

# PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

## FACULTAD DE CIENCIAS E INGENIERÍA



Analizar, diseñar e implementar un sistema de información que soporte el proceso de Gestión de Solicitudes de Servicio del Ministerio de Comunicaciones de una Iglesia Evangélica

Tesis para optar por el Título de Ingeniero Informático, que presenta el bachiller:

Aldo Francisco Román Nureña

ASESOR: Ing. Rony Cueva Moscoso

Lima, Junio de 2019

## RESUMEN

La Iglesia Evangélica caso de estudio es una entidad religiosa conformada por personas vinculadas por una misma fe en Cristo. La iglesia (en adelante, también denominada "la organización") tiene por misión evangelizar a las comunidades con las que tiene contacto, transmitiendo las enseñanzas según la Biblia. En ese sentido, la iglesia ha experimentado crecimiento durante los últimos años en número de miembros, cantidad de sedes, y actividades internas y externas. Actualmente cuenta con 13 localidades y un aproximado de 6 mil miembros.

El principal proceso de la organización es la Evangelización, que consiste en la comunicación de los principios bíblicos a sus comunidades cercanas. Dicho proceso es soportado por distintas actividades, entre las que se encuentran reuniones de celebración, dictado de cursos, retiros, entre otros. Estas actividades son organizadas por alguna de las áreas de la organización, según la competencia de cada una. Para fines de este plan de proyecto, se denomina también Ministerio a cada una de dichas áreas.

Durante la organización de estas actividades, se debe lograr la difusión y cobertura de las mismas, tarea a cargo del Ministerio de Comunicaciones. El ministerio tiene por tarea elaborar publicidad, realizar cobertura audiovisual, y/o crear piezas gráficas de mayor complejidad.

Actualmente, existen procesos y estructuras organizacionales que soportan estas labores, aunque muchos de estos procesos son manuales o de alta intervención humana. El Ministerio de Comunicaciones recepciona las solicitudes de servicio a través de correo electrónico, las registra en hojas de cálculo, asigna un colaborador responsable, un equipo de trabajo y planifica la entrega en un cronograma. Debido a la cantidad de solicitudes recibidas, este trabajo es lento, y altamente expuesto a errores y pérdidas de información. Además, el almacenamiento en hojas de cálculo hace difícil la extracción de información y la generación de reportes, lo que conlleva a un uso poco eficiente de recursos.

Se observa que también es requerido un proceso de planificación que permita priorizar las solicitudes recibidas dado que se cuenta con recursos de producción limitados. La planificación actualmente se realiza procesando las solicitudes en una cola (el primero en entrar, el primero en salir), y considerando una fecha de cierre semanal y un lapso de 15 días para la entrega. Sin embargo, este método de planificación no siempre prioriza las solicitudes correctamente ni otorga una estimación suficiente para la entrega.

Por todo lo mencionado, el Ministerio de Comunicaciones reconoce la dificultad existente en su proceso de gestión de solicitudes de servicio, y desea reducir dicha dificultad, de manera que pueda brindar servicios internos a los demás ministerios de la iglesia con eficiencia y eficacia. Por estos motivos, una de las acciones que la organización desea ejecutar es elaborar un sistema de información que permita registrar las solicitudes de servicio, gestionar dichas solicitudes y generar reportes con información idónea para el área y la iglesia. Asimismo, se requiere de un método adecuado para la planificación de las entregas de cada solicitud, para lo cual se hará uso de un algoritmo heurístico. De esta manera, se busca que el sistema constituya una herramienta de soporte para la gestión de solicitudes de servicio.

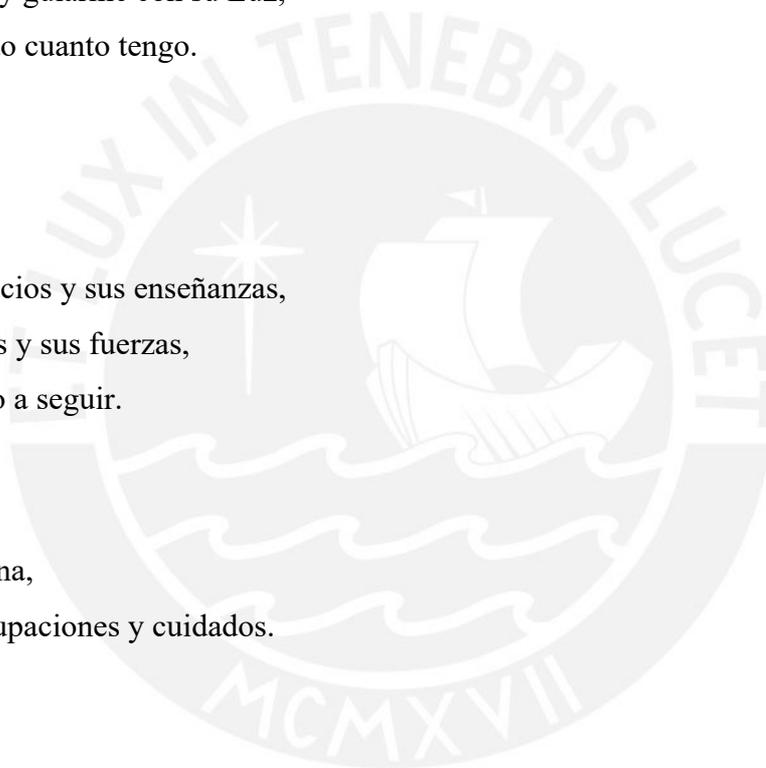
## AGRADECIMIENTOS

*“Estoy seguro de que Dios, que comenzó a hacer su buena obra en ustedes, la irá llevando a buen fin hasta el día en que Jesucristo regrese.” Fil 1:6*

A Dios,  
por sus misericordias e innumerables bendiciones,  
por cuidarme y guiarme con su Luz,  
por darme todo cuanto tengo.

A mamá,  
por sus sacrificios y sus enseñanzas,  
por sus sueños y sus fuerzas,  
por el ejemplo a seguir.

A mamá Albina,  
por sus preocupaciones y cuidados.



## TEMA DE TESIS

<b>PARA OPTAR</b>	: Título profesional de Ingeniero Informático
<b>TEMA</b>	: Analizar, diseñar e implementar un sistema de información que soporte el proceso de gestión de solicitudes de servicio del Ministerio de Comunicaciones de una Iglesia Evangélica
<b>ÁREA</b>	: Sistemas de Información
<b>ASESOR</b>	: Ing. Rony CUEVA MOSCOSO
<b>ALUMNO(S)</b>	: Aldo Francisco ROMÁN NUREÑA - 20100873
<b>FECHA</b>	: 20 de junio de 2019
<b>MÁXIMO</b>	: 100 páginas

---

### DESCRIPCIÓN

La Iglesia Evangélica caso de estudio es una entidad religiosa conformada por personas vinculadas por una misma fe en Cristo. La iglesia (en adelante, también denominada "la organización") tiene por misión evangelizar a las comunidades con las que tiene contacto, transmitiendo las enseñanzas según la Biblia. En ese sentido, la iglesia ha experimentado crecimiento durante los últimos años en número de miembros, cantidad de sedes, y actividades internas y externas. Actualmente cuenta con 13 localidades y un aproximado de 6 mil miembros.

El principal proceso de la organización es la Evangelización, que consiste en la comunicación de los principios bíblicos a sus comunidades cercanas. Dicho proceso es soportado por distintas actividades, entre las que se encuentran reuniones de celebración, dictado de cursos, retiros, entre otros. Estas actividades son organizadas por alguna de las áreas de la organización, según la competencia de cada una. Para fines de este plan de proyecto, se denomina también Ministerio a cada una de dichas áreas.

Durante la organización de estas actividades, se debe lograr la difusión y cobertura de las mismas, tarea a cargo del Ministerio de Comunicaciones. El ministerio tiene por tarea elaborar publicidad, realizar cobertura audiovisual, y/o crear piezas gráficas de mayor complejidad.

Actualmente, existen procesos y estructuras organizacionales que soportan estas labores, aunque muchos de estos procesos son manuales o de alta intervención humana. El Ministerio de Comunicaciones recibe las solicitudes de servicio a través de correo electrónico, las registra en hojas de cálculo, asigna un colaborador responsable, un equipo de trabajo y planifica la entrega en un cronograma. Debido a la cantidad de solicitudes recibidas, este trabajo es lento, y altamente expuesto a errores y pérdidas de información. Además, el almacenamiento en hojas de cálculo

hace difícil la extracción de información y la generación de reportes, lo que conlleva a un uso poco eficiente de recursos.

Se observa que también es requerido un proceso de planificación que permita priorizar las solicitudes recibidas dado que se cuenta con recursos de producción limitados. La planificación actualmente se realiza procesando las solicitudes en una cola (el primero en entrar, el primero en salir), y considerando una fecha de cierre semanal y un lapso de 15 días para la entrega. Sin embargo, este método de planificación no siempre prioriza las solicitudes correctamente ni otorga una estimación suficiente para la entrega.

Por todo lo mencionado, el Ministerio de Comunicaciones reconoce la dificultad existente en su proceso de gestión de solicitudes de servicio, y desea reducir dicha dificultad, de manera que pueda brindar servicios internos a los demás ministerios de la iglesia con eficiencia y eficacia.

Por estos motivos, una de las acciones que la organización desea ejecutar es elaborar un sistema de información que permita registrar las solicitudes de servicio, gestionar dichas solicitudes y generar reportes con información idónea para el área y la iglesia. Asimismo, se requiere de un método adecuado para la planificación de las entregas de cada solicitud, para lo cual se hará uso de un algoritmo heurístico. De esta manera, se busca que el sistema constituya una herramienta de soporte para la gestión de solicitudes de servicio.

#### **OBJETIVO GENERAL**

Analizar, diseñar e implementar un sistema de información que soporte el proceso de gestión de solicitudes de servicio del Ministerio de Comunicaciones de la Iglesia Evangélica caso de estudio.

#### **OBJETIVOS ESPECÍFICOS**

Los objetivos específicos son:

OE1. Modelar el proceso de recolección de solicitudes de servicio. Se requiere definir los datos requeridos, datos opcionales, actores y flujos de datos.

OE2. Diseñar la arquitectura de software del sistema de información que soporte el proceso de gestión de solicitudes de servicio.

OE3. Implementar un algoritmo que permita maximizar la cantidad de solicitudes de servicio que pueden ser completadas con los recursos disponibles.

OE4. Implementar un prototipo funcional del sistema de información para la gestión de solicitudes de servicio.



### ALCANCE

El presente proyecto pretende desarrollar un sistema de información web que permita gestionar las solicitudes de servicio recibidas por el Ministerio de Comunicaciones de la Iglesia Evangélica caso de estudio. Para ello será necesario primero modelar el proceso de gestión de solicitudes de servicio, para un correcto análisis y diseño del sistema de información.

El sistema permitirá el registro de dichas solicitudes en formularios de flujos complejos, asociando la solicitud a un solicitante y a un responsable dentro del equipo. Ambos usuarios serán notificados acerca del estado de la solicitud durante todo el ciclo de vida de la misma.

El sistema permitirá a sus administradores ingresar las características de cada espacio disponible por la Iglesia, tales como aforo, presencia de material multimedia, cantidad de micrófonos, entre otros. Ello permitirá una mayor eficiencia en la atención de las solicitudes de servicio recepcionadas.

Asimismo, el sistema ofrecerá una lista priorizada de las solicitudes recibidas y una estimación de la cantidad de horas-hombre que tomaría cada solicitud. Esto será posible utilizando un algoritmo heurístico, el cual analizará los datos de cada solicitud para otorgar un orden de importancia y estimará la carga de trabajo necesaria según dichos datos.

## TABLA DE CONTENIDO

<b>1.</b>	<b><u>DEFINICION DEL PROBLEMA</u></b>	<b>9</b>
1.	PROBLEMÁTICA.....	9
2.	OBJETIVO GENERAL.....	10
3.	OBJETIVOS ESPECÍFICOS .....	11
4.	RESULTADOS ESPERADOS.....	11
5.	HERRAMIENTAS, MÉTODOS, METODOLOGÍAS Y PROCEDIMIENTOS.....	12
1.	HERRAMIENTAS.....	12
2.	MÉTODOS Y PROCEDIMIENTOS.....	14
3.	METODOLOGÍAS .....	15
6.	ALCANCE .....	15
1.	LIMITACIONES .....	16
2.	RIESGOS .....	16
7.	ESTUDIO DE VIABILIDAD.....	16
1.	JUSTIFICATIVA .....	16
2.	ANÁLISIS DE ALTERNATIVAS.....	17
3.	VIABILIDAD .....	18
<b>2.</b>	<b><u>MARCO CONCEPTUAL</u></b>	<b>20</b>
1.	ESTRUCTURA ORGANIZACIONAL .....	20
2.	SOLICITUDES DE SERVICIO.....	21
3.	PROCESO DE GESTIÓN DE SOLICITUD DE SERVICIO.....	21
4.	CONCLUSIÓN .....	22
<b>3.</b>	<b><u>ESTADO DEL ARTE</u></b>	<b>23</b>
1.	SOFTWARE DE GESTIÓN Y RECOLECCIÓN DE DATOS.....	23
1.	TOUCHPOINT SOFTWARE .....	23
2.	CHURCH WINDOWS .....	23
3.	GOOGLE FORMS.....	24
2.	ALGORITMOS PARA PLANIFICACIÓN DE TAREAS.....	25
1.	ALGORITMO BASADO EN RESERVAS.....	25
2.	ALGORITMO PARA TAREAS DEPENDIENTES.....	26
3.	ALGORITMO BASADO EN RESTRICCIONES DE PLAZO .....	26
4.	ALGORITMO PARA LA PLANIFICACIÓN DE PROYECTOS .....	26
3.	CONCLUSIONES SOBRE EL ESTADO DEL ARTE.....	27
<b>4.</b>	<b><u>MODELADO DE PROCESO</u></b>	<b>28</b>
1.	PROCESO DE GESTIÓN DE SOLICITUDES DE SERVICIO .....	28

<b>2.</b>	<b>REGISTRO DE SOLICITUDES DE SERVICIO.....</b>	<b>29</b>
1.	INFORMACIÓN GENERAL.....	29
2.	PUBLICIDAD.....	30
3.	ATENCIÓN DE EVENTO .....	30
4.	PIEZAS GRÁFICAS .....	32
<b>3.</b>	<b>CONCLUSIONES .....</b>	<b>33</b>
<b><u>5.</u></b>	<b><u>ARQUITECTURA DE SOFTWARE</u></b>	<b><u>34</u></b>
1.	VISTA LÓGICA .....	34
2.	VISTA DE PROCESOS.....	35
3.	VISTA DE IMPLEMENTACIÓN.....	36
4.	VISTA FÍSICA.....	38
5.	VISTA DE ESCENARIOS .....	39
6.	CONCLUSIONES .....	40
<b><u>6.</u></b>	<b><u>ALGORITMO DE PLANIFICACIÓN DE TAREAS</u></b>	<b><u>42</u></b>
1.	DESCRIPCIÓN.....	42
2.	SEUDOCÓDIGO.....	42
3.	CONCLUSIONES .....	45
<b><u>7.</u></b>	<b><u>PROTOTIPO FUNCIONAL</u></b>	<b><u>46</u></b>
1.	PANTALLA INICIAL.....	46
2.	REGISTRO DE SOLICITUD DE SERVICIO.....	47
1.	INFORMACIÓN GENERAL .....	48
2.	PUBLICIDAD.....	49
3.	ATENCIÓN DE EVENTO .....	49
4.	PIEZAS GRÁFICAS .....	51
5.	ENVIAR SOLICITUD.....	51
<b>3.</b>	<b>CONCLUSIONES .....</b>	<b>52</b>
<b><u>8.</u></b>	<b><u>REFERENCIAS BIBLIOGRÁFICAS</u></b>	<b><u>55</u></b>

## 1. DEFINICIÓN DEL PROBLEMA

En esta sección se describe el problema tratado en el presente proyecto de fin de carrera, así como a la organización y el contexto en el cual se desenvuelve dicho problema. A continuación, se presentan los objetivos y resultados esperados para este proyecto, así como las herramientas, métodos y metodologías seleccionadas para su consecución, el alcance del proyecto y el estudio de viabilidad realizado.

### 1. Problemática

La Iglesia Evangélica, cuyo nombre se mantendrá en reserva por confidencialidad y será denominada como Iglesia Evangélica o la Organización, es una entidad religiosa conformada por personas vinculadas por una misma fe en Cristo y tiene por misión evangelizar a las comunidades con las que tiene contacto, transmitiendo las enseñanzas según la Biblia. En ese sentido, la Organización ha experimentado crecimiento durante los últimos años en materia de número de miembros, cantidad de sedes, y actividades internas y externas.

En el Perú, la iglesia evangélica tiene presencia desde los años previos a la Independencia y desde entonces ha crecido paulatinamente (De La Puente, 1974; Amat, Perez, 2007). Según el último Censo de Población y Vivienda del Instituto Nacional de Estadística e Informática (INEI), realizado en 2007, el 12.5% de la población de 12 y más años de edad profesa la religión evangélica. Asimismo, según el censo, la cantidad de congregantes evangélicos se ha duplicado en los últimos 15 años, siendo el porcentaje de congregantes de 6.8% en el Censo de 1993. Además, los resultados del censo declaran un crecimiento promedio anual de 6.6% (INEI, 2008).

Por su parte, la Iglesia Evangélica fue fundada en 1987, cuenta con 11 localidades en Lima, 1 localidad en Santiago de Chile y 1 localidad en New Jersey. Entre todas las localidades, la Iglesia Evangélica cuenta con un aproximado de 6 mil miembros (Román, 2015).

El principal proceso de la Organización es la Evangelización, que consiste en la comunicación de los principios bíblicos a los miembros, visitantes y comunidades cercanas a la Organización. Dicho proceso es soportado por distintas actividades, entre las que se encuentran reuniones de celebración, dictado de cursos, celebraciones especiales, actividades especiales, entre otros. Estas actividades, que pueden ser orientadas a un público interno o externo, son organizadas por alguna de las áreas de la organización, según la competencia de cada área. Para fines de este plan de proyecto, se denomina también Ministerio a cada una de dichas áreas.

Dentro del proceso de la organización de estas actividades, se debe lograr la difusión y cobertura de las mismas, tarea a cargo del Ministerio de Comunicaciones. En esta competencia, el Ministerio tiene por labor elaborar la publicidad de las actividades

organizadas, realizar la cobertura audiovisual de la actividad, y diseñar y desarrollar piezas gráficas de mayor complejidad, como libros o vídeos, en colaboración con el ministerio encargado de la actividad.

Actualmente, para dicho fin, el Ministerio de Comunicaciones tiene procesos y estructuras organizacionales que soportan sus labores, aunque muchos de estos procesos son manuales o de alta intervención humana. El Ministerio de Comunicaciones recepciona las solicitudes de servicio a través de comunicación vía correo electrónico, las registra en hojas de cálculo, asigna un colaborador responsable de la solicitud, un equipo de trabajo, de ser necesario, y planifica la entrega en un cronograma. Debido a la cantidad de solicitudes recibidas por el área, este trabajo es lento, y altamente expuesto a errores y pérdidas de información. Además, el almacenamiento en hojas de cálculo hace difícil la extracción de información y la generación de reportes, lo que conlleva a un uso poco eficiente de sus recursos.

Se observa que también es requerido un proceso de planificación que permita priorizar las solicitudes recibidas dado que se cuenta con recursos humanos limitados. La planificación actualmente se realiza procesando las solicitudes en una cola (el primero en entrar, el primero en salir), y considerando una fecha de cierre semanal y un lapso de 15 días para la entrega. Sin embargo, este método de planificación no siempre prioriza las solicitudes correctamente ni otorga una estimación suficiente para la entrega.

Por todo lo mencionado, el Ministerio de Comunicaciones reconoce la dificultad existente en su proceso de gestión de solicitudes de servicio, y desea reducir dicha dificultad, de manera que pueda brindar servicios internos a los demás ministerios de la Iglesia Evangélica con eficiencia y eficacia.

Cabe resaltar que la Organización cuenta con un sistema de información interno, desarrollado *in-house* por el equipo de Soporte Técnico. Sin embargo, este equipo no cuenta con disponibilidad para llevar a cabo este sistema informático, dado que actualmente atiende otros requerimientos solicitados por la Organización.

Por estos motivos, una de las acciones que el Ministerio de Comunicaciones desea adoptar para un tratamiento eficaz del problema planteado es elaborar un sistema de información que permita registrar las solicitudes de servicio, gestionar dichas solicitudes y generar reportes con información idónea para el área y la Iglesia. Asimismo, se requiere de un método adecuado para la planificación de las entregas de cada solicitud, para lo cual se hará uso de un algoritmo heurístico. De esta manera, se busca que el sistema constituya una herramienta de soporte para la gestión de solicitudes de servicio.

## **2. Objetivo general**

Analizar, diseñar e implementar un sistema de información que soporte el proceso de gestión de solicitudes de servicio del Ministerio de Comunicaciones de la Iglesia Evangélica.

### **3. Objetivos específicos**

Los objetivos específicos son:

1. Modelar el proceso de recolección de solicitudes de servicio. Se requiere definir los datos requeridos, datos opcionales, actores y flujos de datos.
2. Diseñar la arquitectura de software del sistema de información que soporte el proceso de gestión de solicitudes de servicio.
3. Implementar un algoritmo que permita maximizar la cantidad de solicitudes de servicio que pueden ser completadas con los recursos disponibles, a través de la priorización y la estimación de las horas-hombre de solicitudes de servicio.
4. Implementar un prototipo funcional del sistema de información para la gestión de solicitudes de servicio.

### **4. Resultados esperados**

Los resultados esperados son:

1. Para el objetivo 1: Diagramas de proceso de gestión de solicitudes de servicio, que refleje los procesos nuevos o modificados para la gestión de solicitudes de servicio.
2. Para el objetivo 1: Modelado de procesos aprobado mediante Carta de conformidad, firmado por representante de la Organización.
3. Para el objetivo 2: Documento de arquitectura de software, indicando las características de arquitectura del sistema de información y los escenarios caso de uso críticos validados por la Iglesia Evangélica.
4. Para el objetivo 3: Módulo de planificación, para sugerir la lista de solicitudes de servicio recibidas que pueden cubrirse con los recursos disponibles, acompañado de una estimación de la prioridad y las horas-hombre necesarias para cada solicitud. Se desarrollará un algoritmo heurístico para maximizar el uso de recursos y tiempo.
5. Para el objetivo 4: Prototipo funcional del sistema de información que permita administrar las solicitudes de servicio recibidas.

## 5. Herramientas, métodos, metodologías y procedimientos

En esta sección se describen las herramientas, métodos y procedimientos que serán utilizados para lograr los resultados esperados. En la Tabla 1.1 se listan las herramientas que serán empleadas para lograr cada resultado esperado.

Tabla 1.1. Herramientas, métodos y procedimientos

Resultados esperados	Herramientas, métodos o procedimientos
RE1: Diagramas de proceso de gestión de solicitudes de servicio.	- Business Process Modeling Notation o BPMN (Notación para el Modelado de Procesos de Negocio) - Bizagi BPMN Modeler
RE3: Documento de arquitectura de software	- Modelo de Arquitectura 4+1
RE4: Módulo de planificación (algoritmo heurístico)	- Java - Play Framework - IntelliJ IDEA - Algoritmo heurístico
RE5: Prototipo funcional de sistema de información	- Java - Play Framework - IntelliJ IDEA - AngularJS - Material Design

### 1. Herramientas

#### 1. Bizagi BPMN Modeler

Bizagi BPMN Modeler es una herramienta para la creación de diagramas, así como para la documentación y simulación de procesos. La herramienta utiliza el formato estándar BPMN y permite exportar los diagramas generados en formatos PDF, MS Visio, JPG, PNG, entre otros (Bizagi, 2015).

Se ha seleccionado esta herramienta dado su reconocimiento en la industria para la generación de diagramas BPMN (BPMF, 2014), y dada la experiencia del tesista en su uso.

#### 2. Lenguaje de programación Java

Java es un lenguaje de programación diseñado por James Gosling y liberado en 1995, de propósito general, orientado a clases, estáticamente tipado, diseñado para tener pocas dependencias de implementación (Oracle, 2015). Su sintaxis deriva y es similar a

la sintaxis de los lenguajes de programación C y C++ y propone el principio “Escribir una vez, ejecutar en todos lados”; esto es, los programas escritos en Java pueden ser ejecutados en una gran variedad de plataformas, desde teléfonos móviles hasta servidores empresariales (Oracle, 2015).

En agosto de 2015, Java es el lenguaje de programación más popular, con un incremento de 4.3% respecto a junio de 2014. (TIOBE, 2015).

Se ha seleccionado este lenguaje de programación dada la sencillez que presenta para el desarrollo de software, la experiencia técnica en el uso del lenguaje por parte del tesista y la capacidad de ejecutar el código tanto en computadores de escritorio como en servidores.

### **3. Play Framework**

Es un software utilizado para escribir aplicaciones web en Java y Scala, construido para ser ligero, predecible y de bajo consumo de recursos, lo que lo hace ideal para sistemas altamente escalables (Typesafe, 2015).

Play Framework propone el uso del patrón MVC, que, según lo describe Microsoft (Microsoft, 2016), supone la separación de responsabilidades de cada componente de software en Modelo (conocimiento acerca de los datos), Vista (conocimiento de cómo se muestran los datos a los usuarios) y Controlador (conocimiento de la lógica de negocio del sistema). Esto ayuda a mantener orden, cohesión y coherencia dentro del código de la aplicación.

Las aplicaciones web escritas con Play Framework pueden ser posteriormente empaquetadas en diversos formatos, entre ellos ejecutables de Windows, paquetes RPM para Red Hat Linux y derivados, paquetes DEB para Debian Linux y derivados, y paquetes JAR para ser ejecutados en cualquier ambiente soportado por la Java Virtual Machine (Typesafe, 2015).

Se ha seleccionado esta herramienta dada la sencillez de la arquitectura propuesta (basada en MVC) y la experiencia técnica en su uso por parte del tesista.

### **4. IntelliJ IDEA**

IntelliJ IDEA es un entorno de desarrollo de software (IDE, por sus siglas en inglés) para desarrollo de software computadora, desarrollado por JetBrains (JetBrains, 2015). Según su fabricante, esta herramienta permite la escritura, compilación, depuración y ejecución de software; así como soporta una gran variedad de lenguajes y marcos de trabajo, entre ellos, soporta Java 8 completamente.

Según Krochmalski, el objetivo clave de IntelliJ IDEA es incrementar y asistir la productividad del desarrollador; y afirma que el IDE muestra sugerencias inteligentes y relevantes, así como completado de código, análisis de código sobre la marcha (Krochmalski, 2014).

Existe una versión gratuita denominada Community Edition, así como una versión comercial llamada Ultimate Edition (JetBrains, 2016). Según la web del fabricante, ambas versiones permiten desarrollar software comercial; esto es, que luego será vendido. Sin embargo, la versión comercial posee funcionalidades no disponibles para la versión gratuita.

Se ha seleccionado esta herramienta por su enfoque en mejorar la productividad del desarrollador, su integración con Java 8 y a la disponibilidad de una licencia por parte del tesista.

## **5. AngularJS**

AngularJS es un framework de código abierto utilizado para la creación de aplicaciones web. AngularJS propone la arquitectura MVC para aplicaciones web, y ofrece como características ligadura de datos, plantillas en el lado de cliente, entre otros (Angular, 2015).

Se ha seleccionado esta herramienta dado que según la experiencia del tesista ayuda a estructurar aplicaciones web complejas e incrementa la productividad.

## **6. Material Design**

Material Design es un lenguaje visual elaborado por Google que propone los principios clásicos del diseño gráfico considerando la innovación y la posibilidad de la ciencia y la tecnología actual (Google, 2016). El objetivo de este lenguaje visual es “desarrollar un único sistema subyacente que permita una experiencia unificada a través de diversas plataformas y tamaños de dispositivos” (Google, 2016).

## **2. Métodos y Procedimientos**

### **1. Business Process Modeling Notation (BPMN)**

BPMN es una notación utilizada para representar de forma gráfica procesos de negocio, útil para soportar todo proceso de negocio posible, siendo suficientemente comprensible tanto para usuarios técnicos como para usuarios de negocios (OMG, 2013).

Se ha elegido esta notación dado que es una notación validada para la especificación de procesos de negocio y dada la experiencia en su uso por parte del tesista.

### **2. Algoritmo GRASP**

GRASP es un algoritmo metaheurístico que permite generar diferentes soluciones a un problema de manera iterativa, perfeccionando la solución encontrada en cada iteración, que contiene un componente aleatorio que permite alterar las soluciones con el fin de acercarlas a la solución óptima global (Feo, Resende, 1995). Los algoritmos GRASP se caracterizan por lograr un gran acercamiento al óptimo global y a su rápida convergencia en los resultados (Feo, Resende, 1995).

El algoritmo plantea el uso de un coeficiente de relajación, a través del cual se selecciona un conjunto de posibles soluciones que serán evaluadas posteriormente, lo cual representa una ventaja sobre los algoritmos voraces ya que permitirá evaluar la función objetivo sobre diversos elementos (y sus vecindades) y no solo sobre uno –aparentemente el mejor- como en los algoritmos voraces (Feo, Resende, 1995).

Se ha seleccionado este algoritmo dado que ayuda a obtener una solución, no necesariamente óptima global, a problemas de optimización combinatoria, caso del presente proyecto.

### **3. Metodologías**

#### **1. Modelo de Arquitectura 4+1**

El modelo 4+1 es un modelo propuesto por Philippe Krutchen para describir la arquitectura de sistemas de información utilizando múltiples vistas, que son descripciones de la arquitectura desde los puntos de vista de distintas partes interesadas, como desarrolladores, usuarios y gestores de proyecto (Krutchen, 1995).

Las vistas propuestas son (Krutchen, 1995):

- Vista lógica: muestra la funcionalidad que el sistema da a los usuarios finales.
- Vista de implementación: enfocada en los desarrolladores, busca describir los componentes del software.
- Vista de procesos: se enfoca en el comportamiento del sistema cuando está en funcionamiento, detallando los procesos, flujos de información y comunicación del sistema.
- Vista física: describe la topología del sistema en la capa física, así como las conexiones físicas entre distintos componentes.
- Vista de escenarios: describe la secuencia de interacciones entre el sistema y otros agentes externos, como los usuarios u otros sistemas.

Se ha elegido este Modelo debido a que es uno validado para la documentación de arquitecturas de software. Además, el tesista cuenta con experiencia en su uso aplicado a sistemas de información. Se utilizarán las cinco vistas propuestas para la elaboración de este proyecto, dado que son necesarias en su estructuración.

### **6. Alcance**

El presente proyecto pretende desarrollar un sistema de información web que permita gestionar las solicitudes de servicio recibidas por el Ministerio de Comunicaciones de la Iglesia Evangélica. El sistema permite el registro de dichas solicitudes en formularios de flujos complejos, asociando la solicitud a una solicitante y

un responsable dentro del equipo. Ambos usuarios serán notificados acerca del estado de la solicitud durante todo el ciclo de vida de la misma.

El sistema permite a sus administradores ingresar las características de cada espacio disponible por la Iglesia, tales como aforo, presencia de material multimedia, cantidad de micrófonos, entre otros. Ello permite una mayor eficiencia en la atención de las solicitudes de servicio recepcionadas.

Asimismo, el sistema permite el procesamiento de las solicitudes recibidas para proponer una lista priorizada de las solicitudes que deben atenderse, acompañadas de la cantidad de horas-hombre que tomaría cada una. Esto será posible utilizando un algoritmo basado en reglas, el cual identificará los datos de cada solicitud para determinar su importancia y el esfuerzo necesario de cada solicitud. Con estos datos, un algoritmo GRASP buscará la mejor combinación de solicitudes que deben atenderse.

## **1. Limitaciones**

Las limitaciones establecidas al inicio del Proyecto fueron:

- El sistema será desarrollado únicamente en el idioma español.
- El sistema será diseñado para ejecutarse en el sistema operativo Linux, distribución CentOS, sobre la Máquina Virtual Java versión 8.
- El sistema será desarrollado en un entorno Web, por lo que solo podrá ser accedido por navegadores web modernos, sea Internet Explorer 10, Chrome, Firefox y Safari en sus últimas versiones.
- No está garantizada la correcta ejecución y acceso al sistema a través de navegadores móviles.

## **2. Riesgos**

A continuación, se presenta en la Tabla 1.2 los riesgos identificados para el presente Proyecto, así como el impacto que podría ocasionar su materialización y las medidas tomadas para mitigarlos.

## **7. Estudio de viabilidad**

A continuación, se detalla la justificativa y análisis de viabilidad del presente plan de proyecto.

### **1. Justificativa**

Como se comentó en la problemática del presente plan de proyecto, el Ministerio de Comunicaciones de la Iglesia Evangélica recibe una cantidad de solicitudes de servicio que no puede gestionar de forma manual. Esto se debe a la complejidad de cada solicitud, la frecuencia con que se reciben y el alto grado de error humano al que está expuesto el proceso.

Tabla 1.2. Riesgos del proyecto

Riesgo identificado	Impacto en el proyecto	Medidas de mitigación
Pérdida de la información	Alto. Imposibilidad de evaluar el proyecto	Hospedar una copia de la información en Google Drive
Pérdida del código fuente	Alto. Imposibilidad de continuar el proyecto	Utilizar un repositorio remoto para hospedar una copia del código fuente.
Modelado erróneo del proceso de gestión de solicitudes de servicio	Alto. Retraso en cronograma de proyecto para ejecutar correcciones.	Validar de forma frecuente y continua los procesos modelados, con representante de la Organización.
Baja tasa de acierto en la planificación de fechas de entrega por parte del algoritmo propuesto	Alto. Retraso en cronograma de proyecto para ejecutar correcciones.	Investigación del algoritmo. Recibir opiniones y ayuda por parte de docentes calificados.

Por esta razón, la Organización ve necesaria la implementación de un sistema de información que permita gestionar este proceso de forma automática. En ese sentido, el proyecto propuesto tiene una justificación práctica, resolviendo los problemas de procesar dicha información y mantener la integridad de los datos a lo largo del proceso.

Adicionalmente, la organización afirma que es difícil estimar qué solicitudes deben atenderse primero y cuáles después, dado que se dispone de recursos limitados para este análisis y a que la cantidad de solicitudes recibidas va en aumento. Finalmente, el análisis de estimación de la cantidad de horas-hombre que toma cada solicitud se realiza de forma subjetiva, lo cual produce estimaciones erróneas. Este es el motivo por el cual se propone el uso de un algoritmo heurístico que permita priorizar la atención cada solicitud de servicio y estimar el esfuerzo en horas que tomaría cubrir cada solicitud.

## 2. Análisis de alternativas

En el presente proyecto se consideró la posibilidad de proponer, para el mismo problema y objetivo general, algunas otras alternativas de solución.

En primer lugar, se consideró la implantación o adecuación de un software ya existente. Sin embargo, tal como se describe en la revisión del estado del arte, todas las alternativas halladas cubren parcialmente las necesidades de la organización, mas ninguna las cubre en su totalidad.

Por tal motivo se concluyó que era necesario un sistema de información desarrollado a la medida, diseñado para cubrir todas las necesidades de la organización. Sin embargo, posteriormente se tuvo que decidir entre desarrollar el sistema en un entorno de escritorio (cliente-servidor) o en un entorno web. Se optó por un entorno web dada la

facilidad de despliegue que este ofrece. Un sistema en entorno de escritorio requiere un desarrollo compatible para distintos sistemas operativos, la instalación de un cliente en cada equipo que desee acceder al sistema y el aprendizaje de tecnologías por parte del tesista. Por su parte, un entorno web requiere únicamente de la instalación de un navegador web moderno y permite acceder al sistema desde cualquier dispositivo con conectividad al servidor web. Finalmente, el tesista ya cuenta con experiencia en el desarrollo de sistemas de información en entornos web.

### **3. Viabilidad**

Respecto a los aspectos técnicos, luego de un estudio de los lenguajes y plataformas que pudieran soportar la implementación de este plan de proyecto, se ha seleccionado Java como lenguaje de programación y Play Framework como plataforma para el desarrollo de un sistema de información web. Asimismo, la elección de una plataforma de desarrollo en el lado de cliente de la comunicación web se debe a la necesidad de presentar a los usuarios una aplicación web dinámica, con elementos cambiantes. Para ello, se seleccionó AngularJS dado que permite organizar estos elementos con una estructura bastante clara y concisa.

En todo este conjunto de tecnología, se dispone de una amplia documentación acerca de su funcionamiento y uso (Play Framework, 2015; JetBrains, 2015; Angular, 2015). Por ello, y dada la experiencia del tesista en el uso de estas tecnologías a un nivel de competencia bastante alto, se estima la viabilidad técnica.

Por su parte, respecto a las restricciones de tiempo, se estima también la viabilidad del proyecto dado que se han realizado avances en los trabajos de campo con la organización, extrayendo información y modelando los procesos necesarios. El diagrama de actividades detallado se presenta en la Tabla 1.3.

Finalmente, respecto a las restricciones de costo, queda confirmada la viabilidad del proyecto, dado que ninguno de los procedimientos o herramientas necesarias para la consecución del mismo requieren de un desembolso. Todas las herramientas seleccionadas son gratuitas, posee permisos de uso para fines académicos o ya se cuenta con una licencia adquirida. Asimismo, el hardware y servicios necesarios para la ejecución y verificación del proyecto también se encuentran disponibles.

Tabla 1.3. Diagrama de Actividades del proyecto

Tarea	Inicio	Fin	Duración
<b>1. Proyecto de Fin de Carrera</b>	<b>04/01/2016</b>	<b>15/04/2016</b>	<b>102</b>
<b>1.1. Iteración 1</b>	<b>04/01/2016</b>	<b>25/01/2016</b>	<b>21</b>
Entrevistas	04/01/2016	11/01/2016	7
Modelado de procesos	11/01/2016	18/01/2016	7
Documento de visión	18/01/2016	25/01/2016	7
Documento de estándares de programación	18/01/2016	25/01/2016	7
Documento de estándares de diseño	18/01/2016	25/01/2016	7
<b>1.2. Iteración 2</b>	<b>25/01/2016</b>	<b>18/02/2016</b>	<b>24</b>
Lista de requerimientos	25/01/2016	28/01/2016	3
Definición de casos de uso	28/01/2016	08/02/2016	11
Diagramas de casos de uso	08/02/2016	11/02/2016	3
Plan de pruebas	11/02/2016	18/02/2016	7
<b>1.3. Iteración 3</b>	<b>18/02/2016</b>	<b>03/03/2016</b>	<b>14</b>
Modelado Diagrama de clases	18/02/2016	22/02/2016	4
Modelado Diagrama de despliegue	18/02/2016	22/02/2016	4
Modelado Diagrama de componentes	18/02/2016	22/02/2016	4
Prototipo de Arquitectura	22/02/2016	03/03/2016	10
<b>1.4. Iteración 4</b>	<b>03/03/2016</b>	<b>24/03/2016</b>	<b>21</b>
Análisis algoritmo heurístico	03/03/2016	06/03/2016	3
Diseño algoritmo heurístico	06/03/2016	10/03/2016	4
Implementación algoritmo heurístico	10/03/2016	20/03/2016	10
Integración algoritmo a SI	20/03/2016	24/03/2016	4
<b>1.5. Iteración 5</b>	<b>24/03/2016</b>	<b>14/04/2016</b>	<b>21</b>
Ejecución plan de pruebas	24/03/2016	31/03/2016	7
Correcciones y afinaciones	31/03/2016	14/04/2016	14



## 2. MARCO CONCEPTUAL

El Ministerio de Comunicaciones de la Iglesia Evangélica caso de estudio ha determinado la necesidad de automatizar el proceso de gestión de solicitudes de servicio, con el propósito de optimizar la productividad del área y recuperar información de dicho proceso. Este plan de proyecto de fin de carrera pretende brindar una alternativa de solución a dicha necesidad y, para ello, es necesaria la comprensión de la situación actual del área.

La presente sección tiene por objetivo describir y comprender la organización, el proceso actual de gestión de solicitudes de servicio, y las necesidades del Ministerio de Comunicaciones de la Iglesia Evangélica.

### 1. Estructura organizacional

La Iglesia Evangélica es dirigida por un Pastor principal, quien define los principios y políticas que la Organización desea cumplir, así como los objetivos que deben alcanzarse. Asimismo, se dispone de Ministerios, comparables a las gerencias de una empresa tradicional, que ejecuta planes de acción para la consecución de dichos objetivos. Cada ministerio posee una misión particular y realiza actividades afines a dicha misión. La lista de los trece ministerios presentes en la organización se muestra en la Tabla 2.1.

Tabla 2.1. Lista de ministerios

Ministerio
Redes
Evangelismo
Compasión
Apasionados
Intercesión y Misiones
Comunicaciones
Educación Cristiana
Seminario
Gente Sabia
Niños
Escuela de Música
Ser Mujer
Matrimonios

## **2. Solicitudes de Servicio**

Se denomina Solicitud de Servicio al documento que describe el trabajo o servicio que el solicitante desea recibir por parte el Ministerio de Comunicaciones.

Una solicitud de servicio es un documento que puede ser comparado con una orden de trabajo en una planta, ya que al igual que la orden de trabajo, genera todo un movimiento hasta el momento de su culminación (Paredes, 1986).

La solicitud se genera cuando un miembro de algún ministerio dentro de la Organización solicita un servicio al Ministerio de Comunicaciones. De acuerdo al tipo de trabajo solicitado, se debe brindar información específica. Los tipos de trabajo o servicio que pueden solicitarse son: (i) difusión, (ii) cobertura, (iii) piezas gráficas.

Las solicitudes de Difusión contienen los datos relacionados a la actividad que se desea difundir, como nombre, fecha, lugar, público objetivo, entre otros. Las solicitudes de Cobertura, además, deben indicar los requerimientos de cobertura, tales como fotografía, registro de vídeo, proyección, equipos de audio, entre otros. Finalmente, las solicitudes de Piezas gráficas consisten en pedidos de elaboración compleja, como libros y manuales. Estas solicitudes indican qué tipos de pieza gráfica son requeridos y deben luego ser detallados en una reunión presencial.

## **3. Proceso de gestión de solicitud de servicio**

Hasta ahora, la gestión de Solicitud de Servicio se ha realizado de manera manual y con apoyo de algunas herramientas de TI. Las solicitudes de servicio son recibidas y aceptadas en el buzón de correo electrónico del Ministerio de Comunicaciones, donde son revisadas por el asistente del área y luego derivadas al líder de equipo relevante. Este último puede ser del Equipo Creativo o Equipo de Producción.

El Asistente del área realiza una compilación de las solicitudes recibidas en una hoja de cálculo, donde ingresa todos los datos recibidos en la solicitud. En una frecuencia muy alta, las solicitudes recibidas presentan campos requeridos ausentes, por lo que hace falta que el Asistente del área contacte a la persona que registró la solicitud. También es frecuente que la solicitud requiera cambios o adiciones por parte del solicitante, y estos cambios suelen solicitarse de manera oral o telefónica, por lo que la probabilidad de pérdida o traspapelo es alta.

Posteriormente, el Asistente del área registra la solicitud en el cronograma de entregas del área. Dicho cronograma es un calendario digital visible a todos los miembros del área. El Asistente también es responsable de incluir en el cronograma los eventos de frecuencia fija (por ejemplo, de frecuencia semanal o mensual) sin recibir una solicitud de servicio de otro ministerio.

Cada servicio es designado a uno o más miembros del equipo, según las funciones que el asistente conoce que cada miembro desempeña. Cuando el servicio es entregado, el Asistente del área actualiza la información del pedido a estado "Entregado".

En el proceso actual no es viable la elaboración de reportes que muestren el número y tipo de solicitudes recibidas. Sin embargo, se estima que se reciben entre 15 y 20 solicitudes semanales.

#### **4. Conclusión**

Se observa que la Organización ha optado por alternativas de solución precarias para gestionar las Solicitudes de Servicio, basándose en actividades manuales. Ello conlleva una serie de desventajas y problemas en la recopilación, consolidación, organización y generación de reportes de solicitudes de servicio. Asimismo, se observa que es inviable, en este proceso, la elaboración de reportes que permitan obtener información del proceso y apoyar la gestión del mismo.



### 3. ESTADO DEL ARTE

En la actualidad existe una gran variedad de sistemas de información orientados a iglesias, entre gratuitos y comerciales, que cubren parte de las necesidades de cada organización. Asimismo, existen algoritmos de planificación en diversos campos. En esta sección se presentan algunos de los sistemas de información y algoritmos relevantes para la problemática expuesta.

El objetivo de la revisión del estado del arte es identificar las soluciones existentes para problemas similares al planteado anteriormente, de manera que sean útiles para definir el alcance, así como los métodos y procedimientos más óptimos a usar.

#### 1. Software de gestión y recolección de datos

El primer conjunto de elementos encontrados está compuesto por software diseñado para la gestión y recolección de datos; es decir, sistemas de información transaccionales.

##### 1. TouchPoint Software

Esta solución permite a las iglesias mantener un registro de sus miembros y asignar tareas a cada uno, así como comunicarse con cada miembro utilizando correo electrónico y mensajes de texto (SMS); también ofrece la posibilidad de personalizar el contenido de los correos electrónicos con plantillas HTML, para un mejor formato gráfico (TouchPoint, 2015).

La solución permite el acceso a través de un navegador web o desde aplicaciones móviles, permite la generación de reportes personalizables que en varios formatos (tabla, gráficos de barra, gráficos de torta, entre otros), ofrece funcionalidades de gestión de miembros, grupos, reuniones, asistencia, ofrendas y gestión de voluntarios (TouchPoint, 2015).

En cuanto a los costos, el fabricante ofrece planes que varían según la cantidad de registros en la base de datos, a razón de US\$2 por los primeros 25,000 registros y US\$1 por cada registro adicional; por su parte, las aplicaciones móviles autorizadas varían en planes que restringen ciertas funcionalidades, entre los US\$ 1,500 y US\$ 4,000 (TouchPoint, 2015).

##### 2. Church Windows

Este software permite gestionar miembros, eventos y ofrendas de una organización eclesial (Church Windows, 2015), para lo cual, ofrece las siguientes funcionalidades:

- Módulo de Planificación, que permite registrar eventos, instalaciones, equipos y servicios, así como designar responsables, asistentes y organizadores a cada evento.

Con este módulo es posible acceder a los datos de cada persona registrada en el evento directamente, así como exportar estos datos con facilidad.

- Visualización del calendario de la organización por día, semana y mes. Permite filtrar los eventos del calendario por criterios de búsqueda, así como marcar cada evento con un código de color.
- Generación de reportes, que incluyen la lista de voluntarios y su información de contacto, identificar los eventos para los cuales se requieren más voluntarios, identificar eventos para los cuales el aforo superan la capacidad máxima en términos de espacio o equipos, y listar los deberes de cada persona en determinado evento.
- Acceso a través de un cliente de escritorio para el sistema operativo Windows y requiere de un servidor centralizado con el sistema operativo Windows Server.

El costo del software varía según los módulos contratados y según el plan ("Iglesia pequeña" o "Iglesia"), en un rango entre US\$ 227 y US\$ 850 (Church Windows, 2015).

### 3. Google Forms

Este producto forma parte de la suite de ofimática de Google Inc (Google, 2015). La solución se ofrece gratuitamente bajo el modelo Software as a Service y permite el diseño de formularios personalizados, que luego pueden ser utilizados para recolectar información (Google, 2015). Según Google, estos formularios pueden contar una o más páginas, mostrándose según los datos ingresados en el formulario, que pueden ser de tipo:

- Texto
- Párrafo
- Opción múltiple
- Lista
- Escala
- Grilla
- Fecha
- Hora

Asimismo, los datos registrados son visualizados en una hoja de cálculo y se accede a través de un navegador web. En la hoja de cálculo de respuestas, es posible observar gráficos de reporte de los datos. Por este motivo, es ideal para la elaboración de encuestas.

Al igual que otros productos de la suite ofimática de Google, es posible acceder tanto al diseño como a la revisión de datos recolectados junto a otros colaboradores al mismo tiempo. La imagen 3.2 muestra la vista de diseño de un formulario en Google Forms.

Page 1 of 1

## Form 1

Form Description

Question Title: Untitled Question

Help Text:

Question Type: Multiple choice  Go to page based on answer

Option 1

Click to add option or Add "Other"

Advanced settings

Done  Required question

Add Item

Imagen 3.2. Vista de diseño de un formulario en Google Forms.

Permite el diseño de formularios simples a través de un editor web. Se observa la posibilidad de editar múltiples propiedades en cada pregunta.

## 2. Algoritmos para planificación de tareas

El segundo conjunto de elementos encontrados está compuesto por algoritmos para planificación de tareas.

### 1. Algoritmo basado en reservas

El algoritmo considera el problema de planificar la entrega de tareas periódicas y aperiódicas; es decir, de tareas de frecuencia fija y de tareas esporádicas (Shin; Yi-Chieh, 1995). Se propone un algoritmo basado en reserva para ordenar la ejecución de las tareas necesarias, cuyas características son (Shin; Yi-Chieh, 1995):

- Prioriza las tareas periódicas, de modo que los plazos propuestos para estas cumplen con los plazos límites establecidos, minimizando así la probabilidad de retrasos para las tareas aperiódicas
- Las tareas periódicas se planifican según la duración del trabajo, programando primero los trabajos más cortos. Las tareas aperiódicas son programadas en los lapsos disponibles.
- El algoritmo especifica una variable  $u$  como el máximo común denominador de la duración de todas las tareas, y asume que cada tarea será iniciada al inicio de algún periodo  $u$ .
- El algoritmo reserva un tiempo  $R$  para cada tarea aperiódica, sin perder los plazos máximos de las tareas periódicas; esto es, luego de reservar periodos para estas tareas.

## 2. Algoritmo para tareas dependientes

Se propone un algoritmo para planificación de tareas dependientes, representadas en grafo dirigido acíclico, cuyas características son (Mingsheng, Shang; Shixin, Sun; Qingxian, Wang, 2003):

- Su principal objetivo es minimizar la duración del cronograma total y la duración del proceso de planificación en sí mismo.
- Basar la programación en rutas críticas, programando primero las tareas de mayor influencia en la duración del cronograma tan pronto como se pueda.
- Las rutas críticas son representadas en Redes de Petri Coloreadas (Coloured-Petri nets, CPNs, en inglés).

## 3. Algoritmo basado en restricciones de plazo

Este algoritmo toma como referencia restricciones de plazo para la planificación de ejecución de tareas y, según sus autores, sus características son: (Tang, et al., 2012):

- Permite al usuario especificar una fecha de entrega máxima e intenta programar las tareas necesarias para cumplir con dicho plazo.
- Analiza la capacidad de producción o potencial de trabajo para programar las tareas, de manera que asigna las tareas según los recursos disponibles.
- Presenta una alta tasa de precisión para calcular el tiempo restante a cada tarea.
- Está diseñado para ejecutarse en clústeres, utilizando la plataforma Hadoop y el modelo MapReduce para su ejecución. Esto le permite procesar grandes cantidades de información en un corto plazo, haciendo uso de capacidad de procesamiento de múltiples computadores en paralelo.

## 4. Algoritmo para la planificación de proyectos

Se encuentra también en esta revisión del estado del arte un grupo de propuestas para la óptima planificación de cronogramas de proyectos; en especial, para proyectos con recursos limitados y restricciones de tiempo (Smith, Pyle, 2004). Modelar los proyectos de esta forma permite simular situaciones reales como fechas de inicio de proyecto fijas, fechas de entregas parciales y final, actividades que inician en simultáneo, entre otros (Smith, Pyle, 2004).

Se observa, por ejemplo, el algoritmo Optimización de rueda chirriante (SWO, por sus siglas en inglés), cuyas características son (Smith, Pyle, 2004):

- El algoritmo tiene como componente principal un constructor de cronograma voraz, basado en prioridades.
- Cuenta con un mecanismo de resolución de conflictos –llamado *demolición* o *bulldozing*, en inglés-, dado que las restricciones de tiempo de inicio de las actividades pueden tener un impacto negativo sobre el algoritmo voraz. Esto provoca que el

algoritmo voraz planifique un conjunto de actividades más tarde en el cronograma para mantener la factibilidad de proyecto.

- Considera, además de las restricciones de tiempo de inicio y fin, la duración de las actividades, la capacidad de los recursos y los requerimientos de los recursos.
- Considera las restricciones de dependencia entre una tarea y otra, de manera que es posible encadenarlas.

### **3. Conclusiones sobre el estado del arte**

Se encuentra en esta revisión que existen productos para la gestión de Iglesias orientados principalmente al seguimiento de miembros, eventos, ofrendas y contabilidad. Algunas de estas soluciones permiten asignar recursos humanos a sus eventos e incluso identificar eventos para los cuales hay recursos insuficientes. La principal limitante de estas soluciones es el costo que conllevan para su uso.

Estas soluciones no se amoldan a las necesidades de la problemática expuesta, dado que no permiten un registro personalizado de los requerimientos de cada evento. Si bien Google Forms permite crear formularios personalizados, sus funcionalidades son insuficientes, dado que no permite el registro de solicitudes recurrentes (i.e., registros con frecuencia fija semanal, mensual o anual). Además, no permite la edición de los registros realizados al registrante.

Respecto a los algoritmos de planificación, se observa que existen alternativas de solución bastante complejas, que podrían ser adaptadas al problema presentado. Por la revisión llevada a cabo, se concluye que el diseño un algoritmo para la planificación de las tareas de cada solicitud de servicio es viable.

Se concluye que existen productos que cubren partes específicas de la problemática planteada, mas no en su totalidad.

#### 4. MODELADO DE PROCESO

Esta sección tiene como finalidad exponer el proceso de Gestión de Solicitudes de Servicio modelado a partir de las necesidades expuestas por la Organización. En especial, se detalla el subproceso de Registro de Solicitudes de Servicio, pues consta de mayor complejidad. El proceso modelado fue la base para el diseño del sistema de información que soporta las actividades de la organización.

##### 1. Proceso de Gestión de Solicitudes de Servicio

El proceso inicia cuando un miembro de la Organización, de acuerdo a las necesidades del ministerio al que pertenezca, registra una nueva Solicitud de Servicio, por lo cual es llamado Registrante. La solicitud es luego evaluada por el Director del Ministerio de Comunicaciones, quien tras verificar que es viable ejecutarla y que se dispone de toda la información requerida, planifica la fecha de entrega y designa un Equipo de Trabajo. Este equipo será quien en última instancia ejecute las labores necesarias para cumplir con la solicitud. Finalmente, tanto el Director del Ministerio de Comunicaciones como, posteriormente, el Registrante observan el resultado entregado por el Equipo de Trabajo y dan conformidad al mismo. En caso de requerirse algún cambio o no disponer de conformidad, el Director vuelve a evaluar y planificar la Solicitud para cubrir con los requerimientos. La imagen 4.1 muestra el proceso de Gestión de Solicitudes de Servicio en una vista general.

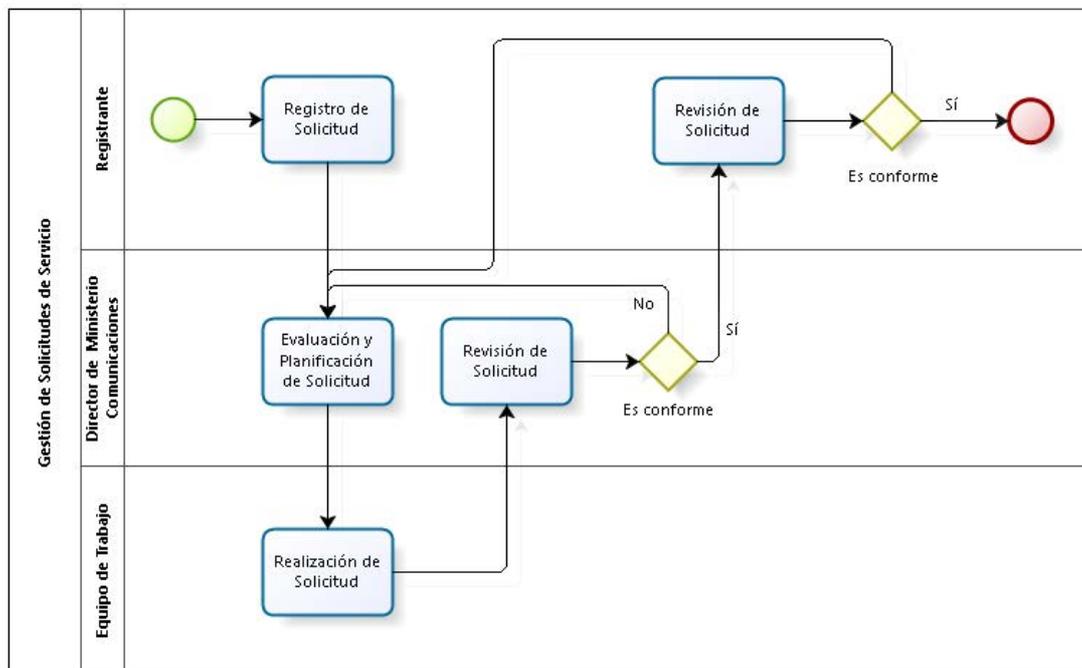


Imagen 4.1. Proceso de Gestión de Solicitudes de Servicio

## 2. Registro de Solicitudes de Servicio

El Registro de Solicitudes de Servicio es realizado únicamente por el Registrante. Durante el proceso, él consignará información general y específica acerca de la Solicitud. La información requerida variará de acuerdo al tipo de solicitud que registre, que puede ser Publicidad, Atención de Evento o Piezas Gráficas. La imagen 4.2 muestra el registro de una Solicitud de Servicio a nivel macro.

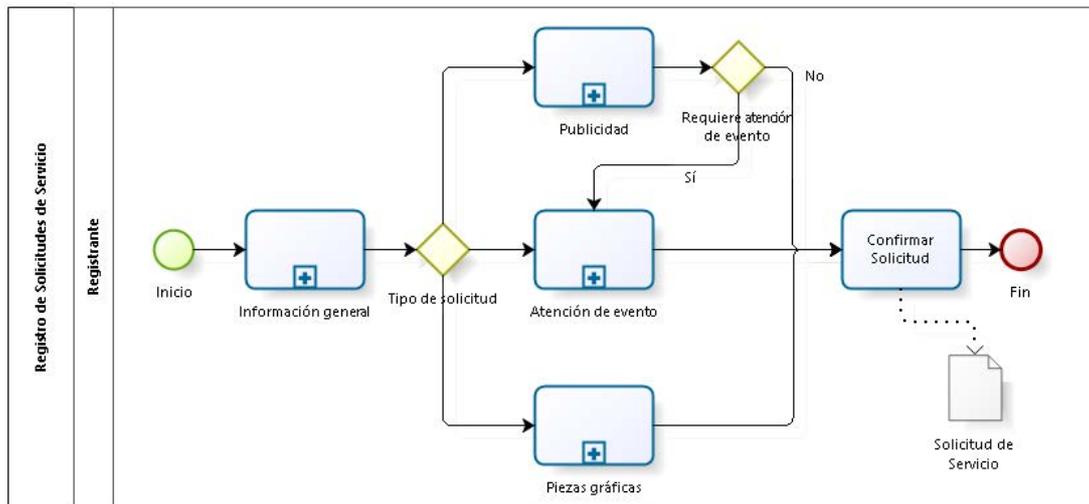


Imagen 4.2. Proceso de Registro de Solicitudes de Servicio

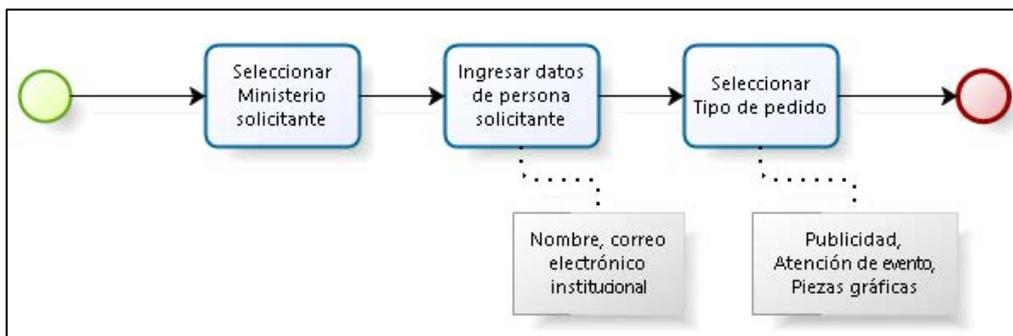
### 1. Información general

En el subproceso de Información general, el registrante debe consignar:

- Ministerio que registra la solicitud. Debe ser uno de los listados en la Tabla 2.1.
- Datos de la persona solicitante: nombre completo y dirección de correo electrónico institucional.
- Tipo de pedido: Publicidad, Atención de evento o Pieza gráfica.

La imagen 4.3 muestra el flujograma del subproceso de Información general.

Imagen 4.3. Flujograma del subproceso Información general.



## **2. Publicidad**

Las solicitudes de tipo Publicidad son aquellas en las que se requiere que el Ministerio de Comunicaciones realice material publicitario para un evento específico. Para ello, se requieren los datos específicos del evento, de manera que el equipo creativo del área puede confeccionar la publicidad correspondiente.

En el subproceso de Publicidad, el registrante debe consignar:

- Tipo de evento: curso, taller, evento, retiro, viaje misionero o campamento.
- Nombre del evento.
- Duración del evento: Fecha, hora de inicio, hora de fin, frecuencia del evento (opcional, solo si el evento se repite).
- Localidad a la que se dirige el evento.
- Localidad y aula sede del evento.
- Si el tipo de evento fuera Retiro o Campamento, fecha y hora de partida, y fecha y hora de retorno.
- Costo de participación.
- Datos de público objetivo: Rango de edad, género, otro. Estos datos son opcionales.
- Frecuencia anual, esto es, número de veces por año que ocurre el evento.
- Si el tipo de evento fuera Curso, Taller o Evento, Expositor o invitado especial. Este campo es opcional.
- Corazón del evento, esto es, el objetivo del evento.
- Tamaño de folleto. Este campo es opcional.
- Tamaño de banner. Este campo es opcional.
- Requisitos de los participantes: discipulado, membresía, otros. Estos datos son opcionales.
- Datos de contacto: nombre de persona de contacto, teléfono y correo electrónico de contacto, lugar y horario de inscripciones.

La imagen 4.4 muestra el flujograma del subproceso de Publicidad.

## **3. Atención de Evento**

Las solicitudes de tipo Atención de Evento son aquellas en las que se requiere de algún tipo de cobertura audiovisual por parte del Ministerio de Comunicaciones para un evento específico. Es posible registrar solicitudes por eventos que solo requieren atención de evento; no obstante, los eventos de tipo Publicidad tienen la opción de solicitar Atención de Evento también. En este último caso, los datos repetidos del subproceso de Publicidad deben coincidir.

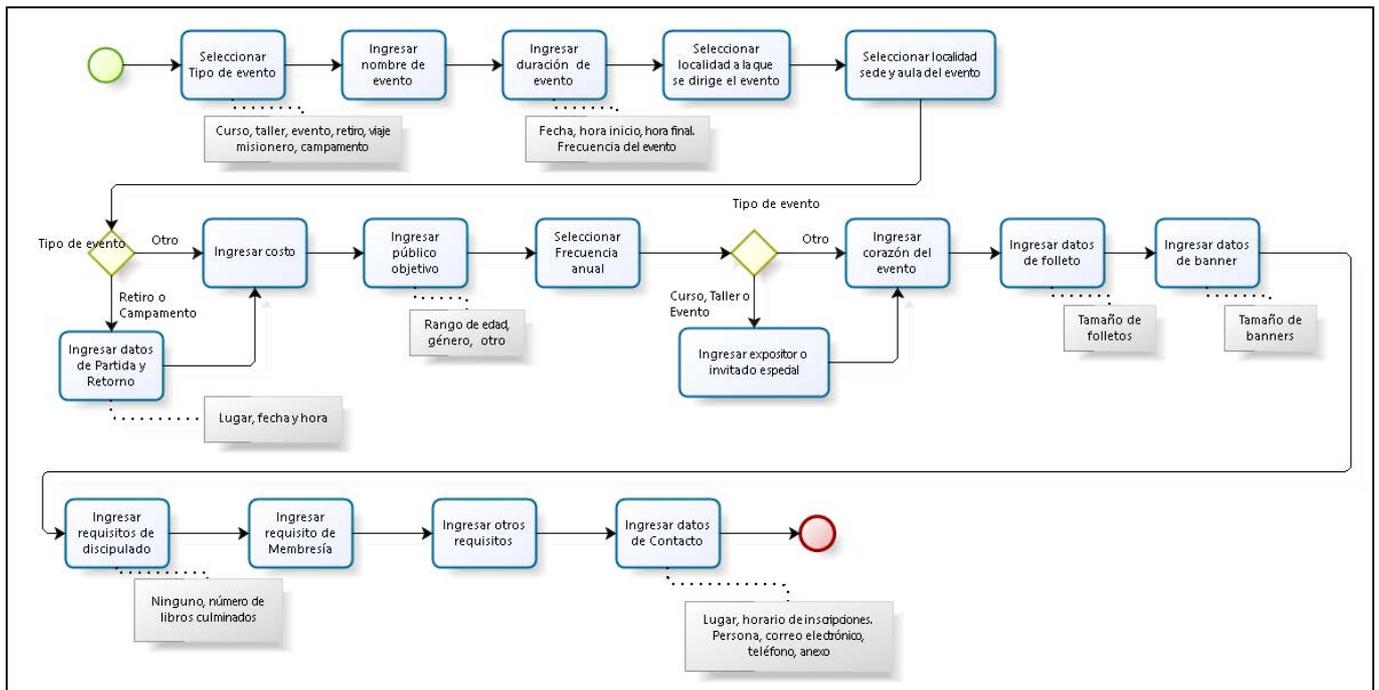


Imagen 4.4. Flujograma del subproceso Publicidad.

En el subproceso de Atención de Evento, el registrante debe consignar:

- Nombre del evento.
- Duración del evento: Fecha, hora de inicio, hora de fin, frecuencia del evento (opcional, solo si el evento se repite).
- Localidad a la que se dirige el evento.
- Localidad y aula sede del evento.
- Asistencia estimada (número de personas).
- Requerimiento de instalación de proyector, solo si el aula sede no dispone de un proyector permanente.
- Requerimiento de instalación de equipo de sonido, solo si el aula sede no dispone de equipo de sonido permanente.
- Requerimiento de laptop, solo si el aula sede no dispone de laptop o computador permanente.
- Requerimiento de grabación de vídeo.
- Requerimiento de cobertura fotográfica.

La imagen 4.5 muestra el flujograma del subproceso de Atención de Evento.

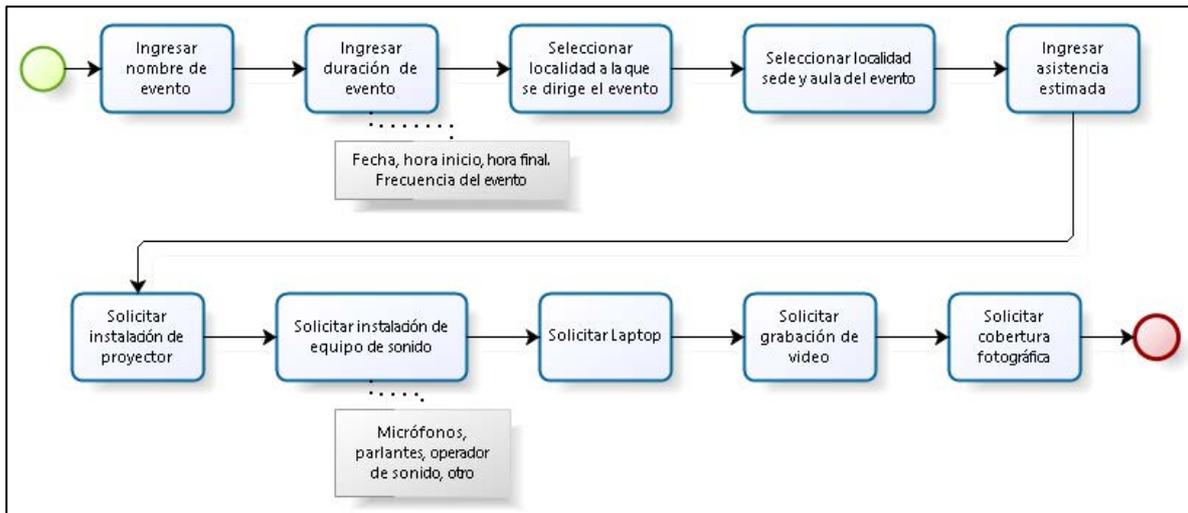


Imagen 4.5. Flujograma del subproceso de Atención de Evento.

#### 4. Piezas Gráficas

Las solicitudes de tipo Piezas Gráficas son aquellas en las que se requiere que el Ministerio de Comunicaciones elabore algún material con cierta complejidad, tales como libros o *merchandising*. Debido a la gran variedad de materiales que pueden solicitarse, se optó por dar libertad al registrante a suministrar información que estime necesaria para cubrir la solicitud.

La información provista servirá para que el Ministerio de Comunicaciones dimensione la solicitud, se reúna con la persona responsable de la solicitud y finalmente programe una fecha de entrega de manera manual.

En el subproceso de Piezas Gráficas, el registrante debe consignar:

- Tipo de material solicitado: Libro, manual, brochure, tarjeta de invitación, diploma, logo o merchandising.
- Documento con información necesaria para dimensionar la solicitud. El contenido de este documento es libre.

La imagen 4.6 muestra el flujograma del subproceso de Piezas Gráficas.

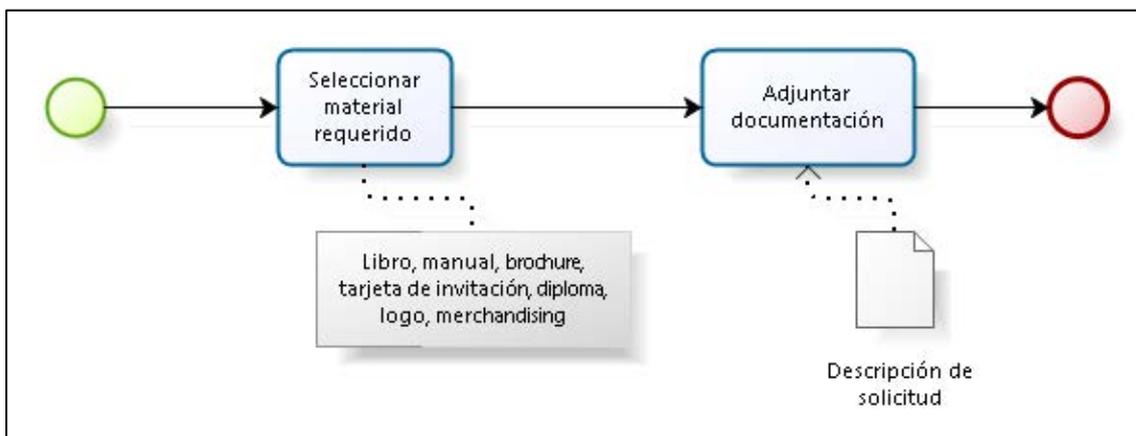


Imagen 4.6. Flujograma del proceso de Piezas Gráficas.

### 3. Conclusiones

El proceso modelado permite ordenar el registro de solicitudes de servicio, de manera que los datos requeridos no sean obviados o ingresados con errores. Asimismo, se observa que los datos requeridos varían según el tipo de solicitud que un registrante haga, y ello depende del objetivo de su solicitud.

Finalmente, se observa que el proceso modelado cumple con las necesidades expuestas por la Organización durante el relevamiento de información.



## 5. ARQUITECTURA DE SOFTWARE

El propósito de esta sección es presentar las características de arquitectura del software propuesto. La definición de la arquitectura siguió la metodología Modelo 4+1, por lo que esta sección incluye subdivisiones por cada vista arquitectónica propuesta por dicha metodología y las representaciones gráficas que corresponden.

Siguiendo la metodología, no se ha representado todos los procesos, tareas o requerimientos del software, sino únicamente aquellos que son críticos y necesarios para comprender el mismo. En este proyecto, se consideró crítico el proceso de Gestión de Solicitudes de Servicio y el proceso de Planificación de tareas.

### 1. Vista lógica

Esta vista se enfoca en cubrir la funcionalidad del sistema en términos de elementos estructurados, esto es, conectados. Estos elementos se representan siguiendo el paradigma de Orientación a Objetos y se grafican utilizando un Diagrama de Clases, que refleja las relaciones entre cada clase. La Figura 5.1 muestra el Diagrama de Clases del Sistema de Información.

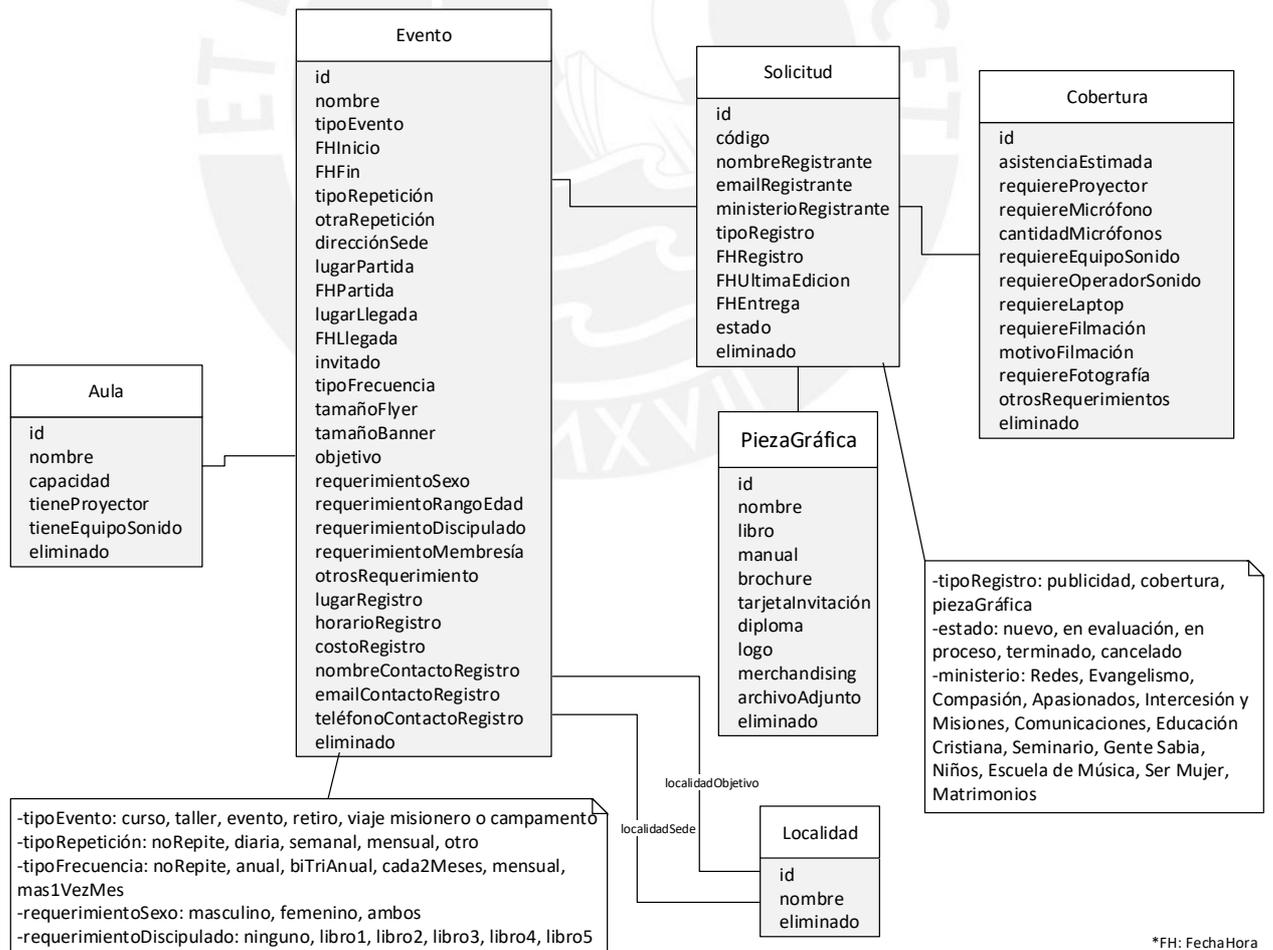


Figura 5.1. Diagrama de Clases del Sistema de Información

## 2. Vista de procesos

Esta vista muestra las operaciones a las que serán sometidos los objetos reflejados en la Vista Lógica. En tal sentido, el objetivo de esta vista es describir el comportamiento del sistema. La Figura 5.2 muestra el Diagrama de Actividades del Proceso de Registro de Solicitud de Servicio.

Cabe resaltar que, para el presente proyecto, no se registran interacciones con otros sistemas, subsistemas o componentes externos; todos los procesos son cubiertos por el propio sistema.

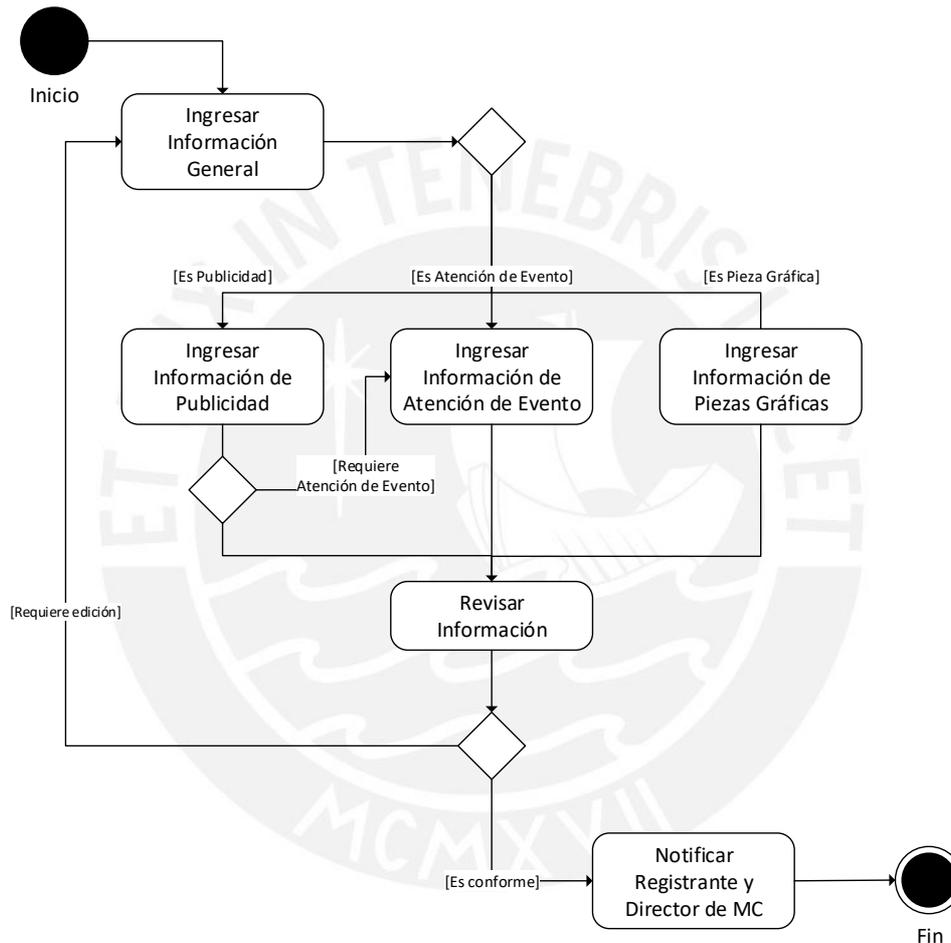


Figura 5.2. Diagrama de Actividades del Proceso de Registro de Solicitud de Servicio

Finalmente, la Figura 5.3 muestra el Diagrama de Actividades del Proceso de Planificación. Cabe mencionar que para la ejecución de una Solicitud de Servicio se estima una lista de tareas necesarias y la duración de cada tarea. Esta generación está basada en reglas condicionales, lo que se detallará más adelante.

Asimismo, cada Solicitud de Servicio recibe un valor de importancia en base a reglas condicionales como frecuencia del evento (e.g., evento anual es más importante que evento semanal), presencia de invitado(s), entre otros.

Se observa que el proceso de planificación sigue el algoritmo “El primero el mejor”, iterando sobre cada tarea obtenida en la estimación y asignando las horas de cada persona que puede realizar dicha tarea hasta cubrir todas las horas necesarias estimadas para completar la Solicitud de Servicio.

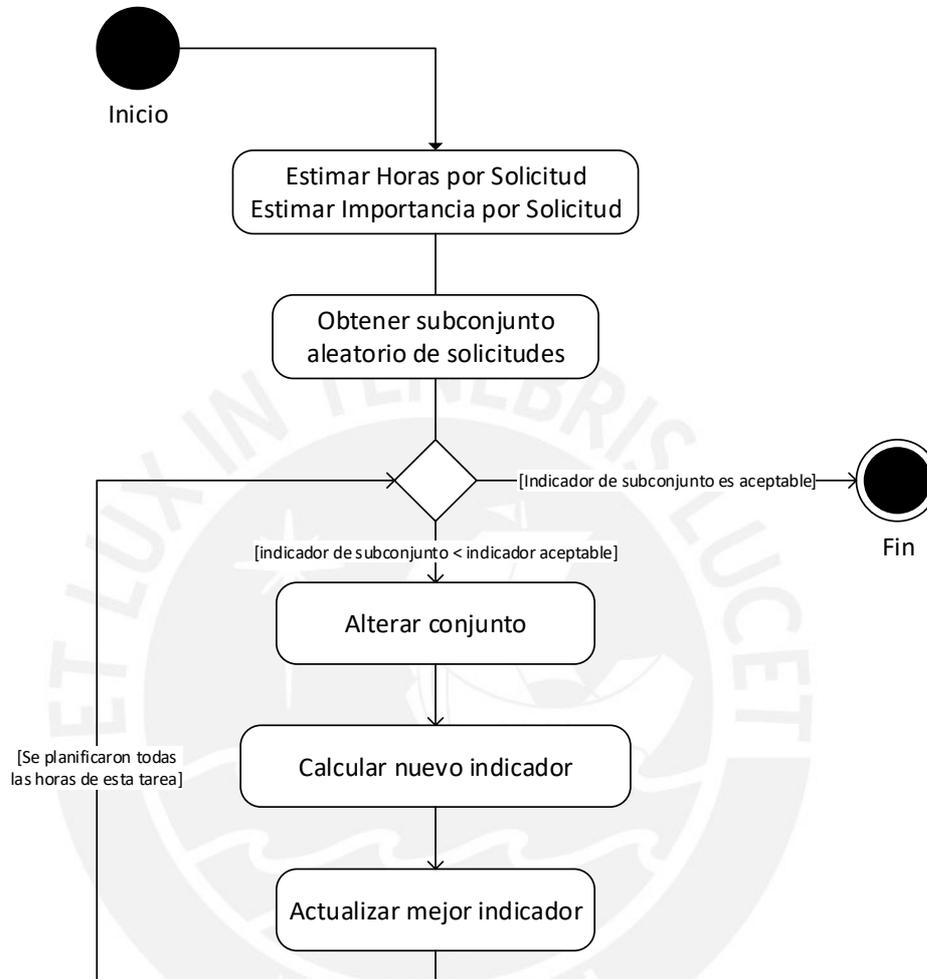


Figura 5.3. Diagrama de Actividades del Proceso de Planificación.

### 3. Vista de implementación

Esta vista se retrata la división del software en módulos y las relaciones que existen entre ellos. El objetivo de esta vista es definir los módulos que componen el sistema y las comunicaciones que establecen entre sí, de manera que sea posible conocer cómo agrupar los elementos hallados en la Vista Lógica y qué interfaces desarrollar en cada módulo. La figura 5.4 muestra el Diagrama de Componentes del Sistema de Información.

Del diagrama destaca que el Sistema sigue el patrón MVC, lo cual ayudará a separación de conceptos en capas de Vista, Controlador y Modelo. Además, el sistema hace uso de recursos externos, en especial, librerías para la capa de Vista. Finalmente, el sistema interactúa con componentes externos, estos son nginx, servidor web; y Play Framework, servidor de transacciones (backend); y MySQL, servidor de base de datos.

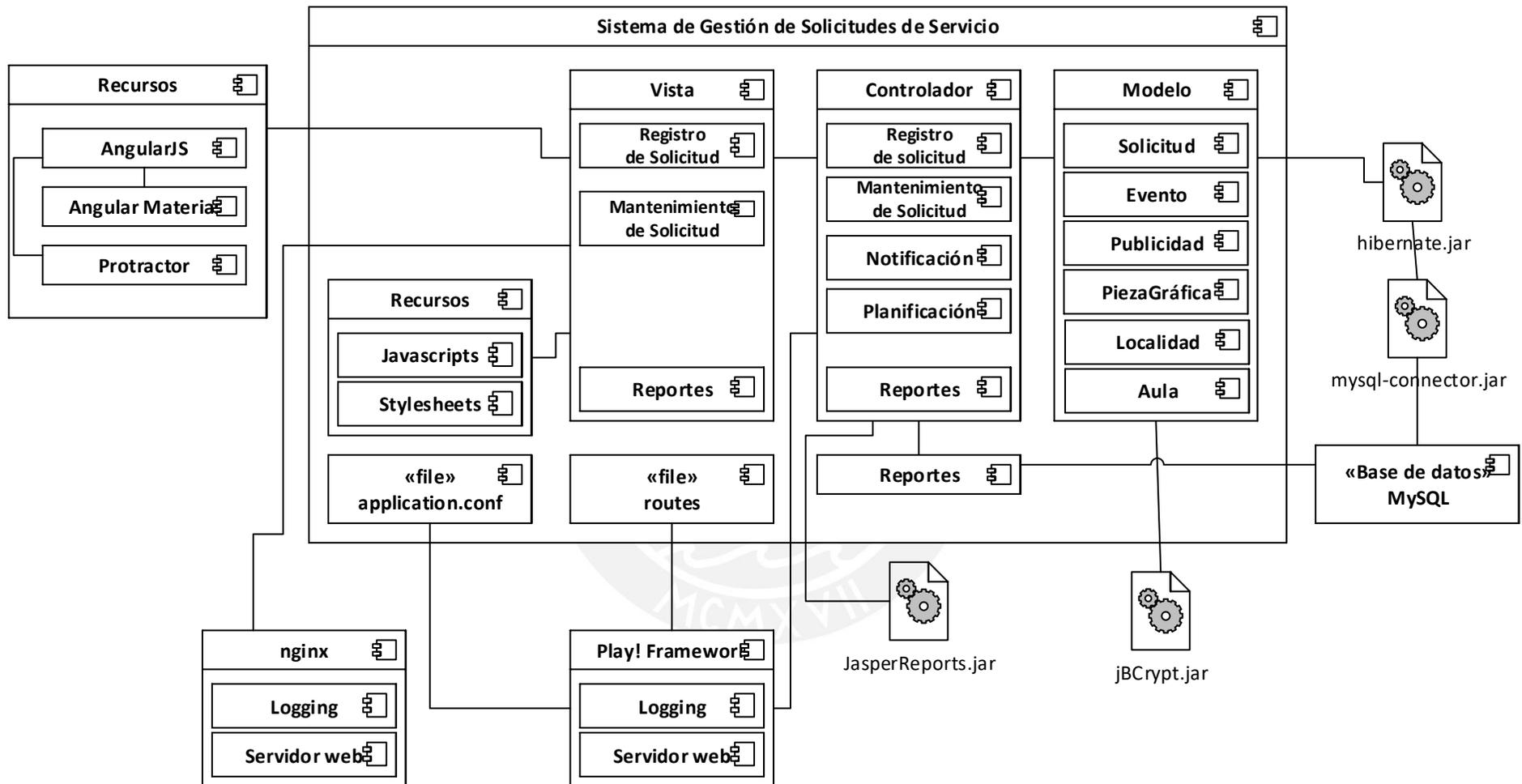


Figura 5.4. Diagrama de Componentes del Sistema de Información.

#### 4. Vista física

Esta vista ofrece detalle de los nodos sobre los cuales se ejecutará el sistema; es decir, representa la topología del hardware. El objetivo de esta vista es conocer qué nodos serán necesarios para la ejecución del sistema, así como conocer su distribución y comunicación. La figura 5.5 muestra el Diagrama de Despliegue del Sistema de Información.

Del diagrama, cabe destacar que no se insertó la capa frontend dentro del servidor backend (servidor de transacciones) ni expuso directamente dicho servidor directamente a Internet, a pesar de que Play Framework dispone de esta funcionalidad. En su lugar, se introdujo un servidor web que opera como servidor proxy, delegando las solicitudes recibidas tanto a la capa frontend como backend.

Esto permitirá, en el futuro, desplegar ambas capas en nodos físicos (dispositivos) separados, según la necesidad de rendimiento de cada una. Por ejemplo, la capa backend podría requerir un dispositivo de mayor rendimiento, por lo que únicamente esta sería instalada en un servidor externo de rendimiento mediano en lugar de trasladar todo el sistema a un servidor de rendimiento grande. Esto permitirá el ahorro de costos de operación a la Organización.

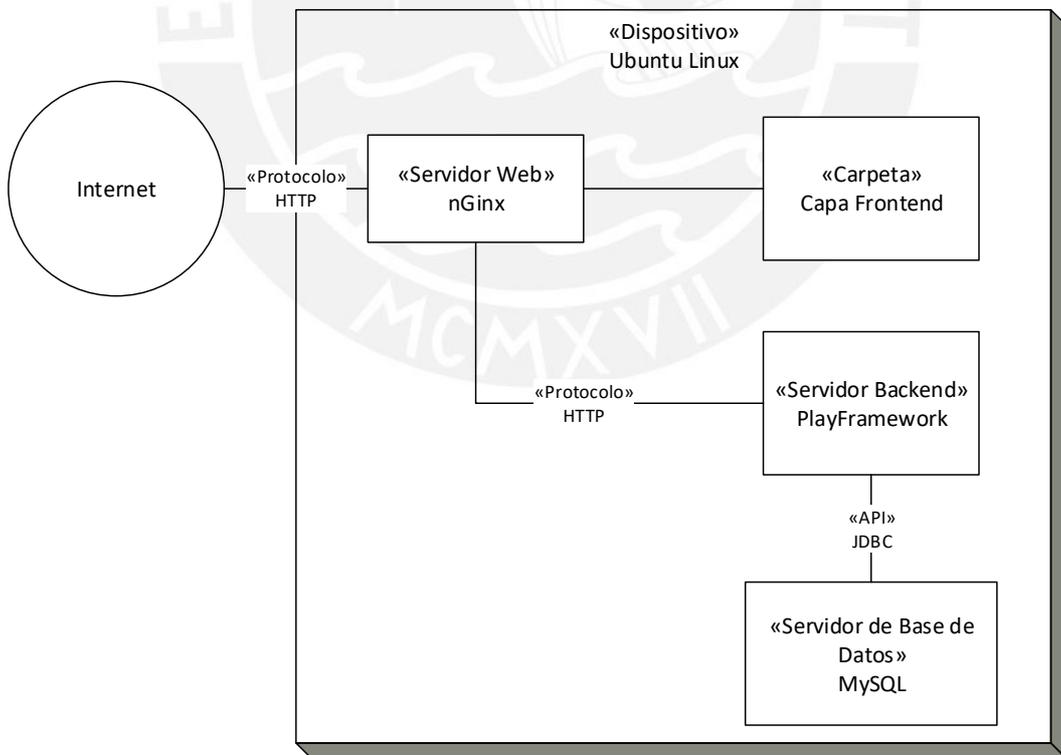


Figura 5.5. Diagrama de Despliegue del Sistema de Información

## 5. Vista de escenarios

Esta vista representa los casos de uso que describen el comportamiento del sistema desde el punto de vista de sus usuarios. El objetivo de esta vista es poder conocer los posibles usos que puedan darle los usuarios al sistema. La figura 5.6 muestra el Diagrama de Casos de Uso del Sistema de Información.

Se observa que las tareas más comunes para los usuarios son Registrar Solicitud de Servicio, en el caso del Registrante; Editar Solicitud de Servicio, tanto para Registrante como para el Director de Ministerio de Comunicaciones; y Generar Reporte, para el Director de Ministerio de Comunicaciones. Asimismo, el registro y edición de una solicitud debe originar el envío de una notificación electrónica a las partes interesadas; esto es, al Registrante, Director del Ministerio de comunicaciones y al Equipo de Trabajo.

Se observa también que el Equipo de Trabajo no es usuario del Sistema, sino que es el Director de Ministerio de Comunicaciones el encargado de revisar, evaluar, modificar o autorizar acciones pertinentes sobre las Solicitudes de Servicio. Por este motivo, el Equipo de Trabajo no fue incluido en el Diagrama de Casos de Uso.

Se observa que únicamente el Director de Ministerio de Comunicaciones es capaz de autenticarse en el Sistema, mas no así el Registrante. Esto se hizo para evitar la complejidad al registrar una solicitud. De esta manera, cualquier miembro con acceso a la red de la Organización tiene la posibilidad de registrar una Solicitud de Servicio. Dado que existe suficiente nivel de confiabilidad en los miembros con acceso a la red interna, esto no supone un riesgo de seguridad relevante para el Sistema.

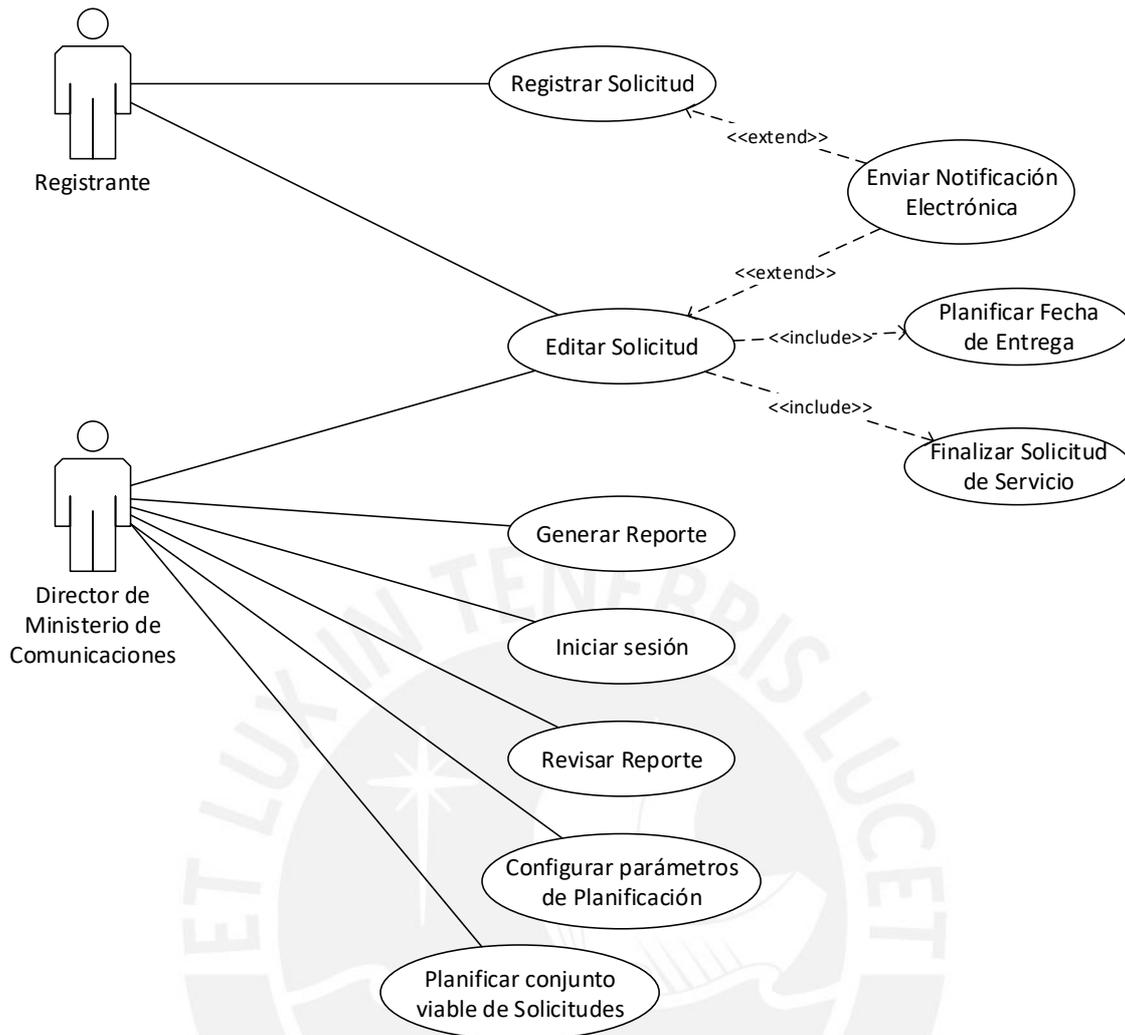


Figura 5.5. Diagrama de Casos de Uso del Sistema de Información

## 6. Conclusiones

Se observa que la arquitectura describe un sistema de información que cubre con los requerimientos de la organización, específicamente, con los procesos de Gestión de Solicitudes de Servicio y de Planificación de tareas.

Se tiene como trabajo futuro la modificación de esta arquitectura de software para interactuar con otros sistemas de la Organización. Por ejemplo, el sistema podría consumir un servicio de autenticación y autorización que permita identificar usuarios desde un sistema o servicio externo (por ejemplo, Active Directory) existente en la Organización.

También, como se mencionó, se tiene como trabajo futuro desplegar el Sistema en nodos físicos separados, a fin de incrementar el rendimiento con el menor costo posible. Esto se efectuará según la demanda del Sistema lo requiera; sin embargo, la arquitectura fue diseñada previendo dicha posibilidad.

Finalmente, como mejora a la comunicación web, se podría en el futuro implementar conexión segura utilizando una conexión HTTPS desde el navegador hasta el servidor web en lugar del actual HTTP (se observa en la capa física). Esto permitiría por un lado tener una conexión cifrada entre cada usuario y el servidor, reduciendo el riesgo de pérdida de integridad de los datos enviados. Por otro lado, permitiría la utilización del protocolo HTTP/2, que supone varias mejoras a la comunicación web no detalladas en este documento, y que tiene por requisito una conexión cifrada.



## 6. ALGORITMO DE PLANIFICACIÓN

Esta sección tiene por objetivo describir el algoritmo diseñado para la planificación de atención de solicitudes de servicio, esto es, generación de una lista de las solicitudes que deben atenderse, así como el esfuerzo en horas-hombre que tomaría cubrir cada solicitud.

### 1. Descripción

El algoritmo desarrollado considera la complejidad de cada Solicitud de Servicio registrada, en base a sus datos y requerimientos, para generar una lista de las tareas necesarias a realizar y estimar la cantidad de horas-hombre que tomaría cada una. De esta manera, al sumar la duración de cada tarea es posible obtener la duración total estimada para cubrir la Solicitud de Servicio recibida.

Tras obtener dicho estimado, se asigna un coeficiente de importancia basado en reglas que son afectadas según los datos de la solicitud. Por ejemplo, las solicitudes de eventos que se realizan solo una vez por año son de mayor importancia.

Cabe destacar que, en caso la solicitud requiera Atención de Evento, el algoritmo debe incluir en el cálculo de horas-hombre requeridas a la cantidad de horas que tomará el evento.

Finalmente, el algoritmo itera hasta encontrar una combinación de solicitudes de servicio aceptable para los requerimientos especificados.

### 2. Función objetivo

El algoritmo desarrollado tiene por objetivo maximizar la cantidad de solicitudes de servicio que pueden atenderse, considerando la cantidad de horas-hombre disponibles y la importancia de cada solicitud. La función, por tanto, produce un conjunto viable de solicitudes de servicio.

```
obtenerCjtoViableSolicitudes (solicitudes, horasHombreDisponibles)
```

La variable “solicitudes” representa las Solicitudes de Servicio recibidas que no han sido marcadas como “planificada” hasta el momento de la ejecución del algoritmo. Cada objeto “solicitud”, elemento del arreglo “solicitudes”, contiene en sus propiedades todos los datos ingresados por el usuario en el formulario de Registro de solicitud, según lo especificado en el proceso modelado anteriormente en este documento.

La variable “horasHombreDisponibles” es un número entero ingresado por el usuario, que representa la cantidad de horas-hombre disponible por el Ministerio de Comunicaciones para determinado periodo.

La función objetivo genera una lista de tareas en base a cada objeto “solicitud” y estima la cantidad de horas de cada una. Posteriormente, suma la duración de cada tarea para obtener la duración estimada total de la solicitud.

### 3. Estructuras de datos

A continuación, se presentan las estructuras de datos utilizadas en el algoritmo:

- MapaSolicitudEstimación: Mapa que contiene cada la Solicitud de Servicio (llave, objeto tipo Solicitud) y el número de horas hombre estimado para dicha solicitud (valor, variable tipo Número entero).
- MapaSolicitudPriorización: Mapa que contiene cada la Solicitud de Servicio (llave, objeto tipo Solicitud) y el coeficiente de importancia asignado a dicha solicitud (valor, variable tipo Número racional entre 0 y 1).
- Tarea: Variable que describe una tarea. Variable tipo Cadena de texto.
- ListaViableTarea: Lista de objetos Tarea. Representa el resultado final del algoritmo explicado anteriormente.

### 4. Seudocódigo

En esta sección se presentan las partes más relevantes del algoritmo, expresado en pseudocódigo.

```
función obtenerCjtoViableTareas (solicitudes, horasHombreDisponibles)
  mapaSolicHoras = calcularHorasPorSolicitud(solicitudes)
  mapaSolicImp = calcularImportanciaPorSolicitud(solicitudes)

  indicadorAceptable = 0.75

  mejorConjunto = solicitudes.subconjuntoAleatorio()
  mejorIndicador = calcularIndicador (mejorConjunto, mapaSolicHoras, mapaSolicImp)

  mientras(mejorIndicador < indicadorAceptable)
    conjunto = conjunto.alterar(solicitudes)
    indicador = calcularIndicador (conjunto, mapaSolicHoras, mapaSolicImp)

    si mejorIndicador < indicador
      mejorIndicador = indicador

  finMientras

  retornar mejorConjunto

finFunción

función calcularIndicador (conjuntoSolicitudes, estimacionHoras, estimacionImportancia)

  horas = estimacionHoras
  .mapear( (k,v) => conjuntoSolicitudes.incluye(k) )
  .reducir( ((k1,v1), (k2,v2)) => v1 + v2 )

  importancia = estimacionImportancia
  .mapear( (k,v) => conjuntoSolicitudes.incluye(k) )
  .reducir( ((k1,v1), (k2,v2)) => v1 * v2 )

  retornar horas*importancia

finFunción
```

```

función calcularHorasPorSolicitud (solicitud)
  mapaTareaEstimación = nuevo Mapa()

  si solicitud.tipo == 'publicidad'
    mapaTareaEstimación.agregar(generarTareasEstimacionesPublicidad(solicitud))

  si solicitud.tipo == 'atención de evento' o solicitud.requiereAtenciónEvento
    mapaTareaEstimación.agregar(generarTareasEstimacionesAtenciónEvento(solicitud))

  si solicitud.tipo == 'pieza gráfica'
    mapaTareaEstimación.agregar(generarTareasEstimacionesPiezaGráfica(solicitud))

  retornar mapaTareaEstimación
finFunción

```

```

función generarTareasEstimacionesPublicidad (solicitud)

```

```

  tareas = nuevo Mapa()

  /** Planificación
   * Consiste en determinar el nombre público y la línea gráfica
   * del evento solicitado. Los eventos de menor frecuencia anual
   * (pe. el Congreso Anual o Navidad) requieren de mayor planificación
   */
  si solicitud.frecuenciaAnual == 1
    tareas.agregar('planificación', 12)
  o si solicitud.frecuenciaAnual <= 3
    tareas.agregar('planificación', 8)
  o si solicitud.frecuenciaAnual <= 12
    tareas.agregar('planificación', 6)
  o
    tareas.agregar('planificación', 4)

  /** Fotografía
   * Consiste en realizar una o más sesiones fotográficas, a fin de
   * disponer de imágenes para el diseño gráfico. Los eventos de frecuencia
   * anual suelen tener dos sesiones, y los eventos de frecuencia hasta
   * trimestral suelen tener una sesión fotográfica.
   */
  si solicitud.frecuenciaAnual == 1
    tareas.agregar('fotografía', 14)
  o si solicitud.frecuenciaAnual <= 4
    tareas.agregar('fotografía', 8)

  /** Vídeo
   * Consiste en realizar un cortometrajes, a fin utilizarlo para
   * la publicidad.
   * Los eventos de frecuencia anual suelen tener disponer de un vídeo.
   */
  si solicitud.frecuenciaAnual == 1
    tareas.agregar('vídeo', 16)

  /** Diseño gráfico
   * Consiste en elaborar los elementos gráficos necesarios para
   * el evento, tales como banners y afiches, tanto para medios impresos
   * como digitales.
   */
  horasDiseñoGráfico = 0
  si solicitud.requiereBanner
    horasDiseñoGráfico += 6
  si solicitud.requiereAfiche
    horasDiseñoGráfico += 6

  tareas.agregar('Diseño gráfico', horasDiseñoGráfico)

  retornar tarea
finFunción

```

**función generarTareasEstimacionesAtenciónEvento (solicitud)**

```
tareas = nuevo Mapa()

/** Acondicionamiento
 * Consiste en acondicionar la sede del evento. Los eventos de menor frecuencia
 * anual (pe. el Congreso Anual o Navidad) requieren de mayor acondicionamiento.
 */
si solicitud.frecuenciaAnual == 1
    tareas.agregar('cobertura-acondicionamiento', 12)
o si solicitud.frecuenciaAnual <= 3
    tareas.agregar('cobertura-acondicionamiento', 6)
o si solicitud.frecuenciaAnual <= 12
    tareas.agregar('cobertura-acondicionamiento', 2)
o
    tareas.agregar('cobertura-acondicionamiento', 1)

/** Cobertura
 * Compuesto por las horas invertidas en dar cobertura al evento. En todo caso,
 * es igual a la cantidad de horas que dura el evento.
 */

// en base a fechas y horas de inicio y fin
duraciónEvento = calcularDuraciónEvento(solicitud)

tareas.agregar('cobertura-general', duraciónEvento)

si solicitud.atenciónEvento.requiereFotografía
    tareas.agregar('cobertura-fotografía', duraciónEvento)
si solicitud.atenciónEvento.requiereVideo
    tareas.agregar('cobertura-video', duraciónEvento)

retornar tareas
finFunción
```

**función generarMapaTareaEstimaciónPiezaGráfica(solicitud)**

```
tareas = nuevo Mapa()

/** según indicado anteriormente, este tipo de solicitud es complejo
 * y es altamente variable. Por ello, solo se estima de manera general
 * asumiendo escenarios de solicitudes complejas.
 */
tareas.agregar('planificación', 18)
tareas.agregar('diseño', 12)
tareas.agregar('ejecución', 24) // imprenta, construcción, etc.
tareas.agregar('publicación', 8) // venta, liberación, apertura, etc.

retornar tareas
finFunción
```

## 5. Conclusiones

Se observa que el algoritmo especificado permite estimar una fecha de entrega de acuerdo al tipo y a la complejidad de cada Solicitud de Servicio. De esta manera, la Organización puede estimar de manera más tipificada cada caso.

## 7. PROTOTIPO FUNCIONAL

El propósito de esta sección es mostrar el prototipo funcional desarrollado del Sistema de Información. Para ello se presentan capturas de las pantallas más representativas para los casos de uso presentados anteriormente.

### 1. Estándar de diseño

En general, se ha seguido la guía de diseño propuesto por Google en la Guía de Diseño Material Design (Google, 2016). Para ello, se dispone de elementos comunes como Barra de navegación, Tarjeta, Botón, Caja de texto, Pestaña, entre otros. A continuación se detallan las elecciones más resaltantes que se realizaron para este Sistema.

El color primario elegido para el sistema es el Azul; y el color secundario, el Naranja. Ambos colores en la tonalidad 500, según lo especificado por Material Design para la elección de colores primario y secundario. El color primario se utiliza para los elementos principales, tales como color de fondo de la barra de navegación, color de fondo de botones, color de texto de los enlaces, entre otros. El color secundario se utiliza para denotar el estado de algunos elementos, tales como el borde inferior de la pestaña activa.

Asimismo, el color del texto es en su mayoría negro, a excepción de los textos con algún significado especial (por ejemplo, el texto de los enlaces es del color secundario), también según lo especificado por la guía de diseño.

La barra de navegación se ubica siempre en la parte superior de la pantalla ocupando todo el ancho de la misma. Esta barra incluye el título del Sistema y un botón para agregar una nueva Solicitud de Servicio.

Al interior del formulario de registro o edición, el proceso se divide en pestañas, una cada una de los subprocesos especificados en la imagen 4.2. Estas pestañas se muestran u ocultan según sea necesario, siguiendo el proceso modelado anteriormente. Para navegar entre cada pestaña, se ubicaron botones con el texto “Anterior” y “Siguiete” en la esquina inferior derecha de cada pestaña. La última pestaña tiene, en lugar del botón “Siguiete” un botón “Enviar”, para indicar al usuario que se almacenarán los datos indicados. Este botón solo está activo si todos los datos requeridos están ingresados y son válidos.

## 2. Pantalla inicial

Esta es la primera pantalla mostrada al usuario al ingresar al Sistema. Se muestra bajo la sección “Esta semana” las Solicitudes de Servicio que, según la fecha planificada, deben ser entregadas durante la semana presente. Asimismo, se muestra las Solicitudes de Servicio cuya fecha de entrega planificada es posterior a la semana presente, bajo la sección “Próximos”. El usuario es capaz de ver o editar cada solicitud al dar click sobre una de ellas. Además, el usuario puede ingresar una nueva solicitud haciendo click en el ícono “+” de la barra de navegación principal (esquina superior derecha).

La imagen 7.1 muestra una captura de la Pantalla inicial.

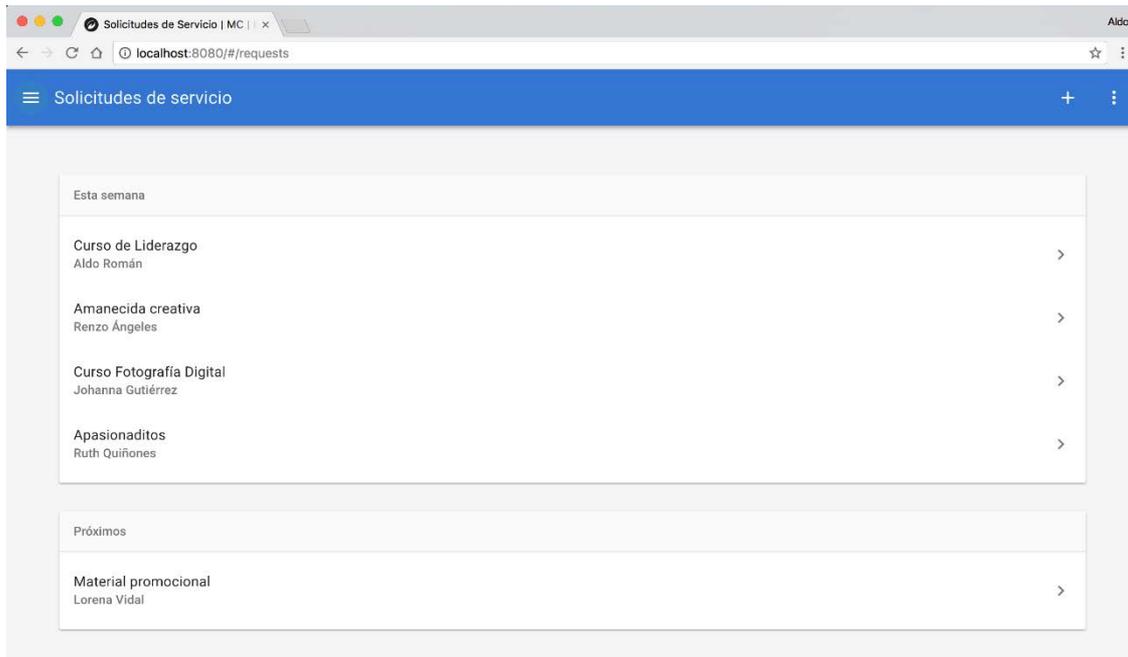


Imagen 7.1. Pantalla inicial del Sistema.

## 3. Registro de Solicitud de Servicio

Esta sección contiene diversas capturas de pantalla acerca del proceso de Registro de Solicitudes de Servicio. Se ha dividido en subsecciones según las diversas etapas de este proceso. Debido a la extensión y complejidad del proceso, se presentan extractos de pantalla relevantes.

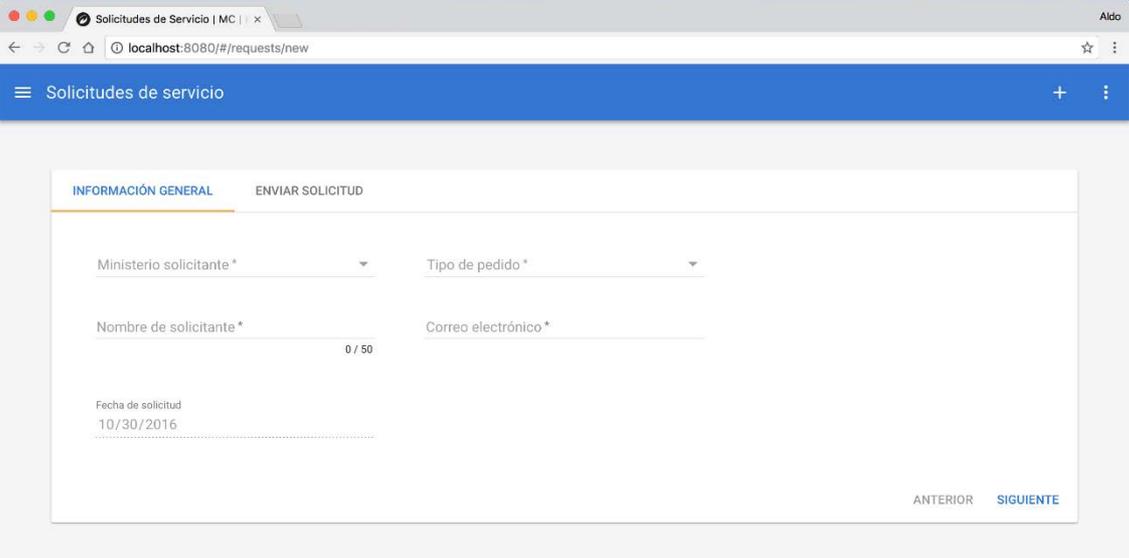
Se observa que, para una mayor facilidad en el llenado de información, las diversas etapas del proceso se han dividido en pestañas ubicadas en la barra superior, las cuales son:

- Información General
- Publicidad
- Atención de Evento
- Piezas Gráficas
- Enviar Solicitud

Al inicio, únicamente las pestañas Información General y Enviar Solicitud se visualizan. Tras completar los campos de la primera, las pestañas Publicidad, Atención de Evento y Piezas Gráficas aparecen en la barra de navegación según la elección del usuario y siguiendo los flujos de proceso descritos anteriormente.

## 1. Información General

Esta es la primera pantalla mostrada al usuario al ingresar a la opción “Ingresar una nueva Solicitud de Servicio”. La figura 7.2 muestra esta pantalla en su estado inicial. El usuario puede seleccionar el Ministerio solicitante y Tipo de Pedido desde una lista desplegable, así como especificar el Nombre de solicitante y Correo electrónico. La pantalla muestra la Fecha de solicitud de acuerdo a la fecha del computador en uso.



The screenshot shows a web browser window with the URL 'localhost:8080/#requests/new'. The page title is 'Solicitudes de Servicio | MC | x'. The main content area has two tabs: 'INFORMACIÓN GENERAL' (active) and 'ENVIAR SOLICITUD'. The form contains the following fields:

- Ministerio solicitante\* (dropdown menu)
- Tipo de pedido\* (dropdown menu)
- Nombre de solicitante\* (text input, 0 / 50 characters)
- Correo electrónico\* (text input)
- Fecha de solicitud (text input, 10/30/2016)

At the bottom right, there are two buttons: 'ANTERIOR' and 'SIGUIENTE'.

Imagen 7.2. Pantalla de Información General en su estado inicial.

De igual modo, la figura 7.3 muestra esta pantalla en su estado final. Se observa que, al haber seleccionado “Publicidad” como Tipo de pedido, las pestañas Publicidad y Atención de Evento son mostradas en la barra de navegación del formulario.

Adicionalmente, se provee al usuario de un texto de ayuda al lado de la lista desplegable de Tipo de pedido. Este texto sirve para indicar cuál es el Tipo de pedido óptimo para sus necesidades, especificando en qué casos debe seleccionar cada uno.

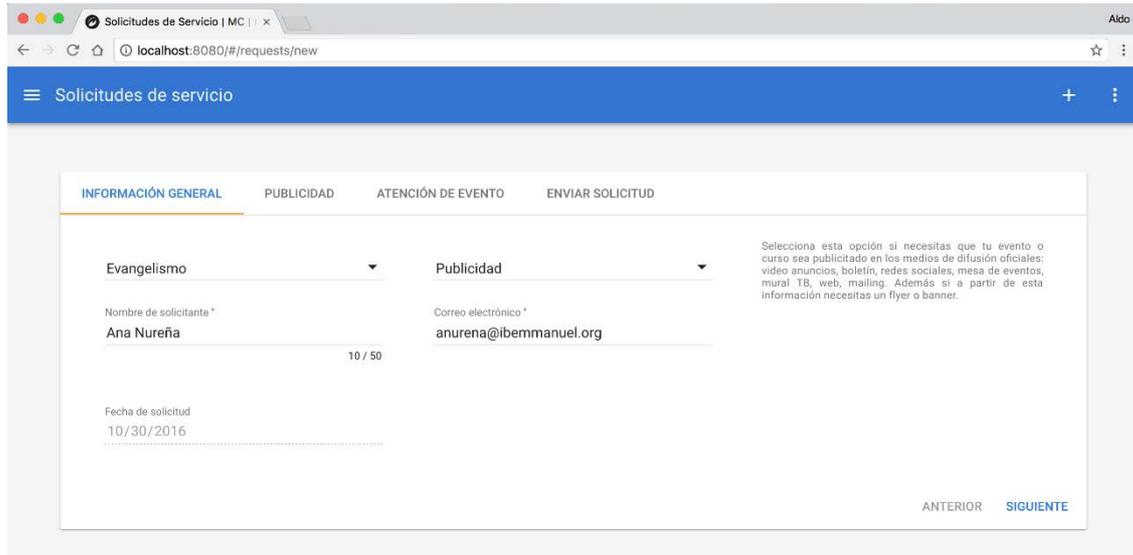


Imagen 7.3. Pantalla de Información General completada.

## 2. Publicidad

Esta pantalla permite al usuario ingresar los datos para Solicitudes de tipo Publicidad. La imagen 7.4 muestra una captura de la Pantalla de Publicidad en su estado inicial.

Se observa que el formulario es extenso, de acuerdo al flujo de proceso detallado anteriormente. Además, se observa que no son mostrados todos los campos de texto posibles, por ejemplo, “Invitado especial”, pues este campo es opcional solo si el Tipo de evento seleccionado es Curso, Taller o Evento. Esto permite ocultar información no relevante al usuario según los datos ingresados, de manera que pueda proseguir con el proceso de manera sencilla y rápida.

## 3. Atención de Evento

Esta pantalla es, según los flujos de proceso mostrados anteriormente, bastante similar a la pantalla Publicidad, por lo cual no se presenta en este documento en su totalidad. De esta pantalla, la imagen 7.5 muestra una captura de los componentes que permiten indicar al usuario los requerimientos de la Atención de evento.

Solicitudes de servicio

INFORMACIÓN GENERAL PUBLICIDAD ATENCIÓN DE EVENTO ENVIAR SOLICITUD

Tipo de evento \*  Nombre del evento \*  0 / 50

Fecha del Evento \*  Hora de inicio \*  Hora de fin \*

Repetición del evento \*

Seleccionar localidad \*  Seleccionar localidad \*

Invitado especial  0 / 50 Frecuencia anual del evento

Tamaño de folleto (si se requiere)  0 / 50 Tamaño de banner (si se requiere)  0 / 50

Corazón del evento \*  0 / 250

**Público objetivo**

Rango de edad \*  Sexo \*  Otros detalles  0 / 50

**Requisitos**

Disciplinado \*   No requiere membresía Otros requisitos:  0 / 50

**Inscripciones**

Dónde \*  Horario \*  Costo \*

Persona de contacto \*  Correo electrónico \*  Teléfono - Anexo \*

ANTERIOR SIGUIENTE

Imagen 7.4. Pantalla de Publicidad.

**Requerimientos**

No requiero proyector

No requiero micrófonos

No requiero parlantes

No requiero operador de sonido Seleccione esta opción solo si no cuenta con un operador de sonido

No requiero laptop

No requiero grabación de vídeo

No requiero fotografías del evento

Otros requerimientos adicionales  0 / 250

Imagen 7.5. Extracto de la Pantalla de Atención de Evento.

#### 4. Piezas Gráficas

Esta pantalla permite al usuario indicar el nombre del material que desea solicitar e indicar los tipos de material requeridos (uno o más). Este componente se muestra en un grupo de casillas de selección (*checkbox*). Asimismo, esta pantalla permite adjuntar un archivo con mayor información acerca del pedido, de manera que el Director de Ministerio de Comunicaciones pueda evaluar y comunicarse con el equipo relevante para coordinar la elaboración del pedido.

La imagen 7.6 muestra una captura de la pantalla de Piezas Gráficas.

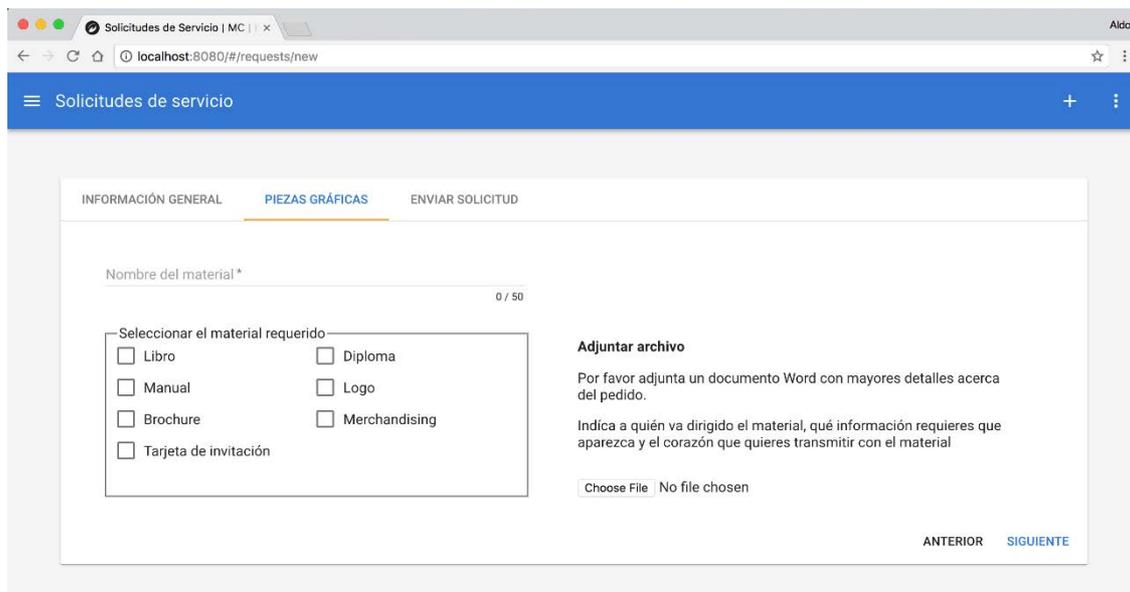


Imagen 7.6. Pantalla de Piezas Gráficas.

#### 5. Enviar Solicitud

Esta pantalla permite al usuario confirmar que todos los campos han sido ingresados correctamente. En caso de no ser así, un mensaje de error será mostrado. En su defecto, si los datos son correctos, se muestra un mensaje confirmando la dirección de correo electrónico a la cual se comunicará el sistema confirmando la Solicitud de Servicio. La imagen 7.7 muestra una captura de la pantalla de Enviar Solicitud.

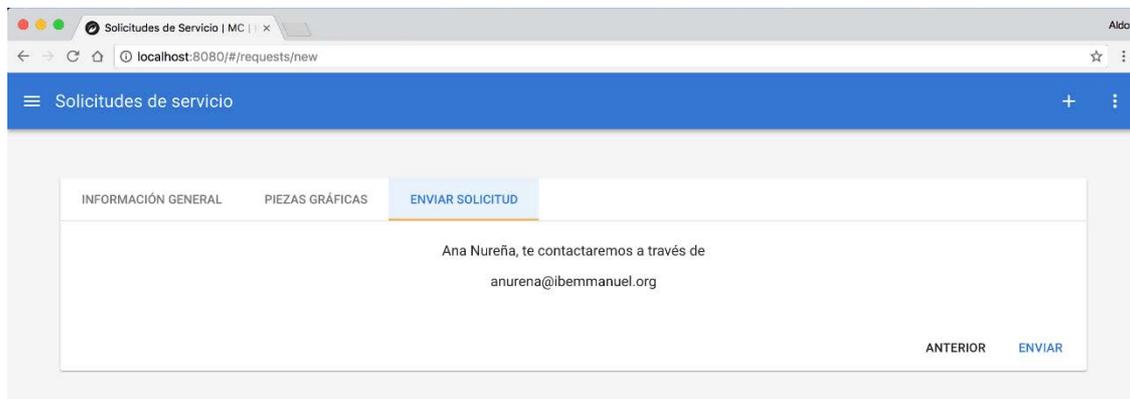
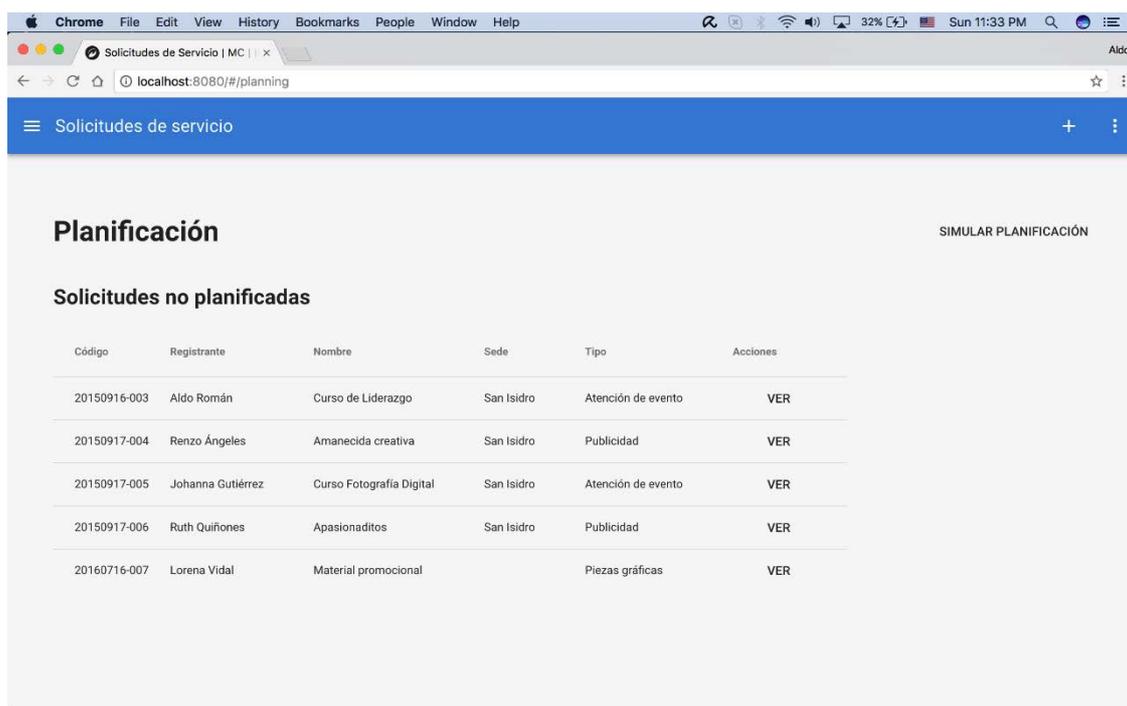


Imagen 7.7. Pantalla de Enviar Solicitud.

## 6. Planificación

Esta pantalla permite al usuario revisar qué Solicitudes de Servicio aún no han sido evaluadas por el algoritmo de Planificación y recibido una fecha de entrega luego de este proceso. Posteriormente, permite al usuario simular una Planificación, ejecutando el algoritmo descrito anteriormente y mostrando el resultado nuevamente al usuario.

La imagen 7.7 muestra una captura de la pantalla de Configuración de la Planificación.



Código	Registrante	Nombre	Sede	Tipo	Acciones
20150916-003	Aldo Román	Curso de Liderazgo	San Isidro	Atención de evento	VER
20150917-004	Renzo Ángeles	Amanecida creativa	San Isidro	Publicidad	VER
20150917-005	Johanna Gutiérrez	Curso Fotografía Digital	San Isidro	Atención de evento	VER
20150917-006	Ruth Quiñones	Apasionaditos	San Isidro	Publicidad	VER
20160716-007	Lorena Vidal	Material promocional		Piezas gráficas	VER

Imagen 7.7. Configuración de la Planificación.

## 4. Conclusiones

Se observa que, con el software presentado, el usuario es capaz de cumplir con el flujo de Registro de Solicitudes de Servicio. El formulario mostrado y sus campos requeridos varía según los datos que el usuario consigna, siguiendo el flujo y las condiciones especificadas anteriormente en este documento.

Asimismo, se observa que tanto el usuario Registrante como el Administrador del Sistema son capaces de conocer la fecha estimada para la Solicitud de Servicio según los datos consignados en el formulario de registro.

## 8. CONCLUSIONES

El proceso de Gestión de Solicitudes de Servicio representa para el Ministerio de Comunicaciones de la Iglesia Evangélica un proceso clave para cumplir sus labores. Este proyecto ha logrado modelar dicho proceso, de manera que ahora están mejor delimitados los pasos a seguir y los datos envueltos en cada etapa del proceso.

Esto ha permitido ordenar el registro de solicitudes de servicio, de manera que los datos requeridos no sean obviados o ingresados con errores. Asimismo, los datos requeridos varían según el tipo de solicitud que el registrante haga, ajustándose a los requerimientos de la Organización expuestos durante el relevamiento de información.

Además, el Sistema de Información se diseñó con una arquitectura de software que cubra los requerimientos de la organización, específicamente, con los procesos de Gestión de Solicitudes de Servicio y de Planificación de tareas.

En cuanto a la planificación de atención de solicitudes, se observa que el algoritmo especificado permite estimar la cantidad de horas-hombre que tomará cada solicitud, así como proponer un orden en que dichas solicitudes deben ser atendidas. De esta manera, la Organización puede estimar de manera más tipificada cada caso.

Finalmente, se observa que con el prototipo funcional el usuario es capaz de cumplir con el flujo de Registro de Solicitudes de Servicio utilizando interfaces gráficas, entre otras tareas. El formulario mostrado y sus campos requeridos varían según los datos que el usuario consigna, siguiendo el flujo y las condiciones especificadas anteriormente en este documento.

Asimismo, se observa que tanto el usuario Registrante como el Administrador del Sistema son capaces de conocer la fecha estimada para la Solicitud de Servicio según los datos consignados en el formulario de registro.

## 9. TRABAJOS FUTUROS

Si bien el proyecto logró cubrir las necesidades expuestas por la Organización, durante su elaboración se detectaron posibles mejoras que podrían otorgarle mayor valor a la Organización y que no fueron parte de este proyecto.

Uno de estos trabajos futuros es la modificación de la arquitectura de software planteada para que sea posible interactuar con otros sistemas de la Organización. Por ejemplo, el sistema podría consumir un servicio de autenticación y autorización que permita identificar usuarios desde un sistema o servicio externo (por ejemplo, Active Directory) existente en la Organización.

Otro trabajo futuro es implementar una comunicación cifrada entre el navegador y el servidor web, utilizando el protocolo HTTPS. Esto permitirá reducir el riesgo de pérdida de integridad de los datos al registrar o editar una Solicitud de Servicio. Asimismo, permitirá la implementación del protocolo HTTP/2, que supone varias mejoras al rendimiento de la comunicación entre navegador y servidor web, y que tiene como requisito la existencia de una comunicación cifrada.

También se tiene como trabajo futuro desplegar el Sistema en nodos físicos separados, a fin de incrementar el rendimiento con el menor costo posible. Esto se efectuará según la demanda del Sistema lo requiera; sin embargo, cabe destacar que la arquitectura fue diseñada previendo dicha posibilidad.

## 10. REFERENCIAS BIBLIOGRÁFICAS

Amat, O.; Pérez, L.

2007 Presencia evangélica en la sociedad peruana. Instituto para la Misión Transformadora. Consulta: 06-07-2013.

Angular

2015 What is Angular? Consulta: 01-11-2015

<https://docs.angularjs.org/guide/introduction>

Bizagi

2015 BPMN Software for Process Modeling. Consulta: 01-11-2015

<http://www.bizagi.com/en/products/bpm-suite/modeler>

BPMF

2014 WfMC Global Awards for Excellence in BPM and Workflow.

Consulta: 01-11-2015. <http://bpmf.org/slides/2014/index14.htm>

Church Windows Software

2015 Church Windows Management Software. Consulta: 11-09-2015.

<http://www.churchwindows.com>

De La Puente Candamo, José

1974 Obra Gubernativa y Epistolario de San Martín. Colección Documental de la Independencia del Perú, Volumen 13, Tomo 2. Publicado por Comisión Nacional del Sesquicentenario de la Independencia del Perú.

Feo, Thomas; Mauricio, Resende

1995 Greedy Randomized Adaptive Search Procedures. Journal of Global Optimization, 6, 109–134 (1995). Kluwer Academic Publishers, Boston.

Google Inc.

2015 Google Forms – create and analyze surveys, for free. Consulta: 11-09-2015.

<https://www.google.com/forms/about>

Google Inc.

2016 Material Design – Introduction – Google design guidelines. Consulta: 08-06-2016

<https://material.google.com/>

Instituto Nacional de Estadística e Informática (INEI)

2008 Perfil Social Demográfico del Perú. Dirección Técnica de Demografía y Estudios Sociales y Centro de Investigación y Desarrollo del Instituto Nacional de Estadística e Informática (INEI). Segunda Edición. Lima, Perú.

Jetbrains

2015 Jetbrains IntelliJ IDEA. Getting Started. Consulta: 12-11-15

<https://www.jetbrains.com/idea/documentation/>

Jetbrains

2016 Jetbrains IntelliJ IDEA. Download. Consulta: 27-04-16

<https://www.jetbrains.com/idea/download>

Krochmalski J.

2014 IntelliJ IDEA Essentials. Develop better software fast with IntelliJ IDEA. Birmingham, Packt Publishing Ltd.

Kroll, Per

2003 The rational unified process made easy: a practitioner's guide to the RUP. Editorial Addison-Wesley.

Krutchen, Philippe

1995 Architectural Blueprint —The “4+1” View Model of Software Architecture. IEEE Software 12 (6), pp. 42-50.

Microsoft Inc.

2016 Model-View-Controller. Consulta 27-04-2016

<https://msdn.microsoft.com/en-us/library/ff649643.aspx>

Object Management Group (OMG)

2013 Business Process Model Notation (BPMN). Versión 2.0.2. Consulta 01-11-2015.

<http://www.omg.org/spec/BPMN/>

Oracle Corporation

2015 Java Technologies: Develop with global standard. Consulta: 24-08-2015.

<https://www.oracle.com/java/technologies/index.html>

Paredes, David

1986 Diseño de un sistema de información y control para el circuito de solicitudes de servicio de una compañía telefónica. Tesis de licenciatura en Ciencias e Ingeniería con mención en Ingeniería Industrial. Lima: Pontificia Universidad Católica del Perú, Facultad de Ciencias e Ingeniería.

Román Nureña, Aldo

2015 "Entrevista realizada a Iglesia Evangélica". 24 de Agosto de 2015.

Shin, K.G.; Yi-Chieh

1995 A reservation-based algorithm for scheduling both periodic and aperiodic real-time tasks. En *Computers, IEEE Transactions on*, vol.44, no.12, pp.1405-1419.

Smith, T.; Pyle, J.

2004 An effective algorithm for project scheduling with arbitrary temporal constraints. Publicado en Proceedings of the 19th national conference on Artificial intelligence. pp. 544-549.

Mingsheng, Shang; Shixin, Sun; Qingxian, Wang

2003 An efficient parallel scheduling algorithm of dependent task graphs. En *Parallel and Distributed Computing, Applications and Technologies*, 2003. PDCAT'2003. Proceedings of the Fourth International Conference on, vol., no., pp.595-598, 27-29.

Play Framework

2015 Play 2.4.x Documentation. Consulta 12-11-15

<https://playframework.com/documentation/2.4.x/Home>

TIOBE Software

2015 TIOBE index for June 2015. Consulta: 08-06-2015

<http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>

Tang, Zhuo; Zhou, Junqing; Li, Kenli; Li, Ruixuan

2012 MTSD: A Task Scheduling Algorithm for MapReduce Base on Deadline Constraints. En Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), 2012 IEEE 26th International, pp. 21-25.

TouchPoint Software

2015 TouchPoint Software. Consulta: 11-09-2015.

<http://www.touchpointsoftware.com>

Typesafe Inc.

2015 Play Framework - Build Modern & Scalable Web Apps with Java and Scala. Consulta: 01-11-2015.

<https://www.playframework.com/>

