Descripción	
El actor desea registrar una Publicación ya sea foto o solo comentario.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de publicaciones recientes.	
Flujo	
<ol style="list-style-type: none">1. El actor presiona el botón “Añadir publicación”.2. El actor ingresa el comentario o carga una imagen, selecciona el nivel de privacidad y da click en “Registrar”.3. El sistema registra la publicación en la Base de Datos.4. El actor es transferido a la vista de “Publicaciones”.	
Post-condición	
La publicación se muestra en la vista de Publicaciones del sistema.	

ID del caso de Uso	Nombre
113	Eliminar Publicación
Descripción	
El actor desea eliminar una Publicación suya del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de publicaciones recientes.	
Flujo	
<ol style="list-style-type: none">1. EL actor da click sobre el ícono “Borrar”.2. El sistema presenta una vista para que el actor confirme la eliminación.3. El actor confirma la eliminación.4. El sistema elimina el registro de la Publicación de la Base de Datos.5. El actor es transferido a la vista de Publicaciones.	
Post-condición	
La publicación no aparece en la lista del sistema.	

ID del caso de Uso	Nombre
114	Modificar Publicación
Descripción	
El actor desea actualizar una Publicación suya del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de publicaciones recientes.	
Flujo	
<ol style="list-style-type: none">1. EL actor da click sobre el ícono del lápiz “Editar”.2. El actor actualiza el cometido de la Publicación y da click sobre el botón “Actualizar”.	

3. El sistema actualiza el registro de la Publicación de la Base de Datos.
4. El actor es transferido a la vista de Publicaciones.
Post-condición
La publicación está actualizada en el sistema.

ID del caso de Uso	Nombre
115	Ver Publicación
Descripción	
El actor desea ver el contenido de la Publicación, asimismo los comentarios recibidos.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de publicaciones recientes.	
Flujo	
1. EL actor da click el link “comentarios” de la Publicación. 2. El sistema presenta una vista con la información completa de la Publicación.	
Post-condición	

ID del caso de Uso	Nombre
116	Crear mensaje grupal
Descripción	
El actor desea registrar un mensaje en el chat que le pertenece al grupo.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
1. El actor presiona sobre el botón “Chat del grupo”. 2. El actor visualiza los mensajes recientes. Ingresa el mensaje y da click sobre el botón “Enviar”. 3. El sistema registra el contenido del mensaje en la Base de Datos. 4. El sistema muestra el mensaje registrado en el sistema.	
Post-condición	
El mensaje está registrado en el sistema.	

ID del caso de Uso	Nombre
117	Crear Incidencia
Descripción	
EL actor desea registrar una incidencia de una Persona en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
1. El actor ingresa a la opción “Incidencias” de la sección “Administrar”. 2. El actor busca a la persona filtrado ya sea por: Nombres, Apellido paterno, apellido materno, código, número de documento o nombre completo. 3. El sistema busca las personas en la base de datos automáticamente. 4. El actor selecciona a la persona y visualiza las incidencias registradas. 5. El actor da click sobre el ícono ya sea mano “positiva” o “negativa”.	

<ol style="list-style-type: none"> 6. El actor el tipo de incidencia, ingresa la descripción, lugar, fecha y hora, y da click en “Registrar”. 7. El sistema registra la incidencia en la Base de Datos. 8. El actor es transferido a la lista de incidencias de la Persona.
Post-condición
La incidencia está registrada en el sistema.

ID del caso de Uso	Nombre
118	Modificar Incidencia
Descripción	
EL actor desea modificar una incidencia de una Persona en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de incidencias de una Persona.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el ícono “Editar”. 2. El sistema presenta toda la información relacionada con la incidencia. 3. El actor da click sobre el botón “Editar”. 4. El actor ingresa los cambios y da click en el botón “Actualizar”. 5. El sistema actualiza el registro de la incidencia en la Base de Datos. 6. El actor es transferido a la vista de la información de la incidencia. 	
Post-condición	
La incidencia está actualizada en el sistema.	

ID del caso de Uso	Nombre
119	Eliminar Incidencia
Descripción	
EL actor desea eliminar una incidencia de una Persona en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de incidencias de una Persona.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el botón “Borrar”. 2. El sistema elimina el registro de la incidencia en la Base de Datos. 3. El actor es transferido a la vista de incidencias. 	
Post-condición	
La incidencia no aparece en la lista del sistema.	

ID del caso de Uso	Nombre
120	Ver Incidencia
Descripción	
EL actor desea ver la información de una incidencia.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de incidencias de una Persona.	

Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre el tipo de incidencia. 2. El sistema muestra en una vista toda la información de la incidencia. 	
Post-condición	

ID del caso de Uso	Nombre
121	Asignar Evidencia a Incidencia
Descripción	
El actor desea registra una evidencia a una incidencia.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de la edición de la incidencia.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el botón “Añadir” archivos adjuntos como evidencia. 2. EL actor selecciona archivo con extensiones: .png, .pdf, .mp3,.mp4,.jpg,.jpeg y .gif. y da click sobre el botón “Subir archivo”. 3. El sistema sube el archivo al servidor y registra la información en la Base de datos. 4. El actor es transferido a la vista de la incidencia. 	
Post-condición	
La evidencia aparece en la lista en el sistema.	

ID del caso de Uso	Nombre
122	Eliminar Evidencia de Incidencia
Descripción	
El actor desea eliminar una evidencia registrada en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de la edición de la incidencia.	
Flujo	
<ol style="list-style-type: none"> 1. El actor presiona sobre el link “Eliminar” evidencia. 2. El sistema presenta un mensaje de confirmación. 3. El actor acepta la confirmación. 4. El sistema elimina el registro de la evidencia de la Base de Datos. 5. El actor es transferido a la vista de la incidencia. 	
Post-condición	
La evidencia no aparece en la lista en el sistema.	

ID del caso de Uso	Nombre
123	Ver Evidencias en Incidencia
Descripción	
El actor desea ver el archivo de la evidencia y su información.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de información de la incidencia.	

Flujo	
1. El actor da click sobre la incidencia. 2. El sistema muestra el archivo según formato correspondiente y sus detalles.	
Post-condición	
ID del caso de Uso	Nombre
124	Crear Perfil del sistema
Descripción	
El actor desea crear un perfil (rol) en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
1. El actor ingresa a la opción “Perfiles” de la sección “Seguridad” del sistema. 2. El actor da click sobre el botón “Nuevo”. 3. El actor ingresa nombre, descripción y da click sobre el botón “Registrar”. 4. El sistema registra al perfil en la Base de Datos. 5. El actor es transferido a la lista de perfiles del sistema.	
Post-condición	
El perfil está en la lista de Perfiles del sistema.	

ID del caso de Uso	Nombre
125	Modificar Perfil
Descripción	
El actor desea modificar un perfil (rol) en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor está en la vista de perfiles del sistema.	
Flujo	
1. El actor da click sobre el link “Editar” perfil. 2. El actor ingresa los cambios y da click sobre el botón “Actualizar”. 3. El sistema actualiza el registro del perfil en la Base de Datos. 4. El actor es transferido a la lista de perfiles del sistema.	
Post-condición	
El perfil está actualizado en el sistema.	

ID del caso de Uso	Nombre
126	Eliminar Perfil
Descripción	
El actor desea eliminar un Perfil del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor está en la vista de perfiles del sistema.	
Flujo	
1. El actor da click sobre el link “Borrar” perfil. 2. El sistema elimina el registro del perfil en la Base de Datos.	

3. El actor es transferido a la lista de perfiles del sistema.
Post-condición
El perfil no aparece en la lista de perfiles del sistema.

ID del caso de Uso	Nombre
127	Ver Perfil
Descripción	
El actor desea ver la información del perfil del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor está en la vista de perfiles del sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre el nombre del perfil. 2. El sistema presenta una vista con la información del perfil. 	
Post-condición	

ID del caso de Uso	Nombre
128	Crear Módulo
Descripción	
El actor desea crear un Módulo en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 6. El actor ingresa a la opción “Módulos” de la sección “Seguridad” del sistema. 7. El actor da click sobre el botón “Nuevo”. 8. El actor selecciona el tipo de módulo, el módulo padre, ingresa el nombre, url y descripción, y da click sobre el botón “Registrar”. 9. El sistema registra al Módulo en la Base de Datos. 10. El actor es transferido a la lista de módulos del sistema. 	
Post-condición	
El módulo está en la lista de módulos del sistema.	

ID del caso de Uso	Nombre
129	Modificar Módulo
Descripción	
El actor desea modificar un módulo en el sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor está en la vista de módulos del sistema.	
Flujo	
<ol style="list-style-type: none"> 5. El actor da click sobre el link “Editar” módulo. 6. El actor ingresa los cambios y da click sobre el botón “Actualizar”. 7. El sistema actualiza el registro del módulo en la Base de Datos. 8. El actor es transferido a la lista de módulos del sistema. 	
Post-condición	

El módulo está actualizado en el sistema.

ID del caso de Uso	Nombre
130	Eliminar Módulo
Descripción	
El actor desea eliminar un módulo del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor está en la vista de módulos del sistema.	
Flujo	
<ol style="list-style-type: none"> 4. El actor da click sobre el link “Borrar” módulo. 5. El sistema elimina el registro del módulo en la Base de Datos. 6. El actor es transferido a la lista de módulos del sistema. 	
Post-condición	
El módulo no aparece en la lista de módulos del sistema.	

ID del caso de Uso	Nombre
131	Ver Módulo
Descripción	
El actor desea ver la información del módulo del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor está en la vista de módulos del sistema.	
Flujo	
<ol style="list-style-type: none"> 3. El actor da click sobre el nombre del módulo. 4. El sistema presenta una vista con la información del módulo. 	
Post-condición	

ID del caso de Uso	Nombre
132	Asignar Permiso a Perfil
Descripción	
El actor desea añadir permisos a un perfil (rol) del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa a la opción “Perfil - accesos” de la sección “Seguridad”. 2. El actor selecciona un Perfil. 3. El sistema lista los accesos de manera organizada. 4. El actor checkea los accesos a las cuales desea añadir y da click en el botón “Guardar”. 5. El sistema guarda el registro de los accesos en la Base de Datos. 6. El sistema actualiza la vista de accesos. 	
Post-condición	
El Perfil tiene los nuevos permisos asignados en el sistema.	

ID del caso de Uso	Nombre
133	Eliminar Permiso de Perfil
Descripción	
El actor desea añadir permisos a un perfil (rol) del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de accesos de un Perfil.	
Flujo	
<ol style="list-style-type: none"> 1. El actor quita los accesos seleccionados y da click en el botón “Guardar”.. 2. El sistema actualiza la vista de accesos. 	
Post-condición	
El Perfil tiene no tiene permisos quitados.	

ID del caso de Uso	Nombre
134	Ver Permiso
Descripción	
El actor desea visualizar los permisos que tiene un Perfil (Rol).	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la vista de accesos de un Perfil.	
Flujo	
<ol style="list-style-type: none"> 1. El actor selecciona un Perfil 2. El sistema muestra los permisos asignados al Perfil. 	
Post-condición	

ID del caso de Uso	Nombre
135	Crear Usuario
Descripción	
El actor desea crear un usuario de acceso al sistema.	
Pre-condición	
El actor se encuentra logeado al sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor ingresa a la opción “Usuarios” de la sección “Seguridad” del sistema. 2. El actor presiona sobre el botón “Nuevo usuario” 3. El actor busca a la persona de quien desea crear el usuario, ingresa usuario y contraseña. 4. El sistema valida el nombre del usuario. 5. Si es correcto, el actor presiona sobre el botón “Crear usuario”. 6. El sistema registra el nuevo usuario en la Base de Datos. 7. El actor es transferido a la lista de usuarios del sistema. 	
Post-condición	
El nuevo usuario está registrado en el sistema.	

ID del caso de Uso	Nombre
136	Modificar Usuario

Descripción
El actor desea modificar un usuario de acceso al sistema.
Pre-condición
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de usuarios del sistema.
Flujo
<ol style="list-style-type: none"> 1. El actor da click sobre el link "Editar". 2. El actor ingresa la información a actualizar. 3. El sistema actualiza el registro del usuario en la Base de Datos. 4. El actor es transferido a la lista de usuarios del sistema.
Post-condición
El usuario tiene su información actualizada en el sistema.

ID del caso de Uso	Nombre
137	Eliminar Usuario
Descripción	
El actor desea eliminar un usuario del sistema.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de usuarios del sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre el link "Borrar" y confirma la eliminación. 2. El sistema elimina el registro del usuario de la Base de Datos. 3. El actor es transferido a la lista de usuarios del sistema. 	
Post-condición	
El usuario no aparece en la lista del sistema.	

ID del caso de Uso	Nombre
138	Ver usuario
Descripción	
El actor desea ver la información de un usuario.	
Pre-condición	
El actor se encuentra logeado al sistema. El actor se encuentra en la lista de usuarios del sistema.	
Flujo	
<ol style="list-style-type: none"> 1. El actor da click sobre el nombre del usuario. 2. El sistema presenta la información del usuario y el acceso que tiene a los grupos organizados. 	
Post-condición	

5. Especificaciones suplementarias

5.1. Usabilidad

5.1.1. Cumplimiento de las expectativas de Aplicaciones Web

El sistema debe cumplir con las expectativas de los usuarios hacia aplicaciones web. Incluye el acceso exclusivo a través de un navegador web, uso de links y botones para el proceso de navegación, y solo botones como mecanismos de ejecución de acciones.

5.1.2. Tiempo de aprendizaje para el usuario

El sistema debe ser muy intuitivo, permitiendo el fácil aprendizaje para cualquier persona con conocimiento demasiado básico en la institución. Además debe permitir a los usuarios la navegación simple según las opciones que tienen acceso.

5.2. Confiabilidad

5.2.1. Integridad de la información

El sistema debe asegurar que siempre la información se encuentre consistente en cada proceso de los Grupos Organizados.

5.2.2. Seguimiento de las transacciones

El sistema debe llevar un registro básico de las transacciones que se registran en él.

5.3. Mantenibilidad

5.3.1. Uso de un mecanismo de control de versiones.

Todo código del sistema en cada de desarrollo y mantenimiento, se debe guardar en cada versión en un espacio de memoria.

5.4. Seguridad

5.4.1. Acceso restringido a usuarios autenticados

El sistema solo debe permitir a los usuarios autenticados la realización de cualquier operación. Solo los usuarios con accesos permitidos podrán realizar dichas operaciones.