Portada

**Índice General**

[Índice de Tablas 4](#_gjdgxs)

[Índice de Figuras 5](#_30j0zll)

[Resumen 6](#_1fob9te)

[Abstract 7](#_3znysh7)

[1.- Introducción 8](#_2et92p0)

[2.- Definición de la institución 9](#_tyjcwt)

2.1.- Descripción de la institución 9

2.2.- Descripción del área de estudio 10

2.3.- Descripción de la problemática 10

2.3.1.- Situación actual 10

2.3.2.- Problemas detectados 10

[3.- Definición del proyecto 11](#_3dy6vkm)

[3.1.- Objetivos del proyecto 11](#_1t3h5sf)

[3.2.- Ambiente de Ingeniería de Software 11](#_4d34og8)

[3.2.1 Metodología de desarrollo: 11](#_2s8eyo1)

[3.2.2 Técnicas: 11](#_17dp8vu)

[3.2.3 Notaciones: 11](#_3rdcrjn)

[3.2.4 Estándares de documentación, producto o proceso 11](#_26in1rg)

[3.2.5 Herramientas de apoyo al desarrollo de software que serán utilizadas 11](#_lnxbz9)

[3.3.- Definiciones, siglas y abreviaciones 11](#_35nkun2)

[4.- Especificación de requerimientos de software 12](#_1ksv4uv)

[4.1.- Alcances 12](#_44sinio)

[4.2.- Objetivo del software 12](#_2jxsxqh)

[4.3.- Descripción global del producto 12](#_z337ya)

[4.3.1.- Interfaz de usuario 12](#_3j2qqm3)

[4.3.2.- Interfaz de hardware 12](#_1y810tw)

[4.3.3.- Interfaz de software 12](#_4i7ojhp)

[4.4.- Requerimientos específicos 12](#_2xcytpi)

[4.4.1.- Requerimientos funcionales del sistema 12](#_1ci93xb)

[4.4.2.- Interfaces externas de entrada 12](#_3whwml4)

[4.4.3.- Interfaces externas de salida 12](#_2bn6wsx)

[4.4.4.- Atributos del producto 12](#_qsh70q)

[5.- Factibilidad 13](#_3as4poj)

[5.1.- Factibilidad Técnica 13](#_1pxezwc)

[5.2.- Factibilidad Operativa 13](#_49x2ik5)

[5.3.- Factibilidad Económica 13](#_2p2csry)

[5.4.- Conclusión de la factibilidad 13](#_147n2zr)

[6.- Análisis 14](#_3o7alnk)

6.1.- Diagrama de flujo de datos 14

[6.2.- Casos de Uso 14](#_23ckvvd)

[6.2.1.- Actores 14](#_ihv636)

[6.2.2.- Diagrama de casos de uso y descripción 14](#_32hioqz)

[6.2.3.- Especificación de los casos de uso 14](#_1hmsyys)

[6.3.- Modelamiento de datos 14](#_41mghml)

[Bibliografía 15](#_vx1227)

# Índice de Tablas

Índice de Figuras

# 

# Resumen

# Abstract

# 1.- Introducción

# 2 Definición del problema

## 2.1 Descripción de la problemática

A continuación se detallará la problemática para comprender de mejor manera cuales son las aristas que el proyecto tratará.

“El ser humano es un ser social por naturaleza”, y también “un ser curioso”, juntando esas dos premisas hay un problema en común, este es que al momento, y sobre todo cuando se llega a un nuevo entorno social (que puede ser una nueva ciudad, o Chile si se viene del extranjero), existe la dificultad, frente a la necesidad de conocer y socializar, una guía para tal propósito, no hay ayuda a la hora de querer encontrar o a lo menos saber que ocurre cerca, y por lo tanto se suelen hacer preguntas como estás: “¿Qué es eso que se escucha?” , “¿Es algún tipo de evento social y público masivo?”, “¿Será de mi gusto?”, “¿Existirá algún tipo de feria local cercana?”. Esas son algunas de las preguntas que pueden ocurrir y se buscan responder.

## 2.2 Situación actual

Actualmente si alguna persona quiere saber sobre algún evento social, una de las primeras fuentes de información son Facebook, Google (con su buscador o Maps), en resumen; Internet.

Si la persona está interesada en buscar sobre un evento que aún no se lleva a cabo y tiene cierta noción de que tipo es, un concierto, cultural, etc., hay tres posibles resultados; no encuentra nada, muy poco al respecto o toda la información requerida.

Ahora si la persona está interesada en conocer qué tipos de eventos se realizan cercanos a su ubicación, igualmente lo hace a través de los medios antes nombrados, con resultados similares a los descritos en el apartado anterior.

## 2.3 Problemas detectados

En este apartado se listan los principales problemas detectados en la problemática, los cuales son vitales para el desarrollo de este proyecto.

En general las herramientas nombradas son buenas y aportan con la información sobre eventos muy masivos. No así con los de menor envergadura o con eventos sociales que son de un aspecto más familiar o social.

Aún así los principales problemas detectados durante el proceso de búsqueda de algún evento son:

* Al buscar un evento de una envergadura inferior al promedio de los eventos actuales, no se encuentra información suficiente para la satisfacción del usuario.
* Al querer buscar un evento social cercano a la ubicación, no existe una forma, si no se conoce a lo menos el nombre o descripción del evento.
* Hay nula o muy poca información respecto a ferias periódicas y locales, en distintas ubicaciones.
* No hay manera de listar todos los eventos cercanos a una ubicación.
* No hay información respecto a la ubicación de algunos eventos.
* No existe un historial global de eventos que fueron realizados en una determinada ubicación.

# 3 Definición del proyecto

En este apartado se define el proyecto…

## 3.1 Objetivos del proyecto

Un objetivo es el planteamiento de una meta o un propósito a alcanzar y es fundamental en el proceso de planificación.

A continuación se muestran los objetivos generales y específicos del proyecto que son el propósito de lo que se quiere lograr.

Objetivo General

Diseñar e implementar un servicio web que permita gestionar y notificar la información de eventos sociales en espacios públicos para ser consumido por una aplicación móvil Android y una página web.

Objetivos Específicos

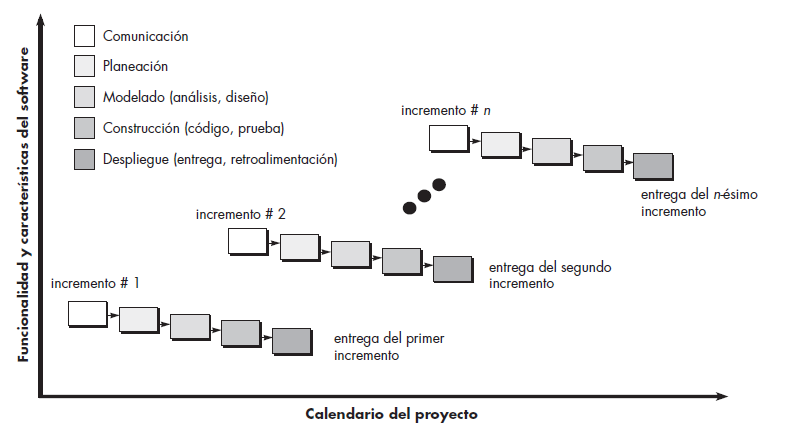
* Crear y mantener una base de datos relacional con la información de los eventos sociales que se realizan en espacios públicos, como pueden ser conciertos, ferias artesanales, etc.
* Diseñar e implementar un servicio web utilizando la arquitectura REST que disponga la información almacenada sobre los eventos sociales en espacios públicos y pueda ser consumido por cualquier aplicación y sistema compatible.
* Integrar la API proporcionada por Facebook con el servicio web a desarrollar, para complementar la información de los eventos sociales.
* Diseñar e implementar una aplicación móvil Android que consuma el servicio web desarrollado, para mostrar la información y ubicación de los eventos sociales, además de notificar si existe algún evento cercano a la ubicación actual según el GPS del dispositivo.
* Diseñar e implementar una página web que consuma el servicio web desarrollado, para visualizar y administrar la información de los eventos sociales.
* Integrar las APIs proporcionadas por Facebook y Google con el servicio web a desarrollar para lograr la autentificación y posterior inicio de sesión en la aplicación móvil y página web.

## 3.2 Ambiente de Ingeniería de Software

En este apartado se expone el ambiente en el que está inmerso el proyecto, es decir, los diversos elementos que interactúan directamente con la realización del mismo.

### 3.2.1 Metodología de desarrollo:

La metodología de desarrollo a utilizar es el modelo incremental, el modelo incremental ejecuta una serie de avances, llamados incrementos, que en forma progresiva dan más funcionalidad al cliente conforme se entrega cada incremento (Pressman, 2010). Este es uno de los factores claves de la elección de esta metodología, puesto que la coordinadora del programa de aprendizaje y servicio, requería entregas de funcionalidades sujetas a evaluación a lo largo del desarrollo del proyecto, esto se adapta a las entregas de funcionalidades por incrementos de esta metodología.



*Figura N°1: Muestra las etapas de desarrollo en diferentes incrementos.*

Este proceso de desarrollo incremental tiene varias ventajas:

1. No se tiene que esperar hasta que el sistema completo se entregue para sacar provecho de él. El primer incremento satisface los requerimientos más críticos de tal forma que pueden utilizar el software inmediatamente.

2. Se pueden utilizar los incrementos iniciales como prototipos y obtener experiencia sobre los requerimientos de los incrementos posteriores al sistema.

3. Existe un bajo riesgo de un fallo total del proyecto. Aunque se pueden encontrar problemas de algunos incrementos, lo normal es que el sistema se entregue de forma satisfactoria.

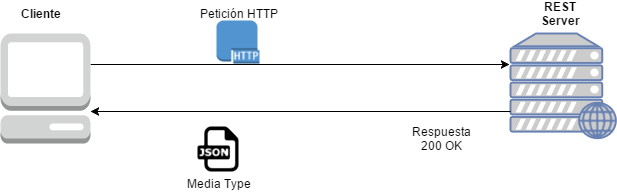
4. Puesto que los servicios de más alta prioridad se entregan primero, y los incrementos posteriores se integran en ellos, es inevitable que los servicios más importantes del sistema sean a los que se le hagan más pruebas. Esto significa que es menos probable que los clientes encuentren fallos de funcionamiento del software en las partes más importantes del sistema.

(Sommerville, 2005)

### 3.2.2 Arquitectura de Software

#### 3.2.2.1 Arquitectura del servicio web

Para el desarrollo del servicio web se utilizará la arquitectura REST, Acrónimo de REpresentational State Transfer, es un estilo de arquitectura software que define un conjunto de principios, por los cuales se diseñan servicios web haciendo foco en los recursos del sistema, además de cómo se acceden a dichos recursos y cómo se transfieren por el protocolo de transporte HTTP hacia clientes escritos en diversos lenguajes.



#### 3.2.2.2 Arquitectura de la aplicación móvil

#### 3.2.2.3 Arquitectura de la página web

Probablemente sea MVC…

### 3.2.3 Patrones de Diseño

Un patrón de diseño es una descripción de un problema que ocurre una y otra vez en nuestro entorno, y de la esencia de la solución a dicho problema, de tal forma que se pueda usar esta solución muchas veces a modo de reutilizar código fuente, reduciendo tiempo de diseño (Pressman, 2005).

En el desarrollo de la solución se utilizan los siguientes patrones de diseño:

* Singleton:
* Builder:
* Dependency Injection:
* Template Method:

### 3.2.4 Tecnologías

En esta sección se definen brevemente las tecnologías a utilizar durante el proceso de  
desarrollo de la solución.

* Android: Es un sistema operativo desarrollado por Google Inc. para dispositivos móviles. Tal como se menciona en la sección Situación Actual del Capítulo I, es una de las tecnologías más usadas en smartphones, debido a su funcionamiento en una amplia gama de dispositivos.
* Java: Es un lenguaje de programación Orientado a Objetos (OO). Su principal característica es ser multiplataforma, por lo que puede ser ejecutado en prácticamente cualquier sistema.
* Spring Framework: es un marco de aplicaciones Java que facilita enormemente la implementación de distintos patrones de diseño y la integración con una gran cantidad de tecnologías.
* Spring Boot: es un proyecto de Spring que facilita la creación de aplicaciones independientes basadas en el Framework de Spring. De esa manera se puede iniciar el desarrollo sin mayores complicaciones en la configuración inicial.
* HTML: Es la sigla que representa HyperText Markup Language. Es un lenguaje de marcado que permite la elaboración de páginas webs.
* CSS: Es la sigla que representa Cascading Style Sheet. Es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas.
* SQL: Es la sigla que representa Structured Query Language. Es un lenguaje diseñado para gestionar bases de datos relacionales mediante consultas y sentencias de ejecución.

### 3.2.2 Técnicas y Notaciones

**Unified Modeling Language (UML):** Lenguaje que será utilizado para modelar el diagrama de clases conceptuales.

### 3.2.4 Estándares de documentación

* **Estándar ISO/IEC/IEEE 29418:** Dirigido a la correcta especificación de requisitos funcionales y no funcionales.
* **Plantilla de Documentación:** Específica estándares en la realización del proyecto, específicamente:
  + Requisitos funcionales, no funcionales y restricciones
  + Especificación de los Casos de uso.

### 3.2.5 Herramientas de apoyo al desarrollo de software que serán utilizadas

* **Microsoft Project:** Permite crear y modificar la carta Gantt de manera fácil y ordenada.
* **JetBrains IntelliJ IDEA Ultimate:** Permite desarrollar aplicaciones en el lenguaje JAVA.
* **JetBrains DataGrip:** Herramienta cuyo propósito sirve para la gestión de las base de datos relacionales tales como sqlite, mysql, oracle, entre otras.
* **MySQL:** Provee al proyecto una base de datos que se comunica con el sistema.
* **API de Google Maps:** Esta API provee al proyecto la posibilidad de mostrar a los usuarios las ubicaciones exactas de los recintos deportivos.
* **Google Docs:** Permite colaborar y editar de manera online en documentos de manera simultánea.

## 3.3 Definiciones, siglas y abreviaciones

* **REST:** La Transferencia de Estado Representacional (Representational State Transfer) es un estilo de arquitectura software para sistemas hipermedia distribuidos como la World Wide Web que utiliza el protocolo HTTP o HTTPS para enviar o recibir información en cualquier formato (XML, JSON, etc…).
* **API:** La interfaz de programación de aplicaciones, abreviada como *API* (*Application Programming Interface)*, es un conjunto de subrutinas, funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro *software* como una capa de abstracción.
* **Endpoint**: El punto final es un punto de conexión donde se exponen los archivos HTML o páginas Active Server. Los extremos proporcionan la información necesaria para hacer frente a un servicio de punto final web.

# 4.- Especificación de requerimientos de software

En este apartado se define qué es lo que se requiere para solucionar la problemática en términos de software, esto guiará el desarrollo para lograr un software funcional que satisfaga sus necesidades.

## 

## 4.1 Alcances y Limitaciones

Los alcances definidos para el proyecto son los siguientes;

* Un evento puede ser agrega por cualquier usuario registrado, no necesariamente por su propietario.
* Dentro del proyecto se incluye sólamente ciudades chilenas con la finalidad de acotar su alcance y ajustar a plazo definido por el desarrollo
* Se considera “Eventos Cercanos” los que estén ubicados hasta 1km de distancia (de radio teniendo como punto central el usuario). Dicha distancia debe ser calculada con la fórmula de Harvesine.
* (La fórmula del semiverseno es una importante ecuación para la navegación astronómica, en cuanto al cálculo de la distancia de círculo máximo entre dos puntos de un globo sabiendo su longitud y su latitud. Es un caso especial de una fórmula más general de trigonometría esférica, la ley de los semiversenos , que relaciona los lados y ángulos de los "triángulos esféricos". http://www.movable-type.co.uk/scripts/latlong.html1)

## 4.2 Objetivo del software

En esta sección se definen los objetivos del software que son el propósito que quiere lograr el sistema una vez que esté implementado.

### 4.2.1 Objetivo general

Desarrollar un sistema que permita gestionar y geolocalizar eventos públicos que se encuentren dentro del país.

### 4.2.2 Objetivos Específicos

* Buscar eventos por categorías a partir de una ubicación origen definida por GPS o de forma manual.
* Explorar la zona geográfica en donde se encuentra el usuario, es decir, mostrar todos los eventos cercanos a él sin filtrar por categorías.
* Publicar eventos nuevos, ingresando información como nombre, descripción, fecha inicio, fecha fin, valor, servicios prestados y una o varias imágenes representativa del evento capturada desde la cámara del dispositivo o almacenada en la memoria.
* Gestionar los eventos ingresados por los usuarios, es decir, añadir, editar, visualizar y eliminar los eventos publicados por los usuarios.

## 4.3 Descripción global del producto

### 4.3.1 Interfaz de usuario

### 4.3.2 Interfaz de hardware

### 4.3.3 Interfaz de software

## 4.4 Requerimientos específicos

En esta sección se presentan los requisitos funcionales que definen el comportamiento del sistema, es decir, que hace el sistema. Además de la especificación mencionada, se definen criterios en la tabla estructurada para mostrar no solo el requisito, sino que su estado y posición en el proyecto.

Cabe mencionar, que al ser la metodología iterativa e incremental la elegida, existirán un total de 3 incrementos, que se traducirán en un incremento para cada hito importante de la solución, que son; el servicio web, la aplicación móvil y la página web. Por lo tanto a continuación se definen los requisitos correspondientes a cada uno de los hitos antes mencionados.

### 4.4.1 Requerimientos funcionales del servicio web

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Nombre | Descripción | Prioridad | Método de verificación |
| RF\_STKSWE\_\_S\_01 | Iniciar Sesión con  Facebook. | El servicio web deberá proveer la opción de iniciar sesión utilizando la API de autenticación de Facebook. | Alta | P |
| RF\_STKSWE\_S\_02 | Iniciar Sesión con Google. | El servicio web debe proveer la opción de iniciar sesión utilizando la API de autenticación de Google. | Alta | P |
| RF\_STKSWE\_S\_03 | Registrar usuario | El servicio web permitirá registrar la información de un usuario. | Alta | P |
| RF\_STKSWE\_S\_04 | Iniciar sesión | El servicio web permitirá iniciar sesión con nombre usuario y contraseña registrados. | Alta | P |
| RF\_STKSWE\_S\_05 | Cerrar sesión | El servicio web permitirá cerrar una sesión ya iniciada por un administrador o usuario registrado. | Alta | P |
| RF\_STKSWE\_S\_06 | Suspender cuenta usuario | El servicio web permitirá al administrador suspender la cuenta de un usuario registrado. | Alta | P |
| RF\_STKSWE\_S\_07 | Mostrar información de un usuario | El servicio web deberá proveer la información básica de un perfil de usuario, a la que solo podrán acceder un administrador o usuario registrado con la sesión iniciada. | Media | P |
| RF\_STKSWE\_S\_08 | Modificar perfil del usuario. | El servicio web permitirá al usuario de la sesión modificar los datos de su perfil. | Alta | p |
| RF\_STKSWE\_\_S09 | Ingresar evento | El servicio web permitirá ingresar un nuevo evento a un usuario registrado y que tenga su sesión iniciada. | Alta | P |
| RF\_STKSWE\_S\_10 | Complementar información de evento. | El servicio web podrá complementar la información de un evento al momento de crearlo. | Baja | P |
| RF\_STKSWE\_S\_11 | Agregar eventos Facebook | El servicio web deberá comunicarse y consumir la API de Facebook que provee eventos de Facebook. | Baja | P |
| RF\_STKSWE\_S\_12 | Modificar evento | El servicio web permitirá al administrador y al usuario que creó el evento modificar la información ingresada. | Alta | P |
| RF\_STKSWE\_S\_13 | Brindar información de evento. | El servicio web deberá brindar la información de un evento a usuarios registrados y no registrados. | Alta | P |
| RF\_STKSWE\_S\_14 | Brindar la información de eventos cercanos a un área. | El servicio web deberá brindar la información de todos los eventos cercanos a un área dada a usuarios registrados y no registrados. | Alta | P |
| RF\_STKSWE\_S\_15 | Buscar eventos | El servicio web deberá ser capaz de entregar uno o más eventos de acuerdo a un criterio de búsqueda otorgado por usuarios registrados y no registrados. | Alta | P |
| RF\_STKSWE\_S\_16 | Notificar evento cercano | El servicio web deberá notificar cuando exista un evento cercano a la ubicación obtenida de un usuario registrado. | Alta | P |
| RF\_STKSWE\_S\_17 | Actividades de un usuario | El servicio web permitirá al administrador obtener la actividad o las actividades realizadas por un usuario. | Media | p |
| RF\_STKSWE\_S\_18 | Promedio de actividades realizadas | El servicio web permitirá al administrador obtener un promedio de las actividades realizadas por los usuarios registrados y no registrados. | Baja | P |
| RF\_STKSWE\_S\_19 | Eliminar evento. | El servicio web permitirá al administrador y al creado eliminar un evento previamente seleccionado. | Alta | P |
| RF\_STKSWE\_S\_20 | Cantidad de eventos a realizarse | El servicio web deberá entregar la cantidad de eventos a realizarse respecto a la fecha y hora que se consulte dicha información. | Media | P |
| RF\_STKSWE\_S\_21 | Listar usuarios | El servicio web deberá entregar una lista de los usuarios registrados en el sistema. | Alta | P |
| RF\_STKSWE\_S\_22 | Listar eventos | El servicio web deberá entregar una lista de los eventos registrados en el sistema. | Alta | P |

*\*SWE: Servicio web para notificaciones de eventos sociales en espacios públicos consumido por una aplicación móvil y una página web.*

*\*RF: Requisito funcional*

*\*STK: Requisito de grupo de interés*

*\*S: Servicio web*

*\*P: Prueba, D: Demostración, I: Inspección, A: Análisis*

### 4.4.2 Requerimientos funcionales de la aplicación móvil

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Nombre | Descripción | Prioridad | Método de verificación |
| RF\_STKSWE\_A\_01 | Iniciar sesión | La aplicación móvil deberá ser capaz de iniciar sesión con las opciones proveídas por el servicio web. | Alta | P |
| RF\_STKSWE\_A\_02 | Cerrar sesión | La aplicación móvil deberá consumir el servicio web y ser capaz de cerrar una sesión iniciada | Alta | P |
| RF\_STKSWE\_A\_03 | Registrar usuario | La aplicación móvil deberá ser capaz de enviar la información al servicio web para registrar un usuario. | Alta | P |
| RF\_STKSWE\_A\_04 | Mostrar información de usuario | La aplicación móvil deberá ser capaz de consumir el servicio web para mostrar a usuarios registrados y administrados la información de un perfil de usuario en específico. | Media | P |
| RF\_STKSWE\_A\_05 | Modificar perfil de usuario | La aplicación móvil deberá ser capaz de consumir el servicio web para modificar el perfil de usuario. | Alta | P |
| RF\_STKSWE\_A\_06 | Ingresar evento | La aplicación móvil deberá ser capaz de consumir el servicio web para permitir ingresar un evento a los usuarios registrados. | Alta | P |
| RF\_STKSWE\_A\_07 | Visualizar información evento. | La aplicación móvil deberá consumir el servicio web para visualizar la información de un evento en conjunto con un mapa indicando la ubicación del mismo. | Alta | P |
| RF\_STKSWE\_A\_08 | Visualizar eventos cercanos | La aplicación móvil deberá consumir el servicio web para visualizar en un mapa los eventos cercanos a una ubicación específica. | Alta | P |
| RF\_STKSWE\_A\_09 | Notificar evento cercano | La aplicación móvil deberá consumir el servicio web para entregar la ubicación del GPS y notificar cuando exista un evento cercano a los usuarios registrados y con la sesión iniciada. | Alta | P |
| RF\_STKSWE\_A\_10 | Buscar evento | La aplicación móvil deberá consumir el servicio web para entregar los criterios de búsqueda de un evento y visualizar los resultados en una lista o mapa. | Alta | P |

*\*SWE: Servicio web para notificaciones de eventos sociales en espacios públicos consumido por una aplicación móvil y una página web.*

*\*RF: Requisito funcional*

*\*STK: Requisito de grupo de interés*

*\*A: Aplicación móvil*

*\*P: Prueba, D: Demostración, I: Inspección, A: Análisis*

### 4.4.3 Requerimientos funcionales de la página web

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Nombre | Descripción | Prioridad | Método de verificación |
| RF\_STKSWE\_W\_01 | Iniciar sesión | La página web deberá ser capaz de iniciar sesión con las opciones proveídas por el servicio web. | Alta | P |
| RF\_STKSWE\_W\_02 | Cerrar sesión | La página web deberá consumir el servicio web y ser capaz de cerrar una sesión iniciada. | Alta | P |
| RF\_STKSWE\_W\_03 | Registrar usuario | La página web deberá ser capaz de enviar la información al servicio web para registrar un usuario. | Alta | P |
| RF\_STKSWE\_W\_04 | Modificar perfil de usuario | La página web deberá ser capaz de consumir el servicio web para modificar el perfil de usuario. | Alta | P |
| RF\_STKSWE\_W\_05 | Ver perfil de usuario | La página web deberá ser capaz de consumir el servicio web y mostrar la información de un usuario. | Alta | P |
| RF\_STKSWE\_W\_06 | Ingresar evento | La página web deberá ser capaz de consumir el servicio web que permita ingresar un evento a los usuarios registrados y con la sesión iniciada | Alta | P |
| RF\_STKSWE\_W\_07 | Visualizar información evento. | La página web deberá consumir el servicio web para visualizar la información de un evento en conjunto con un mapa indicando la ubicación del mismo. | Alta | P |
| RF\_STKSWE\_W\_08 | Visualizar eventos cercanos | La página web deberá consumir el servicio web para visualizar en un mapa los eventos cercanos a una ubicación específica. | Alta | P |
| RF\_STKSWE\_W\_09 | Buscar evento | La página web deberá consumir el servicio web para entregar los criterios de búsqueda de un evento y visualizar los resultados en una lista o mapa. | Alta | P |
| RF\_STKSWE\_W\_10 | Eliminar evento | La página web deberá consumir el servicio web y ser capaz de eliminar un evento, siempre que sea una acción de un administrador o del usuario que ingresó el evento. | Media | P |
| RF\_STKSWE\_W\_11 | Suspender cuenta usuario | La página web deberá consumir el servicio web y ser capaz de suspender la cuenta de un usuario, siempre que sea una acción de un administrador. | Alta | P |
| RF\_STKSWE\_W\_12 | Modificar evento | La página web deberá consumir el servicio web y ser capaz de modificar un evento al usuario que lo creó o al administrador. | Alta | P |
| RF\_STKSWE\_W\_13 | Listar usuarios | La página web deberá consumir el servicio web y ser capaz de mostrar una lista con los usuarios registrados. | Alta | P |
| RF\_STKSWE\_W\_14 | Listar eventos | La página web deberá consumir el servicio web y ser capaz de mostrar una lista de los eventos registrados en el sistema. | Alta | p |
| RF\_STKSWE\_W\_15 | Ver actividades de un usuario | La página web deberá consumir el servicio web y ser capaz de mostrar la o las actividades de un usuario registrado. | Media | P |
| RF\_STKSWE\_W\_16 | Ver promedio de actividades realizadas | La página web deberá consumir el servicio web y ser capaz de mostrar el promedio de actividades más realizadas por usuarios registrados y no registrados. | Baja | P |

*\*SWE: Servicio web para notificaciones de eventos sociales en espacios públicos consumido por una aplicación móvil y una página web.*

*\*RF: Requisito funcional*

*\*STK: Requisito de grupo de interés*

*\*W: Página web*

*\*P: Prueba, D: Demostración, I: Inspección, A: Análisis*

### 4.4.4 Interfaces externas de entrada

Cada interfaz de entrada indica todos los grupos de datos que serán ingresados al sistema independiente del medio de ingreso.

|  |  |  |
| --- | --- | --- |
| ID | Nombre | Datos o atributos |
| IEE\_01 | Ingresar al sistema | Nombre de usuario (email), contraseña. |
| IEE\_02 | Registrarse en el sistema | Nombres, apellidos, email, foto, contraseña. |
| IEE\_03 | Datos evento | Nombre evento, categoría, lugar, dirección, descripción, fotos. |
| IEE\_04 | Datos modificar evento | Nombre evento, categoría, lugar, dirección, descripción, fotos. |
| IEE\_05 | Datos perfil | Nombres, apellidos, email, foto. |
| IEE\_06 | Datos modificar perfil | Nombres, apellidos, email, foto, contraseña. |
| IEE\_07 | Coordenadas GPS | Coordenada GPS |
| IEE\_08 | Dirección | Dirección |
| IEE\_09 | Buscar evento | Nombre evento, categoría, lugar, dirección |
| IEE\_10 | Buscar perfil | Nombre usuario, email |
|  |  |  |

### 4.4.5 Interfaces externas de salida

Se especifica cada salida del sistema, indicando en cada caso el formato o medio de salida.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Nombre | Datos o atributos | Medio Salida |
| IES\_01 | Lista eventos | Nombre evento, foto o imagen. | Pantalla |
| IES\_02 | Datos evento | Nombre evento, categoría, lugar, dirección, descripción, fotos. | Pantalla |
| IES\_03 | Mapa con eventos | Nombre evento, punto en el mapa. | Pantalla |
| IES\_04 | Notificación evento cercano | Nombre evento, ubicación. | Pantalla |
| IES\_05 | Coordenadas de eventos | Coordenadas eventos | Http |
| IES\_06 | Lista usuarios | Nombre usuario, foto | Pantalla |
| IES\_07 | Perfil usuario | Nombre usuario, apellidos, email, foto | Pantalla |
| IES\_08 | Actividad de usuario | Nombre, descripción, tipo, IP | Pantalla |
|  |  |  |  |

### 4.4.6 Atributos del producto

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Nombre | Descripción | Subcaracteristica de calidad | Prioridad | Metrica | Método de verificación |
| RNF\_STKSWE\_001 | Acceso al sistema | El acceso al sistema debe estar restringido por el uso de clave y login por cada uno de los usuarios. | Autenticidad | Alta | Todos los usuarios registrados pueden ingresar al sistema | P |
| RNF\_STKSWE\_002 | Disponibilidad del sistema | El servicio web debe estar disponible las 24 horas del día los 7 días de la semana. | Disponibilidad | Alta | Horas totales del sistema funcionando a la semana. | P |
| RNF\_STKSWE\_003 | Notificación de errores | Cuando exista un error, el servicio web debe presentar mensajes que permitan a los usuarios identificar el tipo de error. | Protección frente a errores del usuario | Media | Número total de protección de campos implementados con respecto al total. | P |
| RNF\_STKSWE\_004 | Portabilidad del servicio web | Portabilidad del servicio web. | Adaptabilidad | Alta | Funcionar en distintos sistemas capaces de consumir un servicio web REST | D |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

*\*SWE: Servicio web para notificaciones de eventos sociales en espacios públicos consumido por una aplicación móvil y una página web.*

*\*RNF: Requisito no funcional*

*\*STK: Requisito de grupo de interés*

# 5 Factibilidad

## 5.1 Factibilidad Técnica

## 5.2 Factibilidad Operativa

## 5.3 Factibilidad Económica

## 5.4 Conclusión de la factibilidad

# 6 Análisis

## 6.1 Casos de Uso

### 6.1.1 Actores

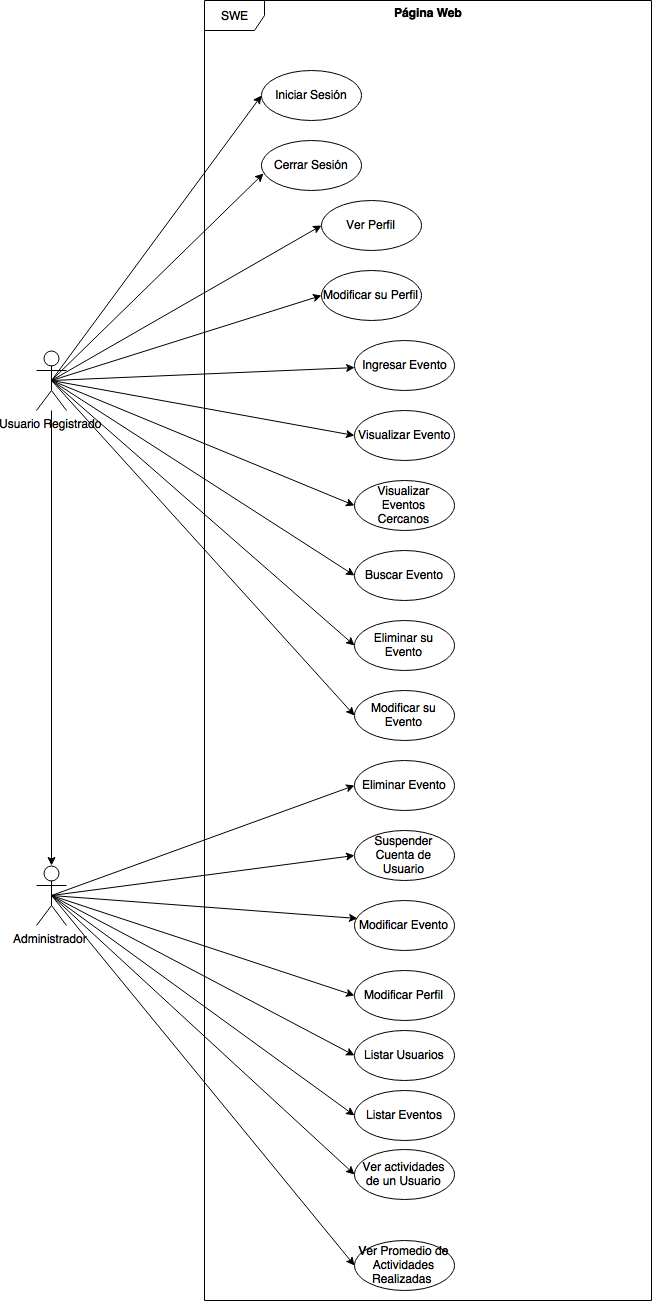
* **Usuario invitado:** Rol desempeñado por una persona que visita el sistema y tiene un acceso restringido. Requiere de conocimientos técnicos básicos de uso de páginas web y aplicaciones móviles.
* **Usuario registrado:** Rol desempeñado por una persona que se registra en el sistema para acceder a todas las funciones disponibles. Requiere de conocimientos técnicos básicos de uso de páginas web y aplicaciones móviles.
* **Administrador:** Rol desempeñado por una persona que controla y administra la información del sistema. Requiere altos conocimientos técnicos para modificar la información que estará disponible en el sistema.

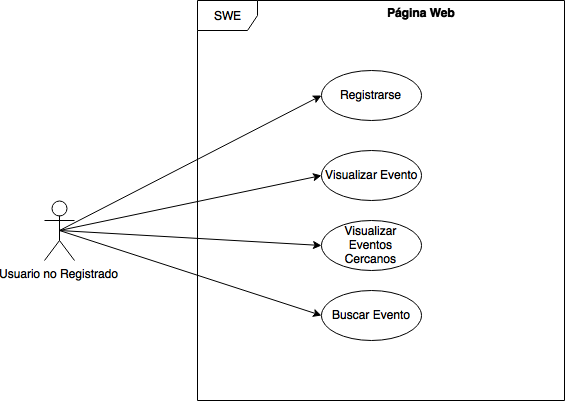
### 

### 

### 6.1.2 Diagramas de casos de uso y descripción

#### 6.1.2.1 Diagramas de casos de uso para la página web

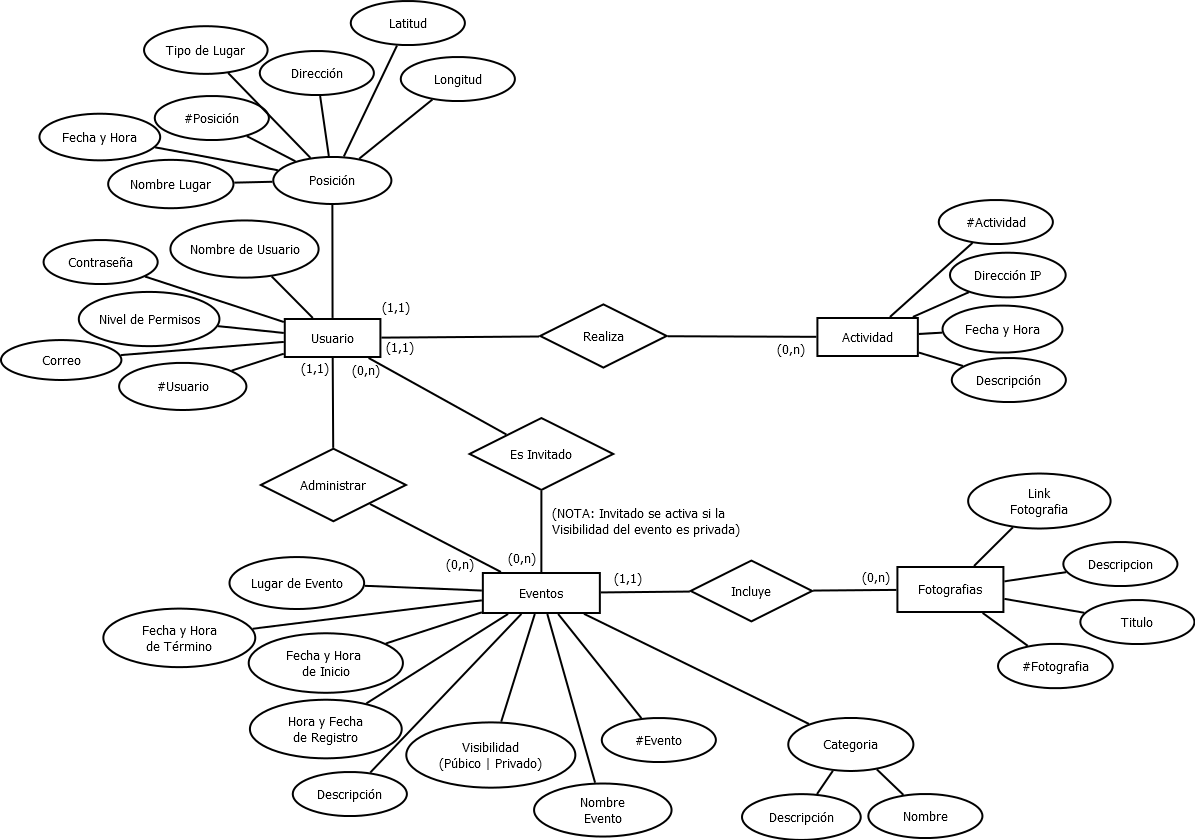




### 6.1.3 Especificación de los casos de uso de la página web

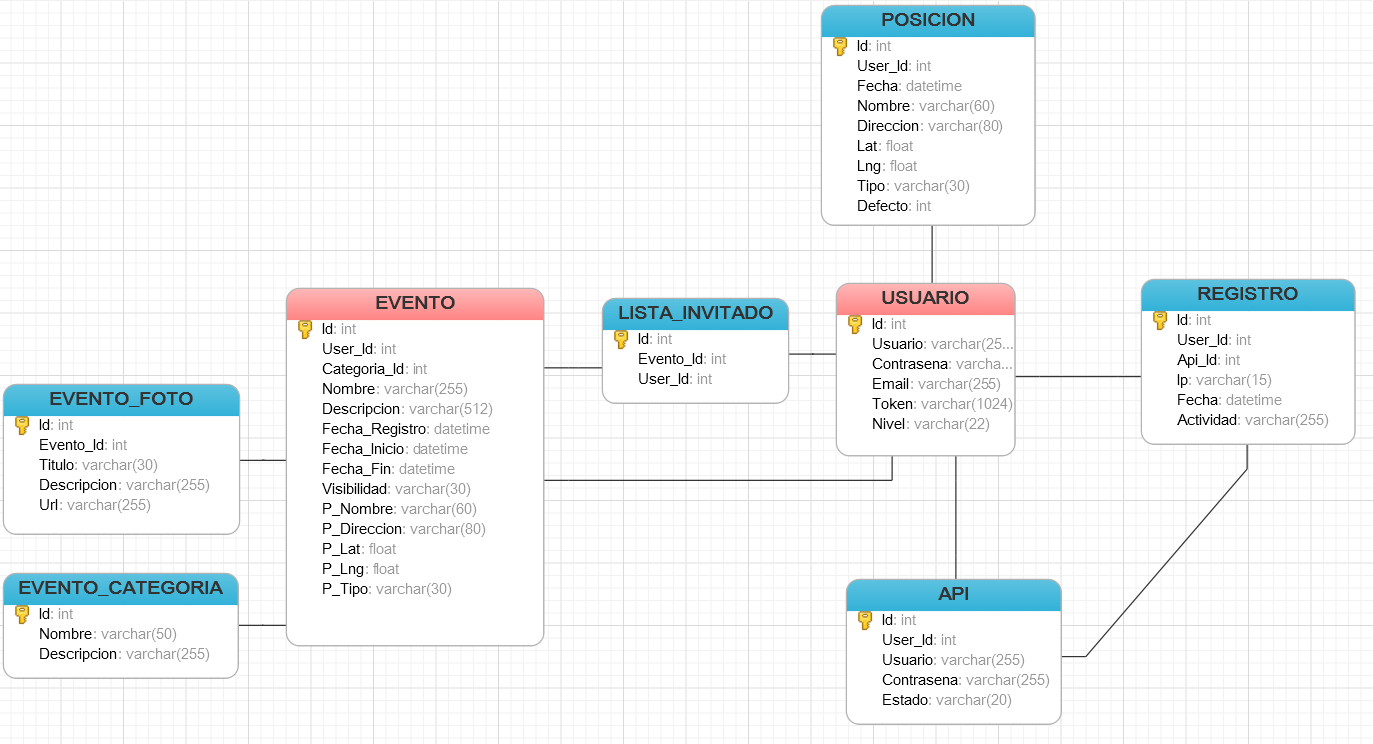
#### 6.1.3.1 Caso de uso: Registrarse

## 6.2 Modelamiento de datos



# 7 Diseño

## 7.1 Diseño Físico de la Base de datos



## 7.2 Diseño de arquitectura funcional

## 7.3 Diseño interfaz y navegación

## 7.4 Especificación de módulos

# 8 Pruebas

## 8.1 Elementos de prueba

## 8.2 Especificación de las pruebas

## 8.3 Responsable de las pruebas

## 8.4 Calendario de pruebas

## 8.5 Detalle de las pruebas

## 8.6 Conclusiones de las pruebas

# 9 Plan de capacitación y entrenamiento

# 10 Plan de implantación y puesta en marcha

# Bibliografía