

UNIVERSIDAD DEL BÍO-BÍO
FACULTAD DE CIENCIAS EMPRESARIALES
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN Y
TECNOLOGÍA DE LA INFORMACIÓN



Alarma Vecinal

Norma Contreras

UNIVERSIDAD DEL BÍO-BÍO
FACULTAD DE CIENCIAS EMPRESARIALES
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
Y TECNOLOGÍA DE LA INFORMACIÓN

Alarma Vecinal

Norma Contreras

Profesor Guía : Sylvia Marcela Pinto Fernández

Profesor Informante : Alfonso Rodríguez Ríos

Memoria para optar el Título de Ingeniero Civil en Informática

Chillán, 2016

Agradecimientos

Dedico esta sección para darle el más sentido agradecimiento a mi familia. Por haber estado presente siempre, dedicándome todo su apoyo y su confianza de manera incondicional. Por el tenaz acompañamiento que siempre me han brindado solo por quererme tanto.

Especialmente quiero agradecer a mis padres Norma Baeza Salazar y Mario Contreras Caro, por el constante sacrificio realizado por mi bienestar y cuidado.

A mi hermana Katherinne Contreras Baeza y a mi primo Leonardo Soto Baeza, mis regalones, por los buenos momentos vividos y porque siempre han estado ahí para mí.

A mi madrina Uberlinda Baeza Salazar, quien también siempre estuvo apoyándome y preocupándose de mi bienestar, a lo largo de toda mi carrera.

A mi abuelita Fidelina Salazar Contreras por su paciencia, amor y cariño. Y a mi tío Claudio Baeza Salazar por las historias y enseñanzas que me entrego.

Y a mis abuelos paternos José Contreras Henríquez y Elena Caro Baeza, por simpatía y preocupación por mí.

También quiero agradecer a mis compañeros y amigos que siempre me apoyaron y me ayudaron cuando los necesite, a lo largo de toda mi carrera.

A mi prima Fidelina Ferrada quien animó y apoyo a lo largo de toda mi carrera.

A Anita Casanova, Víctor Inostroza, Felipe Cantillana, Luis Cid, Cristopher Alvear, Cristofer Gajardo y Miguel Castillo, y a las muchas personas que me han apoyado en este camino que comenzó hace ya cinco años y que han estado conmigo tanto en las buenas como en las malas, que de alguna manera siempre estuvieron ahí cuando los he necesitado.

Gracias a Dios por la vida, la salud y porque nunca me faltó nada, por darme la oportunidad de estudiar y hacerme una profesional, gracias por todo.

Gracias totales...

***“Dedico esta tesis en especial a
mi mami Fide y mi tío Claudio, quienes fueron
un gran apoyo emocional todo el tiempo
y que partieron de este mundo
hace poco, pero que estoy segura que
estarán orgullosos.”***

Norma. -

Resumen

El presente documento tiene como fin detallar el desarrollo de un proyecto para telefonía móvil denominado “S.O.S”, el cual consiste en una aplicación para dispositivos móviles, que busca facilitar el acceso rápido a los centros de emergencias, siendo una herramienta útil en situaciones de emergencia, en donde se necesite contactar a la brevedad a alguien que pueda acudir en su ayuda (juntas de vecinos, familiares, amigos, plan cuadrante, consultorio o bomberos).

La aplicación móvil busca ser una solución al alcance de la mano para cualquier persona que disponga de un Smartphone, tanto para Android como para IOs y que tenga acceso a internet, sacando el máximo de rendimiento al dispositivo utilizando las capacidades de éste. Esta aplicación es la encargada de: enviar alertas a todos los contactos pertenecientes a las juntas de vecinos a través de una notificación, ver a los vecinos de los grupos vecinales en los que se encuentra registrado el usuario y crear un grupo familiar a quienes también se les contactará.

Lo novedoso de este proyecto es la mezcla de las tecnologías emergentes al servicio de las personas, ya que los dispositivos móviles permiten la utilización de GPS, mapas, acceso a internet, acceso a un almacenamientos locales y bases de datos remotas vía web services.

Este proyecto, que combina varias tecnologías emergentes, pretende ser una herramienta que permita satisfacer la necesidad de seguridad, brindando un medio de difusión de información y ahorro de tiempo en casos de necesidad.

Índice

Agradecimientos	1
Resumen	3
Índice de tablas.....	8
Índice de Figuras.....	10
1. Introducción.....	11
2. Descripción de la problemática.....	12
3. Descripción de la solución.....	13
3.1. Diagramas explicativos del funcionamiento	14
3.1.1. Diagrama Registro en Firebase Cloud Messaging (FCM)	14
3.1.2. Diagrama Seguimiento de Dispositivo Móvil.....	15
3.2. Diagrama de Menú.....	16
4. Definición del proyecto.....	17
4.1. Objetivos del proyecto	17
4.1.1. Objetivo general.....	17
4.1.2. Objetivos específicos.....	17
4.1.3. Seguridad.....	18
4.2. Ambiente de Ingeniería de Software.....	18
4.2.1. Arquitectura de software	19
4.2.2. Metodología	20
4.3. Definiciones siglas y abreviaciones	22
5. Especificación de requisitos del software.....	24
5.1. Alcances.....	24
5.2. Objetivos del software.....	24
5.3. Descripción global del producto	25
5.3.1. Interfaz de usuario.....	25
5.3.2. Tecnologías	29
5.4. Requisitos Específicos	31
5.4.1. Requisitos funcionales.....	32
5.4.2. Requisitos no funcionales del sistema.....	34
5.4.3. Restricciones	35
5.4.4. Interfaces externas de entrada	36
5.4.5. Interfaces externas de salida	36
6. Factibilidad	37

Alarma Vecinal

6.1.	Factibilidad técnica.....	37
6.2.	Factibilidad operativa.....	38
6.2.1.	Obtención de información de los usuarios finales.....	39
6.2.1.1.	Entrevista.....	39
6.2.1.2.	Resultados de la encuesta	40
6.2.1.3.	Conclusión de entrevista	41
6.3.	Factibilidad económica.....	43
6.3.1.	Recursos necesarios.....	43
6.3.2.	Determinación de costos.....	43
6.3.2.1.	Costos de desarrollo.....	43
6.3.2.2.	Costos de implementación.....	44
6.3.2.3.	Costos de operación.....	44
6.3.2.4.	Resumen de costos.....	45
6.3.3.	Determinación de ingresos y beneficios	46
6.3.4.	Flujo de caja	47
6.4.	Calculo de VAN	48
6.4.1.	Caso tradicional.....	48
6.4.2.	Caso real.....	49
6.5.	Conclusión de factibilidad.....	49
CAPITULO 7 “INCREMENTO I”		50
7.	Incremento I.....	51
7.1.	Análisis Incremento I.....	51
7.1.1.	Casos de uso de Incremento I.....	51
7.1.1.1.	Actores	52
7.1.1.2.	Diagrama de casos de uso de Incremento I.....	53
7.1.1.3.	Especificación de casos de uso de Incremento I.....	54
7.1.1.3.1.	Registrarse como institución de seguridad	54
7.1.1.3.2.	Registrarse como vecino.....	55
7.1.1.3.3.	Registrar junta de vecinos.....	56
7.1.1.3.4.	Listar Vecinos.....	57
7.1.1.3.5.	Añadirse a junta vecinal.....	58
7.1.2.	Diseño.....	59
7.1.2.1.	Modelamiento de datos del incremento I.....	59
7.1.2.1.1.	Modelo Entidad-Relación	59

7.1.2.1.2.	Modelo físico de la base de datos.....	60
7.1.2.1.3.	Descripción de cada tabla.....	61
7.1.2.1.3.1.	Descripción tabla: grupo.....	61
7.1.2.1.3.2.	Descripción tabla: vecino_has_grupo.....	61
7.1.2.1.3.3.	Descripción tabla: vecino.....	62
7.1.2.1.3.4.	Descripción tabla: Emergencia.....	62
7.1.2.1.3.5.	Descripción de la tabla: emergencia_has_grupo	63
7.2.	Pruebas Incremento I	64
7.2.1.	Aspectos Generales.....	64
7.2.2.	Especificación de pruebas del Incremento I.....	65
7.2.3.	Casos de pruebas de caja negra	66
7.2.3.1.	Caso de prueba: Registrarse como usuario.....	66
7.2.3.2.	Caso de prueba: Registrarse como plan cuadrante	67
7.2.3.3.	Caso de prueba: Registrarse como consultorio.....	68
7.2.3.4.	Caso de prueba: Registrarse como Bombero	69
7.2.3.5.	Caso de prueba: Registrar Junta de vecinos.....	70
7.2.3.6.	Caso de prueba: Añadirse a junta de vecinos	71
7.2.3.7.	Caso de prueba: Ver juntas de vecinos.....	72
7.2.4.	Conclusiones de las pruebas	73
CAPITULO 8 “INCREMENTO II”		74
8.	Incremento II	75
8.1.1.	Casos de Uso	75
8.1.1.1.	Actores	75
8.1.1.2.	Diagrama de Casos de Uso incremento II.....	76
8.1.1.3.	Diagrama de casos de uso portal web.....	77
8.1.1.4.	Especificación de casos de uso.....	78
8.1.1.4.1.	Contactar centro médico	78
8.1.1.4.2.	Contactar plan cuadrante	79
8.1.1.4.3.	Contactar bomberos	80
8.1.1.4.4.	Contactar grupo vecinal:.....	81
8.1.1.4.5.	Recibir alarma.....	82
8.1.1.5.	Especificación de casos de uso del portal web	83
8.1.1.5.1.	Modificar vecino.....	83
8.1.1.5.2.	Eliminar vecino.....	84

Alarma Vecinal

8.1.1.5.3.	Ver vecino	85
8.1.1.5.4.	Modificar institución de seguridad.....	86
8.1.1.5.5.	Eliminar institución de seguridad	87
8.1.1.5.6.	Ver institución de seguridad	88
8.1.1.5.7.	Registrar grupo.....	89
8.1.1.5.8.	Modificar grupo	90
8.1.1.5.9.	Eliminar grupo	91
8.1.2.	Pruebas	92
8.1.2.1.	Aspectos Generales.....	92
8.1.3.	Especificación de pruebas.....	93
8.1.4.	Casos de pruebas de caja negra	94
8.1.4.1.	Caso de prueba: Contactar plan cuadrante	94
8.1.4.2.	Caso de prueba: Contactar a consultorio	95
8.1.4.3.	Caso de prueba: Contactar a bomberos	96
8.1.4.4.	Caso de prueba: Contactar a juntas de vecinos	97
8.1.4.5.	Caso de prueba: Ubicación geográfica.....	98
8.1.5.	Conclusiones de las pruebas	99
9.	Conclusiones Generales.....	100
10.	Trabajos futuros	101
11.	Bibliografía.....	102
	ANEXOS.....	104
	Anexo A: Capturas reales del sistema.....	105
	Anexo B: Encuesta de Interfaz.....	110

Índice de tablas

Tabla 1 Requisitos Funcionales de la aplicación móvil.....	32
Tabla 2 Requisitos funcionales de portal web.....	33
Tabla 3 Requisitos no funcionales.....	34
Tabla 4 Restricciones.....	35
Tabla 5 Interfaces externas de entrada.....	36
Tabla 6 Interfaces externas de salida.....	36
Tabla 7 Resumen de resultados de entrevista.....	40
Tabla 8 Resumen de costos.....	45
Tabla 9 Flujo de caja.....	47
Tabla 10 Actor "Usuario".....	52
Tabla 11 Actor "Institución de Seguridad".....	52
Tabla 12 Especificación del caso de uso "Registrarse como institución de seguridad".....	54
Tabla 13 Especificación del caso de uso "Registrar Junta de Vecinos".....	56
Tabla 14 Especificación del caso de uso "Listar Vecinos".....	57
Tabla 15 Especificación de caso de uso "Añadirse a junta de vecinos".....	58
Tabla 16 Descripción de la tabla grupo.....	61
Tabla 17 Descripción de la tabla vecino_has_grupo.....	61
Tabla 18 descripción de la tabla vecino.....	62
Tabla 19 Descripción de la tabla Emergencia.....	62
Tabla 20 Descripción de la tabla emergencia_has_grupo.....	63
Tabla 21 Especificación de prueba "Registrarse como vecino".....	66
Tabla 22 Especificación de prueba "Registrarse como plan cuadrante".....	67
Tabla 23 Especificación de prueba "Registrarse como consultorio".....	68
Tabla 24 Especificación de prueba "Registrarse como bombero".....	69
Tabla 25 Especificación de la prueba "Registrar junta de vecinos".....	70
Tabla 26 Especificación de prueba "Añadirse a junta de vecinos".....	71
Tabla 27 Especificación de caso de uso Ver junta de vecinos.....	72
Tabla 28 Actor: "Administrador".....	75
Tabla 29 Diagrama de caso de uso incremento II del portal web.....	77
Tabla 30 Especificación de casos de uso "Contactar centro médico".....	78
Tabla 31 Especificación de casos de uso "Contactar plan cuadrante".....	79

Alarma Vecinal

Tabla 32 Especificación de casos de uso "Contactar Bomberos"	80
Tabla 33 Especificación de casos de uso "Contactar grupo vecinal"	81
Tabla 34 Especificación de casos de uso "Recibir alarma"	82
Tabla 35 Especificación del caso de uso "Modificar vecino"	83
Tabla 36 Especificación del caso de uso "Eliminar vecino"	84
Tabla 37 Especificación del caso de uso "Ver vecino"	85
Tabla 38 Especificación del caso de uso "Modificar institución de seguridad"	86
Tabla 39 Especificación del caso de uso " Eliminar institución de seguridad"	87
Tabla 40 Especificación del caso de uso "Ver institución de seguridad"	88
Tabla 41 Especificación del caso de uso "Registrar grupo"	89
Tabla 42 Especificación del caso de uso "Modificar grupo"	90
Tabla 43 Especificación del caso de uso "Eliminar grupo"	91
Tabla 44 Especificación de pruebas.....	93
Tabla 45 Especificación de la prueba "Contactar plan cuadrante"	94
Tabla 46Especificación de prueba "Contactar a consultorio"	95
Tabla 47 Especificación de prueba "Contactar a bomberos"	96
Tabla 48 Especificación de prueba "Contactar a juntas de vecinos"	97
Tabla 49 Especificación de la prueba "Ubicación Geográfica"	98

Índice de Figuras

Ilustración 1 Diagrama de registro de seguimiento de dispositivo móvil	15
Ilustración 2 Diagrama del menú de la aplicación móvil	16
Ilustración 3 Modelo arquitectónico de la aplicación.....	19
Ilustración 4 Modelo de desarrollo	21
Ilustración 5 Interfaz de usuario de emitir alarma.....	25
Ilustración 6 Interfaz de usuario Menú	26
Ilustración 7 Interfaz de usuario para registrarse en la aplicación	26
Ilustración 8 Interfaz de Usuario para crear junta de vecinos.....	27
Ilustración 9 interfaz de usuario para ver junta de vecinos.....	27
Ilustración 10 Interfaz de usuario notificación	28
Ilustración 11 Aceptación de iconos para emitir alarmas	42
Ilustración 12 Formula de la VAN.....	48
Ilustración 13 Diagrama de casos de uso incremento I	53
Ilustración 14 Especificación del caso de uso "Registrarse como vecino"	55
Ilustración 15 Modelo entidad relación incremento I.....	59
Ilustración 16 Modelo físico del incremento I	60
Ilustración 17 Especificación de pruebas incremento I.....	65
Ilustración 18 Diagrama de casos de uso del incremento II	76

1. Introducción

Por más de 40 años, la comunidad empresarial ha buscado la automatización de sus procesos de negocio, tanto por la eficiencia como por las ventajas competitivas que consigue el despliegue de un sistema informático. Sin ir más lejos, según estadísticas del INE ¹ en el 2010, demuestran que en Chile la cantidad de teléfonos móviles es mayor a la población total, por cada habitante se encuentra 1.13 teléfonos, esto se debe a que actualmente estos dispositivos son utilizados como minicomputadores, facilitando la portabilidad y el acceso a la información en todo momento.

El siguiente documento consta de once capítulos incluido el actual. El Capítulo 2 consiste en la definición de la problemática y motivación que originó el desarrollo del proyecto. En el Capítulo 3 se describe una posible solución correspondiente al problema abordado en el Capítulo 2. En el Capítulo 4 se definen aspectos relevantes a la definición del proyecto en donde se especifican los objetivos, la arquitectura del software, la metodología y la definición de siglas y abreviaciones. El Capítulo 5 corresponde al análisis del proyecto, en donde se describen los siguientes aspectos: el alcance, objetivos del software, descripción global del producto, las tecnologías utilizadas y la especificación de los requisitos del proyecto. En el Capítulo 6 se realiza un estudio de factibilidad desde los ámbitos técnico, operacional y económico. En el Capítulo 7 se trata el incremento en el cual se realiza un análisis del proyecto mediante casos de uso, el diseño mediante el modelamiento de datos y las pruebas correspondiente al primer incremento. El Capítulo 8 comprende lo relacionado al incremento II, el cual se realiza un análisis mediante nuevos casos de usos y la especificación de las pruebas correspondientes al segundo incremento. En el Capítulo 9 se plantean las conclusiones generales, mientras que el Capítulo 10 se describen trabajos futuros del proyecto.

Finalmente, en el Capítulo 11 se encuentra la bibliografía. Todo esto con el fin de describir la implementación de una aplicación móvil con la cual se pretende facilitar la comunicación con los servicios de emergencia de la ciudad de Chillán, como plan cuadrante, consultorio, bomberos y un botón de pánico que permitirá la comunicación con un grupo vecinal, por medio de una alarma de emergencia que será transmitida de un dispositivo móvil a otro.

¹ INE Instituto Nacional de Estadísticas.

2. Descripción de la problemática

Una ventaja muy importante que puede proporcionarle una aplicación móvil a los servicios de emergencia de la ciudad de Chillán, es la agilidad en la comunicación entre el usuario que presenta la urgencia con el servicio correspondiente a la emergencia.

Sin ir más lejos, los resultados de ENUSC ² 2015 demuestran que el 30,9% de la población ha sido víctima de un delito en más de una oportunidad, lo cual señala la importancia de la conexión entre población y las instituciones que resguardan la seguridad. Además, de acuerdo con las estadísticas del INE y el censo 2002, indican que el 11,3% de la población corresponde a adultos mayores quienes requieren de un sistema fácil de utilizar que les permitan una comunicación más rápida en caso de una emergencia. Para lo cual, en Chillán, se dispone de 7 cuadrantes los cuales pueden ser accedidos por cualquier ciudadano que lo requiera. Sin embargo, este sistema es efectivo si se contacta el cuadrante adecuado, es decir, el que corresponde al sector en que se encuentra la emergencia. Por otro lado, según estadísticas del INE declara que el 33% de la población es diagnosticada con una enfermedad crónica y el 22% de la población posee alguna discapacidad, lo cual indica la importancia de crear un vínculo de comunicación entre las personas que padecen las enfermedades con los centros médicos en donde son atendidos.

Algunas comunidades o vecindarios optan por utilizar redes sociales como WhatsApp realizando grupos para suplir la deficiencia de comunicación y asegurarse entre ellos frente a una emergencia. Existen números de emergencia para recurrir a estos servicios de manera expedita, sin embargo, son desaprovechadas al momento de entrar en pánico, interfiriendo en el vínculo entre el servicio y el ciudadano que lo necesita.

Entre los problemas se destacan los siguientes:

- Desinformación sobre los números de emergencia.
- Desinformación sobre donde ocurre la emergencia.

Para los problemas antes mencionados existen posibles soluciones tales como, la utilización de redes sociales, anotar números de emergencia en papel, etc. Es por ese motivo que se optó por la solución de alarma vecinal, con el fin de automatizar el proceso incorporando las tecnologías ya existentes.

² Encuesta Nacional Urbana de Seguridad Ciudadana

3. Descripción de la solución

Con respecto a la solución planteada, ésta consiste en la implementación de un sistema llamado “S.O.S” en el cual se permite al usuario emitir una notificación de alerta, tanto a centro de emergencia como a grupos de contactos, indicando por medio de la geolocalización su ubicación actual al usuario que reciba la notificación.

Este sistema está compuesto por dos módulos, el primero consiste en una aplicación móvil multiplataforma, en la cual el usuario podrá registrarse en la aplicación, ya sea como un vecino o como un centro de emergencia, registrando el dispositivo móvil designado por cada institución, almacenando así el id único que posee cada Smartphone, el cual será utilizado posteriormente para emitir la alarma. Además, el usuario podrá registrarse en más de un grupo para así poder informar una situación de emergencia tanto a sus vecinos como a sus familiares, por lo cual el sistema también permite al usuario crear nuevos grupos. El segundo módulo consiste en un portal web, el cual será utilizado por un administrador el cual tiene como objetivo permitir a este administrador gestionar la base de datos.

Finalmente, y lo más importante, es que el sistema permite que el usuario pueda enviar una notificación de auxilio tanto a centros de emergencia (plan cuadrante, bomberos y consultorio) como a los contactos que se encuentren registrados en los grupos vecinales en los que el usuario se registró, sin costo utilizando la conexión a internet.

3.1. Diagramas explicativos del funcionamiento

En ésta sección se explica cómo se llevan a cabo los procesos internos de la Aplicación Móvil para enviar una alerta, con la finalidad de otorgar una visión general del funcionamiento del Sistema.

3.1.1. Diagrama Registro en Firebase Cloud Messaging (FCM)

En la Figura 1 Diagrama de registro en FCM se analiza el proceso de registro del dispositivo móvil en el servicio de FCM, el cual consiste en una solución multiplataforma que permite enviar mensaje y notificaciones de forma gratuita, es necesario para poder enviar información mediante notificaciones PUSH al emitir una alarma posteriormente (requerido para realizar una petición desde el servidor al cliente, otorgando un importante ahorro de recursos) hacia el móvil. Cabe destacar que este proceso se realiza solo una vez, cuando se inicia por primera vez la aplicación móvil.

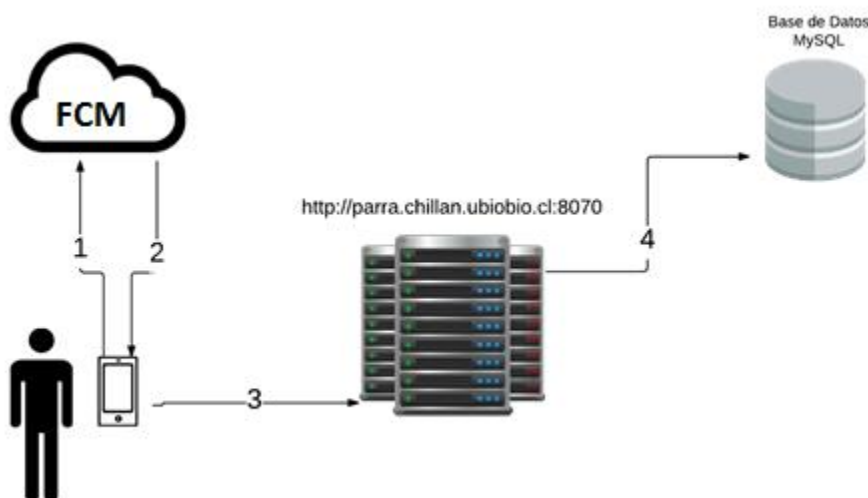


Figura 1 Diagrama de registro en FCM

Los pasos para llevar a cabo este proceso son:

- El móvil realiza una petición de registro en el servicio FCM.
- El servicio FCM asigna y entrega un identificador único al móvil.
- El móvil envía al servidor su identificador.
- El servidor se encarga de guardar el identificador de su móvil.

3.1.2. Diagrama Seguimiento de Dispositivo Móvil

En la **Ilustración 1** Diagrama de registro de seguimiento de dispositivo móvil se analiza el proceso de seguimiento de un dispositivo, en donde se muestra cómo el servidor se comunica con cada contacto del usuario que se encuentra en alerta, es decir que el este proceso se lleva a

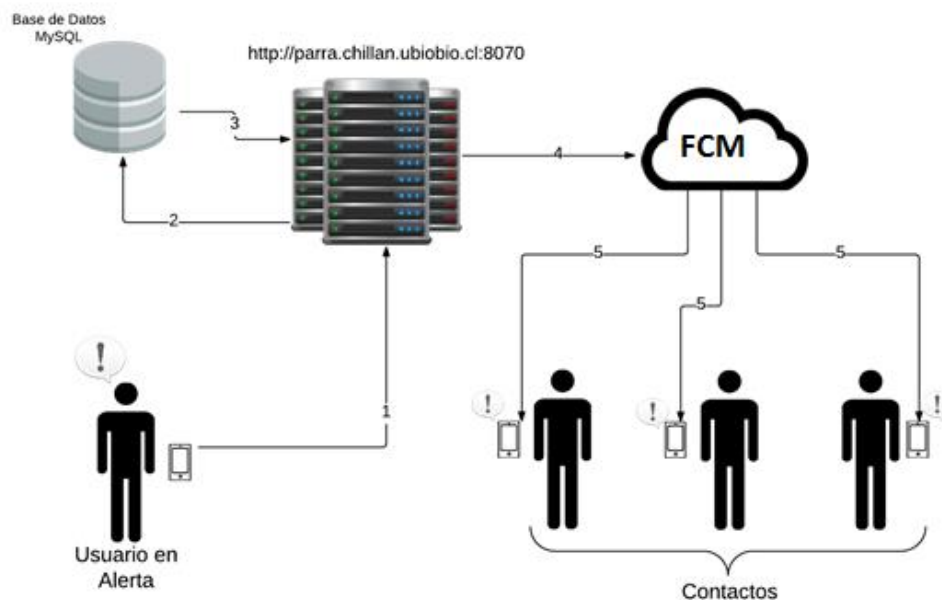


Ilustración 1 Diagrama de registro de seguimiento de dispositivo móvil

cabo cuando el usuario emite una alarma.

La serie de pasos para llevar a cabo este proceso son los siguientes:

- El usuario en alerta envía una notificación al servidor.
- El servidor consulta en la base de datos por los contactos del usuario en alerta.
- La base de datos retorna los contactos del usuario al servidor.
- El servidor obtiene el identificador FCM de cada contacto y los envía junto a un grupo de datos de éste.
- El servicio FCM se encarga de enviar una notificación a cada contacto.

3.2. Diagrama de Menú

En la siguiente sección, se muestra un diagrama que tiene como finalidad exponer las diversas funcionalidades que posee la aplicación móvil.

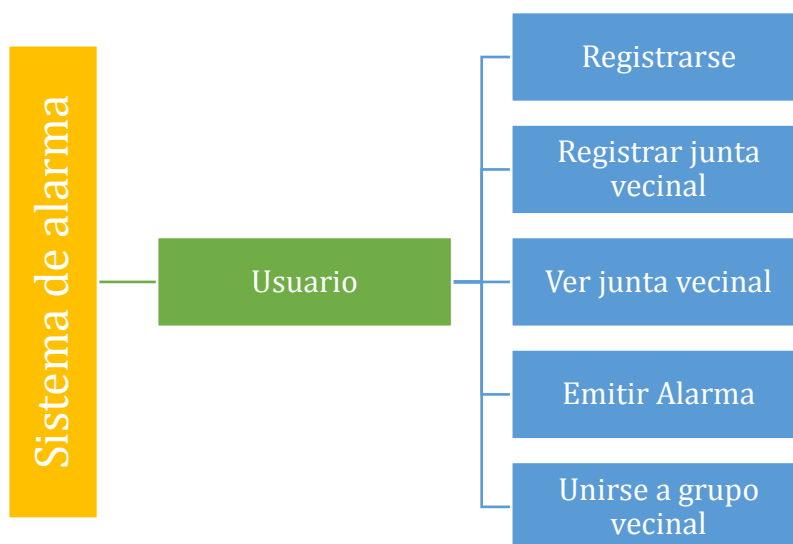


Ilustración 2 Diagrama del menú de la aplicación móvil

4. Definición del proyecto

El proyecto tiene como objetivo fundamental aumentar la eficiencia y eficacia al momento de contactar, tanto a un grupo de vecinos como a las instituciones que resguardan la seguridad (plan cuadrante, consultorio y bomberos) en Chillán. Por lo cual, se pretende desarrollar una aplicación móvil para automatizar este procedimiento en situaciones de emergencia y facilitar así un poco la comunicación.

4.1. Objetivos del proyecto

Con el fin de realizar una construcción de la aplicación correctamente, es necesario definir aspectos relevantes como lo son los requisitos, modelos de diseño, etc. Sin embargo, no menos importante es definir adecuadamente los objetivos del proyecto, para así alinear bien lo que se quiere conseguir del proyecto con los requisitos y actividades.

4.1.1. Objetivo general

- Desarrollar una aplicación de comunicación vecinal, mediante una aplicación móvil en donde todos los implicados (carabineros, bomberos, consultorios y vecinos) puedan acceder a esta, con el fin de agilizar y facilitar la seguridad de la comunidad vecinal de manera expedita y respuesta oportuna.

4.1.2. Objetivos específicos

- Crear una aplicación que pueda ser visualizada en cualquier dispositivo móvil, para emitir alarmas de emergencias.
- Permitir a los usuarios que puedan registrarse, registrar un nuevo grupo y añadirse a uno o más grupos, tanto de vecinos como familiares y amigos.
- Generar una base de datos en donde se almacenen los FCM de los dispositivos móviles y la información de cada grupo y usuario registrado.

4.1.3. Seguridad

El tema de la seguridad forma parte importante durante el desarrollo de un sistema, debido a que cada vez hay más personas con conocimientos, sin embargo, no todos son expertos. En consideración a lo anterior, el sistema de alarma vecinal, es un sistema que será utilizado por todo tipo de personas, lo que podría ocasionar un mal uso de este.

Entre los problemas encontrados destacan los siguientes con una descripción de la posible solución:

- Enviar una notificación por accidente, para la cual se implementó un botón extra que notifique un error luego de ser enviada la notificación de auxilio.
- Verificar que el dispositivo móvil registrado como un servicio de seguridad, efectivamente corresponde a la institución, la posible solución es investigar el identificador único correspondiente a todos los móviles utilizados por estas instituciones de emergencia para verificarlo al momento que éstos se registren en las juntas de vecino, sin embargo, por limitaciones de tiempo esto será considerado como un trabajo futuro.

4.2. Ambiente de Ingeniería de Software

Con el fin de establecer una estrategia de desarrollo que permita abordar el problema eficazmente, se seleccionó un modelo de desarrollo, que permite asegurar la calidad del producto final.

El ambiente real en el que el proyecto se lleva a cabo es en la Universidad del Bío-Bío, sede Fernando May en la ciudad de Chillán. Todas las reuniones con la profesora guía se realizan en la oficina de dicha profesora dentro la universidad, aunque la mayor parte de las actividades o tareas son realizadas de manera independiente por el estudiante.

4.2.1. Arquitectura de software

Para el desarrollo de esta aplicación se utilizará el patrón de diseño **MVC**³, el cual separa de manera clara y precisa los componentes de una aplicación en el modelo (Objetos del negocio), la vista (interfaz de usuario) y el controlador (Controlador de flujos de datos). Su objetivo principal es separar la lógica del negocio de la lógica de la presentación para darle estructura a la implementación y facilitar con esto su posterior mantenimiento.

Entre los principales beneficios de utilizar éste patrón tenemos:

- Claridad en el diseño.
- Facilita el mantenimiento.
- Clara definición y especialización de tareas de cada elemento.
- Mayor flexibilidad, principalmente en las vistas.

Además, el framework ionic (herramienta para el desarrollo de aplicaciones híbridas) que se utilizará para el desarrollo de esta aplicación utiliza **MVVM**⁴, la cual indica que la interacción entre la vista y el controlador es bidireccional, es decir, que el controlador muestra los datos en la vista y si en la vista se efectúan cambios de datos, estos serán actualizados en el modelo automáticamente.

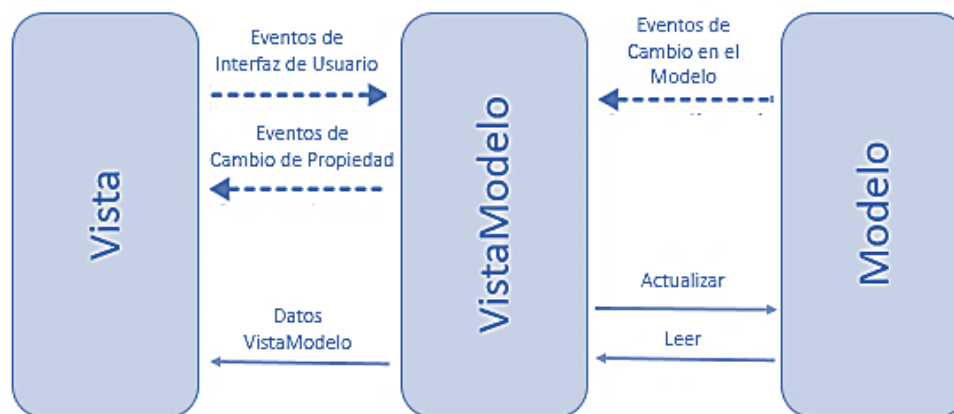


Ilustración 3 Modelo arquitectónico de la aplicación

³ Modelo Vista Controlador

⁴ Modelo-Vista-Vista-Modelo

4.2.2. Metodología

Para este proyecto se decidió seguir la metodología iterativa incremental, dado que se adecua mejor a la necesidad que representa desarrollar este proyecto. Tales como, el corto periodo de tiempo que se dispone para el desarrollo de esta aplicación, además al hacer uso de esta metodología permite realizar mejoras y análisis en todas las iteraciones, haciendo de esta aplicación más adecuada para los requerimientos.

Entre las principales ventajas que posee esta metodología destacan las siguientes:

- Mitigación tan pronto como sea posible de riesgos altos (técnicos, requisitos, objetivos, usabilidad y demás).
- Progreso visible en las primeras etapas.
- Temprana retroalimentación, compromiso de los usuarios y adaptación, que produce un sistema refinado que se ajusta a las necesidades reales del personal involucrado.
- Gestión de la complejidad; el equipo no se ve abrumado por la “parálisis del análisis” o pasos muy largos y complejos.
- El conocimiento adquirido en una iteración se puede utilizar metódicamente para mejorar el propio proceso de desarrollo, iteración a iteración.

Para la realización de las iteraciones, se construye el siguiente modelo en respuesta a las necesidades del proyecto.

En primer lugar, debido a que se incursiona en nuevas tecnologías, se comienza con una etapa de investigación la cual se lleva a cabo a lo largo de todo el proyecto, luego se clarifican los requisitos para desarrollar el proyecto de forma correcta y obtener un producto final de calidad.

En segundo lugar, respecto a los requisitos del usuario se incorporaron características de modelos ágiles ⁵ tales como la planificación y diseño, por medio de casos de uso, diagramas de flujo, modelos de datos y pruebas, postergando la documentación importante de usuario para la fase final.

⁵ Modelos de desarrollo de software que pretende disminuir la demora en un proyecto en cuanto al desarrollo.

En la Ilustración 4 Modelo de desarrollo se observan las iteraciones un poco más detalladas, las cuales consisten básicamente en las fases de diseño, implementación y pruebas. Finalmente, al acabar las dos iteraciones se completa el proyecto con la entrega de la documentación y el despliegue del software.

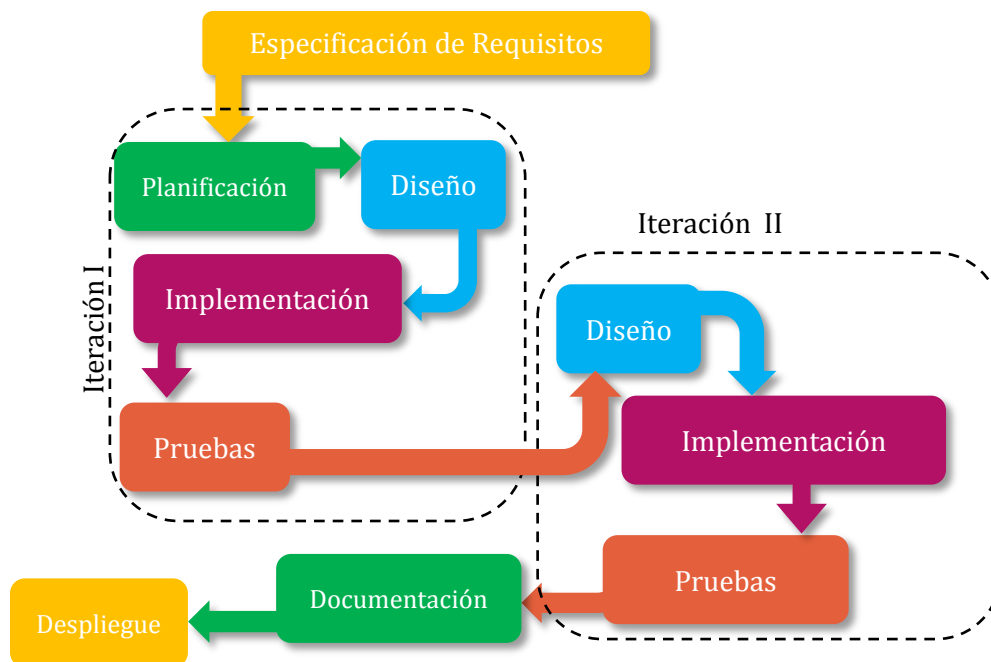


Ilustración 4 Modelo de desarrollo

Con respecto al modelo de desarrollo, se llevan a cabo dos iteraciones, las cuales dan como resultado un incremento del sistema, es decir, parte de la aplicación funcional. La primera iteración genera los módulos de almacenamiento de los datos tanto en la local storage como en la base de datos MySQL. Al final la segunda iteración se realiza la geolocalización y el envío de notificaciones push.

Por último, para finalizar el proyecto se realizan las correcciones necesarias y se cierra la documentación.

4.3. Definiciones siglas y abreviaciones

1. Actor: entidad externa al sistema, personas u otro sistema, que interacciona con este directamente. En este proyecto representarán solamente a los usuarios del sistema.
2. Angular: es un framework MVC de código abierto desarrollado por Google y escrito en Javascript, que trabaja del lado del cliente (client-side) y nos permite hacer más dinámica nuestra aplicación web
3. APK: básicamente representa la generación de la aplicación.
4. Cordova: es un framework para el desarrollo de aplicaciones móviles.
5. CRUD: operaciones básicas con datos en un Sistema de Gestión de Base de Datos. Crear, Leer, Actualizar y Borrar (*del inglés Create, Read, Update, Delete*).
6. CSS: hoja de estilos en cascada (*del inglés Cascade Style Sheet*). Lenguaje que permite definir los estilos visuales para HTML.
7. ECU: especificación de casos de uso. Tabla que describe el proceso de uso, en lenguaje natural.
8. HTML: lenguaje de marcado de Hipertexto (*del inglés Hypertext Markup Language*). Lenguaje que permite la creación de interfaces visuales en páginas web.
9. Informática: conjunto de conocimientos técnicos que se ocupan del tratamiento automático de la información por medio de computadoras.
10. Modelo de desarrollo: una guía con una serie de pasos, sugerencias y métodos que permiten un desarrollo sistemático del software, brindando mayor eficiencia y eficacia en las actividades relacionadas.
11. SASS: básicamente consiste en un preprocesador de CSS.

Alarma Vecinal

12. SPA: aplicación de página única es una aplicación web o es un sitio web que cabe en una sola página con el propósito de dar una experiencia más fluida a los usuarios como una aplicación de escritorio.

13. Stakeholder: personas, roles u organizaciones que les concierne el sistema a desarrollar. Los principales stakeholder dentro de un desarrollo de software son el cliente y los usuarios finales.

14. UCA: actor de caso de uso (*del inglés Use Case Actor*).

15. UC o CU: caso de uso (*del inglés Use Case*). Diagrama UML que permite la representación gráfica de las funcionalidades del sistema.

5. Especificación de requisitos del software

5.1. Alcances

En relación a la problemática expuesta en la Sección 2, se diseña e implementa una aplicación de comunicación entre las instituciones que resguardan la seguridad y las comunidades vecinales de la ciudad de Chillán.

Específicamente el desarrollo de una aplicación móvil, tanto para tecnologías Android como Apple iOS incluyendo acceso a la información de contactos (números telefónicos) instituciones de seguridad (números telefónicos de carabineros, bomberos y consultorio), y utilizar información de geolocalización de los usuarios.

Entre los límites que presenta la aplicación, es que ésta no contempla la recepción de alguna respuesta por parte de las instituciones de seguridad ni de parte del grupo vecinal, debido a que como es para situaciones de emergencia, la ayuda debe ser lo más expedita posible sin hacer cuestionamientos. Además, la aplicación debe ser utilizada con internet, para poder capturar la ubicación geográfica del usuario y emitir la alarma.

5.2. Objetivos del software

Desarrollar una herramienta de comunicación con los servicios de emergencia que se encuentran en la ciudad de Chillán y la comunidad que necesita de sus servicios.

El sistema puede gestionar las alarmas de los servicios de emergencia como lo son: plan cuadrante, consultorio, bomberos, grupo vecinal y favorecer los procesos de comunicación en casos de emergencia con la utilización de tecnología y procurando que participen de ellas el máximo de comunidades y servicios de emergencia. Se espera paulatinamente que tales servicios logren mayor autonomía y eficacia en su trabajo acudiendo a las alarmas de manera más expedita.

5.3. Descripción global del producto

5.3.1. Interfaz de usuario

A continuación, se revela lo que cada tipo de usuario puede ver en cada una de sus respectivas interfaces.

Usuario:

- Emitir alarma y pantalla de inicio.



Ilustración 5 Interfaz de usuario de emitir alarma y pantalla de inicio

- Menú

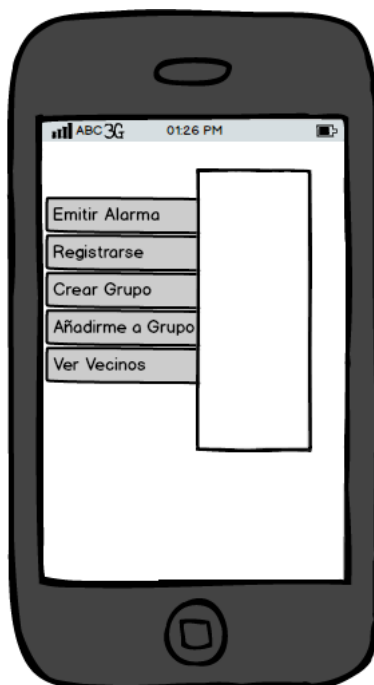


Ilustración 6 Interfaz de usuario Menú

- Registrarse



Ilustración 7 Interfaz de usuario para registrarse en la aplicación

- Crear Junta de Vecinos

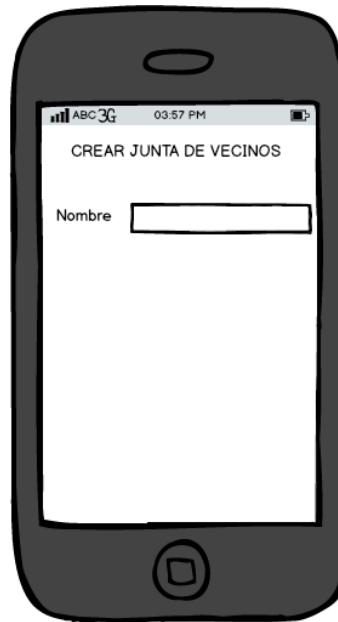


Ilustración 8 Interfaz de Usuario para crear junta de vecinos

- Ver vecinos de Junta de Vecinos

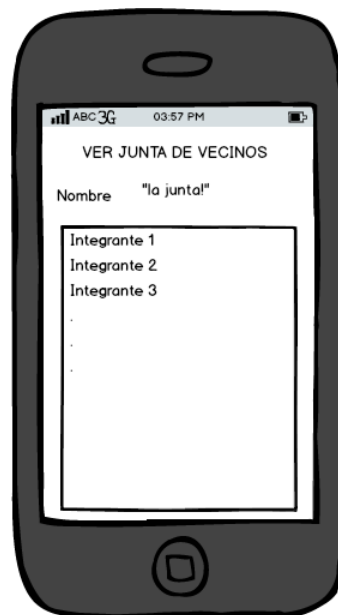


Ilustración 9 interfaz de usuario para ver junta de vecinos

Institución de seguridad:

- Recibir notificación de emergencia.

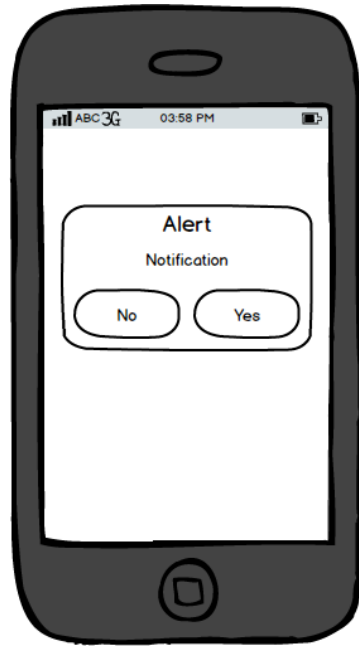


Ilustración 10 Interfaz de usuario notificación

5.3.2. Tecnologías

Los recursos de software necesarios que se utilizan para el desarrollo de esta aplicación móvil son:

- Angular JS: es un framework MVC de JavaScript para el Desarrollo Web Front End que permite crear aplicaciones SPA⁶.
- API Google maps: corresponde a la API lanzada por Google mediante la cual se pueden visualizar los mapas en un servidor web, a través de peticiones AJAX al servidor.
- Cordova: es un framework para el desarrollo de aplicaciones móviles propiedad de Adobe Systems que permite a los programadores desarrollar aplicaciones para dispositivos móviles utilizando herramientas web genéricas como JavaScript, HTML5 y CSS3, resultando aplicaciones híbridas.
- Css: es un lenguaje de hojas de estilo creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML.
- Entorno de desarrollo
 - SublimeText, Brackets: son dos software conocidos para el desarrollo de aplicaciones web.
- Firebase cloud messaging: es una solución multiplataforma que te permite enviar, de forma gratuita y segura, mensajes y notificaciones.
- Framework para aplicación móvil
 - Ionic: ofrece una web de calidad y componentes de aplicaciones nativas para la construcción de aplicaciones web altamente interactivas nativas y progresivas con angular.

⁶ Single Page Application. (Véase 12 de la sección 4.3)

Alarma Vecinal

- Yii2: framework PHP basado en componentes de alta performance para desarrollar aplicaciones Web de gran escala. El mismo permite la máxima reutilización en la programación web y puede acelerar el proceso de desarrollo.
- Google Chrome: es un navegador web desarrollado en el Proyecto Chromium.
- Html5: corresponde a la quinta revisión de HTML, el cual es un lenguaje de etiqueta que permite la elaboración de páginas webs.
- Javascript: es un lenguaje interpretado que se utiliza principalmente para crear páginas o aplicaciones dinámicas. La interpretación de este lenguaje se realiza por parte del cliente en vez del servidor, por lo cual se puede realizar cálculos u operaciones sin sobrecargar el lado del servidor.
- Json: es un formato ligero para el traspaso de datos, utilizado en llamadas AJAX. Es considerada una alternativa al uso de XML.
- Servidor de aplicaciones:
 - Xampp: se utiliza este software porque tiene una fácil instalación, además porque nos permite trabajar con mySql, que es el lenguaje que se utiliza para el desarrollo de la aplicación móvil, debido a que cuenta con un administrador de base de datos llamado PhpMyAdmin
- Sistema de gestión de Base de Datos:
 - MySql: debido a que se puede trabajar con una base de datos en la nube, la cual permite emitir las alarmas con conexión a internet.
 - Local Storage: se trata de un espacio de almacenamiento local, tiene un espacio menos limitado entre 5 MB a 10 MB, además la información almacenada no es enviada por cada petición al servidor y no existe caducidad de la información almacenada.

5.4. Requisitos Específicos

En la siguiente sección se detallan los requerimientos del proyecto clasificándolos en dos grandes grupos:

- **Requerimientos Funcionales:** Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares.
- **Requerimientos No Funcionales:** Son restricciones de los servicios o funciones ofrecidas por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares.

5.4.1. Requisitos funcionales

A continuación, la Tabla 1 Requisitos Funcionales muestra la lista de requisitos funcionales de usuario, que fueron extraídos en las

Id	Nombre	Descripción
RFU01	Registrarse como usuario (vecino)	El sistema debe permitir al usuario registrarse como vecino, almacenando automáticamente el id del dispositivo e ingresando los siguientes datos: nombre y la junta de vecino, que ya se encuentra registrada.
RFU02	Registrarse como usuario (Institución)	El sistema debe permitir al usuario registrarse como institución, almacenando automáticamente el id del dispositivo e indicando a que institución pertenece: carabineros, consultorio, o bomberos y la junta de vecino, que ya se encuentra registrada.
RFU03	Registrar Junta de Vecino	El sistema debe permitir al usuario registrar una nueva junta de vecinos, ingresando el nombre de la junta de vecinos.
RFU04	Ver vecinos	El sistema debe permitir al usuario visualizar todos los integrantes de las juntas de vecinos a las cuales se registró.
RFU05	Añadirse a Junta vecinal	El sistema debe permitir al usuario registrarse en más de una junta vecinal o grupo.
RFU06	Emitir alarma	El sistema debe permitir enviar una alarma a la institución antes registrada en la aplicación, que el usuario pretenda avisar, mediante una notificación.
RFU07	Menú principal	El sistema debe mostrar a todos los usuarios una pantalla de inicio, con un panel de opciones que permita a los usuarios realizar las funcionalidades establecidas para cada usuario.

conversaciones y entrevistas realizadas al cliente en la fase pre-proyecto.

Id	Nombre	Descripción
RFS01	Modificar vecino	El sistema debe permitir al usuario Modificar el registro de vecino que ya se encuentre registrado previamente.
RFS02	Eliminar vecino	El sistema debe permitir al usuario eliminar los datos de un vecino.
RFS03	Ver vecino	El sistema debe permitir al usuario visualizar todos los vecinos registrados.
RFS04	Modificar Institución de seguridad	El sistema debe permitir al usuario modificar el registro de institución de seguridad que ya se encuentre registrado previamente.
RFS05	Eliminar institución de seguridad	El sistema debe permitir al usuario eliminar los datos de una institución de seguridad.
RFS06	Ver instituciones de seguridad	El sistema debe permitir al usuario ver las instituciones de seguridad registradas.
RFS07	Registrar grupo	El sistema debe permitir al usuario registrar un nuevo grupo de junta de vecino, indicando el nombre que tendrá el grupo.
RFS08	Modificar grupo	El sistema debe permitir al usuario modificar el nombre de un grupo de junta de vecino ya registrado.
RFS09	Eliminar grupo	El sistema debe permitir al usuario eliminar los datos de un grupo vecinal.

5.4.2. Requisitos no funcionales del sistema

Los requerimientos no funcionales del sistema, corresponden a una característica que tiene el sistema, sin embargo, no es una funcionalidad, en esta sección, se visualizan los propuestos por el desarrollador dado sus necesidades, las necesidades del cliente y la naturaleza de la aplicación, luego del análisis de requerimientos.

A continuación, la Tabla **4 Restricciones**

ID	Nombre	Descripción
RNF01	Usabilidad	El sistema móvil debe ser de fácil uso para los funcionarios al momento de emitir una alarma.
RNF02	Formato de colores representativos	El sistema deberá utilizar los colores e imágenes representativas de las instituciones de seguridad.
RNF04	Eficiencia	El sistema debe emitir rápidamente las alarmas.

Tabla 3 Requisitos no funcionales

5.4.3. Restricciones

Las restricciones de un sistema, son las que delimitan el alcance que tendrán los requisitos funcionales, con respecto al desarrollo de la aplicación

La **Tabla 4** Restricciones corresponde a las restricciones de la aplicación.

Id	Nombre	Descripción
RES01	Sistema para dispositivo móvil	El sistema debe desarrollarse mediante una arquitectura web, con la cual posteriormente se construye la aplicación móvil, facilitando y ampliando de esta forma la creación de la aplicación a diferentes plataformas.
RESS02	Centralizado en datos	El sistema debe gestionar la información con una base de datos.
RESS03	Lenguaje de programación JavaScript	El lenguaje de programación es JavaScript, ya que permite realizar sistemas web robustos.
RES04	Base de datos MySQL	La base de datos del sistema debe ser implementada en MySQL, debido a que es libre de costos.
RES05	Local storage	Se utiliza para almacenar de manera local los id de los usuarios y de los grupos a los que pertenece. Debido a que es menos limitada y la información persiste, aunque se cierre la aplicación, a menos que se declare eliminada.

Tabla 4 Restricciones

5.4.4. Interfaces externas de entrada

La **Tabla 5** Interfaces externas de entrada muestra todos los datos de entrada que la aplicación debe gestionar, y que el usuario debe ingresar a través de la entrada estándar (pantalla del Smartphone) sobre la interfaz gráfica.

ID	Nombre	Datos o Atributos.
IEE01	Datos del usuario vecino	Nombre y junta de vecinos.
IEE02	Datos del usuario institución	Nombre y junta de vecinos.
IEE03	Datos junta de vecinos	Nombre junta de vecinos.
IEE04	Emitir alarma	Seleccionar el botón correspondiente a la urgencia que desea reportar.

Tabla 5 Interfaces externas de entrada

5.4.5. Interfaces externas de salida

La **Tabla 6** Interfaces externas de salida muestra los datos de salida que el sistema muestra, en este caso por la pantalla del dispositivo móvil.

ID	Nombre	Datos o Atributos.
IES01	Notificación	Mensaje señalando la dirección en donde se desarrolla la situación de emergencia.
IES02	Lista de Vecinos	Lista con el nombre de todos los vecinos de las juntas de vecinos en las que se encuentra registrado.

Tabla 6 Interfaces externas de salida

6. Factibilidad

A continuación, se realiza un estudio de factibilidad, el cual permite conocer que tan viable es el proyecto, en el cual se ve analizan los siguientes aspectos.

Factibilidad Técnica: En esta área se analizan los aspectos relacionado con los recursos que se requieren para un correcto desarrollo del proyecto, como lo son: recursos de Software, Hardware y Recursos Humanos competentes

Factibilidad Operacional: La cual se centra al uso que se le da la aplicación y la aceptación de esta por parte de los usuarios.

Factibilidad Económica: Evalúa la viabilidad respecto a los costos del Proyecto. Adicionalmente, este tipo de factibilidad evalúa los beneficios futuros que se obtendrán una vez puesto en marcha el sistema.

El resultado de los estudios permite concluir la factibilidad del proyecto.

6.1. Factibilidad técnica

En esta sección se determina el equipamiento de software, hardware y recurso humano necesario para desarrollar correctamente el proyecto.

6.1.1. Lenguaje y herramientas a utilizar.

La aplicación de sistema de alarma se desarrolla mediante el framework ionic el cual trabaja con HTML 5, CSS y Angular y como SGBD para el sistema se utiliza MySQL para emitir las alarmas y SQLite para almacenar los FCM de los dispositivos móviles.

Ionic Framework es una capa que trabaja por encima de cordova y que permite la creación de aplicaciones híbridas dinámicas gracias a Angular.js. Éste framework está constituido con Angular y SASS por lo que se puede trabajar de manera más fácil con los elementos CSS y HTML.

6.1.2. Hardware

Actualmente se cuenta con equipos computacionales con las siguientes características:

Servidor:

- Procesador: 4 núcleos Intel (R) Core(TM) i3-3110M – 2.40 GHz.
- Memoria: 2 GB de RAM.
- Disco Duro: 60 GB
- Sistema operativo: Windows 8, 64 bits.

Móvil:

- CPU: Dual Core 1.3 GHZ.
- Memoria: 339 MB.
- Almacenamiento: 2GB.
- Sistema Operativo: Android 4.2.2

Con respecto al análisis realizado sobre los requisitos técnicos del sistema, se puede concluir que efectivamente se dispone con las herramientas técnicas necesarias para llevar a cabo el desarrollo de la aplicación. Por lo tanto, es factible técnicamente.

6.2. Factibilidad operativa

La factibilidad Operativa ayuda a saber si el sistema puesto en marcha es utilizado por los usuarios finales. Lo anterior se logra realizando un estudio de éstos últimos respecto a la posesión de Smartphone ya sea con Android o Apple IOS como sistema operativo. Según estadísticas del INE indican que Chile es uno de los países en donde se encuentra la mayor cantidad de usuarios Smartphone.

Además, mediante una entrevista sobre la interfaz que se les realizó a las personas a quien va dirigido el sistema, al futuro usuario de la aplicación propiamente tal, se analizó la usabilidad de ésta a través de los siguientes criterios:

- La aplicación debe ser fácil de aprender a utilizar.

- La aplicación debe ser de uso rápido, es decir que las tareas se puedan ejecutar de forma expedita.
- El sistema debe ser sencillo y fácil de recordar.
- La aplicación debe ser agradable de utilizar.

A las personas a quienes va dirigida la aplicación deben tener conocimientos básicos sobre el manejo de Smartphone y debe ser utilizada por todo tipo de usuarios, por ende, la aplicación debe tener una interfaz sencilla, sin sobrecarga de contenido.

6.2.1. Obtención de información de los usuarios finales

La información que se recopiló de los usuarios finales fue mediante una entrevista aplicada a 11 personas, de las cuales 8 de ellos era niños con discapacidad y 3 personas adultas mayores. Todos ellos se mostraron expectantes en la puesta en marcha del sistema, ya que indicaban que es algo novedoso y que sería de gran ayuda sobre todo a los adultos mayores entrevistados, ya que ellos en lo personal no habían escuchado de alguna otra aplicación similar al proyecto a desarrollar

6.2.1.1. Entrevista

Con respecto a las preguntas realizadas en la entrevista, se encuentran presentadas en el Anexo B: Encuesta de Interfaz.

6.2.1.2. Resultados de la encuesta



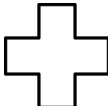









	Aciertos		Aciertos
	5		7
	5		8
	2		7
	6		8
	8		6
	9		9

Tabla 7 Resumen de resultados de entrevista

6.2.1.3. Conclusión de entrevista

Con respecto a los resultados obtenidos en la sección anterior se puede concluir que la interfaz más adecuada para emitir las notificaciones es con los siguientes iconos.

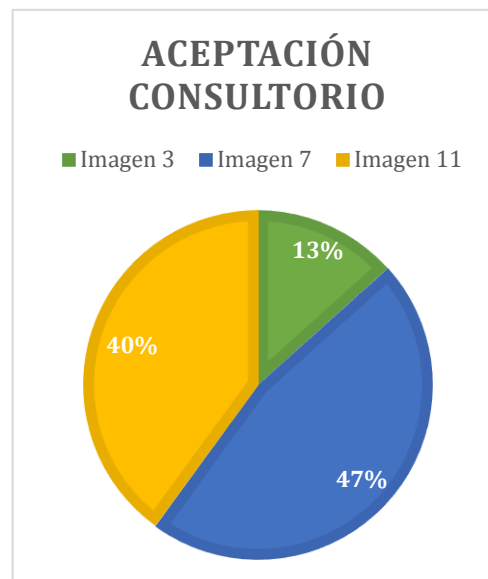
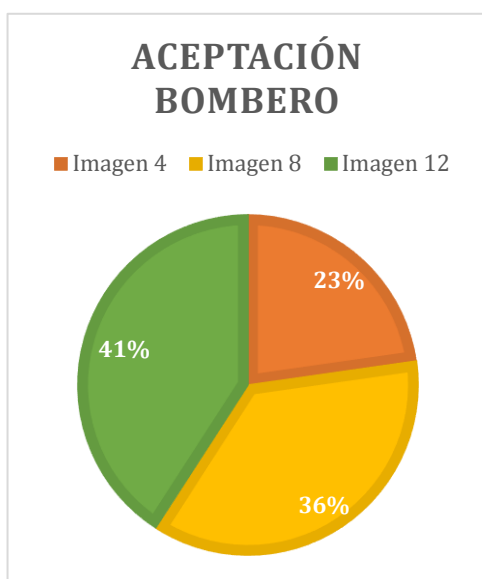
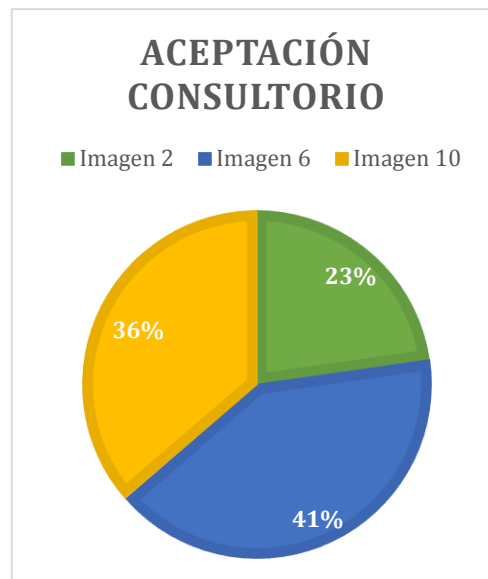
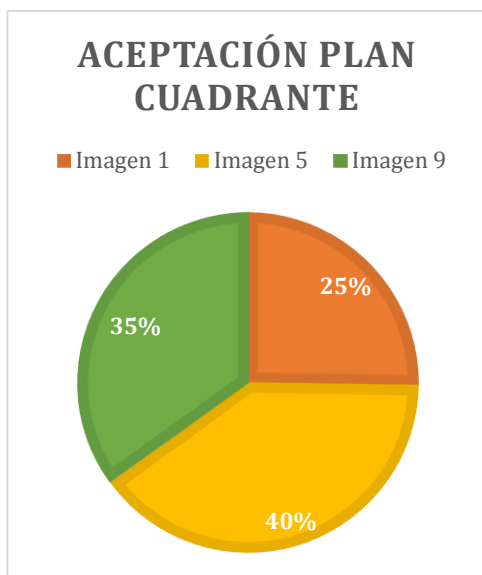


Ilustración 11 Aceptación de iconos para emitir alarmas



La entrevista fue una herramienta de gran utilidad en esta sección, ya que, por medio de esta, se pudo ver la aceptación que tiene la aplicación, además la interfaz será implementada de acuerdo a los resultados obtenidos, lo cual indica que la aplicación es factible operativamente

6.3. Factibilidad económica

A continuación, se realiza un estudio sobre la factibilidad económica la permitirá determinar la posibilidad de desarrollar el proyecto, por medio de estimaciones de costos y beneficios económicos que se obtendrán luego de la puesta en marcha de la aplicación.

Para determinar que es un proyecto viable económicamente se utilizara el indicador VAN (Valor Actual Neto), cuyo resultado permitirá concluir si el proyecto es rentable.

El horizonte de tiempo con el que se evaluará el proyecto será de 5 años.

6.3.1. Recursos necesarios

A continuación, se detallarán los recursos necesarios para el desarrollo de este proyecto. Sin embargo, existen algunos recursos que son actualmente propiedad del desarrollador, por lo tanto, no es necesario adquirirlos.

- Notebook Lenovo Intel Core i3, 4GB RAM, 500 GB, Windows 8.1 pro.
- Apple Mac Intel Core i5, 8GB RAM, 128 GB SSD, OS Yosemite.
- Horas hombre de programador.

6.3.2. Determinación de costos

6.3.2.1. Costos de desarrollo

Para el desarrollo de este proyecto se requiere de un Ingeniero Civil en Informática, el cual cuesta \$10.000.- horas/hombre aproximadamente. La duración estimada del proyecto es de 3 meses aproximadamente, considerando 45 horas semanales es un total de 12 semanas, de las cuales genera un total de 540 horas y un total de \$ 5.400.000.-

Se requiere de un técnico en informática para la realización de las pruebas el cual tiene un valor de \$ 5.000.- aproximadamente. Se estima que el proceso de pruebas tenga una duración de 2 semana con 45 horas semanales, lo que genera un total de \$450.000.- aproximadamente.

Junto con eso se necesita un Técnico en diseño gráfico para la estética de la aplicación con un costo de \$3500.- horas/hombre, además el proceso requiere de un periodo de una semana trabajando 45 horas semanales lo que genera un total de \$ 157.500.-

6.3.2.2. Costos de implementación

Con respecto a los costos de implementación se debe considerar los requerimientos técnicos para el desarrollo (Sección 5.1.2). El equipo con las características necesarias para desarrollar el proyecto tiene un valor de \$289.990.- y el equipo necesario para generar la APK del proyecto para dispositivos con IOS, tiene un valor de \$799.990.-.

Respecto a la utilización de software, en su mayoría son gratuitos a excepción de la licencia de Windows 8.1 pro, el cual tiene un valor de \$43.990.-

6.3.2.3. Costos de operación

Para la puesta en marcha del sistema e requiere de un servidor para los web services utilizados en la aplicación móvil. Dicho hosting tiene un costo de \$45.000.- aproximadamente sin IVA, por lo que se estima que luego de aplicarle el 19% correspondiente al IVA el costo real es de \$53.550.- por año.

Además, estos costos serán considerados en el flujo de caja por cada año dentro de la evaluación.

6.3.2.4. Resumen de costos

A continuación, por medio de la Tabla 8 Resumen de costos se mostrará un resumen de los costos divididos en las tres categorías antes mencionadas

Costos de Desarrollo	
Costo personal de desarrollo	\$ 5.400.000.-
Costo personal de pruebas	\$ 450.000.-
Costo personal de diseño	\$ 157.500.-
COSTO TOTAL DE DESARROLLO	\$ 6.007.500.-
Costos de Implementación	
Equipo de desarrollo	\$ 289.990.-
Equipo de generación APK	\$799.990.-
Licencia de Software	\$ 43.990.-
COSTO TOTAL DE INVERSIÓN	\$ 7.141.470.-
Costo de Operación	
Servidor Web Service	\$ 53.550.-
COSTO TOTAL DE OPERACIÓN	\$ 1.653.550.-

Tabla 8 Resumen de costos

6.3.3. Determinación de ingresos y beneficios

El ingreso percibido es mediante la descarga de la aplicación, ya que PLAY STORE en Android genera una ganancia de 3 pesos por descargas, además considerando un costo de \$1.399.- tomando como referencia la aplicación española MY11.

Se calcula que la aplicación será instalada por 40 dispositivos distintos cada mes aproximadamente, lo cual conlleva un ingreso aproximadamente de \$56.080.- mensuales. Por lo cual se estima que al año se generarían \$672.960.- aproximadamente. No obstante, cabe destacar que estas cifras son una estimación de lo que se espera vender anualmente, esto implica considerar un costo de administración mensual, sin embargo en este caso el mismo desarrollador puede ser el encargado de realizar la mantención, por lo cual es un costo ahorrado.

Además, se presentan beneficios intangibles tales como:

- Acceso a la localización de los usuarios por parte de las instituciones, siendo un gran apoyo a la hora de entregar correctamente la dirección en donde ocurre la urgencia.
- Ahorro de tiempo por parte de las instituciones, ya que con esta aplicación se encontrará en menos tiempo la dirección en donde se encuentra la víctima, además como se podrá visualizar la navegación, lo que facilitará la llegada es estos lugares.
- Accesos del usuario a un medio de difusión y canal de contacto gratuito con las instituciones de seguridad.

6.3.4. Flujo de caja

En esta sección, tal y como fue mencionado para evaluar la factibilidad económica se utiliza el indicador VAN y así tener un criterio de decisión frente a éste.

	Año 0	Año 1	Año 2	Año 3	Año 4	Año 5
Ingresos (+)						
Beneficios	\$ 0	\$ 672.960	\$ 672.960	\$ 672.960	\$ 672.960	\$ 672.960
Costos (-)						
Servidor	\$ 0	(\$ 53.550)	(\$ 53.550)	(\$ 53.550)	(\$ 53.550)	(\$ 53.550)
Inversión (-)						
Personal	(\$ 6.007.500)					
Computador	(\$ 1.089.980)					
Licencia Software	(\$ 43.990)					
TOTAL	(\$ 7.141.470)	\$ 619.410	\$ 619.410	\$ 619.410	\$ 619.410	\$ 619.410

Tabla 9 Flujo de caja

6.4. Cálculo de VAN

Para el cálculo del indicador VAN se utiliza la siguiente ecuación que se muestra en la Ilustración 12 Fórmula de la VAN:

$$VAN = \sum_{t=1}^n \frac{V_t}{(1+k)^t} - I_0$$

Ilustración 12 Fórmula de la VAN

Donde:

- V_t representa los flujos de caja para el periodo t .
- I_0 representa la inversión total inicial.
- n es el número de periodos considerados.
- k es la tasa de descuento o mínima rentabilidad exigida.

Para efectos del cálculo de considerará una tasa de descuento de 12%.

6.4.1. Caso tradicional

Realizando los cálculos para el caso tradicional donde se desea realizar el proyecto de forma particular, se deben considerar todos los gastos, entonces tenemos que:

$$VAN(12\%) = -(0) + \frac{619.410}{(1+0.12)^1} + \frac{619.410}{(1+0.12)^2} + \frac{619.410}{(1+0.12)^3} + \frac{619.410}{(1+0.12)^4} + \frac{619.410}{(1+0.12)^5}$$

$$VAN(12\%) = -(0) + 553.044 + 493.789,9 + 440.883,8 + 393.646 + 351.469,9$$

$$VAN(12\%) = 2.232.833,6$$

6.4.2. Caso real

Luego de haber realizado los cálculos respectivos, cabe destacar algunos aspectos importantes sobre una memoria de título, los cuales deben ser considerados:

- El costo de desarrollo y pruebas es cero ya que esta labor es desarrollada por el alumno tesista.
- El alumno memorista cuenta con un equipo apto para el desarrollo que incluye la licencia de software de Windows, por lo que éste costo igual debe ser descartado.
- La carrera de Ingeniería Civil en Informática provee del servicio de apoyo de un diseñador gráfico al proyecto de título, por lo que el costo de diseño gráfico es cero.
- La carrera de ingeniería civil en informática cuenta con servidores dedicados a los proyectos de título y que cumplen con los requerimientos técnicos, por lo que el costo de operación por concepto de hosting es cero.

6.5. Conclusión de factibilidad

Con respecto a la factibilidad técnica y operativa, se puede concluir que la aplicación es viable, además, considerando la factibilidad económica, en donde el indicador VAN es positivo, se podría decir que el proyecto también es viable económicamente.

No obstante, debido a que el proyecto corresponde a una actividad de titulación, para obtener una factibilidad real se requiere considerar un segundo cálculo de VAN, debido a que el equipo de desarrollo, la licencia del software, mano de obra y pruebas son responsabilidad del estudiante por lo que no generan un costo adicional. Además, el costo de diseño gráfico y hosting son servicios otorgados por la carrera de ingeniería civil en informática a los alumnos memoristas, lo cual tampoco sería considerado un costo adicional.

En conclusión, el que se esté realizando un proyecto de título no genera costos extras para los estudiantes ni para la casa de estudios, por otra parte, solo se consideran los beneficios que conlleva el buen funcionamiento del proyecto, por lo que si se considera factible.

CAPITULO 7 “INCREMENTO I”

7. Incremento I

A continuación, se especifica el análisis del proyecto, el cual se divide en tres ámbitos principales los cuales son: procesos de negocios, diagramas de casos de uso y modelamiento de datos. En primera instancia se muestra el proceso de negocios a través de un diagrama de flujo, luego el diagrama de clases de uso con cada una de sus especificaciones, describiendo las secuencias e interacciones con el sistema. Finalmente se visualiza el modelo entidad relación en cual tiene como propósito mostrar un modelo de datos.

7.1. Análisis Incremento I

Conceptualización y representación gráfica de los requerimientos del usuario, mediante diagramas y especificaciones en lenguaje natural, que permiten entender lo que debe hacer el sistema frente a cada usuario, los cuales se denominan actores.

7.1.1. Casos de uso de Incremento I

A continuación, se presenta de manera gráfica las actividades de interacción, que deben ocurrir entre los usuarios y el sistema. Además de explicar cómo debe reaccionar el sistema ante los diversos eventos que suceden según su uso. Este tipo de modelo tiene los siguientes componentes:

- Límite del sistema: corresponde al cuadro dibujado alrededor del modelo el cual indica el límite del sistema modelado.
- Actores: se refiere a los roles que cumplen las personas y/o entidades que utilizarán el sistema.
- Casos de Uso: indican lo que los actores pueden realizar en el sistema.
- Relaciones: se refiere al vínculo que existe entre los actores y los casos de uso.

7.1.1.1. Actores

A continuación, se especifican los actores correspondientes a la aplicación móvil:

Actor	Usuario.	ID	CUA01
Descripción	Publico general que pertenezca a una comunidad vecinal.		
Responsabilidades	Actualizar contactos telefónicos, emitir y recibir alarmas o notificaciones.		

Tabla 10 Actor "Usuario"

Actor	Institución de seguridad.	ID	CUA01
Descripción	Institución que pertenezca a las siguientes entidades: carabineros (plan cuadrante), bomberos y consultorio.		
Responsabilidades	recibir alarmas.		

Tabla 11 Actor "Institución de Seguridad"

7.1.1.2. Diagrama de casos de uso de Incremento I

En esta sección, la Ilustración 13 Diagrama de casos de uso incremento I representa los casos de uso correspondientes al incremento I del proyecto.

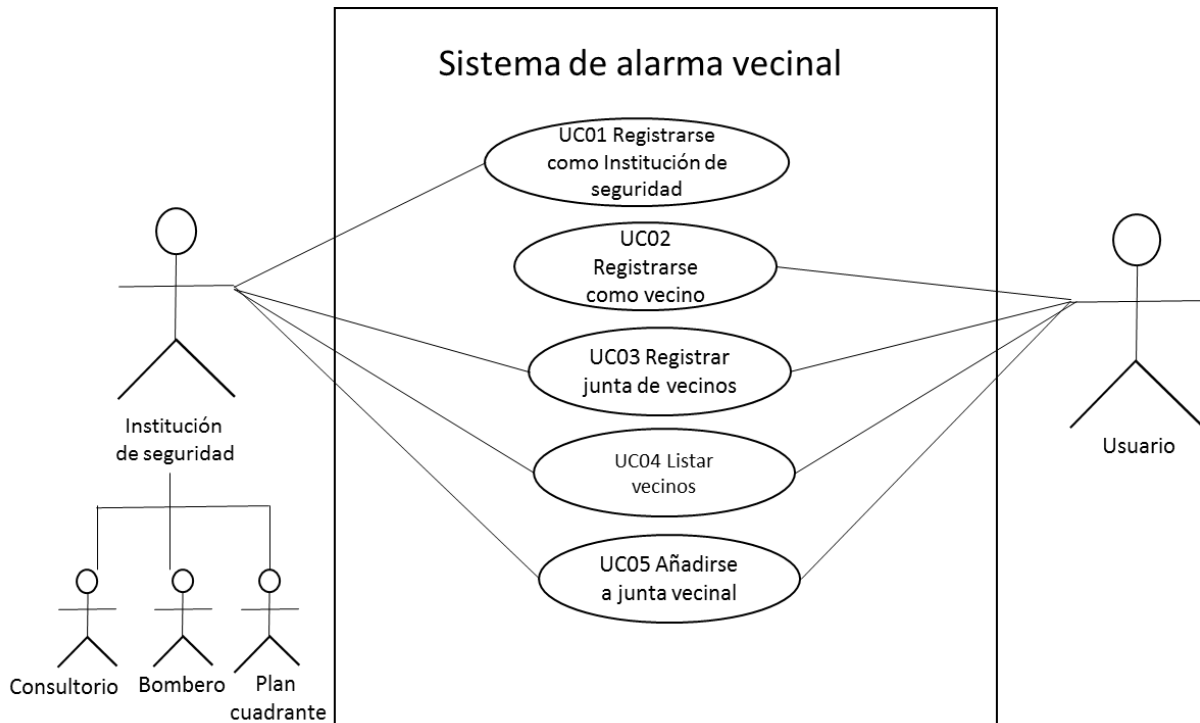


Ilustración 13 Diagrama de casos de uso incremento I

7.1.1.3. Especificación de casos de uso de Incremento I

A continuación, se presenta la especificación de casos de uso, en donde se pretende describir en lenguaje natural y de manera general, las acciones de los actores y el sistema.

7.1.1.3.1. Registrarse como institución de seguridad

En la Tabla 12 Especificación del caso de uso "Registrarse como institución de seguridad" se especifica el caso de uso móvil "Registrarse como una institución de seguridad", donde el actor puede registrar su dispositivo en la aplicación.

Registrarse como institución de seguridad			
ID.	UC01	Requisitos Asociados.	RFU01, RFU02, RFU03.
Descripción.	El Usuario registra sus datos su información y su dispositivo en la aplicación con el rol de institución de seguridad.		
Actores.	Institución de seguridad.	Actores secundarios.	
Pre-condiciones.	Junta de vecinos registrada UC03	Post-condiciones.	Usuario Registrado.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "Registrarse" 2. Sistema muestra el formulario que permite registrar los datos. 3. Usuario confirma el registro de los datos. 4. Sistema muestra mensaje de "Operación Exitosa". 		

Tabla 12 Especificación del caso de uso "Registrarse como institución de seguridad"

7.1.1.3.2. Registrarse como vecino

En Tabla 13 Especificación del caso de uso "Registrar Junta de Vecinos" la se especifica el caso de uso móvil "Registrarse como vecino", donde el actor puede registrar su dispositivo en la aplicación.

Registrarse como vecino			
ID.	UC02	Requisitos Asociados.	RFU01, RFU02, RFU03.
Descripción.	El Usuario registra sus datos su información y su dispositivo en la aplicación con el rol de vecino.		
Actores.	Vecino.	Actores secundarios.	
Pre-condiciones.	Junta de vecinos registrada UC03	Post-condiciones.	Usuario Registrado.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "Registrarse" 2. Sistema muestra el formulario que permite registrar los datos. 3. Usuario confirma el registro de los datos. 4. Sistema muestra mensaje de "Operación Exitosa". 		

Ilustración 14 Especificación del caso de uso "Registrarse como vecino"

7.1.1.3.3. Registrar junta de vecinos

En la Tabla 13 Especificación del caso de uso "Registrar Junta de Vecinos" se especifica el caso de uso móvil "Registrar Junta de Vecinos", donde el actor puede realizar el registro de una nueva junta de vecinos.

Registrar Junta de Vecinos			
ID.	UC03	Requisitos Asociados.	RFU3.
Descripción.	El Usuario puede registrar una nueva junta de vecinos.		
Actores.	Usuario e Institución de seguridad.	Actores secundarios.	
Pre-condiciones.		Post-condiciones.	Junta de vecinos registrada.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "Registrar junta de vecinos" 2. Sistema muestra el formulario que permite registrar los datos de la junta de vecinos. 3. Usuario confirma el registro de los datos. 4. Sistema muestra mensaje de "Operación Exitosa". 		

Tabla 13 Especificación del caso de uso "Registrar Junta de Vecinos"

7.1.1.3.4. Listar Vecinos

En la Tabla 14 Especificación del caso de uso "Listar Vecinos" se especifica el caso de uso móvil "Listar Vecinos", donde el actor puede listar a todos los vecinos que se encuentre registrados en la misma junta de vecinos y que posean la aplicación.

Listar Vecinos			
ID.	UC04	Requisitos Asociados.	
Descripción.	El Usuario visualiza una lista de todos los vecinos pertenecientes a la junta de vecinos en la cual se haya registrado.		
Actores.	Usuario e Institución de seguridad.	Actores secundarios.	
Pre-condiciones.		Post-condiciones.	Lista de Vecinos.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "Ver Junta de Vecinos" 2. Sistema muestra una lista de todos los vecinos de esa junta de vecinos. 		

Tabla 14 Especificación del caso de uso "Listar Vecinos"

7.1.1.3.5. Añadirse a junta vecinal

En la Tabla 15 Especificación de caso de uso "Añadirse a junta de vecinos" se especifica el caso de uso móvil "Añadirse a junta vecinal", donde el actor puede listar a todos los vecinos que se encuentre registrados en la misma junta de vecinos y que posean la aplicación.

Añadirse a Junta Vecinal			
ID.	UC05	Requisitos Asociados.	RF05
Descripción.	El Usuario puede incorporarse a otras juntas de vecinos.		
Actores.	Usuario e Institución de seguridad.	Actores secundarios.	
Pre-condiciones.	Junta de vecinos registrada. UC02	Post-condiciones.	Usuario registrado en otra junta de vecinos.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "Añadirse a junta de Vecinos" 2. Sistema muestra el formulario que permite Añadirse a otra junta de vecinos. 3. Usuario confirma el registro de los datos. 4. Sistema muestra mensaje de "Operación Exitosa". 		

Tabla 15 Especificación de caso de uso "Añadirse a junta de vecinos"

7.1.2. Diseño

A continuación, en esta sección se detallan los aspectos de diseño del proyecto con respecto a los ámbitos del MER ⁷ y diseño físico de la base de datos, la cual corresponde a la evolución del MER y su especificación de tablas.

7.1.2.1. Modelamiento de datos del incremento I

Para definir el modelamiento de datos se utilizará un MER y una descripción breve de las respectivas entidades de dicho modelo. Ambas subsecciones se definen a continuación.

7.1.2.1.1. Modelo Entidad-Relación

En la Ilustración 15 Modelo entidad relación incremento I se ilustra el modelo

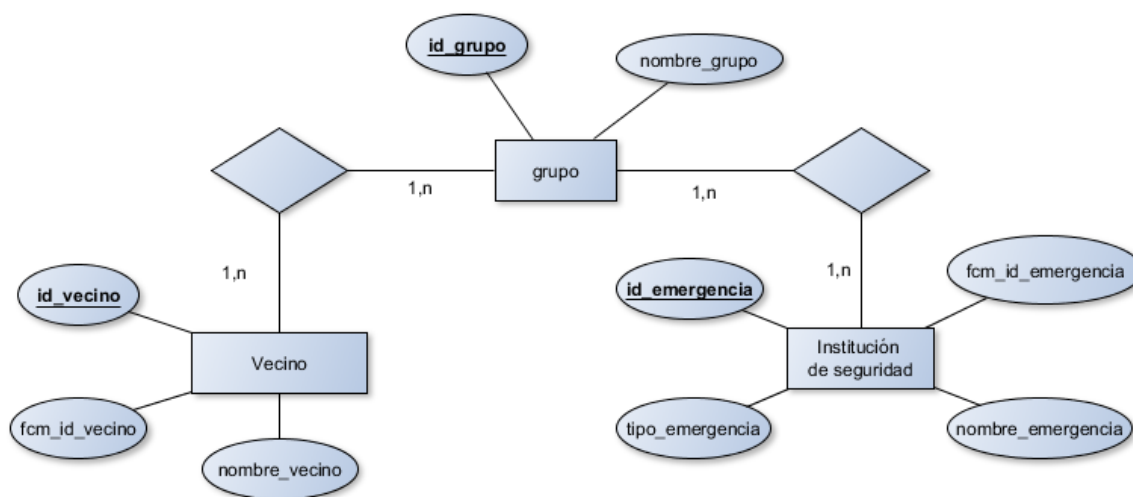


Ilustración 15 Modelo entidad relación incremento I

entidad-relación que da origen al modelo físico de la base de datos.

⁷ Modelo Entidad Relación

7.1.2.1.2. Modelo físico de la base de datos

En esta sección se describe el diseño físico de la base de datos, la cual incluye aspectos de bajo nivel, es decir los atributos y los tipos de datos que se utilizan, estos últimos no son considerados en el MER ilustrado en la sección 7.1.2.1.1.

La Ilustración 16 Modelo físico del incremento I representa el diagrama físico de a base de dato que se obtuvo a partir del modelo entidad relación, la cual da origen a la base de

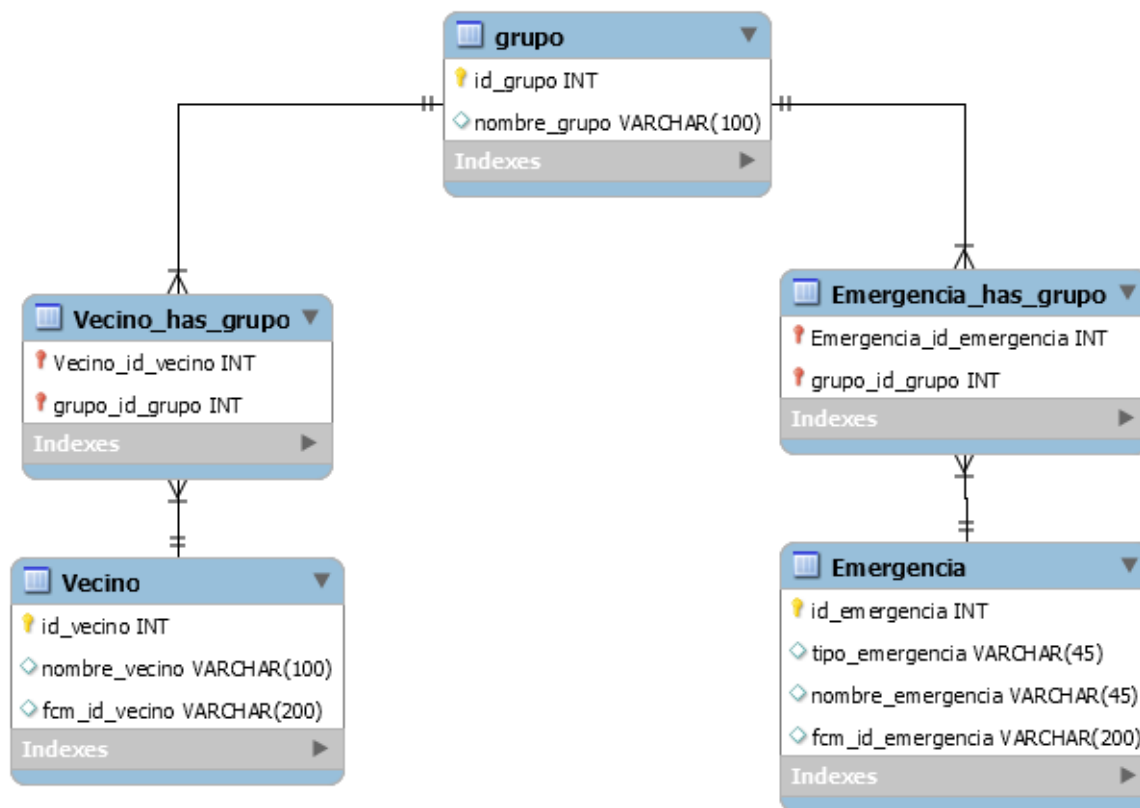


Ilustración 16 Modelo físico del incremento I

dato misma.

7.1.2.1.3. Descripción de cada tabla

A continuación, se ilustra una especificación de las tablas que son utilizadas en el Modelo físico de la base de datos representado en la sección anterior.

7.1.2.1.3.1. Descripción tabla: grupo.

Descripción de la tabla	
Nombre de la Tabla	grupo
Descripción	<p>La tabla grupo contiene los datos correspondientes a un grupo, los cuales corresponden a: id y el nombre de éste. La clave primaria corresponde a “id_grupo”.</p> <ul style="list-style-type: none"> • Id_grupo: INT • Nombre_grupo: VARCHAR(100)
Validaciones	No tiene

Tabla 16 Descripción de la tabla grupo

7.1.2.1.3.2. Descripción tabla: vecino_has_grupo.

Descripción de la tabla	
Nombre de la Tabla	Vecino_has_grupo
Descripción	<p>La tabla vecino_has_grupo contiene la relación entre los datos correspondientes a la tabla grupo y a la tabla vecino de la aplicación debido a la normalización del MER, los cuales corresponden a: vecino_id_vecino, grupo_id_grupo</p> <ul style="list-style-type: none"> • Vecino_id_vecino: INT • Grupo_id_grupo: INT
Validaciones	No tiene

Tabla 17 Descripción de la tabla vecino_has_grupo

7.1.2.1.3.3. Descripción tabla: vecino

Descripción de la tabla	
Nombre de la Tabla	vecino
Descripción	<p>La tabla vecino contiene los dato correspondientes a la tabla vecino, los cuales corresponden a: id_vecino, nombre_vecino, fcm_id_vecino. La clave primaria de esta tabla es : id_vecino.</p> <ul style="list-style-type: none"> • id_vecino: INT • nombre_vecino: VARCHAR(100) • fcm_id_vecino: VARCHAR(200)
Validaciones	No tiene

Tabla 18 descripción de la tabla vecino

7.1.2.1.3.4. Descripción tabla: Emergencia

Descripción de la tabla	
Nombre de la Tabla	Emergencia
Descripción	<p>La tabla emergencia contiene los datos correspondientes a los usuarios que pertenezcan a un centro de emergencia, los cuales corresponden a: id_emergencia, nombre_emergencia, tipo_emergencia y fcm_id_emergencia. La clave primaria corresponde a “id_emergencia”.</p> <ul style="list-style-type: none"> • Id_emergencia: INT • Tipo_emergencia: VARCHAR(45) • Nombre_emergencia: VARCHAR(45) • Fcm_id_emergencia: VARCHAR(200)
Validaciones	No tiene

Tabla 19 Descripción de la tabla Emergencia

7.1.2.1.3.5. Descripción de la tabla: emergencia_has_grupo

Descripción de la tabla	
Nombre de la Tabla	Emergencia_has_grupo
Descripción	<p>La tabla emergencia_has_grupo contiene los datos correspondientes a la relación ente las tablas grupo y emergencia, debido a la normalización del MER, los cuales corresponden a: emergencia_id_emergencia y grupo_id_grupo.</p> <ul style="list-style-type: none"> • Grupo_id_grupo: INT • Emergencia_id_emergencia:INT
Validaciones	No tiene

Tabla 20 Descripción de la tabla emergencia_has_grupo

7.2. Pruebas Incremento I

Los casos de pruebas definidos y/o ejecutados son los siguientes:

- Registrarse como usuario.
- Registrarse como plan cuadrante.
- Registrarse como consultorio.
- Registrarse como bombero.
- Registrar junta de vecinos.
- Ver junta de vecinos.

7.2.1. Aspectos Generales

Se considera como una unidad los métodos de una clase, ya que, son una parte fundamental del proyecto. Para esto se aplican pruebas con particiones en donde se ingresan valores críticos para verificar que cumpla con su función o se ingresan valores erróneos para verificar que los rechace y no los acepte.

Las unidades escogidas para este primer incremento son registrarse como los distintos usuarios (vecino, plan cuadrante, bombero y consultorio), además de registrar una nueva junta de vecino, poder añadirse a más de una junta vecinal y poder ver la lista de vecinos que se encuentran en las juntas de vecinos en las cuales se encuentra registrado.

Las técnicas y herramientas utilizadas serán particiones y valores límites, esto se refiere a las pruebas de caja negra, donde solo se consideran los datos de entradas y se analizan los datos de salida, sin preocuparse de lo que ocurre al interior en el código.

Como configuración para realizar estas pruebas, solo se necesita un notebook o equipo de escritorio, con 2 GB de memoria RAM y 250GB de disco duro como mínimo, un sistema operativo Windows, MAC o Linux y finalmente lo más importante es que el equipo tenga acceso a internet.

7.2.2. Especificación de pruebas del Incremento I

unidad	referencia	Objetivo de las pruebas	Técnicas usadas	Criterio de cumplimiento
Registrarse como usuario	Clase Registro y UC01	Comprobar que el usuario fue registrado como vecino.	Caja negra	100%
Registrarse como Plan Cuadrante	Clase Registro y UC01	Comprobar que el usuario fue registrado como plan cuadrante.	Caja negra	100%
Registrarse como Consultorio	Clase Registro y UC01	Comprobar que el usuario fue registrado como Consultorio	Caja negra	100%
Registrarse como Bombero	Clase Registro y UC01	Comprobar que el usuario fue registrado como bombero.	Caja negra	100%
Registrar Junta de Vecinos	Clase Registro y UC02	Comprobar que se registró una nueva junta vecinal.	Caja negra	100%
Añadirse Junta de Vecinos	Clase Registro y UC04	Comprobar que el usuario se puede registrar en más de una junta de vecinos.	Caja negra	100%
Ver Junta de Vecinos	Clase Registro y UC03	Comprobar se visualizan todos los vecinos de la junta de vecinos.	Caja negra	100%

Ilustración 17 Especificación de pruebas incremento I

7.2.3. Casos de pruebas de caja negra

7.2.3.1. Caso de prueba: Registrarse como usuario

Caso de prueba: CPU01	
Nombre	Registrarse como vecino.
Descripción	Comprobar que el usuario fue registrado como vecino con un nombre del vecino y una junta de vecino a la que pertenece.
Tipo	Caja negra.
Precondiciones	Debe haber una junta de vecinos ya registrada.
Postcondiciones	Registro de usuario exitoso.
Entradas	<ul style="list-style-type: none"> • Nombre del usuario: Norma Contreras. • Nombre de Junta de Vecinos: Doña Francisca.
Referencias	UC01
Procedimiento de la prueba	
Actor	Sistema
1. El usuario ingresa su nombre y el nombre de la junta de vecinos.	2. El sistema graba datos en la base de datos.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 21 Especificación de prueba "Registrarse como vecino"

7.2.3.2. Caso de prueba: Registrarse como plan cuadrante

Caso de prueba: CPU02	
Nombre	Registrarse como plan cuadrante.
Descripción	Comprobar que el usuario fue registrado como emergencia del tipo plan cuadrante y una junta de vecino.
Tipo	Caja negra.
Precondiciones	Debe haber una junta de vecinos ya registrada.
Postcondiciones	Registro de usuario exitoso.
Entradas	<ul style="list-style-type: none"> • Emergencia: plan cuadrante. • Nombre de Junta de Vecinos: Doña Francisca.
Referencias	UC01
Procedimiento de la prueba	
Actor	Sistema
1. El usuario ingresa su emergencia y el nombre de la junta de vecinos.	2. El sistema graba datos en la base de datos.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 22 Especificación de prueba "Registrarse como plan cuadrante"

7.2.3.3. Caso de prueba: Registrarse como consultorio

Caso de prueba: CPU03	
Nombre	Registrarse como consultorio.
Descripción	Comprobar que el usuario fue registrado como emergencia del tipo consultorio y una junta de vecino.
Tipo	Caja negra.
Precondiciones	Debe haber una junta de vecinos ya registrada.
Postcondiciones	Registro de usuario exitoso.
Entradas	<ul style="list-style-type: none"> • Emergencia: consultorio. • Nombre de Junta de Vecinos: Doña Francisca.
Referencias	UC01
Procedimiento de la prueba	
Actor	Sistema
1. El usuario ingresa su emergencia y el nombre de la junta de vecinos.	2. El sistema graba datos en la base de datos.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 23 Especificación de prueba "Registrarse como consultorio"

7.2.3.4. Caso de prueba: Registrarse como Bombero

Caso de prueba: CPU04	
Nombre	Registrarse como bombero.
Descripción	Comprobar que el usuario fue registrado como emergencia del tipo bombero y una junta de vecino.
Tipo	Caja negra.
Precondiciones	Debe haber una junta de vecinos ya registrada.
Postcondiciones	Registro de usuario exitoso.
Entradas	<ul style="list-style-type: none"> • Emergencia: bombero. • Nombre de Junta de Vecinos: Doña Francisca.
Referencias	UC01
Procedimiento de la prueba	
Actor	Sistema
1. El usuario ingresa su emergencia y el nombre de la junta de vecinos.	2. El sistema graba datos en la base de datos.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 24 Especificación de prueba "Registrarse como bombero"

7.2.3.5. Caso de prueba: Registrar Junta de vecinos

Caso de prueba: CPU05	
Nombre	Registrar junta de vecinos.
Descripción	Comprobar que la junta de vecinos fue registrada con su nombre.
Tipo	Caja negra.
Precondiciones	
Postcondiciones	Registro de una nueva junta de vecinos exitosa.
Entradas	<ul style="list-style-type: none"> Nombre de Junta de Vecinos: Santa Clara.
Referencias	UC02
Procedimiento de la prueba	
Actor	Sistema
1. El usuario ingresa el nombre de la junta de vecinos.	2. El sistema graba datos en la base de datos.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 25 Especificación de la prueba "Registrar junta de vecinos"

7.2.3.6. Caso de prueba: Añadirse a junta de vecinos

Caso de prueba: CPU06	
Nombre	Añadirse a junta de vecinos.
Descripción	Comprobar que el usuario fue registrado en una junta de vecino.
Tipo	Caja negra.
Precondiciones	Debe haber una junta de vecinos ya registrada.
Postcondiciones	Registro en la junta de vecinos exitosa.
Entradas	<ul style="list-style-type: none"> Nombre de Junta de Vecinos: Santa Clara.
Referencias	UC04
Procedimiento de la prueba	
Actor	Sistema
1. El usuario busca la junta de vecinos ingresando el nombre de ésta.	2. El sistema graba datos en la base de datos.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 26 Especificación de prueba "Añadirse a junta de vecinos"

7.2.3.7. Caso de prueba: Ver juntas de vecinos

Caso de prueba: CPU07	
Nombre	Ver juntas de vecinos.
Descripción	Comprobar lista de todos los vecinos de las juntas de vecinos en la que se encuentra registrado el usuario.
Tipo	Caja negra.
Precondiciones	Debe haber una junta de vecinos ya registrada y el usuario debe encontrarse registrado en una.
Postcondiciones	Lista de vecinos exitosa.
Entradas	<ul style="list-style-type: none"> Nombre de los vecinos: Norma Contreras.
Referencias	UC03
Procedimiento de la prueba	
Actor	Sistema
1. El usuario ingresa el nombre del vecino.	2. El sistema graba datos en la base de datos.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 27 Especificación de caso de uso Ver junta de vecinos

7.2.4. Conclusiones de las pruebas

En síntesis, se le aplicaron pruebas de caja negra a la aplicación orientadas a comprobar que los datos estaban siendo ingresados correctamente, ya que sin estos la función principal que es enviar notificaciones no se podría llevar a cabo.

Dichas pruebas permitieron realizar una comparación de los resultados esperados con respecto a lo que resultó realmente, dando como resultado el 100% de cumplimiento estipulado.

CAPITULO 8 “INCREMENTO II”

8. Incremento II

A continuación, se desglosa el análisis del proyecto correspondiente al segundo incremento, en donde al igual que en el avance 1 (véase 7) se divide en dos partes. Debido a que el diseño se conserva igual que en el incremento I.

8.1.1. Casos de Uso

A continuación, se presenta de manera gráfica los sucesos de interacción, que debe ocurrir entre los usuarios y el sistema durante el segundo incremento de este proyecto.

8.1.1.1. Actores

Debido a que los actores que interactúan en este proyecto en el segundo incremento son los mismo que interactúan en el primer incremento véase 7.1.1.1. Sin embargo, para el modulo del portal web se especifica el siguiente actor:

Actor	Administrador	ID	CUA03
Descripción	Persona dedicada a la gestión de la base de datos de la aplicación.		
Responsabilidades	Gestionar base de datos.		

Tabla 28 Actor: "Administrador"

8.1.1.2. Diagrama de Casos de Uso incremento II

A continuación, se ilustra el diagrama de casos de usos correspondiente al incremento II, y que tiene como finalidad complementar el análisis de los requisitos.

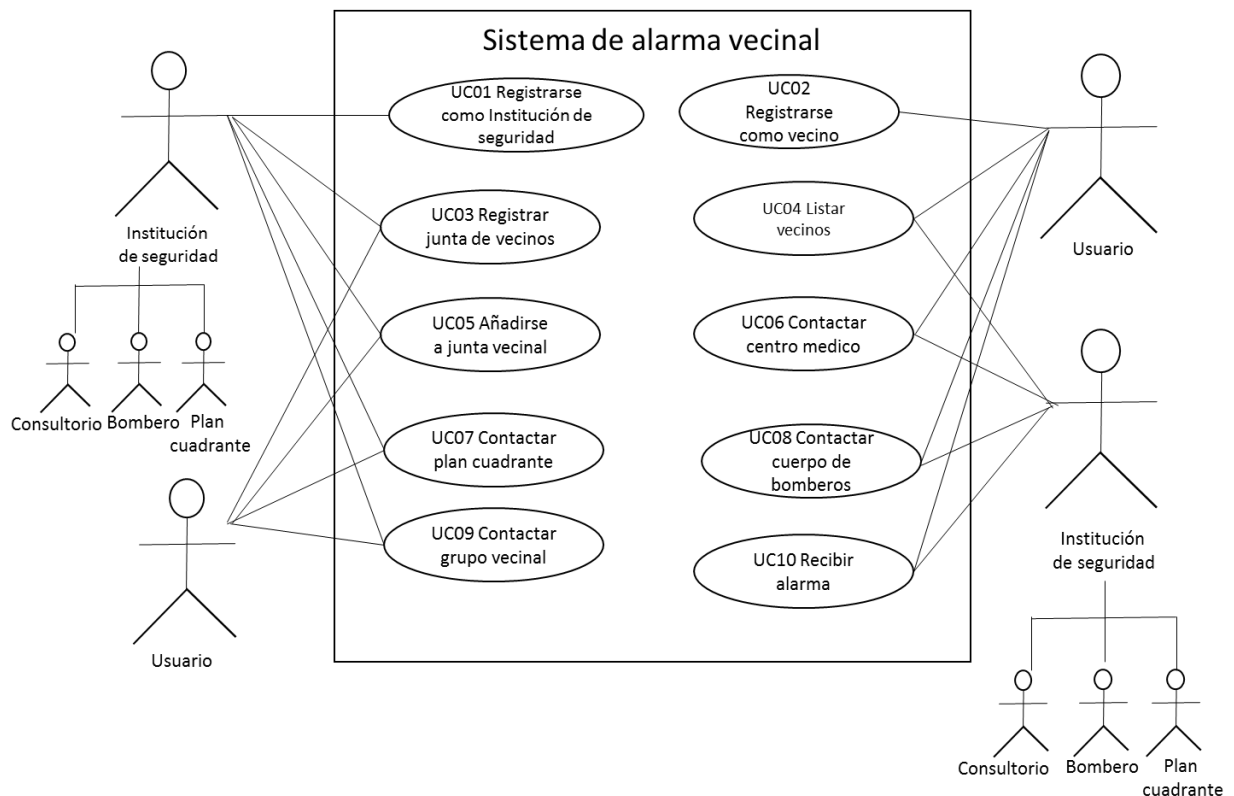


Ilustración 18 Diagrama de casos de uso del incremento II

8.1.1.3. Diagrama de casos de uso portal web

A continuación, se presenta el diagrama de casos de uso correspondiente al portal web, por el cual se gestionará la base de datos.

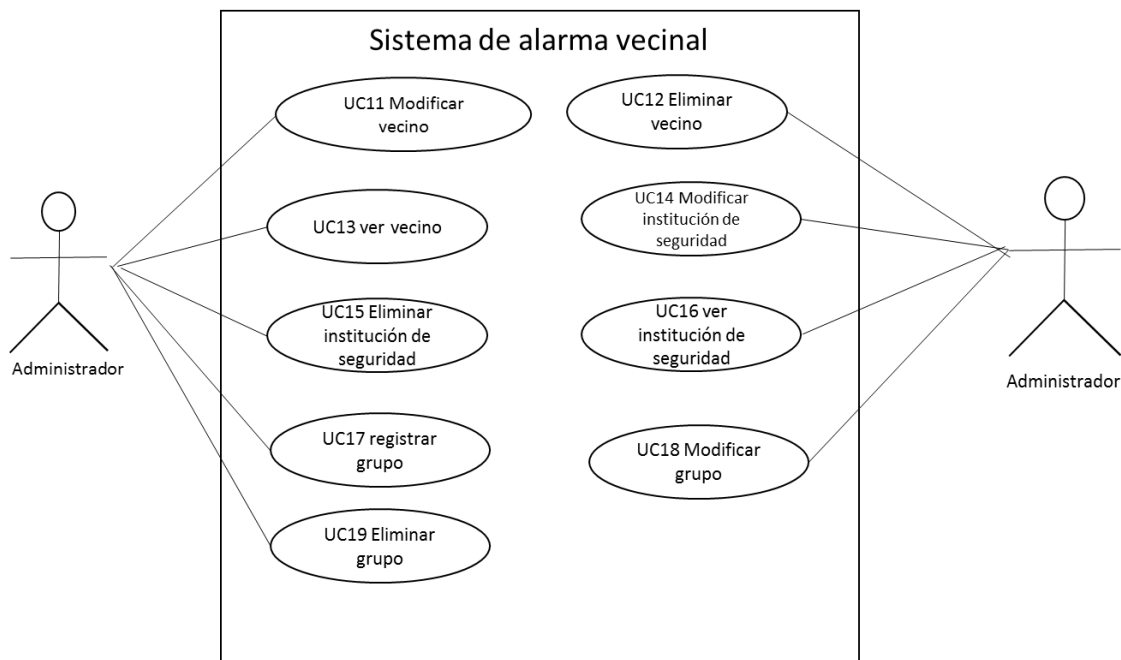


Tabla 29 Diagrama de caso de uso incremento II del portal web

8.1.1.4. Especificación de casos de uso

8.1.1.4.1. Contactar centro médico

En la Tabla 30 Especificación de casos de uso "Contactar centro médico" se especifica el caso de uso móvil "Contactar centro médico", donde el actor puede realizar un contacto correspondiente a un centro médico en la aplicación.

Contactar centro médico			
ID.	UC06	Requisitos Asociados.	RF06
Descripción.	El Usuario emite una alarma al centro médico que le corresponde mediante la aplicación.		
Actores.	Usuario.	Actores secundarios.	
Pre-condiciones.	Tener registrado en el grupo vecinal un usuario del tipo centro médico. UC01	Post-condiciones.	Alarma emitida.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "Emitir una alarma" 2. Sistema muestra cuatro botones para que el usuario seleccione a quien va dirigida la alarma. 3. Usuario selecciona el icono de la enfermera, que corresponde al centro medico 4. Sistema envía la notificación con la geolocalización del usuario. 5. Sistema muestra mensaje de "Operación Exitosa". 		

Tabla 30 Especificación de casos de uso "Contactar centro médico"

8.1.1.4.2. Contactar plan cuadrante

En la Tabla 31 Especificación de casos de uso "Contactar plan cuadrante " se especifica el caso de uso móvil "Contactar plan cuadrante", donde el actor puede realizar un contacto del número telefónico correspondiente a al plan cuadrante en la aplicación.

Contactar plan cuadrante			
ID.	UC07	Requisitos Asociados.	RF02
Descripción.	El Usuario emite una alarma al plan cuadrante mediante la aplicación.		
Actores.	Usuario.	Actores secundarios.	
Pre-condiciones.	Tener registrado en el grupo vecinal un usuario del tipo plan cuadrante. UC01	Post-condiciones.	Alarma emitida
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "Emitir una alarma" 2. Sistema muestra cuatro botones para que el usuario seleccione a quien va dirigida la alarma. 3. Usuario selecciona el icono del carabinero, que corresponde al plan cuadrante de carabineros. 4. Sistema envía la notificación con la geolocalización del usuario. 5. Sistema muestra mensaje de "Operación Exitosa". 		

Tabla 31 Especificación de casos de uso "Contactar plan cuadrante "

8.1.1.4.3. Contactar bomberos

En la Tabla 32 Especificación de casos de uso "Contactar Bomberos" se especifica el caso de uso móvil "Contactar bombero", donde el actor puede realizar un contacto del número telefónico correspondiente a un centro de bomberos en la aplicación.

Contactar bomberos			
ID.	UC08	Requisitos Asociados.	RF02
Descripción.	El Usuario emite una alarma a bomberos mediante la aplicación.		
Actores.	Usuario.	Actores secundarios.	
Pre-condiciones.	Tener registrado en el grupo vecinal un usuario del tipo bombero. UC01	Post-condiciones.	Alarma emitida
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema <i>"Emitir una alarma"</i> 2. Sistema muestra cuatro botones para que el usuario seleccione a quien va dirigida la alarma. 3. Usuario selecciona el botón <i>"bomberos"</i>, que corresponde a bomberos. 4. Sistema envía la notificación con la geolocalización del usuario. 5. Sistema muestra mensaje de <i>"Operación Exitosa"</i>. 		

Tabla 32 Especificación de casos de uso "Contactar Bomberos"

8.1.1.4.4. Contactar grupo vecinal:

En la Tabla 33 Especificación de casos de uso "Contactar grupo vecinal" se especifica el caso de uso móvil "Contactar vecinos", donde el actor puede realizar un contacto del número telefónico correspondiente a los vecinos en la aplicación.

Contactar grupo vecinal			
ID.	UC09	Requisitos Asociados.	RF02
Descripción.	El Usuario emite una alarma al plan cuadrante mediante la aplicación.		
Actores.	Usuario.	Actores secundarios.	
Pre-condiciones.	Estar registrado en algún grupo vecinal. UC01	Post-condiciones.	Alarma emitida
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "Emitir una alarma" 2. Sistema muestra cuatro botones para que el usuario seleccione a quien va dirigida la alarma. 3. Usuario selecciona el botón "pánico", que corresponde al grupo vecinal. 4. Sistema envía la notificación con la geolocalización del usuario. 5. Sistema muestra mensaje de "Operación Exitosa". 		

Tabla 33 Especificación de casos de uso "Contactar grupo vecinal"

8.1.1.4.5. Recibir alarma

En la Tabla 34 Especificación de casos de uso "Recibir alarma" se especifica el caso de uso móvil "recibir alarma", donde el actor puede recibir una notificación del contacto o usuario de que fue emitida en la aplicación.

Recibir alarma			
ID.	UC10	Requisitos Asociados.	RF02,
Descripción.	El Usuario registra modifica y elimina el contacto perteneciente a el centro médico mediante la aplicación.		
Actores.	Institución de seguridad o grupo vecinal.	Actores secundarios.	
Pre-condiciones.		Post-condiciones.	Contacto actualizado.
Flujo Básico.	<ol style="list-style-type: none"> 1. Sistema muestra un mensaje de alerta con ubicación geográfica. 2. Usuario acepta la notificación. 		

Tabla 34 Especificación de casos de uso "Recibir alarma"

8.1.1.5. Especificación de casos de uso del portal web

8.1.1.5.1. Modificar vecino

En la Tabla 35 Especificación del caso de uso “Modificar vecino se especifica el caso de uso “Modificar vecino”, donde el actor puede realizar la modificación de los datos de un vecino.

Modificar vecino			
ID.	UC11	Requisitos Asociados.	RFS01
Descripción.	El administrador puede modificar los datos correspondientes a un vecino que se encuentre registrado.		
Actores.	Administrador.	Actores secundarios.	
Pre-condiciones.	El vecino debe estar registrado. UC01	Post-condiciones.	Vecino modificado.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema “<i>modificar vecino</i>” 2. Sistema muestra el formulario que permite modificar los datos del vecino. 3. Usuario modifica los datos del formulario y presiona editar. 4. Sistema actualiza los datos. 		

Tabla 35 Especificación del caso de uso “Modificar vecino”

8.1.1.5.2. Eliminar vecino

En la Tabla 36 Especificación del caso de uso “Eliminar vecino” se especifica el caso de uso “Eliminar vecino”, donde el actor puede realizar la eliminación de los datos de un vecino.

Eliminar vecino			
ID.	UC12	Requisitos Asociados.	RFS02
Descripción.	El administrador puede eliminar los datos correspondientes a un vecino que se encuentre registrado.		
Actores.	Administrador.	Actores secundarios.	
Pre-condiciones.	El vecino debe estar registrado. UC01	Post-condiciones.	Vecino eliminado.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema “<i>eliminar vecino</i>” 2. Sistema elimina al vecino del sistema. 		

Tabla 36 Especificación del caso de uso “Eliminar vecino”

8.1.1.5.3. Ver vecino

En la Tabla 37 Especificación del caso de uso "Ver vecino" Tabla 35 Especificación del caso de uso "Modificar vecino se especifica el caso de uso "Ver vecino", donde el actor puede realizar la visualizar los datos de un vecino.

Ver vecino			
ID.	UC13	Requisitos Asociados.	RFS03
Descripción.	El administrador puede visualizar los datos correspondientes a un vecino que se encuentre registrado.		
Actores.	Administrador.	Actores secundarios.	
Pre-condiciones.	El vecino debe estar registrado. UC01	Post-condiciones.	
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "Ver vecino" 2. Sistema muestra los datos del vecino en pantalla. 		

Tabla 37 Especificación del caso de uso "Ver vecino"

8.1.1.5.4. Modificar institución de seguridad

En la Tabla 38 Especificación del caso de uso "Modificar institución de seguridad" se especifica el caso de uso "Modificar institución de seguridad", donde el actor puede realizar la modificación de los datos de una institución registrada.

Modificar institución de seguridad			
ID.	UC11	Requisitos Asociados.	RFS01
Descripción.	El administrador puede modificar los datos correspondientes a un vecino que se encuentre registrado.		
Actores.	Administrador.	Actores secundarios.	
Pre-condiciones.	La institución debe estar registrada. UC02	Post-condiciones.	Institución de seguridad modificada.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "modificar institución de seguridad" 2. Sistema muestra el formulario que permite modificar los datos de la institución. 3. Usuario modifica los datos del formulario y presiona editar. 4. Sistema actualiza los datos. 		

Tabla 38 Especificación del caso de uso "Modificar institución de seguridad"

8.1.1.5.5. Eliminar institución de seguridad

En la Tabla 39 Especificación del caso de uso " Eliminar institución de seguridad" se especifica el caso de uso "Eliminar institución de seguridad", donde el actor puede realizar la eliminación de los datos de una institución.

Eliminar institución de seguridad			
ID.	UC15	Requisitos Asociados.	RFS05
Descripción.	El administrador puede eliminar los datos correspondientes a una institución de seguridad que se encuentre registrado.		
Actores.	Administrador.	Actores secundarios.	
Pre-condiciones.	La institución debe estar registrado. UC02	Post-condiciones.	Institución de seguridad eliminada.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "eliminar institución de seguridad" 2. Sistema elimina la institución del sistema. 		

Tabla 39 Especificación del caso de uso " Eliminar institución de seguridad"

8.1.1.5.6. Ver institución de seguridad

En la Tabla 40 Especificación del caso de uso "Ver institución de seguridad" Tabla 35 Especificación del caso de uso "Modificar vecino se especifica el caso de uso "Ver institución de seguridad", donde el actor puede realizar la visualizar los datos de una institución.

Ver institución de seguridad			
ID.	UC16	Requisitos Asociados.	RFS06
Descripción.	El administrador puede visualizar los datos correspondientes a una institución de seguridad que se encuentre registrado.		
Actores.	Administrador.	Actores secundarios.	
Pre-condiciones.	El vecino debe estar registrado. UC01	Post-condiciones.	
Flujo Básico.	<ol style="list-style-type: none"> 3. Usuario solicita al sistema "Ver institución de seguridad" 4. Sistema muestra los datos de la institución en pantalla. 		

Tabla 40 Especificación del caso de uso "Ver institución de seguridad"

8.1.1.5.7. Registrar grupo

En la Tabla 41 Especificación del caso de uso "Registrar grupo" se especifica el caso de uso móvil "Registrar grupo", donde el actor puede registrar un grupo vecinal en la aplicación.

Registrar grupo			
ID.	UC17	Requisitos Asociados.	RFS07.
Descripción.	El administrador registra los datos de un nuevo grupo vecinal.		
Actores.	Administrador.	Actores secundarios.	
Pre-condiciones.		Post-condiciones.	grupo Registrado.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "Registrar junta de vecinos" 2. Sistema muestra el formulario que permite registrar los datos. 3. Usuario confirma el registro de los datos. 4. Sistema guarda los datos del nuevo registro. 		

Tabla 41 Especificación del caso de uso "Registrar grupo"

8.1.1.5.8. Modificar grupo

En la Tabla 42 Especificación del caso de uso "Modificar grupo" se especifica el caso de uso móvil "Modificar vecino", donde el actor puede realizar la modificación de los datos de un vecino.

Modificar grupo			
ID.	UC18	Requisitos Asociados.	RFS08
Descripción.	El administrador puede modificar los datos correspondientes a un grupo de vecinos que se encuentre registrado.		
Actores.	Administrador.	Actores secundarios.	
Pre-condiciones.	El grupo de vecino debe estar registrado. UC17	Post-condiciones.	Grupo vecinal modificado.
Flujo Básico.	5. Usuario solicita al sistema "modificar grupo" 6. Sistema muestra el formulario que permite modificar los datos del grupo. 7. Usuario modifica los datos del formulario y presiona editar. 8. Sistema actualiza los datos.		

Tabla 42 Especificación del caso de uso "Modificar grupo"

8.1.1.5.9. Eliminar grupo

En la Tabla 43 Especificación del caso de uso "Eliminar grupo" se especifica el caso de uso "Eliminar grupo", donde el actor puede realizar la eliminación de los datos de un grupo vecinal.

Eliminar grupo			
ID.	UC19	Requisitos Asociados.	RFS09
Descripción.	El administrador puede eliminar los datos correspondientes a un grupo vecinal que se encuentre registrado.		
Actores.	Administrador.	Actores secundarios.	
Pre-condiciones.	La grupo debe estar registrada. UC17	Post-condiciones.	Grupo vecinal eliminado.
Flujo Básico.	<ol style="list-style-type: none"> 1. Usuario solicita al sistema "eliminar junta de vecinos" 2. Sistema elimina la junta de vecinos del sistema. 		

Tabla 43 Especificación del caso de uso "Eliminar grupo"

8.1.2. Pruebas

- Contactar Bomberos.
- Contactar Plan Cuadrante.
- Contactar Consultorio.
- Contactar Grupo Vecinal.

8.1.2.1. Aspectos Generales.

En esta etapa de pruebas finales, y debido al escaso tiempo, las pruebas son realizadas a la funcionalidad de emitir notificaciones. Para esto se envían notificaciones a los distintos centros de emergencia y a los grupos de vecinos registrados.

Las técnicas y herramientas utilizadas al igual que en el incremento I, se realizaron pruebas de caja negra, donde solo se consideran los datos de entradas y se analizan los datos de salida.

Como configuración para realizar estas pruebas, solo se necesita un notebook o equipo de escritorio, con 2 GB de memoria RAM y 250GB de disco duro como mínimo, un sistema operativo Windows, MAC o Linux y finalmente lo más importante es que el equipo tenga acceso a internet.

8.1.3. Especificación de pruebas

unidad	referencia	Objetivo de las pruebas	Técnicas usadas	Criterio de cumplimiento
Contactar Plan cuadrante	UC09	Comprobar la notificación fue enviada al dispositivo móvil registrado como plan cuadrante.	Caja negra	100%
Contactar Consultorio	UC09	Comprobar la notificación fue enviada al dispositivo móvil registrado como consultorio.	Caja negra	100%
Contactar Bombero	UC09	Comprobar la notificación fue enviada al dispositivo móvil registrado como bombero.	Caja negra	100%
Contactar a grupos vecinales	UC09	Comprobar la notificación fue enviada a los dispositivos móviles registrados como grupos vecinales.	Caja negra	100%
Capturar Ubicación Geográfica	UC09	Comprobar que el usuario se encentra en la posición marcada por el mapa.	Caja negra	100%

Tabla 44 Especificación de pruebas

8.1.4. Casos de pruebas de caja negra

8.1.4.1. Caso de prueba: Contactar plan cuadrante

Caso de prueba: CPU08	
Nombre	Contactar plan cuadrante.
Descripción	Comprobar que se envía una notificación al usuario que fue registrado como plan cuadrante.
Tipo	Caja negra.
Precondiciones	Debe haber un usuario del tipo plan cuadrante registrado.
Postcondiciones	Notificación enviada exitosamente.
Entradas	
Referencias	UC09
Procedimiento de la prueba	
Actor	Sistema
1. El usuario seleccionar icono del carabinero.	2. El sistema emite notificación al dispositivo registrado como plan cuadrante.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 45 Especificación de la prueba "Contactar plan cuadrante"

8.1.4.2. Caso de prueba: Contactar a consultorio

Caso de prueba: CPU09	
Nombre	Contactar a consultorio.
Descripción	Comprobar que se envía una notificación al usuario que fue registrado como consultorio.
Tipo	Caja negra.
Precondiciones	Debe haber un usuario del tipo consultorio registrado.
Postcondiciones	Notificación enviada exitosamente.
Entradas	
Referencias	UC09
Procedimiento de la prueba	
Actor	Sistema
1. El usuario seleccionar icono de la enfermera.	2. El sistema emite notificación al dispositivo registrado como consultorio.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 46Especificación de prueba "Contactar a consultorio"

8.1.4.3. Caso de prueba: Contactar a bomberos

Caso de prueba: CPU10	
Nombre	Contactar a bomberos.
Descripción	Comprobar que se envía una notificación al usuario que fue registrado como bombero.
Tipo	Caja negra.
Precondiciones	Debe haber un usuario del tipo bombero registrado.
Postcondiciones	Notificación enviada exitosamente.
Entradas	
Referencias	UC09
Procedimiento de la prueba	
Actor	Sistema
1. El usuario seleccionar icono del bombero.	2. El sistema emite notificación al dispositivo registrado como bombero.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 47 Especificación de prueba "Contactar a bomberos"

8.1.4.4. Caso de prueba: Contactar a juntas de vecinos

Caso de prueba: CPU11	
Nombre	Contactar a juntas de vecinos.
Descripción	Comprobar que se envía una notificación a todos los usuario que fueron registrados como vecinos y en las mismas juntas de vecinos que el usuario en alerta.
Tipo	Caja negra.
Precondiciones	Debe haber una junta de vecinos ya registrada y con vecinos.
Postcondiciones	Notificaciones enviadas exitosamente.
Entradas	
Referencias	UC09
Procedimiento de la prueba	
Actor	Sistema
1. El usuario seleccionar icono donde sale mucha gente.	2. El sistema notifica a todo aquel que se encuentre registrado como vecino en los mismos grupos que el usuario en alerta.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 48 Especificación de prueba "Contactar a juntas de vecinos"

8.1.4.5. Caso de prueba: Ubicación geográfica

Caso de prueba: CPU12	
Nombre	Ubicación geográfica.
Descripción	Comprobar que al enviar la notificación se capture la posición actual del usuario en alerta.
Tipo	Caja negra.
Precondiciones	
Postcondiciones	Geolocalización exitosa.
Entradas	
Referencias	UC09
Procedimiento de la prueba	
Actor	Sistema
1. El usuario selecciona cualquier icono que le permita emitir una alarma.	2. Muestra la posición actual del usuario por medio de un mapa.
Resultado obtenido	
Cumple	Comentario
<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	

Tabla 49 Especificación de la prueba "Ubicación Geográfica"

8.1.5. Conclusiones de las pruebas

En síntesis, a lo largo de todo el desarrollo de este proyecto se aplicaron alrededor de 12 pruebas de caja negra a la aplicación orientadas a comprobar las distintas funcionalidades de la aplicación, dichas pruebas permitieron comprobar que se estaban enviando las notificaciones correctamente y dirigidas a quien correspondía según lo que necesitaba el usuario en alerta.

9. Conclusiones Generales

Una vez finalizado el proyecto Sistema de alarma vecinal denominado SOS es posible hacer una comparación entre los objetivos que se plantearon inicialmente versus los resultados obtenidos. A partir de esto, se sostiene lo siguiente:

- Se logró satisfacer todos los requerimientos propuestos al sistema y a su vez cumplir con los objetivos generales y específicos planteados. Aunque igual se realizaron ajustes durante la implementación del proyecto.
- La investigación de Ionic framework fue una tarea recurrente a lo largo del proyecto, ya que es una tecnología que relativamente nueva, debido a esta inexperiencia generaron diversos problemas, como lo fue el enviar notificaciones. Durante el desarrollo del proyecto al investigar las posibles soluciones para emitir notificaciones push, se encontró que muchas de ella ya estaban obsoletas, sin embargo, investigando se encontró el servicio Firebase Cloud Messaging, el cual permite enviar las notificaciones, utilizando un enrutamiento y enviando el mensaje al dispositivo de destino.
- Si bien la ejecución de las pruebas no pudo ser exhaustiva, las pruebas de caja negra que se alcanzaron a realizar no arrojaron fallas en el funcionamiento de la aplicación.
- Dado la colaboración de los niños con discapacidades del colegio Reyes de España y otros adultos mayores entrevistados, se pudo determinar la interfaz más óptima para todo tipo de usuarios y así mejorar significativamente la usabilidad de la aplicación.
- La implementación de los módulos fue una tarea ardua, considerando que el desarrollador no poseía experiencia con las tecnologías utilizadas. Sin embargo, los objetivos se consiguieron, a pesar de las variadas complejidades que surgieron, y se espera que este trabajo pueda ser utilizado y motive a los desarrolladores a realizar proyectos enfocados a mejorar el estilo de vida del adulto mayor, ya que cabe destacar que esta idea se originó a partir de la necesidad que tiene los adultos mayores para pedir ayuda en situaciones de emergencia de manera más expedita.

10. Trabajos futuros

A medida que se iba desarrollando el proyecto, fueron apareciendo nuevos detalles y nuevas ideas para mejorar el proyecto, sin embargo, por limitaciones de plazos, los cuales fueron establecidos por la Universidad del Bío-Bío, no se incorporaron en esta versión, postergando su implementación para próximas versiones o futuros proyectos similares basados en este.

Algunas de las ideas que surgieron fueron:

- Realizar una confirmación previa a emitir la alarma, en caso de que por equivocación se pase a emitir.
- Realizar un filtro de los grupos a de vecinos o familias a los que se le desea enviar la notificación.
- Enviar una notificación a más de una institución a la vez, dependiendo de la situación de emergencia.
- Lograr que la aplicación vaya actualizando la ubicación del usuario en alerta, luego de haber enviado la notificación.
- Lanzar la Aplicación al mercado a través de Google Play Store.
- Se implementó un portal web para la administración de la aplicación sin embargo, debido a que fue implementada con poco tiempo quedó abierta a modificaciones que podrían ser de mucha utilidad, tales como identificar los sectores más peligrosos de la ciudad.

Con respecto a otras actividades:

- Realizar un dispositivo aparte, una especie de botonera que sirva para emitir la alarma en la calle sin correr el riesgo de que le roben el Smartphone.

11. Bibliografía

- **Baca Urbina, G.**
Consultado 26/11/2016).
Evaluación de Proyectos. McGraw-Hill.
- **Bizagi.**
Consultado 20/08/2016.
BPMN (Business Process Modeling Notation).
<http://wiki.bizagi.com/es/index.php?title=BPMN>
- **Pressman, R. S.**
Consultado 20/09/2016.
Ingeniería de Software, un enfoque práctico. McGraw-Hill.
- **GitHub.**
Consultado 14/11/2016.
<https://github.com/hollyschinsky/PushNotificationSample>
- **balsamiq, m.**
Consultado 13/10/ 2016.
<https://balsamiq.com/products/mockups/>
- **beFunky.**
Consultado 25/10/2016.
<https://www.befunky.com/es/crear/>
- **Edraw Max.**
Consultado 23/10/2016.
<https://www.edrawsoft.com/es/download-edrawmax.php>

-
- **facilito, c.**
Consultado 16/09, 2016,
<https://www.youtube.com/playlist?list=PLpOqH6AE0tNhdnO11mOBthj4C70HdwQB2>
 - **framework, i.**
Consultado 21/09/2016.
<http://ionicframework.com/docs/components/>
 - **INE.**
Consultado 10/09/2016.
<http://www.ine.cl>
 - **ngCordova.**
Consultado 22/10/ 2016.
<http://ngcordova.com/docs/plugins/geolocation/>

ANEXOS

Anexo A: Capturas reales del sistema

En este anexo se presentan las capturas de la aplicación real, además de una breve descripción de cada una.

En la Ilustración A-A 1 Pantalla de inicio, emitir alarma y geolocalización se presenta la pantalla inicial que se visualiza al momento de abrir la aplicación y con la cual se pueden enviar las notificaciones, además, del mapa en donde se captura la posición del usuario.

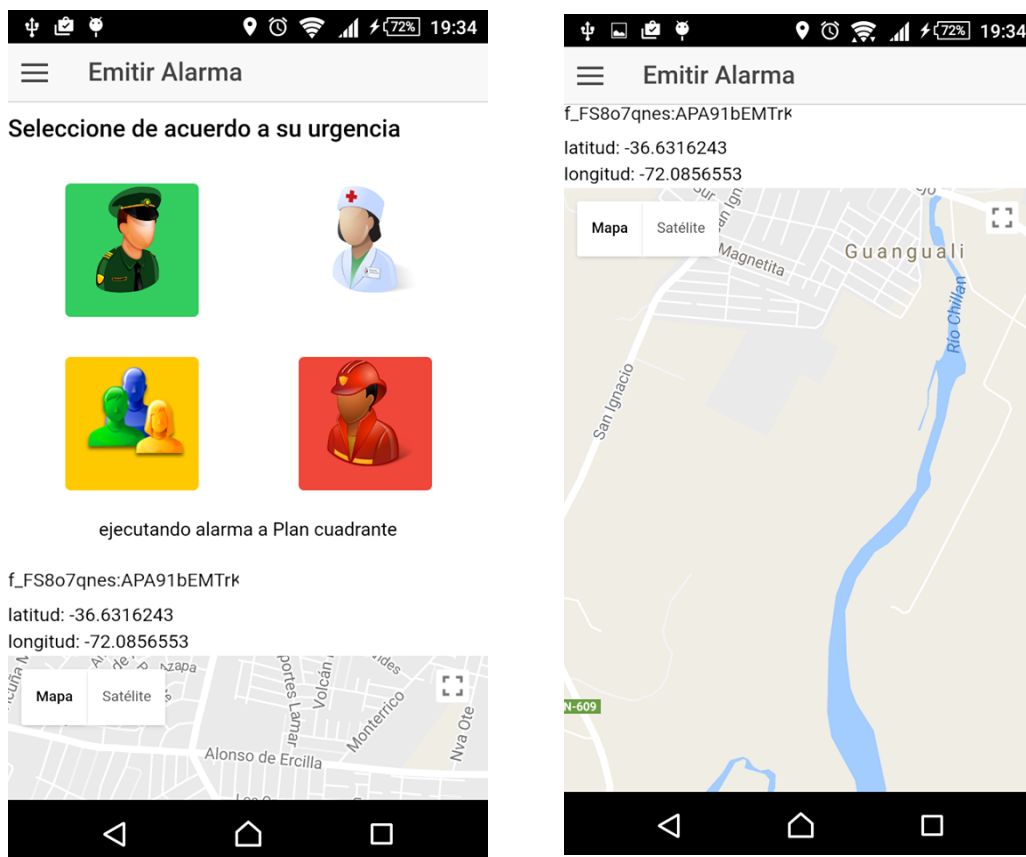


Ilustración A-A 1 Pantalla de inicio, emitir alarma y geolocalización

En la Ilustración A-A 2 Menú de la aplicación y formulario registro nueva Junta vecinal se muestran capturas de pantallas del menú de navegación de la aplicación, y el formulario para crear un nuevo grupo o junta de vecinos.

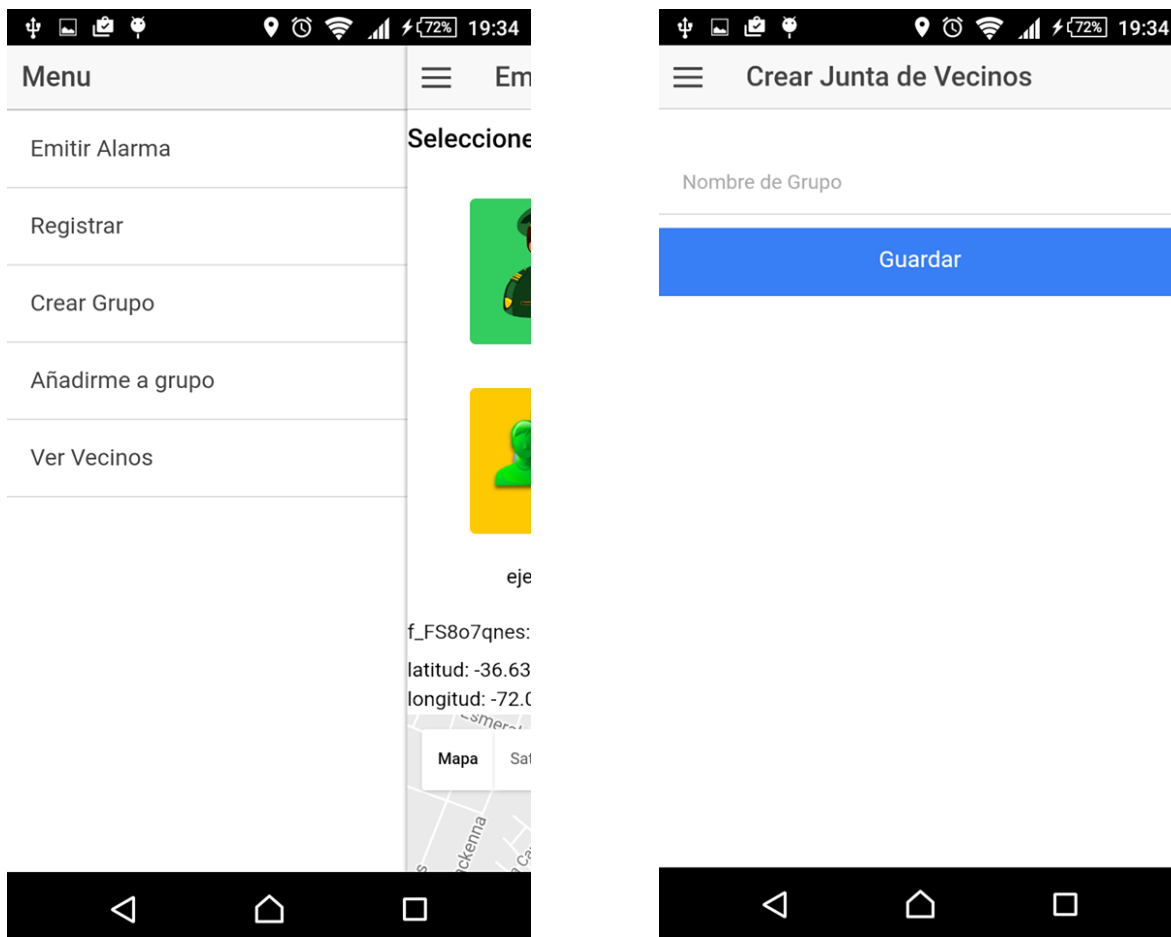


Ilustración A-A 2 Menú de la aplicación y formulario registro nueva Junta vecinal

En la Ilustración A-A 3 Formulario para registrarse en la aplicación móvil muestra las os posibles pantallas que tiene el usuario para registrar su dispositivo móvil en la aplicación, todo dependerá de que responda al señalar que si es vecino o no.

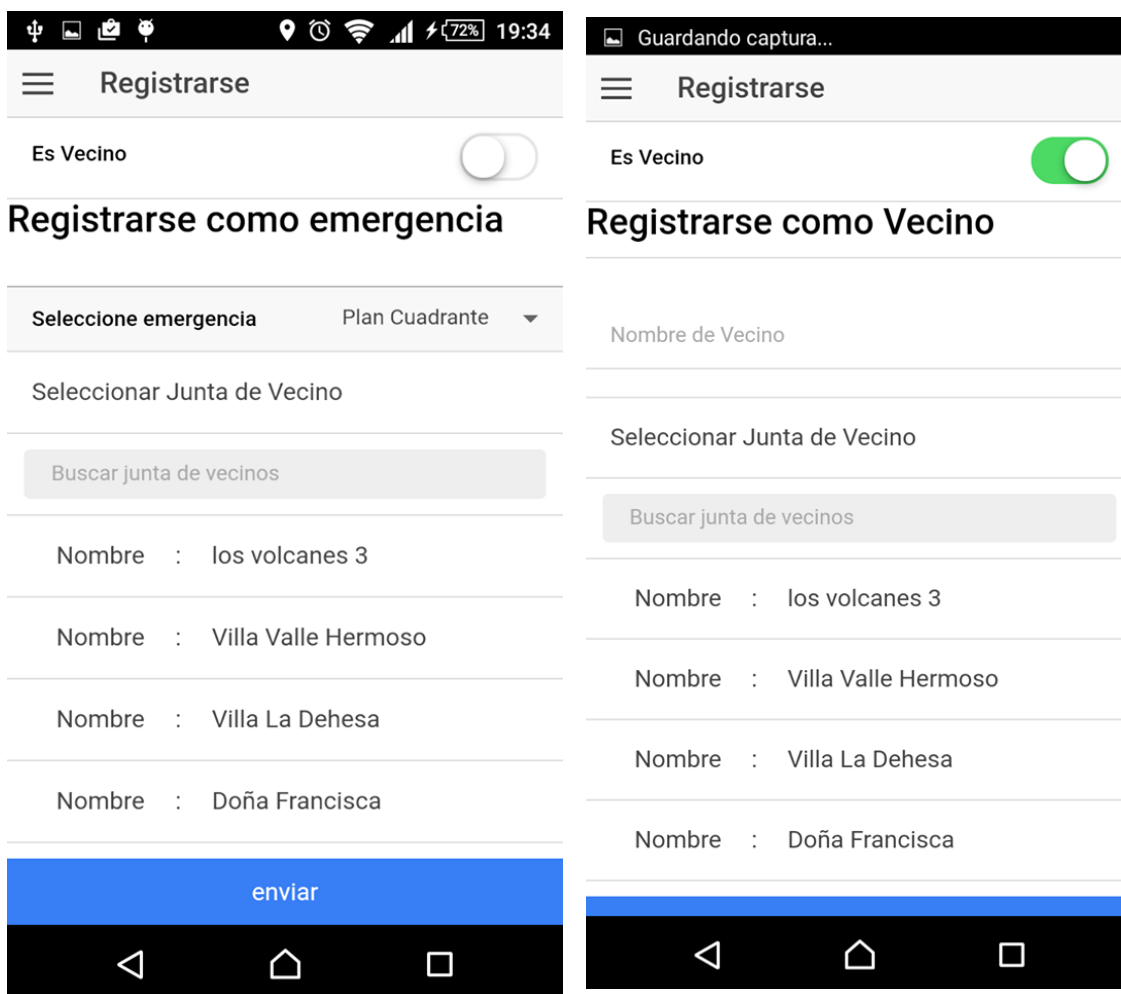


Ilustración A-A 3 Formulario para registrarse en la aplicación móvil

En la Ilustración A-A 4 Añadirse a Junta vecinal y ver junta vecinal muestra dos pantallas, la primera corresponde a la interfaz que permite al usuario incorporarse a un grupo vecinal ya registrado. La segunda pantalla corresponde a la vista en donde se listan todos los vecinos que se encuentran registrados en la misma junta vecinal que el usuario.

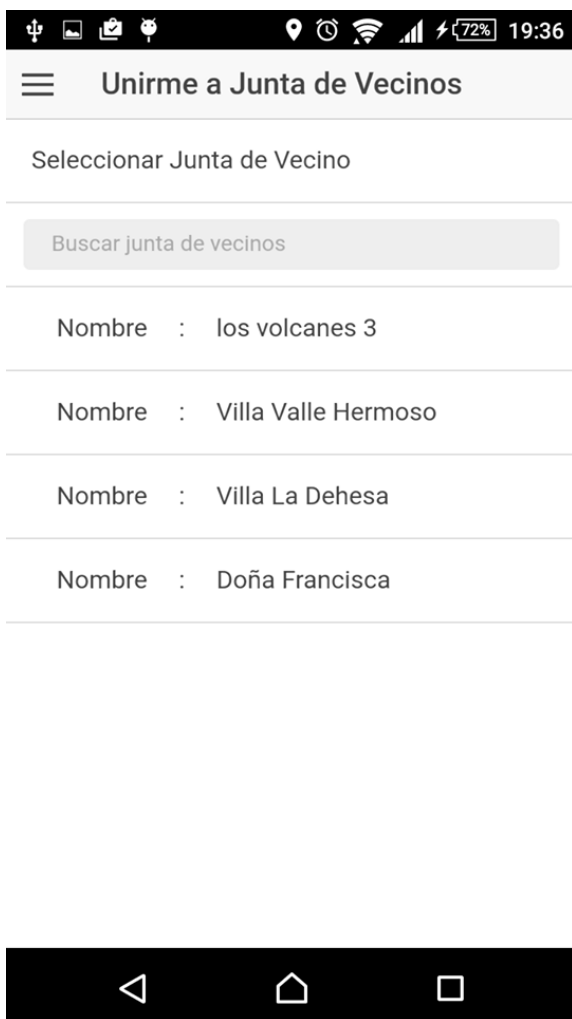


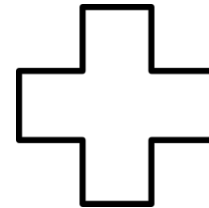
Ilustración A-A 4 Añadirse a Junta vecinal y ver junta vecinal

Anexo B: Encuesta de Interfaz

Indique que representa cada una de las imágenes para usted



1.- _____



2.- _____



3.- _____



4.- _____



5.- _____



6.- _____



7. _____



8.- _____



9.- _____



10.- _____



11.- _____



12.- _____

