



UNIVERSIDAD DEL BÍO-BÍO
FACULTAD DE CIENCIAS EMPRESARIALES
ESCUELA DE INGENIERÍA CIVIL INFORMÁTICA

“Sistema de información para Android que permita conocer lugares y/o descripción, de los distintos espacios de la Universidad del Bío-Bío, Campus Fernando May”.

Por Gustavo Adolfo San Martín Guerrero.

Para optar al título de Ingeniero Civil en Informática.

Profesor guía: Marlene Elena Muñoz Sepúlveda.

Propuesta original por: Sylvia Marcela Pinto Fernández.

Noviembre de 2018.

Chillán, Chile.

Agradecimientos.

A continuación, quiero agradecer a las siguientes personas que me han ayudado y apoyado tanto en este proyecto de tesis como durante el proceso universitario.

Agradecer a mi familia que ha sido un pilar fundamental, ya que me han apoyado y entendido durante toda mi vida y más en este último proceso, que es muy importante, permitiendo finalmente continuar hasta el final y no rendirme.

Agradecer también, a mis amigos Cristian Araya y Jermis Mendoza y compañeros que han sido muy importantes en el proceso universitario y vida en general, puesto que, al divertirnos, compartir, salir y distrayéndonos un rato de las obligaciones, generan energía y motivación para seguir con los estudios.

Finalmente, agradecer a Macarena Carrasco que ha sido muy importante durante este último tiempo, ya que me ha brindado mucho amor y cariño, nunca aburriéndose y esperándome siempre, por lo cual me genera, des estrés, energía y posterior motivación para seguir adelante con los estudios y actualmente en el proyecto de tesis.

Resumen

Este proyecto se presenta para dar conformidad a los requisitos exigidos por la Universidad de Bío-Bío en el proceso de titulación para a la carrera Ingeniería Civil en Informática.

El proyecto titulado “Sistema de información para Android que permita conocer lugares y/o descripción, de los distintos espacios de la Universidad del Bío-Bío, Campus Fernando May”, busca difundir de una mejor manera los diferentes lugares y espacios que tiene la Universidad del Bío-Bío, hablese de aulas, departamentos, escuelas, edificios, etcétera, cuyo propósito es poder lograr ubicar a todas las personas que no conozcan el establecimiento. Para lo anterior se presenta el desarrollo de una aplicación móvil, de carácter informativo, que permite mostrar un plano geográfico de la Universidad del Bío-Bío con los lugares más importantes con que cuenta. La aplicación indica un lugar del establecimiento, solicitado por el usuario, mostrando una ruta de acceso desde la entrada de la Universidad hasta dicho lugar, la distancia, el tiempo aproximado de llegada a pie y bicicleta, y una opción para mostrar información relevante e importante del lugar solicitado. Se menciona también, que la aplicación cuenta con un sistema de posicionamiento global (GPS), para facilitar la búsqueda y la llegada de la persona hacia el lugar.

Para realizar la aplicación, se trabaja con el IDE Android Studio para la plataforma Android, en la que principalmente se hace uso de la API de Google que ofrece un mapa interactivo para explorar el entorno, la posibilidad de establecer marcadores dentro de éste, diferentes rutas de acceso, distancias, ubicación actual del dispositivo que se usa, entre otras funcionalidades. También se trabaja con el hosting gratuito 000webhost el cual ofrece una base de datos MySQL con phpMyAdmin, un administrador de archivos por lo cual se usa Filezilla que es el gestor FTP, etcétera, todo esto para poder almacenar los diferentes espacios que la Universidad del Bío-Bío tiene.

Finalmente, se logra un software de aplicación para teléfonos móviles con sistema operativo Android, que incluye todos los requisitos funcionales y no funcionales solicitados por el cliente, de forma que pueda ser usada por todas las personas

(estudiantes, docentes, apoderados, personas externas, etc.) que deseen encontrar y/o conocer en profundidad la Universidad del Bío-Bío, Campus Fernando May.

Abstract

This project is presented to give accordance to the requirements demanded by the Universidad del Bío-Bío in the process of titulation for the career of Ingeniería Civil en Informática.

The project titled “System information to Android that allows know places and/or descriptions, of the differents spaces of the Universidad del Bío-Bío, Campus Fernando May”, seeks diffuse of a better way the differents places and spaces that the Universidad del Bío-Bío has, speak of classrooms, departments, schools, buildings, and so on, whose purpose is can achieve locate to all the peoples that do not know the establishment. To the previous the development of a mobile application is presented, of informative character, that allows show a geographic plane of the Universidad del Bío-Bío with the most important places it has. The application indicate a place of the establishment, requested by the user, showing a route of access from the entry of the University until said place, the distance, the approximate time of arrival to foot and bicycle, and an option to show information relevant and important of the requested place. It is also mentioned that the application has a global positioning system (GPS), to facilitate the seeks and the arrival of the person until the place.

to perform the application, Android Studio IDE is worked for the Android platform, in the that mainly it is used of the Google API that offers an interactive map to explore the environment, the possibility of stablish markers inside of this, differents access routes, distances, actual locate of the device that is used, between other functionalities. Also free hosting 000webhost is used, wich offers a MySQL data base with phpMyAdmin, a file manager whereby Filezilla is used that is the FTP manager, and so on, all this to can store the differents spaces that Universidad del Bío-Bío has.

Finally, it is achieved an application’s software to mobile phones with Android operative system, that includes all the functional requirements and non-functional requirements requested for the customer, of way that can be used for all the peoples

(students, teachers, parents, external peoples, etc.) who want find and/o know in depth the Universidad del Bío-Bío, Campus Fernando May.

Índice.

2 Descripción del problema.....	16
2.1 Características organizacionales y/o del área de estudio.....	16
2.1.1 Misión.....	16
2.1.2 Visión.....	17
2.1.3 Estructura organizativa.....	17
2.1.4 Otras características organizacionales	18
2.2 Ámbito del problema.	18
2.2.1 Problemas actuales encontrados.....	18
3 Definición proyecto.....	20
3.1 Objetivos del proyecto.	20
3.1.1 Objetivo general.....	20
3.1.2 Objetivos específicos.....	20
3.2 Ambiente de ingeniería de software.....	21
3.2.3 Herramienta para la administración de proyectos.....	22
3.3 Especificación de requerimientos.....	30
3.4 Tecnologías para el desarrollo de la aplicación.....	31
3.5 Herramientas para el desarrollo de la aplicación.....	32
3.6 Técnicas y notaciones.....	33
3.7 Librerías, patrones y otros.....	34
4 Caracterización del software.....	35
4.1 Descripción global del producto.	35
4.2 Objetivo del Software.	36
4.3 Interfaces de la aplicación.....	36
4.3.1 Interfaz de usuario.....	37
4.3.2 Interfaz de hardware.....	38
4.3.3 Interfaz de software.....	39
4.4 Requerimientos del sistema.....	40
4.4.1 Requisitos funcionales.....	40
4.4.2 Requisitos no funcionales.....	42
4.5 Atributos del producto.	42

4.5.1 Atributos de calidad para usuarios.....	42
4.5.1.1 Disponibilidad.....	42
4.5.1.2 Eficiencia.....	43
4.5.1.3 Flexibilidad.....	43
4.5.1.4 Integridad.....	43
4.5.1.5 Confiabilidad.....	43
4.5.1.6 Robustez.....	43
4.5.1.7 Usable.....	44
4.5.2 Atributos de calidad para desarrolladores.....	44
4.5.2.1 Mantenimiento.....	44
4.5.2.2 Portabilidad.....	44
4.5.2.3 Reusabilidad.....	44
4.5.2.4 Corrección.....	45
5 Estudio de factibilidad.....	46
5.1 Factibilidad técnica.....	46
5.1.1 Lenguajes y herramientas a utilizar.....	46
5.1.2 Hardware.....	47
5.2 Factibilidad operacional.....	47
5.3 Factibilidad económica.....	47
5.3.1 Beneficios.....	48
5.3.2 Calculo del VAN.....	49
6. Modelamiento del sistema.....	51
6.1 Modelo y notación de procesos de negocio.....	51
6.2 Modelos de entidad-relación.....	53
6.3 Diseño de base de datos.....	57
6.4 Arquitectura utilizada.....	59
6.5 Interfaz y navegación.....	60
6.5.1 Jerarquías de menú.....	61
6.5.1.1 Jerarquías de menú para usuario.....	61
6.5.1.2 Jerarquías de menú para administrador.....	62
6.5.2 Interfaz de usuario.....	63
6.6 Casos de uso.....	72

7 Pruebas de Sistema.	77
7.1 Elementos y Planificación de pruebas.	77
7.1.1 Pruebas de los requisitos funcionales.	78
7.2 Conclusiones de las pruebas efectuadas.	86
8 Planes de implantación y puesta en marcha del sistema.	88
9 Plan de capacitación a usuarios.	89
10 Conclusiones.	90
11 Referencias bibliográficas.	92
12 anexos.	95
Anexo A. Planificación inicial vs. Esfuerzo real invertido.	95
Anexo B. Especificación de pruebas de software.	98
Anexo C Manual usuario.	105

Índice figuras.

Figura 1. Organigrama de la Universidad del Bío-Bío.....	17
Figura 2. Tablero de tareas en la herramienta Trello	23
Figura 3. Tablero de tareas en la herramienta Trello	23
Figura 4. Tablero de tareas en la herramienta Trello	24
Figura 5. Tablero de tareas en la herramienta Trello	24
Figura 6. Tablero de tareas en la herramienta Trello	25
Figura 7. Tablero de tareas en la herramienta Trello	25
Figura 8. Tablero de tareas en la herramienta Trello	26
Figura 9. Tablero de tareas en la herramienta Trello	27
Figura 10. Tablero de tareas en la herramienta Trello	28
Figura 11. Tablero de tareas en la herramienta Trello	29
Figura 12. Colores usados en la aplicación móvil	37
Figura 13. Logotipo aplicación Android	37
Figura 14. Características dispositivo físico para el desarrollo Android	38
Figura 15. Fórmula para el cálculo del VAN	50
Figura 16. Fórmula para el cálculo del TIR	50
Figura 17. Business Process Model and Notation (BPMN).....	53
Figura 18. Modelo de entidad-relación (MER).....	54
Figura 19. Modelo de datos	57
Figura 20. Base de datos en phpMyAdmin	58
Figura 21. Diagrama modelo-vista-controlador	59
Figura 22. Jerarquía de menú de un usuario	61
Figura 23. Jerarquía de menú de un administrador	62

Figura 24. Pantalla de inicio	63
Figura 25. Login administrador	64
Figura 26. Menú principal	64
Figura 27. Activación GPS cuando está off	65
Figura 28. Menú administrador.....	65
Figura 29. Configuración administrador	66
Figura 30. Creando marcador	66
Figura 31. Marcador creado	67
Figura 32. Actualizando marcador.....	67
Figura 33. Confirmar actualización.....	68
Figura 34. Marcador actualizado	68
Figura 35. Confirmar borrar marcador.....	69
Figura 36. Marcador eliminado	69
Figura 37. Visualización marcadores.....	70
Figura 38. Ruta y tiempo de llegada.....	70
Figura 39. Información de un lugar	71
Figura 40. Diagrama de casos de uso.....	72
Figura 41. Planificación inicial y final en Carta Gantt	95
Figura 42. Esfuerzo real invertido en Carta Gantt.....	96

Índice tablas.

Tabla 1. Requisitos funcionales	41
Tabla 2. Inversión proyecto	48
Tabla 3. Gastos proyecto	48
Tabla 4. Beneficios proyecto.....	49
Tabla 5. Tasa de descuento y años de recuperación de la inversión	49
Tabla 6. Flujo de caja	50
Tabla 7. Entidades del modelo de datos	55
Tabla 8. Atributos del modelo de datos.....	56
Tabla 9. Relaciones del modelo de datos	56
Tabla 10. Caso de uso: Desplazarse por el mapa	73
Tabla 11. Caso de uso: Consultar ubicación	73
Tabla 12. Caso de uso: Consultar Descripción	74
Tabla 13. Caso de uso: Crear marcador	74
Tabla 14. Caso de uso: Editar marcador	75
Tabla 15. Caso de uso: Borrar Marcador	75
Tabla 16. Caso de uso: Cambiar usuario o contraseña.....	76
Tabla 17. Resultado pruebas efectuadas	86
Tabla 18. Prueba de sistema: Desplazarse por el mapa.....	98
Tabla 19. Prueba de sistema: Consultar ubicación	99
Tabla 20. Prueba de sistema: Consultar descripción	100
Tabla 21. Prueba de sistema: Crear marcador.....	101
Tabla 22. Prueba de sistema: Editar marcador.....	102
Tabla 23. Prueba de sistema: Borrar marcador.....	103
Tabla 24. Prueba de sistema: Cambiar usuario y contraseña.....	104

1 Introducción.

Hoy en día el uso de celulares móviles ha aumentado considerablemente día a día debido a la gran importancia que estos poseen, siendo una necesidad importante para los muchos usuarios que usan estos dispositivos, debido a la variada cantidad de funciones que estos traen, como la comunicación a distancia, para entretención, conectividad a internet, acceso a datos e información, entre otras funcionalidades.

Debido a la gran cantidad de personas que usan estos dispositivos, han salido al mercado una infinidad de marcas y modelos, generando competencia entre los que lo proveen; por lo mismo la facilidad de adquirir un dispositivo móvil es mucho mayor, ya que los precios son más baratos y asequibles para los usuarios finales. Además, estos dispositivos cuentan con diferentes sistemas operativos con distintas características que hacen que sus clientes lo escojan. Es así que han surgido dos exponentes importantes en relación a sistemas operativos, que logran que sus fabricantes, operadores y desarrolladores den mayor funcionalidad a sus teléfonos celulares. Ambos exponentes son iOS e Android, siendo este último el sistema operativo con mayores ventas dentro del mercado, logrando alcanzar un aproximado de 81.7%, contra un iOS que alcanza un 17.9% en ventas para los usuarios finales (Ziff Davis, 2018).

Por lo anterior, para poder abarcar la mayor cantidad de usuarios, la Universidad del Bío-Bío ha solicitado la creación de una aplicación para Android que permita conocer en detalle las diferentes instalaciones y espacios que posee. La finalidad de esta aplicación es tratar de mejorar a gran escala, el sistema "Tour Virtual", que la Universidad del Bío-Bío tiene en su página oficial, que permite hacer un recorrido del establecimiento en un diseño de tres dimensiones.

En el presente informe se da a conocer en detalle la empresa para la cual está dirigido este proyecto, como su importancia, relevancia y participación en este proceso de desarrollo. También este documento abarca el proceso completo de desarrollo de la aplicación, como los objetivos principales, se especifica los requerimientos que debe llevar, las herramientas utilizadas necesarias para su

desarrollo, pruebas del sistema, se muestra las interfaces, etc. Finalmente se menciona que se hace un estudio y análisis completo sobre la factibilidad, para comprobar si el producto realizado puede ser exitoso y viable en el futuro.

En el capítulo 2 se presenta información general y particular de la empresa, así como el problema actual con el que cuenta.

En el capítulo 3 se define el proyecto a partir de, los objetivos generales y específicos, los requerimientos, la metodología de desarrollo, herramientas para administración y desarrollo, tecnologías para el desarrollo, técnicas y notaciones, librerías y patrones, entre otros.

En el capítulo 4 se describe globalmente el producto a partir de, el objetivo del software, las interfaces de la aplicación y los requisitos funcionales y no funcionales.

En el capítulo 5 se presenta un estudio completo de la factibilidad del producto y se menciona los lenguajes y herramientas para su desarrollo.

En el capítulo 6 se presenta el modelamiento completo del sistema desarrollado a partir de, el modelo y notación de procesos de negocio, el modelo de entidad-relación, el diseño de la base de datos, la interfaz y navegación, la arquitectura utilizada y los casos de uso.

En el capítulo 7 se presenta las pruebas realizadas al sistema y se utiliza las pruebas de caja negra para verificar las funcionalidades sin tomar en cuenta la estructura interna de código.

En el capítulo 8 se presenta los planes de implantación y puesta en marcha del sistema para poner a disposición de los usuarios.

En el capítulo 9 se menciona el plan de capacitación a usuarios para el uso correcto y adecuado de la aplicación.

En el capítulo 10 se presenta las conclusiones personales y del desarrollo del proyecto y aplicación.

En el capítulo 11 se presenta las referencias bibliográficas que muestra de donde se extrae la información.

En el capítulo 12 se presenta los anexos el cual incluye, la planificación inicial versus el esfuerzo invertido, la especificación de las pruebas de software y el manual de usuario para el uso de la aplicación.

2 Descripción del problema.

En este capítulo se presenta en detalle información particular y general de la empresa para la cual va dirigido este proyecto, así como el ámbito de la problemática actual que logra que se implemente esta aplicación.

2.1 Características organizacionales y/o del área de estudio.

Nombre	:	Universidad del Bío-Bío Campus Fernando May Didier.
RUT	:	60.911.006-6.
Representante legal	:	Rector, Dr. Héctor Guillermo Gaete Féres.
Dirección	:	Avda. Andrés Bello 720.
Sitio Web	:	www.ubiobio.cl
Fono	:	+56-422463400.

2.1.1 Misión.

La Universidad del Bío-Bío, a partir de su naturaleza pública, responsable socialmente y estatal, tiene por misión, desde las Regiones del Biobío y Ñuble, aportar a la sociedad con la formación de personas integrales, a través de una Educación Superior de excelencia. Comprometida con los desafíos regionales y nacionales, contribuye a la movilidad e integración social por medio de la generación y transferencia de conocimiento avanzado, mediante la docencia de pregrado y postgrado de calidad, la investigación fundamental, aplicada y de desarrollo, la vinculación bidireccional con el medio, la formación continua y la extensión. Asimismo, impulsa el emprendimiento y la innovación, el fortalecimiento de la internacionalización y el desarrollo sustentable de sus actividades, basada en una cultura participativa centrada en el respeto a las personas (Bío-Bío, 2018).

2.1.2 Visión.

Ser reconocida a nivel nacional e internacional como una Universidad pública, responsable socialmente y regional que, comprometida con su rol estatal, desde las Regiones del Biobío y Ñuble, forma personas integrales de excelencia y aporta a través de su quehacer al desarrollo sustentable de las regiones y el país (Bío-Bío, 2018).

2.1.3 Estructura organizativa.

En la figura 1, se muestra la estructura orgánica de la Universidad del Bío-Bío.

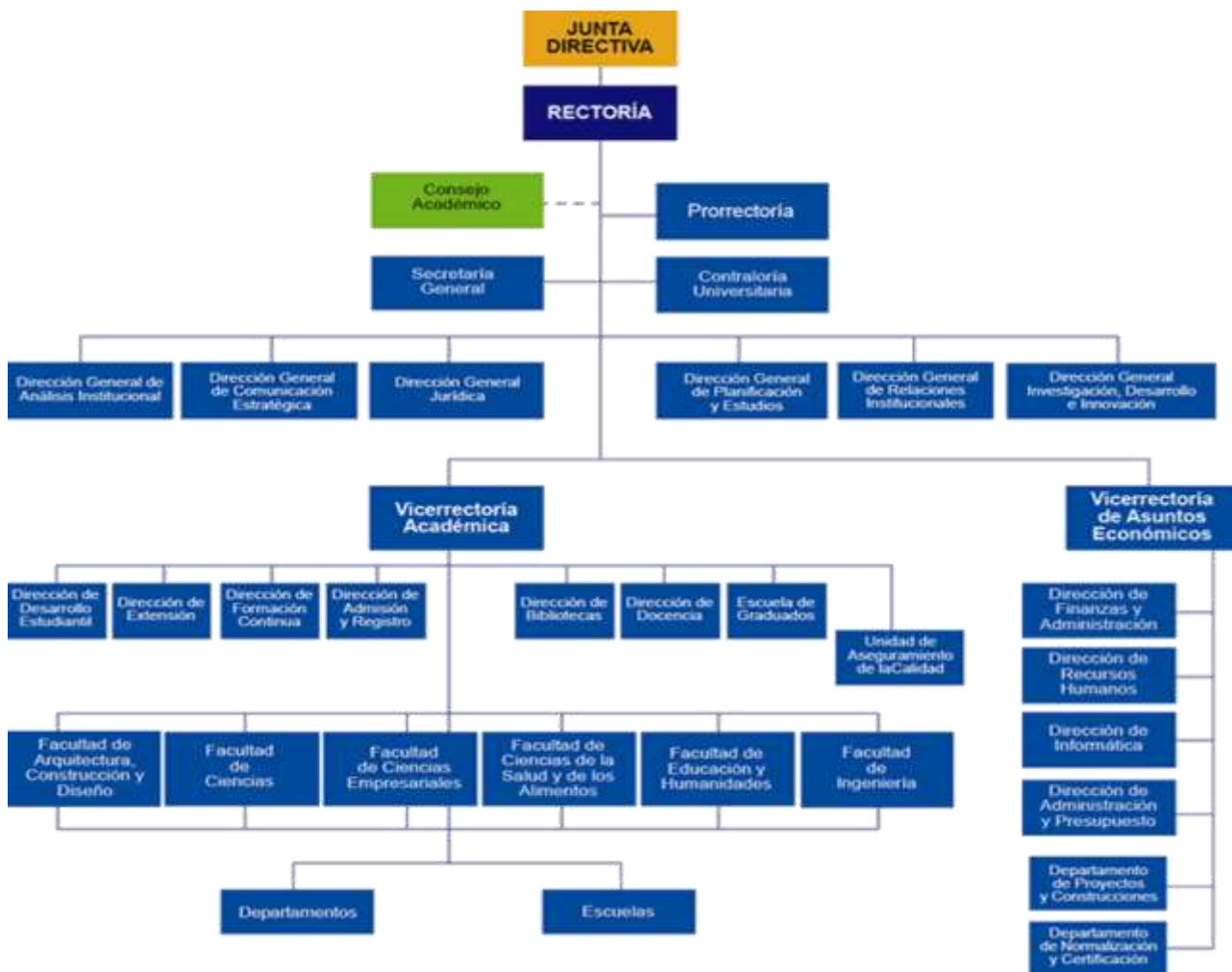


Figura 1. Organigrama de la Universidad del Bío-Bío (Bío-Bío, ubiobio, 2018).

2.1.4 Otras características organizacionales

Como información relevante para este proyecto, actualmente la Universidad del Bío-Bío, en el Campus Fernando May cuenta con 12 carreras universitarias con una cantidad aproximada de 4500 estudiantes, en la que el establecimiento pone a disposición a estas carreras y alumnos, aulas, gimnasios, camarines, laboratorios de física, química y biología, laboratorios de computación, salas de estudio, bibliotecas, amplias zonas de recreación y esparcimiento, casinos y cafetería, entre otros (Bío-Bío, ubiobio, 2018). Aproximadamente serían más de 35 instalaciones que el establecimiento dispone (Bío-Bío, ubiobio, 2018).

2.2 Ámbito del problema.

En diferentes oportunidades, en el interior del establecimiento, se puede percatar que algunas personas realizan consultas sobre la ubicación de una oficina, biblioteca, cafetería, museos, aulas, entre otras, en la que en ocasiones no se logra encontrar con una respuesta confiable, lo que provoca que lleguen atrasados a compromisos importantes o quizás nunca se encuentre el lugar buscado, perdiendo el tiempo y en ocasiones probablemente sus citas.

2.2.1 Problemas actuales encontrados

1.- La Universidad del Bío-Bío actualmente dispone de una escasa señalización sobre la ubicación de sus diferentes espacios, edificios e instalaciones, ocasionando claramente que las personas que no conocen el establecimiento y/o sus alrededores, les dificulte poder llegar a su destino o bien no conozcan la información necesaria sobre que se hace en ese lugar.

2.- No existe difusión alguna con respecto a los lugares más importante del establecimiento; más solo cuando estudiantes de la Universidad trabajan

promocionando las carreras y los lugares con los que cuenta el establecimiento a los alumnos de enseñanza media de algunos colegios de Chillán y de regiones.

3.- Se destaca, que las páginas de la Universidad del Bío-Bío, cuenta con información sobre algunos lugares del establecimiento, sin embargo, es escasa en comparación a la cantidad de instalaciones que posee; más aún las personas no revisan estos sitios cuando necesitan encontrar la ubicación de un lugar o bien conocer información relevante sobre el mismo.

4.- Google Maps muestra algunos espacios o lugares de la Universidad del Bío-Bío, pero es extremadamente escasa. Además, si uno desea puede agregar sitios a este servidor de aplicaciones de mapas en la web, sin embargo, por algún motivo no lo hacen.

3 Definición proyecto.

En este capítulo se presenta la definición del proyecto, con los objetivos generales y específicos, así como los requerimientos, que en conjunto permite guiar el desarrollo del proyecto en cuestión y de la respectiva aplicación.

3.1 Objetivos del proyecto.

En el siguiente capítulo, se presenta los objetivos generales y específicos del proyecto, que se deben cumplir al momento de ser desarrollado.

3.1.1 Objetivo general.

Desarrollar una aplicación móvil para la Universidad del Bío-Bío, Campus Fernando May que permita conocer la ubicación y descripción de los diferentes lugares y edificios que se encuentren dentro del establecimiento, así como también la ubicación actual (mediante GPS) de un usuario al momento de iniciada la aplicación.

3.1.2 Objetivos específicos.

- 1.- Implementar un plano geográfico que identifique la Universidad del Bío-Bío, Campus Fernando May y muestre marcadores en los lugares y edificios con los que cuenta, así como, las rutas, la distancia y los tiempos de demora a pie y bicicleta desde la entrada del establecimiento hasta una ubicación solicitada.
- 2.- Implementar una barra de búsqueda, en la que el usuario pueda ingresar un lugar del establecimiento, para luego ser visualizado dentro del mapa.
- 3.- Implementar un sistema de posicionamiento global (GPS) que permita conocer la posición actual de la personal al iniciar la aplicación.

4.- Implementar diferentes ventanas o interfaces que muestren información importante y relevante del lugar por el cual se ha consultado.

3.2 Ambiente de ingeniería de software.

Las metodologías se caracterizan por hacer énfasis cara a cara, es decir, se basan en una fuerte y constante interacción, donde clientes y desarrolladores trabajan constantemente juntos, estableciéndose así una estrecha comunicación. Estas metodologías están orientadas al resultado del producto, lo cual exige que el proceso sea adaptable, permitiendo realizar cambios a último momento (EcuRed, 2018).

En esta oportunidad se elige la metodología Kanban que es un método para gestionar el trabajo intelectual, con énfasis en la entrega justo a tiempo. Kanban es una aproximación al proceso gradual, evolutivo y al cambio de sistemas para las organizaciones. Utiliza un sistema de extracción limitada del trabajo en curso como mecanismo básico para exponer los problemas de funcionamiento del sistema (o proceso) y estimular la colaboración para la mejora continua del sistema (Fundación Wikimedia, WikipediA, 2018).

Kanban se ha convertido en una técnica muy utilizada por los equipos de desarrollo ágil de software, ya que permite centrarse en qué hay que entregar y cuándo, estimulando la colaboración y evitando hacer cosas innecesarias. En este caso se trabaja de manera individual y es por lo mismo que se ha adaptado para que pueda ser utilizada por una sola persona. A esta adaptación la han llamado Kanban Personal.

Kanban personal es un sistema flexible y tiene dos reglas:

1.- Visualizar el trabajo: Visualizar el trabajo permite tenerlo bajo control. Puesto que es difícil comprender bien todo aquello que no se ve, se necesita registrar todas las tareas que el proyecto demanda a cada momento. De esta forma se puede tomar mejores decisiones y priorizar en función del contexto en que uno se encuentre.

2.- Limitar el trabajo en curso: No se debe hacer más cosas de las que uno es capaz de hacer. Limitando el número de tareas que se puede tener en progreso a la vez, se puede concentrar mejor en lo que se está haciendo y responder tranquilamente a los cambios (SÁEZ, 2018).

3.2.3 Herramienta para la administración de proyectos.

Para la administración de proyectos se trabaja con la herramienta Trello, el cual es un software de administración de proyectos con interfaz web, cliente para iOS y Android para organizar proyectos.

Empleando el sistema Kanban, para el registro de actividades con tarjetas virtuales organiza tareas, permite agregar listas, adjuntar archivos, etiquetar eventos, agregar comentarios y compartir tableros.

Trello es un tablón virtual en el que se puede colgar ideas, tareas, imágenes o enlaces. Es versátil y fácil de usar pudiendo usarse para cualquier tipo de tarea que requiera organizar información (WikipediA, 2018).

A continuación, se presenta el proceso completo de la metodología de desarrollo Kanban a través del tablero virtual Trello.

Se debe mencionar que, el proceso contempla 10 etapas, en las cuales se van realizando las tareas, se agregan nuevas funcionalidades y se corrigen tareas (ver figura 8 y 9).

La figura 2 muestra una lista de las tres tareas principales para el inicio del desarrollo del proyecto y aplicación.

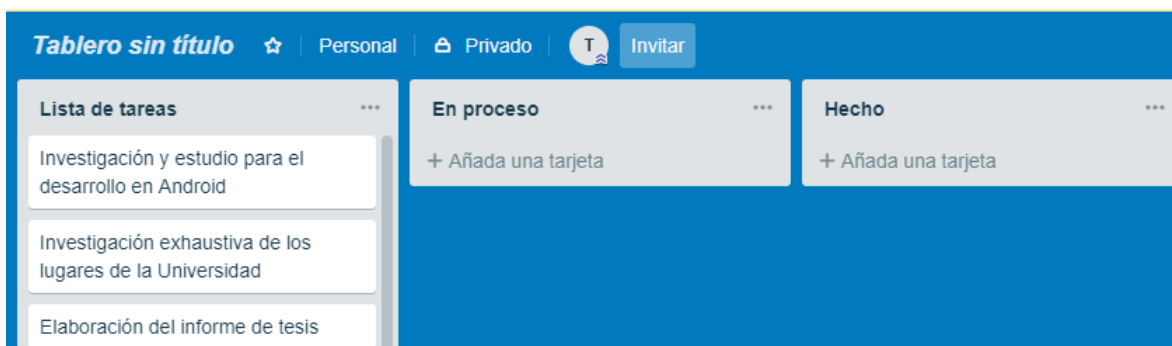


Figura 2. Tablero de tareas en la herramienta Trello (ATLASSIAN, 2018).

La figura 3 muestra que las tres tareas principales ahora están en proceso y a medida que se avanza en el tiempo, se agregan nuevas tareas para el proyecto y aplicación.

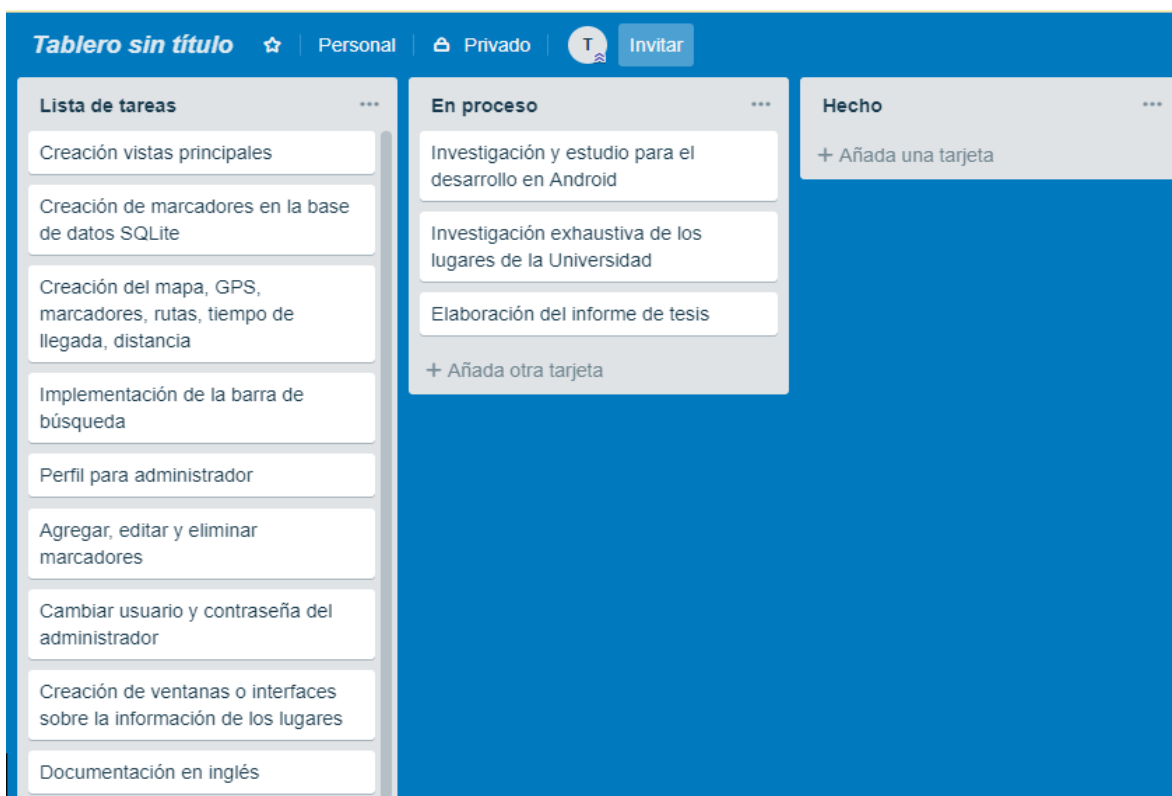


Figura 3. Tablero de tareas en la herramienta Trello (ATLASSIAN, 2018).

La figura 4 añade dos tareas en proceso y se contempla el término de una de las tareas principales, agregadas al inicio del proceso de desarrollo.

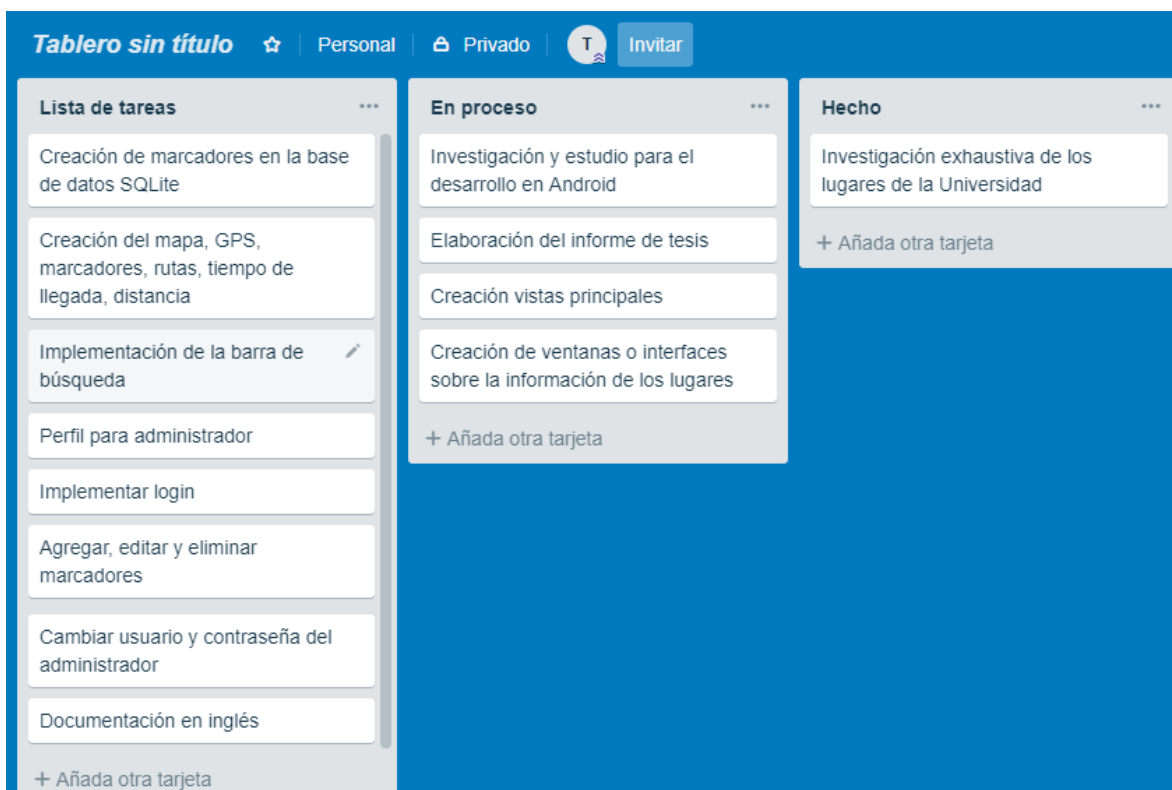


Figura 4. Tablero de tareas en la herramienta Trello (ATLASSIAN, 2018).

La figura 5 representa que a medida que avanza el tiempo, se van desarrollando y completando las tareas añadidas en un comienzo.



Figura 5. Tablero de tareas en la herramienta Trello (ATLASSIAN, 2018).

La figura 6 al igual que la figura 5 muestra que se completan algunas tareas y se van desarrollando las que faltan.

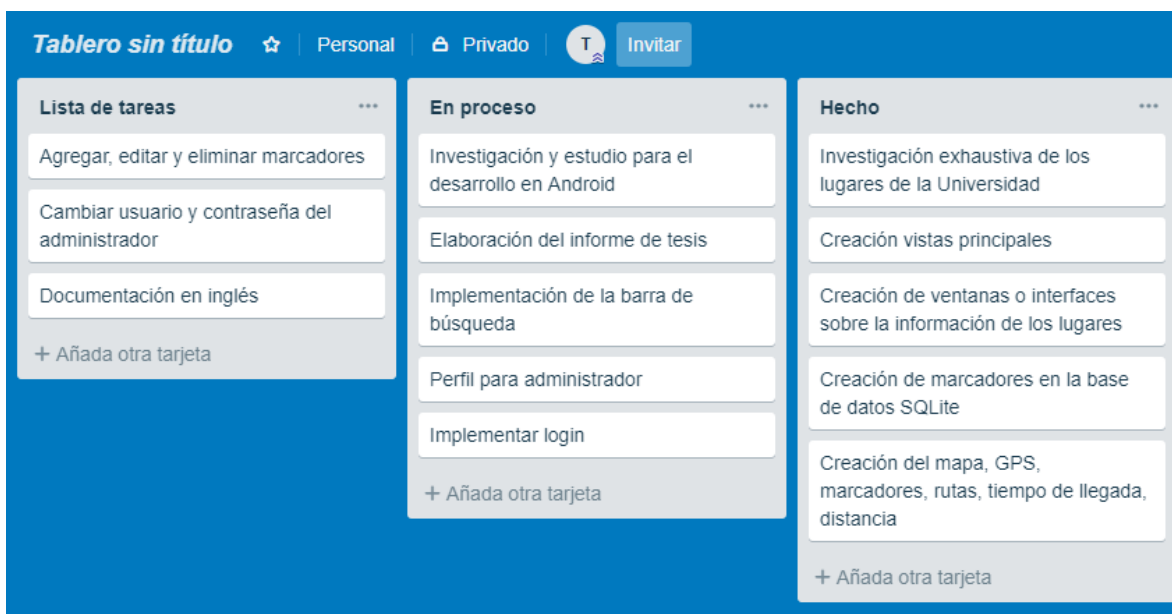


Figura 6. Tablero de tareas en la herramienta Trello (ATLASSIAN, 2018).

La figura 7 evidencia que no falta demasiado para desarrollar completamente todas las tareas añadidas en este proceso de desarrollo del proyecto y aplicación.



Figura 7. Tablero de tareas en la herramienta Trello (ATLASSIAN, 2018).

La figura 8 muestra una vez más que se ha completado otra tarea, quedando menos para el término del proyecto y aplicación.

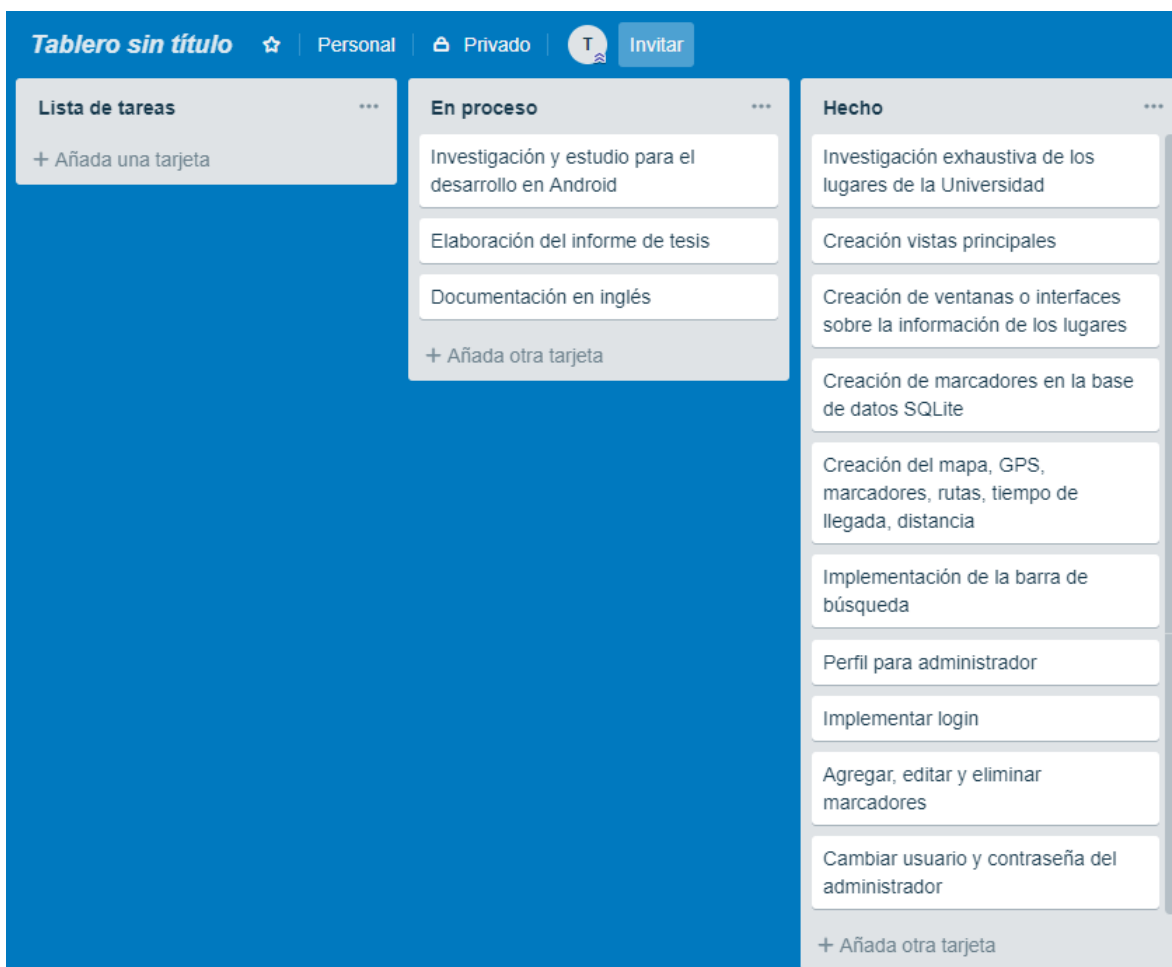


Figura 8. Tablero de tareas en la herramienta Trello (ATLASSIAN, 2018).

Con respecto a la figura 9 se debe mencionar que, a medida que se iba estudiando y desarrollando para la aplicación, sucedió que era más desafiante realizar la aplicación de una forma remota y no local como se pensó en un comienzo. Esto último quiere decir que, era mejor para fines de aprendizaje almacenar los marcadores en una base de datos remota y no local. Por lo tanto, se elimina la tarea completada “creación de marcadores en la base de datos SQLite” y se añaden nuevas tareas en proceso.

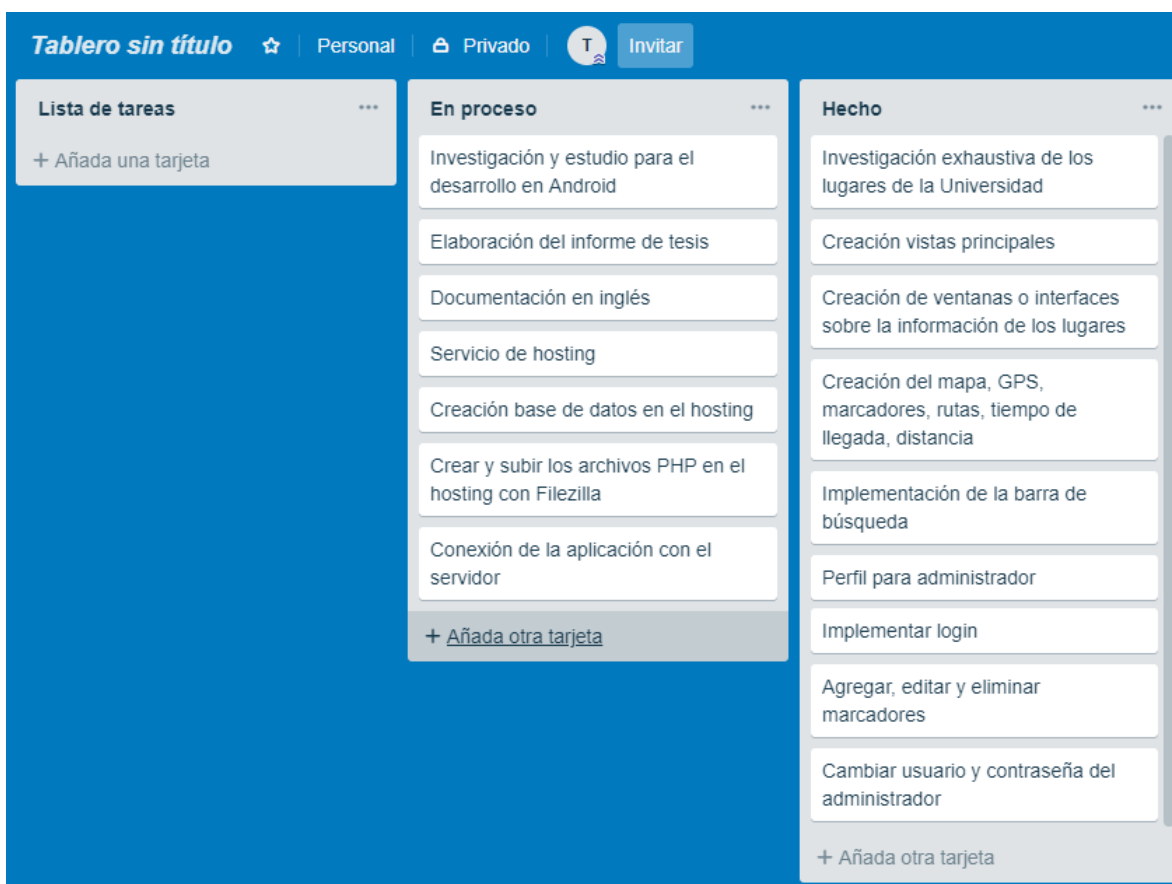


Figura 9. Tablero de tareas en la herramienta Trello (ATLASSIAN, 2018).

La figura 10 muestra que se han completado la mayoría de las tareas con respecto a la aplicación, más solo falta la elaboración del informe y la documentación correspondiente al código de la aplicación móvil.



Figura 10. Tablero de tareas en la herramienta Trello (ATLASSIAN, 2018).

La figura 11 muestra el término completo de las tareas que se han añadido inicialmente para el desarrollo de la aplicación y se da fin a la metodología de desarrollo Kanban a través de la herramienta Trello.

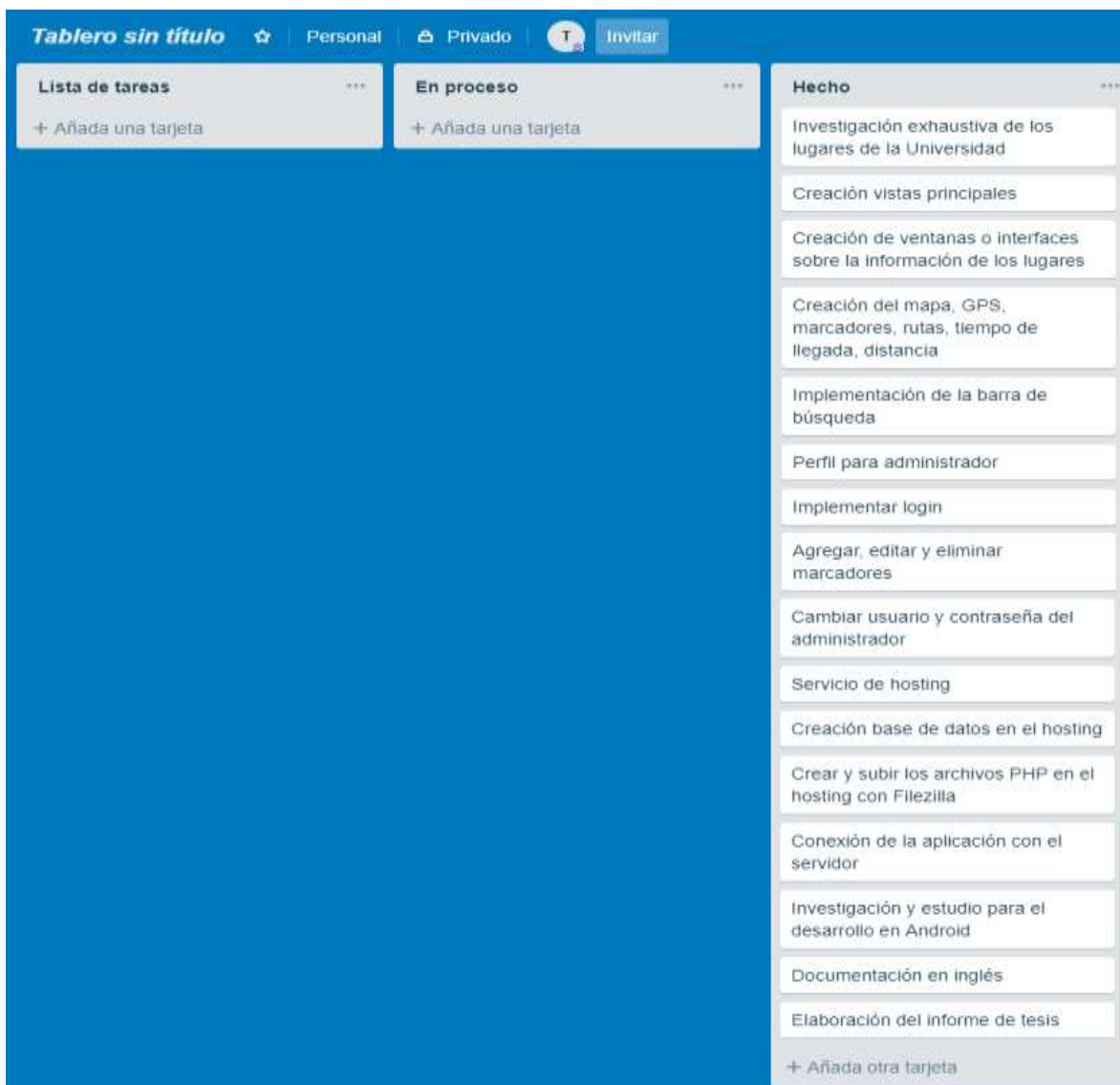


Figura 11. Tablero de tareas en la herramienta Trello (ATLASSIAN, 2018).

Finalmente, se puede decir que, el uso de la metodología Kanban fue de mucha importancia, ya que permite realizar las tareas de una forma clara y ordenada, además de permitir poder corregir y/o añadir nuevas tareas por si existe algún problema o se desea cambiar el desarrollo de algunas funcionalidades.

3.3 Especificación de requerimientos.

1.- Plano UBB: La aplicación debe contener el plano geográfico del establecimiento, para una mayor visualización y usabilidad de la aplicación.

2.- Mostrar lugares: La aplicación debe ser capaz de mostrar cada uno de los lugares del establecimiento, mediante marcas que identifiquen que allí se encuentra un edificio, una instalación o un espacio recreativo importante.

3.- Buscador UBB: La aplicación debe contar con un buscador para que los usuarios puedan encontrar más fácilmente los lugares que buscan, ya sea de algún edificio en particular o algún área específica.

4.- GPS: La aplicación debe contener un sistema de posicionamiento global (GPS), para que los usuarios puedan saber dónde están ubicados en ese momento y así poder seguir las trayectorias que se van generando al momento de consultar por una ubicación.

5.- Rutas, distancia y tiempo de llegada: la aplicación debe proveer de una ruta de acceso desde la entrada principal (hacia cabina de seguridad) y entrada secundaria (hacia diseño y psicología) hasta la ubicación solicitada, mostrando, además, la distancia en metros, así como el tiempo en minutos y segundos que tarda en llegar tanto en bicicleta como caminando.

6.- Descripciones detalladas: El sistema debe contar con descripciones detalladas de los lugares o edificios que existen en el campus, esto quiere decir mostrar información sobre que se hace en ese lugar, que ayuda, facilidades, materiales y/o servicios provee a las personas y/o alumnos, horarios de atención del lugar si corresponde y horarios de atención de secretarías y profesores jefes de las carreras universitarias.

7.- Administrador: La aplicación debe contener un panel de administrador para que él pueda crear, editar y/o borrar las marcas de los lugares en el mapa. Cabe destacar que el administrador debe estar registrado en la base de datos y debe poder acceder mediante su usuario y contraseña.

3.4 Tecnologías para el desarrollo de la aplicación.

En el siguiente capítulo se presentan las diferentes tecnologías o aplicaciones que se utiliza para el desarrollo de la aplicación móvil.

1.- Sublime Text: Es un editor de texto y editor de código fuente está escrito en C++ y Python para los plugins. Desarrollado originalmente como una extensión de Vim, con el tiempo fue creando una identidad propia, por esto aún conserva un modo de edición tipo vi llamado Vintage mode (Fundación Wikimedia, WikipediA, 2018).

Se puede descargar y evaluar de forma gratuita. Sin embargo, no es software libre o de código abierto y se debe obtener una licencia para su uso continuado, aunque la versión de evaluación es plenamente funcional y no tiene fecha de caducidad (Fundación Wikimedia, WikipediA, 2018).

2.- Notepad++: Es un editor de texto y de código fuente libre con soporte para varios lenguajes de programación. Con soporte nativo para Microsoft Windows.

Se parece al Bloc de notas en cuanto al hecho de que puede editar texto sin formato y de forma simple. No obstante, incluye opciones más avanzadas que pueden ser útiles para usuarios avanzados como desarrolladores y programadores (Fundación Wikimedia, WikipediA, 2018).

3.- Android Studio: Es el entorno de desarrollo integrado (IDE) indispensable para crear y mantener una aplicación móvil. El uso de este software es gratuito y está disponible para el público en general mediante su descarga en la web (Fundación Wikimedia, WikipediA, 2018).

4.- Java Development Kit (JDK): Es un software que provee herramientas de desarrollo para la creación de programas en Java. Puede instalarse en una computadora local o en una unidad de red (Fundación Wikimedia, WikipediA, 2018).

5.- PdaNet +: Es una de las mejores aplicaciones de Android de todos los tiempos. Comparte el acceso a Internet de su teléfono Android con su computadora o tableta. Funciona en todos los teléfonos Android sin rooting. Además, admite la conexión

mediante WiFi, USB Tether o Bluetooth DUN. No hay límite de velocidad (Inc. F. T., PdaNet, 2018).

6.- JSON Formatter: Fue creado para ayudar con la depuración. Como los datos JSON a menudo se envían sin saltos de línea para ahorrar espacio, es extremadamente difícil leerlos y darle sentido. Esta pequeña herramienta esperaba resolver el problema al formatear los datos JSON para que los seres humanos los puedan leer y depurar fácilmente (Concept, 2018).

7.- 000webhost: Es un sistema de hosting gratis, con soporte DNS, con panel, acceso FTP, 1500 MB de espacio Web, PHP, MySQL, entre otros (Mis Negocios En Internet, 2018).

8.- phpMyAdmin: Herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando internet. Actualmente puede leer, crear, modificar y eliminar Base de datos, tablas y campos, ya sea manualmente o con sentencias SQL (Fundación Wikimedia, WikipediA, 2018).

9.- Filezilla: Es una aplicación para la transferencia de archivos por FTP. Es una aplicación gratuita y de código abierto. Permite transferir archivos desde una computadora local, hacia uno o más servidores FTP (y viceversa) de forma sencilla (ALEGSA, 2018).

10.- JMeter: Es un proyecto de Apache que puede ser utilizado como una herramienta de prueba de carga para analizar y medir el desempeño de una variedad de servicios, para aplicaciones web y móviles (Fundación Wikimedia, WikipediA, s.f.).

3.5 Herramientas para el desarrollo de la aplicación.

En el siguiente capítulo se presentan las diferentes herramientas o lenguajes utilizados para el desarrollo de la aplicación móvil.

1.- Java: Es un lenguaje de programación de propósito general, concurrente, orientado a objetos, que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que

los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo, lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra (Fundación Wikimedia, WikipediA, 2018).

2.- Cascading Style Sheets (CSS): Es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Es muy usado para establecer el diseño visual de los documentos web, e interfaces de usuario escritas en HTML o XHTML (Fundación Wikimedia, WikipediA, 2018).

3.- eXtensible Markup Language (XML): Es un meta-lenguaje que permite definir lenguajes de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible (Fundación Wikimedia, WikipediA, 2018).

4.- Structured Query Language (SQL): Es un lenguaje específico del dominio utilizado en programación, diseñado para administrar sistemas de gestión de bases de datos relacionales (Fundación Wikimedia, WikipediA, 2018).

5.- Hypertext Preprocessor (PHP): Lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico (Fundación Wikimedia, WikipediA, 2018). Se utiliza en este caso para realizar las conexiones con la base de datos en phpMyAdmin.

3.6 Técnicas y notaciones.

En el siguiente capítulo se presentan las diferentes técnicas y notaciones para lograr un buen modelado de los datos generados por este proyecto y aplicación móvil.

1.- Business Process Model and Notation (BPMN): Es una notación gráfica estandarizada que permite el modelado de procesos de negocio, en un formato de flujo de trabajo (workflow) (Fundación Wikimedia, WikipediA, 2018).

2.- Unified Modeling Language (UML): Es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el Object Management Group (OMG).

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema (Fundación Wikimedia, WikipediA, 2018).

3.- Diagrama de casos de uso: Es un diagrama de casos de uso es una forma de diagrama de comportamiento UML mejorado (Fundación Wikimedia, WikipediA, 2018).

4.- Modelo Entidad Relación (MER): Es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información, así como sus interrelaciones y propiedades (Fundación Wikimedia, WikipediA, 2018).

3.7 Librerías, patrones y otros.

En el siguiente capítulo se presenta librerías, patrones, entre otros, que se utiliza para lograr obtener una buena y completa aplicación Android.

1.- Singleton: Es un patrón de diseño que permite restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto. Su intención consiste en garantizar que una clase solo tenga una instancia y proporcionar un punto de acceso global a ella (Fundación Wikimedia, WikipediA, 2018). En este caso se utiliza para evitar realizar por cada clase o actividad una petición al servidor web, sino que las peticiones las haga solo una clase.

2.- JavaScript Object Notation (JSON): Formato de texto ligero para el intercambio de datos (Fundación Wikimedia, WikipediA, 2018). En este caso las respuestas a las peticiones a la base de datos son devueltas en formato JSON y así poder trabajarlas de fácil manera en la aplicación Android a través de Volley.

3.- Volley: Es un cliente Http creado que facilita la comunicación de red en las aplicaciones Android. Volley está enfocado totalmente a las peticiones, evitándola creación de código repetitivo para manejar tareas asíncronas por cada petición o incluso para parsear los datos que vienen del flujo externo (Programación, 2018).

4 Caracterización del software.

A continuación, en el siguiente capítulo se presenta las características relevantes que tendrá el software a desarrollar.

4.1 Descripción global del producto.

1.- El producto a desarrollar es una aplicación móvil para dispositivos con sistema operativo Android, el cual sea capaz de mostrar el mapa del mundo, con un acercamiento visual hacia la Universidad del Bío-Bío Campus Fernando May.

2.- La aplicación debe mostrar a través de marcadores los diferentes lugares y espacios más importantes y relevantes que el establecimiento posea; hablese de relevantes o importantes los lugares que sean más visitados por la comunidad UBB. Además, los lugares mostrados tienen que ser aquellos que puedan ayudar de alguna forma a todas las personas y más a los estudiantes del establecimiento, que sean un aporte para ellos, para sus estudios y formaciones.

3.- La aplicación debe ser capaz de mostrar las diferentes rutas, distancias y tiempos de tardanza que se van generando a la hora de realizar las distintas consultas sobre las ubicaciones de los lugares del establecimiento. Además, la aplicación debe trabajar con los permisos de GPS, que muestre la posición actual del usuario y así poder encontrar sin ningún problema el lugar por el cual ha buscado.

4.- La aplicación debe tener un perfil para administrador, para que sea capaz de poder crear, editar y eliminar los lugares a través de los marcadores en el mapa.

5.- La aplicación debe generar diferentes ventanas o interfaces para los distintos lugares de la Universidad, en el cual sean capaces de mostrar información que oriente a sus alumnos con sus estudios como, por ejemplo, horarios de atención de docentes, de préstamos de material (libros, salas de estudios, computadores, entre otros), así como también donde se puede convalidar y/o botar ramos, elevación de

solicitudes, prestamos de dinero, pagos de matrículas, talleres deportivos, entre otras informaciones relevantes para la comunidad.

4.2 Objetivo del Software.

Se menciona que, la aplicación aparte de ser para los estudiantes, también puede ser ocupada por docentes, apoderados, gente de otras universidades o bien gente externa al establecimiento que desee encontrar un lugar específico o saber que realizan allí.

Por lo anterior el objetivo principal del software de aplicación móvil es facilitar a todas las personas a encontrar el lugar que buscan dentro del establecimiento, evitando que se pierdan o se retiren del establecimiento por no encontrar lo que buscaban. También, la aplicación tiene como objetivo ayudar a las personas a llegar a la hora a sus compromisos importantes, evitando que lleguen atrasados o bien lo pierdan.

Finalmente, se menciona que, la aplicación no solo sirve para ver los lugares de la Universidad del Bío-Bío, además se puede usar con el objetivo de encontrar lugares de la ciudad de Chillán o bien del mundo entero.

4.3 Interfaces de la aplicación.

En el siguiente capítulo se presenta las interfaces de usuario, hardware y software que posee el desarrollo de esta aplicación móvil.

4.3.1 Interfaz de usuario.

Se menciona que, con respecto a los colores, se utiliza aquellos que facilita la usabilidad y legibilidad en la aplicación, sin embargo, se destaca los colores característicos tanto de la Universidad del Bío-Bío, como la de Google Maps.

A continuación, se muestra la figura 12 con algunos de los colores utilizados en el desarrollo de la aplicación.

```

1      <?xml version="1.0" encoding="utf-8"?>
2      <resources>
3
4      <color name="rosado">#FF4081</color>
5      <color name="negro">#000000</color>
6      <color name="verde">#1ce192</color>
7      <color name="azul">#0171c6</color>
8      <color name="blanco">#ffffff</color>
9      <color name="rojo">#f41313</color>
10     <color name="amarillo">#fcaa08</color>
11     </resources>
12

```

Figura 12. Colores usados en la aplicación móvil.

Con respecto al logotipo de la aplicación se ha hecho uso del logo de la Universidad del Bío-Bío con el nombre de “Ubicate UBB”.

En la figura 13 se presenta el logotipo y el nombre de la aplicación, visto desde un dispositivo físico.



Figura 13. Logotipo aplicación Android.

Con respecto al diseño de los textos escritos en la aplicación, se utiliza un estilo de texto normal o en algunos casos en negrita, con un tamaño en letra de 14 (bold y 15sp en notación Android Studio). Además, algunos textos y los botones, están escritos de tal manera que el usuario pueda entender de manera intuitiva que actividad se realiza.

4.3.2 Interfaz de hardware.

La aplicación debe funcionar en dispositivos móviles con sistema operativo Android mayor o igual a 5.0 (Lollipop) que cubre más del 60% del uso global en clientes, y en el desarrollo se utiliza para que no existan errores con algunas de las funciones a programar.

Para la emulación en tiempo real en el desarrollo de la aplicación se puede instalar y crear un dispositivo virtual, sin embargo, en este caso se utiliza un dispositivo físico con características que se muestra en la figura 14.

Características

Pantalla LCD 5.5" táctil (1280x720)
Memoria interna 8 GB
RAM 1 GB
Cámara principal Estándar 13.0 MP
Cámara frontal Estándar 8.0 MP
Almacenamiento externo microSD
Batería 2910 mAh
Sistema operativo Android 5.1 "Lollipop"

Características avanzadas

Procesador MediaTek MT6582 (1300 MHz)
CPU ARM Cortex A7 (Quad core / 1300 MHz)
GPU ARM Mali 400 MP2
Peso 168 g.
Tamaño 78 x 151 x 9 mm
Conectividad
 2G GSM: 850 / 900 / 1800 / 1900 MHz
 3G HSDPA: B1 (2100) / B2 (1900 PCS) / B5 (850) / B8 (900 GSM) MHz

Figura 14. Características dispositivo físico para el desarrollo Android (SOLOTUDO, 2018).

4.3.3 Interfaz de software.

Para el desarrollo de la aplicación se ha descargado Android Studio versión 3.1, para lo cual se descarga Java 8 y Java Development Kit para su correcto funcionamiento.

Se puede instalar Android Virtual Device para emular en un dispositivo móvil virtual lo que se ha desarrollado en la aplicación hasta ese momento.

Para el desarrollo y funcionamiento correcto de la aplicación móvil, se instala los componentes de Google Play Services para usar los servicios de Google, en este caso para usar Google Maps. También se instala el componente de Google USB Driver para emular en un dispositivo físico lo que se está programando en ese momento.

Se utiliza el hosting gratuito 000webhost, para gestionar una base de datos MySQL y almacenar los archivos con Filezilla.

Además, se trabaja con phpMyAdmin que maneja la administración de MySQL y así poder crear la base de datos que lleva las instalaciones que posee la Universidad. Por lo mismo se utiliza el lenguaje de programación PHP para realizar las conexiones pertinentes entre la base de datos y la aplicación Android.

Finalmente, se usa la librería Volley de Android que es un cliente Http creado para facilitar la comunicación de red en las aplicaciones Android y que está enfocado en las peticiones, evitando la creación de código repetitivo para manejar las tareas asíncronas por cada petición.

4.4 Requerimientos del sistema.

En el siguiente capítulo se presentan los requerimientos solicitados por el cliente, la profesora Marcela Pinto, en el cual, el desarrollador de la aplicación ha agendado solamente una cita, para posteriormente captar y deducir a través de la conversación los siguientes requerimientos descritos en la tabla 1.

4.4.1 Requisitos funcionales.

ID	Nombre	Descripción
01	Visualización plano geográfico.	La aplicación debe contener un plano geográfico que muestre la Universidad con sus instalaciones y alrededores.
02	Menú principal.	Antes de comenzar con la visualización del mapa, la aplicación debe contar con un menú principal que muestre información relevante para su uso y de un botón que dé inicio a dicho mapa.
03	Sistema de posicionamiento global.	La aplicación debe contar con los permisos de GPS, que muestre la posición actual del usuario que usa la aplicación. En caso de estar desactivado, la aplicación debe permitir al usuario poder activar la posición.
04	Visualización marcadores.	Se debe colocar marcadores a los diferentes lugares e instalaciones que la Universidad del Bío-Bío tiene.
05	Visualización ruta.	La aplicación debe generar las diferentes rutas que van desde la

		entrada del establecimiento hasta la ubicación por la cual se ha consultado.
06	Visualización distancia.	La aplicación debe mostrar la distancia en metros de las rutas generadas.
07	Visualización tiempo de tardanza.	La aplicación debe mostrar el tiempo en minutos y segundos de tardanza con respecto a las rutas generadas, tanto a pie como en bicicleta
08	Barra Buscadora.	La aplicación debe contener un buscador para que el usuario pueda buscar más rápido los diferentes lugares e instalaciones que el establecimiento tiene. Además, el buscador debe poder autocompletarse para una mayor eficiencia.
09	Descripción lugares.	La aplicación, cuando corresponda, debe mostrar información relevante e importante (descripción general, de lo que se hace, horarios de atención, imágenes, horarios de secretarías y profesores jefes de carreras y servicios y/o materiales que provee a las personas.) del lugar por el cual se ha consultado.
10	Perfil administrador.	La aplicación debe tener una opción para poder entrar en modo administrador (autenticado con su usuario y contraseña), el cual sea capaz de poder editar, borrar y generar marcadores dentro del mapa.

Tabla 1. Requisitos funcionales.

4.4.2 Requisitos no funcionales.

Eficiencia: El sistema debe ser capaz de responder al usuario con sus funcionalidades en menos de 5 segundos.

Seguridad: La administración de los lugares del establecimiento solo puede ser trabajados (crear, editar, borrar) por el administrador del sistema, el cual debe estar autenticado para poder acceder.

Usabilidad: El sistema debe poseer interfaces gráficas bien diseñadas (textos, colores, entre otros.). Además, el sistema debe proporcionar mensajes de errores que sean informativos para el usuario final.

Portabilidad: El sistema debe funcionar sin problemas en cualquier dispositivo móvil con sistema operativo Android con una versión igual o mayor a 5.0(Lollipop).

4.5 Atributos del producto.

En el siguiente capítulo se abarca los atributos de calidad en el desarrollo de la aplicación móvil, tanto para los usuarios como para el desarrollador.

4.5.1 Atributos de calidad para usuarios.

A continuación, se presenta los atributos o cualidades de calidad básicas para usuarios que se debe cuidar en el desarrollo de esta aplicación móvil.

4.5.1.1 Disponibilidad.

La aplicación debe estar 99 por ciento del tiempo disponible para los usuarios que deseen descargarla.

4.5.1.2 Eficiencia.

La aplicación no debe afectar al equipo en términos de procesador, banda ancha, espacio en memoria y/o disco, etc.).

4.5.1.3 Flexibilidad.

La aplicación móvil debe ser susceptible a los cambios que se requieran realizar, incluyendo la modificación de las pruebas y/o códigos.

4.5.1.4 Integridad.

En el caso del panel de administrador, solo él y nadie más tiene el privilegio de acceder al sistema y a los datos para su modificación.

4.5.1.5 Confiabilidad.

La aplicación debe ser capaz de ejecutarse u operarse sin ninguna falla bajo condiciones establecidas por un periodo determinado de tiempo; en caso contrario debe lanzar los pertinentes errores y que éstos sean de carácter informativo para que el usuario no vuelva a cometerlos.

4.5.1.6 Robustez.

La aplicación debe continuar operando con propiedad y no caerse o cerrarse al momento que se ha ingresado algún tipo de entrada invalida o bien existan condiciones inesperadas como, por ejemplo, en este caso, caídas en la conexión a internet o desactivación del GPS.

4.5.1.7 Usable.

La aplicación móvil debe ser amigable con el usuario, pues éste debe entender sin complicaciones de que trata y cómo se usa la aplicación.

4.5.2 Atributos de calidad para desarrolladores.

A continuación, se presenta los atributos o cualidades de calidad básicas para desarrolladores que se debe cuidar en esta aplicación móvil.

4.5.2.1 Mantenimiento.

Cuando corresponda, la aplicación debe ser corregida fácilmente si existiesen errores, así como también debe ser fácil para el programador realizar una modificación al sistema.

4.5.2.2 Portabilidad.

El sistema debe funcionar y operar sin ningún error y esfuerzo en cualquier dispositivo móvil con sistema operativo Android que contenga una versión igual o mayor a “Lollipop” (Version 5.0).

4.5.2.3 Reusabilidad.

El sistema debe poseer códigos que sean capaces de ser reusables en otros proyectos similares.

4.5.2.4 Corrección.

El programa que se desarrolla debe satisfacer y cumplir con las especificaciones de requerimientos, y conseguir los objetivos que el cliente ha propuesto.

5 Estudio de factibilidad.

En este capítulo se presenta el estudio de la factibilidad que se encarga de orientar en la toma de decisiones en la evaluación de un proyecto, y en este caso para medir las posibilidades de éxito o de fracaso en la realización de la aplicación que se ha requerido desarrollar.

5.1 Factibilidad técnica.

A continuación, se presenta los lenguajes, herramientas y hardware utilizadas para el desarrollo de la aplicación.

5.1.1 Lenguajes y herramientas a utilizar.

Para desarrollar la aplicación móvil, se utiliza Android Studio 3.1 para el desarrollo de aplicaciones para dispositivos con sistema operativo Android, así como el uso de phpMyAdmin para el uso de base de datos MySQL proveído por el hosting 000webhost. En este caso la base de datos almacena (Título, Subtítulo, Latitud y Longitud) los distintos lugares de la Universidad del Bío-Bío, Campus Fernando May a través de marcadores en el mapa.

Android Studio es el entorno de desarrollo integrado oficial para la plataforma Android y ha sido publicado de forma gratuita para las plataformas Microsoft Windows, macOS y GNU/Linux. Dicho software es totalmente gratuito y está en internet para llegar y descargar en su página oficial (Fundación Wikimedia, WikipediA, 2018).

5.1.2 Hardware.

Actualmente la mayoría de los estudiantes de la Universidad del Bío-Bío y de las personas cuentan con teléfonos móviles con sistemas operativos Android, suficiente para correr y soportar la instalación y ejecución de la aplicación.

5.2 Factibilidad operacional.

Para los estudiantes de la Universidad del Bío-Bío quienes principalmente va dirigido esta aplicación móvil, éstos poseen los conocimientos básicos necesarios de cómo se usa un teléfono móvil con sus diferentes aplicaciones y funcionalidades. Por lo anterior no será muy difícil entender y hacer uso de la aplicación. Sin embargo, para las personas que no sepan usar muy bien estas tecnologías, no tendrán problemas para poder usar la aplicación, puesto que se pretende diseñar una interfaz completamente amigable, simple y fácil de usar para el usuario y así abarcar la mayor cantidad de personas posibles.

5.3 Factibilidad económica.

Para el desarrollo de la aplicación móvil, es necesario invertir en equipamiento tecnológico, más que nada en un celular de gama media, con el fin de poder emular en tiempo real, lo que se está desarrollando en el software Android Studio; en ese caso se necesita de un Smartphone Galaxy J1 Ace LTE VE Quad Core 8GB, Android 5.1 con valor de \$ 39.990 pesos chilenos, puesto que es el más económico dentro del mercado, y cabe recordar, además, que se requiere de una versión de Android 5.0 en adelante para poder trabar sin complicaciones (S.A., 2018).

Por otra parte, se debe pagar un sueldo por 4 meses aproximadamente al ingeniero civil informático para el desarrollo de la aplicación de \$ 600.000 mensuales (sueldo promedio de un recién egresado) (Indeed, 2018).

Se debe tener en consideración también que para poder darse de alta como desarrollador y poder subir la aplicación a Play Store de Google se necesita de \$16.861 pesos chilenos, y que se paga una sola vez (Antevenio, 2018).

De igual forma, existe un gasto único asociado al proyecto, que corresponde a la mantención del mismo, que consta de actualizaciones o de modificaciones de código. Dicha mantención se debe realizar semestralmente a un costo de \$200.000 pesos chilenos.

Tipo	Costo
Trabajo de ingeniero civil informático.	\$ 3.125 por hora (24 días de trabajo por 8 horas al día).
Smartphone.	\$ 39.990.
Pago de Play Store de Google.	\$ 16.861.
Total inversión.	\$ 2.456.851.

Tabla 2. Inversión proyecto.

Tipo	Detalle	Costo por día	Costo por año
Gastos en mantención.	Se gastará en contratar un ingeniero informático que trabaje 4 días por semestre.	\$ 50.000.	\$ 200.000.
Total gastos.			\$ 200.000.

Tabla 3. Gastos proyecto.

5.3.1 Beneficios.

Se considera el uso de AdMob que crea espacios publicitarios para las aplicaciones, mostrando a los usuarios anuncios con mejores resultados. Cada vez que alguien

pulse en una publicidad se llevará una cantidad de 15 pesos chilenos aproximadamente (AdMob, 2018).

Al año ingresan aproximadamente 600 alumnos a la Universidad del Bío-Bío, de esta cifra se estima que, por diferentes factores, el 60% de ellos utilizará la aplicación, lo cual hace referencia a por lo menos 360 alumnos, de igual forma es de esperar que ésta también sea utilizada por docentes, apoderados, gente de otras universidades, entre otras, agregando un aproximado de 140 personas. Por lo tanto, al año se tendría 500 personas con la aplicación instalada.

Referente a lo anterior se tiene lo siguiente.

Tipo	Detalle	Costo por mes	Costo por año
Utilización gratuita de AdMob.	Cada persona utilizará en promedio 10 veces la aplicación al mes y verá el anuncio.	\$75.000.	\$900.000.
Total beneficios.			\$900.000

Tabla 4. Beneficios proyecto.

5.3.2 Calculo del VAN.

Tasa de descuento.	10%.
Periodo.	5.

Tabla 5. Tasa de descuento y años de recuperación de la inversión.

Periodo	0	1	2	3	4	5
Inversión	2.456.851					
Gastos		-200.000	-200.000	-200.000	-200.000	-200.000
Beneficios		900.000	900.000	900.000	900.000	900.000
Flujo caja neto	-2.456.851	700.000	700.000	700.000	700.000	700.000

Tabla 6. Flujo de caja.

La fórmula para el cálculo del VAN es la siguiente.

$$VAN = -I_0 + \sum_{t=1}^n \frac{F_t}{(1+k)^t} = -I_0 + \frac{F_1}{(1+k)} + \frac{F_2}{(1+k)^2} + \dots + \frac{F_n}{(1+k)^n}$$

Figura 15. Fórmula para el cálculo del VAN (Economipedia, economipedia, 2018).

La fórmula para el cálculo del TIR es la siguiente.

$$VAN = -I_0 + \sum_{t=1}^n \frac{F_t}{(1+TIR)^t} = -I_0 + \frac{F_1}{(1+TIR)} + \frac{F_2}{(1+TIR)^2} + \dots + \frac{F_n}{(1+TIR)^n} = 0$$

Figura 16. Fórmula para el cálculo del TIR (Economipedia, economipedia, 2018).

Con los datos ya entregados, se ha calculado un valor actual neto de 196.149 y una tasa interna de retorno de 13%.

En conclusión, dado que el VAN es positivo se dice que el proyecto y la aplicación desarrollada es viable, y dado que la TIR es mayor que la tasa de descuento en un 3%, podemos decir que éste es rentable en un periodo de 5 años, por lo que se sugiere seguir adelante con el proyecto en cuestión.

6. Modelamiento del sistema.

En el siguiente capítulo se presenta el análisis y diseño de la solución, el modelado de los datos y el diseño de la base de datos, arquitectura e interfaces del sistema que se desarrolla.

6.1 Modelo y notación de procesos de negocio.

La figura 17 muestra el proceso de negocio de la aplicación en un formato de flujo de trabajo, en el cual se observa tres procesos involucrados en el problema a resolver, en el cual el primero; usuarios, tiene un proceso de inicio, que consta en abrir la aplicación, en la que posteriormente se hace revisión de los permisos de conectividad de internet y activación de GPS en su dispositivo, en ambos casos cuando estos están desactivados, se muestra un mensaje informativo para conectarse a internet o una ventana de diálogo con una opción para poder activar el GPS. Se menciona que, la aplicación cumple con el atributo de calidad de robustez que detalla que el sistema sigue operando con propiedad cuando existan condiciones inesperadas, por lo mismo, la aplicación sigue funcionando cuando los permisos ya mencionados no están activados, sin embargo, se requiere activarlos para su correcto funcionamiento y una buena experiencia de usuario. Posteriormente el usuario puede visualizar en un mapa los diferentes lugares que hay en el establecimiento y puede buscarlos ya sea deslizándose o navegando por el mapa, o bien buscar un lugar a través de la barra de búsqueda; en este último caso si el lugar no existe, corresponde enviar un mensaje informativo indicando que el lugar no existe y que lo vuelva a intentar. Por otro lado, de cualquier forma, que el usuario haya encontrado el lugar que desea, éste recibirá una ruta hacia su destino, mostrando además la distancia y el tiempo en llegar. Finalmente, el usuario puede consultar por la información de un lugar si así lo quiere, siendo mostrado en una nueva ventana con datos importantes y relevantes de dicho lugar por el cual se consulta.

El segundo proceso involucrado en este proceso de negocio es la aplicación móvil que gestiona las respuestas y las va mostrando al usuario, como los diálogos de activación de permisos, de errores, generación de rutas y envío de información de un lugar.

El tercer y último proceso denominado administrador, se encarga de entrar a la aplicación y “logear” al perfil de administrador siempre y cuando se quiera hacer modificaciones o creaciones con respecto a los marcadores dentro del mapa en la aplicación, por lo que tendrá que ingresar su usuario y contraseña. Posteriormente, si desea crear un marcador, debe ingresar el título más los datos informativos de éste, aparte de presionar unos segundos en el mapa, para que se cree el marcador. Si quiere borrar o editar, es necesario enviar el título del marcador en el cual es la clave primaria de cada uno, por lo que, en caso de no existir, se envía un mensaje informativo indicando la no existencia de este marcador, en caso contrario se envía la información correspondiente al marcador si es que existe y si es que se quiere editar o bien se envía un diálogo de confirmación si es que se desea eliminar.

Finalmente, se menciona que, la aplicación trabaja con diferentes tareas dependiendo del tipo de usuario, ya sea un administrador o bien un usuario normal. El proceso de negocio finaliza cuando el usuario no desea encontrar más ubicaciones o bien el administrador no quiere seguir editando, creando o borrando marcadores.

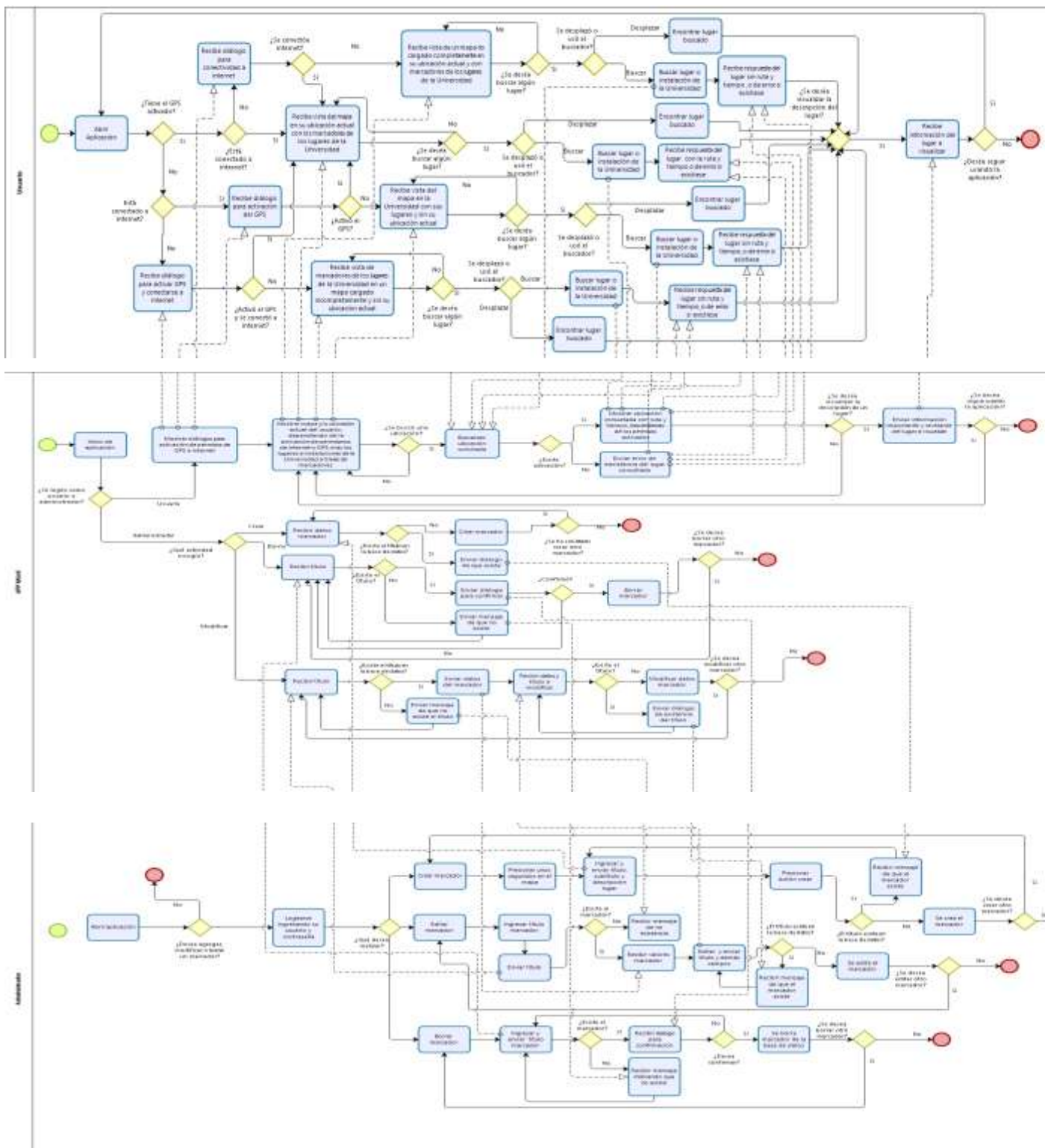


Figura 7. Business Process Model and Notation (BPMN) (Inc. L. S., Lucidchart, 2018).

6.2 Modelos de entidad-relación.

En la figura 18 se presenta el tipo de modelo entidad relación que, en cierto punto para la aplicación, determina la estructura lógica que lleva la base datos y de forma fundamental determina el modo en que se almacena, organiza y se manipula los datos.

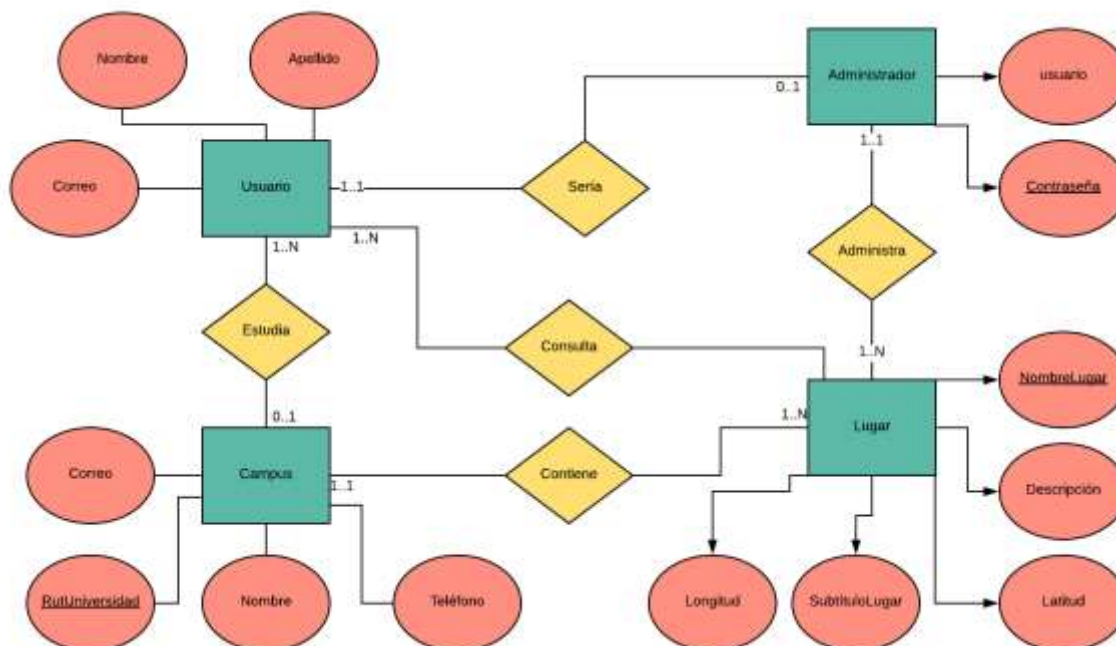


Figura 18. Modelo de entidad-relación (MER) (Inc. L. S., Lucidchart, 2018).

Se debe mencionar que, para el desarrollo de la aplicación y cumplimiento de los requerimientos, no es necesario el almacenamiento de la entidad Usuario y Campus, ya que se ha establecido que la aplicación solo muestre a las personas los lugares del campus Fernando May, sin embargo, en un futuro se podría hacer registro de la entidad Campus para poder mostrar todos los lugares del Campus la castilla y concepción, y también hacer registro de los usuarios para enviar información respecto a cambios de los lugares de los Campus, como por ejemplo, cambios de oficinas a otras instalaciones, cambios de horarios de profesores y/o secretarías o creaciones de nuevas instalaciones, edificios o espacios recreativos.

En la tabla 7, 8 y 9 se explican las entidades, relaciones y atributos más importantes del modelo de datos.

Nombre	Descripción
Usuario.	Entidad que hace referencia a quienes utilizarán la aplicación móvil.
Lugar.	Entidad que hace referencia a los diferentes lugares que cuenta el establecimiento.
Campus.	Entidad que hace referencia a la Universidad del Bío-Bío, Campus Fernando May, que será en cierta parte, dueña de la aplicación.
Administrador.	Entidad que hace referencia al administrador de la aplicación que será el encargado de crear, borrar y editar un marcador dentro del mapa y su correspondiente información.

Tabla 7. Entidades del modelo de datos.

Nombre	Descripción
RutUniversidad.	Identificador de la entidad Universidad, que lo identifica como único.
NombreLugar.	Identificador de la entidad Lugar que describe como único el nombre del lugar del establecimiento.
Descripción.	Atributo de la entidad Lugar que describe un lugar del establecimiento. Dichas descripciones pueden ser descripción general, de lo que se hace, horarios de atención, imágenes, horarios de secretarias y profesores jefes de carreras y servicios y/o materiales que provee a las personas.
SubtituloLugar.	Atributo de la entidad Lugar que puede representar la abreviación , sinónimo o sigla del nombre del lugar.

Latitud.	Atributo de la entidad Lugar que representa la coordenada de una dirección dentro del mapa, y en este caso de algún lugar del establecimiento.
Longitud.	Atributo de la entidad Lugar que representa la coordenada de una dirección dentro del mapa.
Contraseña.	Atributo referente al Administrador que representa la clave necesaria para poder ingresar al perfil de administrador dentro de la aplicación.

Tabla 8. Atributos del modelo de datos.

Nombre	Descripción
Consulta.	Relación entre las entidades Usuario y Lugar, en el cual describe que un usuario puede consultar por uno o muchos lugares, y en caso contrario una ubicación puede ser consultada por uno o muchos usuarios.
Contiene.	Relación entre las entidades Campus y Lugar, en el que describe que el campus contiene uno o muchos lugares, y un lugar está contenido dentro de un campus.
Estudia.	Relación entre las entidades Usuario y Campus, en el cual describe que un usuario puede pertenecer a ninguna o un Campus, y en caso contrario en un Campus pueden estudiar 1 o muchos usuarios.
Sería.	Relación entre las entidades Usuario y Administrador, en el cual describe que un usuario puede o no ser un administrador y en caso contrario un administrador es un usuario.
Administra.	Relación entre las entidad Administrador y Lugar, en el que describe que un administrador puede administrar uno o muchos lugares y en caso contrario un lugar puede ser administrado por un solo administrador.

Tabla 9. Relaciones del modelo de datos.

6.3 Diseño de base de datos.

En la figura 19 se presenta el diseño de la base de datos que se utiliza en la aplicación Android.

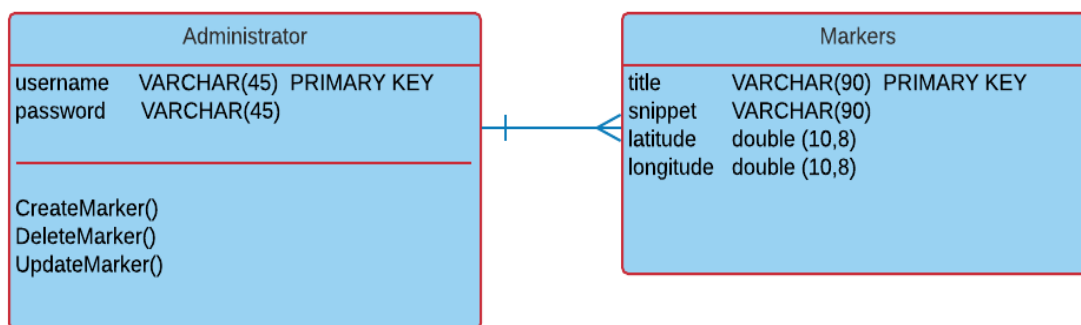


Figura 19. Modelo de datos (Inc. L. S., Lucidchart, 2018).

Se menciona que, no se ha considerado la entidad Usuario y Campus expuestos en la figura 18, puesto que no son por el momento relevantes para el desarrollo y el objetivo principal de la aplicación, sin embargo, a futuro pueden ser incluidos si se desea, por ejemplo, sería importante tener el registro de todos los usuarios de la aplicación o bien tener más Campus y mostrar sus instalaciones, lugares, departamentos, entre otros.

Con respecto al modelo, se observa que, existe una tabla “administrador” con atributos usuario y contraseña, que sirven para entrar en modo administrativo en la aplicación desarrollada. La tabla además cuenta con 3 métodos los cuales el administrador puede utilizar, los cuales son poder crear, editar y borrar un marcador dentro del mapa o bien su información.

Finalmente, se muestra una tabla “coordenadas”, el cual almacena los atributos necesarios para crear un marcador dentro del mapa, que son el título, un pequeño extraño que defina el marcador (subtítulo), latitud y longitud. Los dos últimos atributos muestran que son de tipo “double (10,8)”, en el cual estos números

significan que la latitud y longitud contendrán 10 caracteres y que 8 de ellos serán decimales.

Para entender más este tema, en la figura 20 se presenta una imagen de la base de datos real con algunos datos, hecha en la herramienta phpMyAdmin.

idbib	nombre	latitud	longitud
	Académicos Del Departamento De Ciencias Básicas	-36.60191600	-72.07650930
	Área De Desarrollo Pedagógico Y Tecnológico	-36.60239800	-72.07702930
	Administración Central	-36.60176300	-72.07734930
	Aula Magna Fernando Mey	-36.60176300	-72.07734930
	Ceremonias	-36.60176300	-72.07734930
	Asias A	-36.60319400	-72.07718930
	A1FM Hasta A1FM	-36.60319500	-72.07636190
	B1FM Hasta B1FM	-36.60319500	-72.07636190
	C1FM Hasta C1FM	-36.60319500	-72.07636190
	D1FM Hasta D1FM	-36.60319500	-72.07636190
	E1FM Hasta E1FM	-36.60319500	-72.07636190
	Biblioteca Principal	-36.60279200	-72.07610490
	Sala PIESDA Programa De Inclusión De Estudiantes C...	-36.60279200	-72.07610490
	Cabesa De Vigilancia De La Universidad Del Bío-Bío	-36.60272600	-72.07608930
	Snack-Bar	-36.60272600	-72.07608930
	Tarhau Calentita	-36.60015600	-72.07620930
	Pabellón De Dibujo Y Psicología	-36.59982700	-72.07460930
	Tasa	-36.60179400	-72.07544930

Figura 20. Base de datos en phpMyAdmin (phpMyAdmin, 2018).

6.4 Arquitectura utilizada.

Para la realización de la aplicación Android se usa el patrón de arquitectura de software MVC, que separa los datos y la lógica de negocio de una aplicación de su representación y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, vista y controlador, es decir, por un lado, define componentes para la representación de la información, y por otro lado para la interacción del usuario. Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento (Fundación Wikimedia, WikipediA, 2018).

En la figura 21 se aprecia la relación que existe entre el modelo, el controlador y la vista.

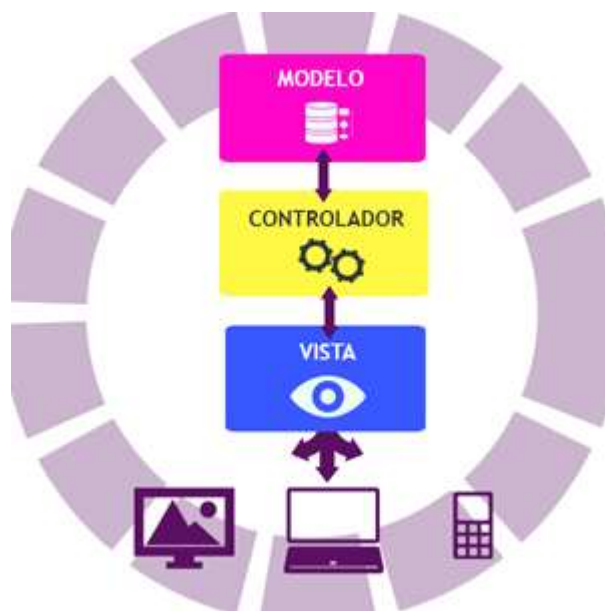


Figura 21. Diagrama modelo-vista-controlador¹.

¹ Fuente: <https://programarfacil.com/podcast/19-patron-modelo-vista-controlador/>

Con respecto a los tres componentes antes mencionados y la aplicación Android, podemos decir que:

1.- Modelo: Tiene que ver con la lógica de negocio; es la representación de la información con la cual el sistema opera, por lo tanto, gestiona todos los accesos a dicha información. Por ejemplo, en la aplicación el modelo sería la clase que gestiona la base de datos que almacena los marcadores y las peticiones de acceso o manipulación a la información de un usuario a través de la interfaz llegando al modelo a través del controlador.

2.- Vista: La vista hace referencia a la interfaz de usuario, a lo que el usuario ve por pantalla al momento de ejecutar la aplicación. En el caso de Android, serían los Layouts, textos, imágenes y botones; lenguaje XML en este caso. La vista realiza peticiones al modelo para la realización de algún evento, pero primero para aquello debe pasar por el controlador que solicita la información para posteriormente enviar la respuesta del modelo a la vista.

3.- Controlador: El controlador responde usualmente a las acciones de un usuario en la aplicación e invoca peticiones al modelo cuando se hace una solicitud sobre la información. El controlador hace de intermediario entre la vista y el modelo. En el caso de la aplicación, tendríamos la función de algún botón que hace algo, por ejemplo, consultar la información de algún marcador en el mapa.

6.5 Interfaz y navegación.

A continuación, en el siguiente capítulo se muestra las diferentes ventanas o interfaces que lleva la aplicación, así como también las jerarquías de menú tanto para el usuario como para el administrador.

6.5.1 Jerarquías de menú.

La aplicación se compone de dos partes, la primera correspondiente a los usuarios los cuales pueden ver y buscar los lugares, espacios o instalaciones que posee la Universidad, esto a través de marcadores puesto en el mapa. La segunda parte corresponde al perfil administrador, el cual puede crear, editar y modificar los marcadores dentro del mapa, así como el cambio de su usuario y contraseña si se desea.

6.5.1.1 Jerarquías de menú para usuario.

La figura 22 presenta la jerarquía de menú para un usuario normal:

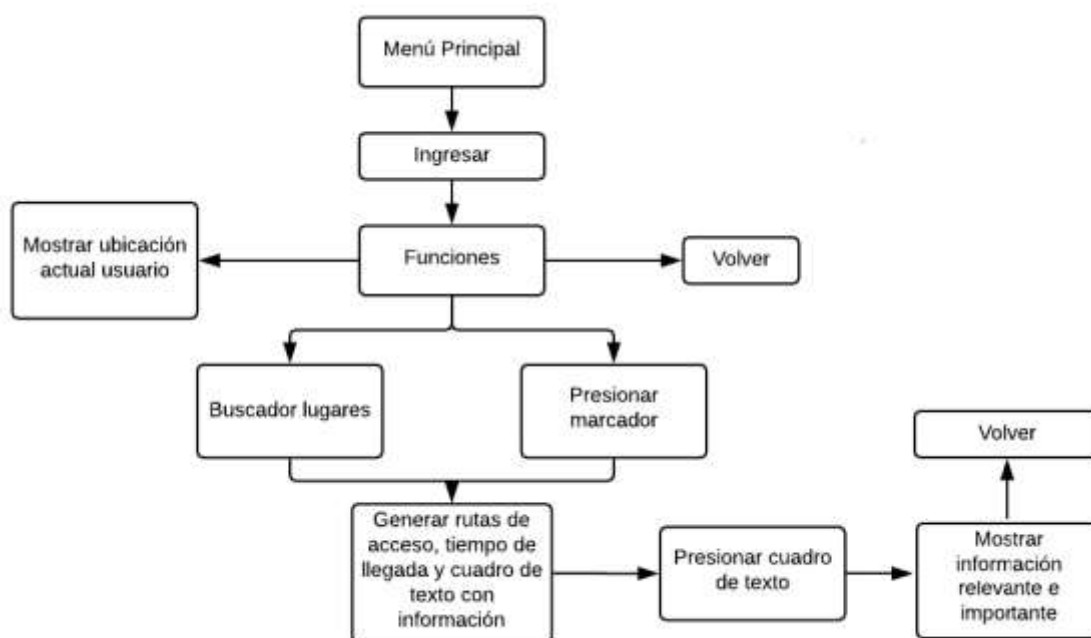


Figura 22. Jerarquía de menú de un usuario².

² Fuente: <https://www.lucidchart.com/pages/es?noHomepageRedirect=true>

6.5.1.2 Jerarquías de menú para administrador.

La figura 23 presenta la jerarquía de menú para un administrador:

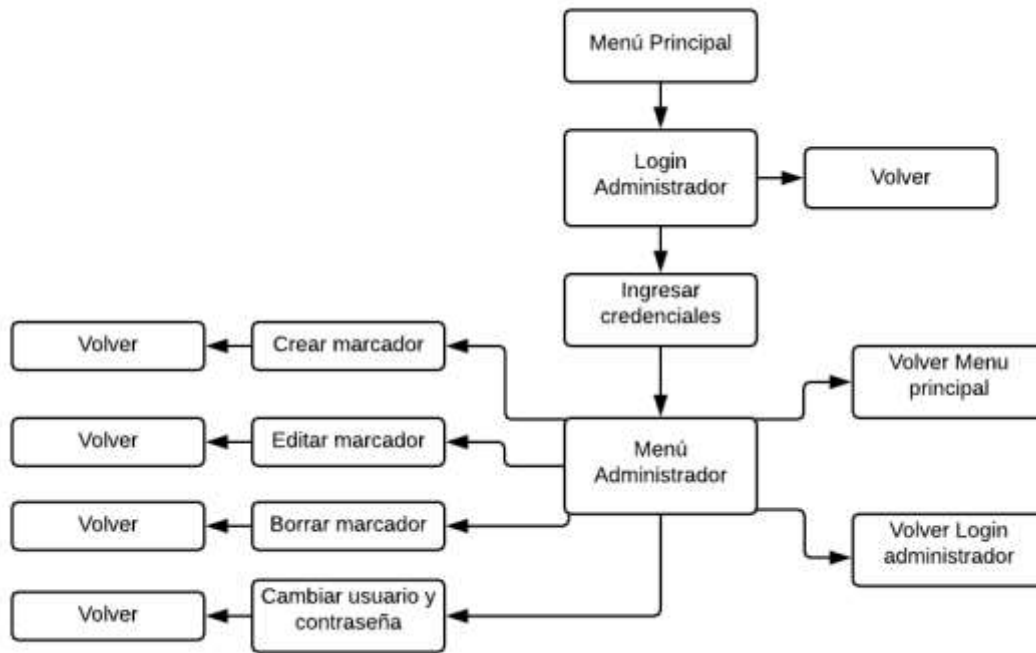


Figura 23. Jerarquía de menú de un administrador³.

³ Fuente: <https://www.lucidchart.com/pages/es>

6.5.2 Interfaz de usuario.

En el siguiente capítulo, se muestra las diferentes ventanas o interfaces tanto para el administrador como para un usuario normal.

La figura 24 muestra la pantalla de inicio de la aplicación.



Figura 24. Pantalla de inicio.

La figura 25 muestra el “login” de administrador con los campos usuario y contraseña para completar.

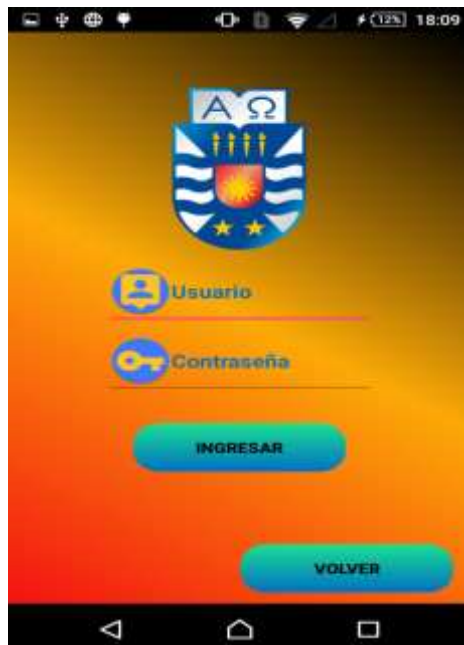


Figura 25. Login administrador.

La figura 26 muestra el menú principal de la aplicación, la cual presenta dos opciones, que son: comenzar la aplicación y “logear” al perfil de administrador.



Figura 26. Menú principal.

La figura 27 muestra una ventana para la activación de GPS cuando se entra al mapa y la ubicación está desactivada.



Figura 27. Activación GPS.

La figura 28 muestra el menú de administrador con las opciones de: crear marcador, editar y borrar marcador, cambiar usuario o contraseña, "login administrador" y menú principal.

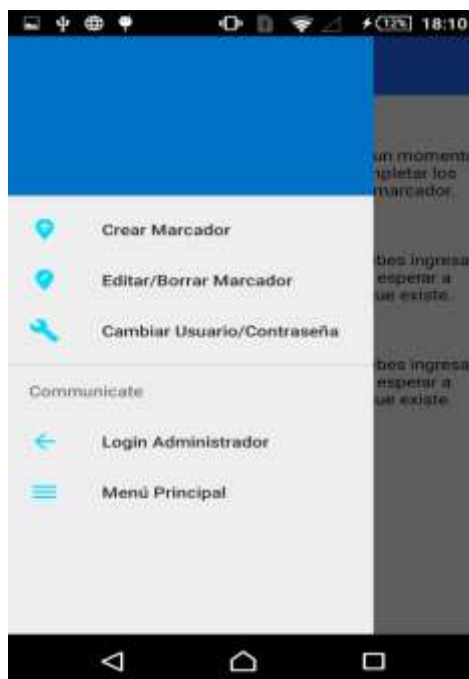


Figura 28. Menú administrador.

La figura 29 muestra la configuración del administrador para cambiar el usuario o contraseña.



Figura 29. Configuración administrador.

La figura 30 muestra la interfaz para crear un marcador dentro del mapa.



Figura 30. Creando marcador.

La figura 31 muestra un marcador ya creado con los datos mostrados en la figura 30.



Figura 31. Marcador creado.

La figura 32 muestra como se actualiza un marcador dentro del mapa.



Figura 32. Actualizando marcador.

La figura 33 muestra un diálogo de confirmación para actualizar un marcador con los datos que presenta la figura 32.



Figura 33. Confirmar actualización.

La figura 34 muestra un marcador actualizado con los datos que presenta la figura 32.



Figura 34: Marcador actualizado.

La figura 35 muestra un diálogo de confirmación para borrar un marcador.



Figura 35. Confirmar borrar marcador.

La figura 36 muestra un mensaje de que se ha eliminado correctamente un marcador en el mapa.



Figura 36. Marcador eliminado.

La figura 37 muestra el mapa geográfico ubicado en el establecimiento y en el cual se aprecia los marcadores correspondientes a las instalaciones del mismo.



Figura 37. Visualización marcadores.

La figura 38 muestra una ruta de acceso desde la entrada de la Universidad hasta un marcador. Además, se aprecia la distancia en metros y el tiempo de llegada a pie y bicicleta.



Figura 38. Ruta y tiempo de llegada.

La figura 39 muestra información de un lugar consultado, en este caso de los Laboratorios Centrales de Computación.



Figura 39. Información de un lugar.

6.6 Casos de uso.

Un caso de uso es una descripción de las acciones de un sistema desde el punto de vista del usuario. Es una herramienta valiosa dado que es una técnica de aciertos y errores para obtener los requerimientos del sistema, justamente desde el punto de vista del usuario.

Los diagramas de caso de uso modelan la funcionalidad del sistema usando actores y casos de uso. Los casos de uso son servicios o funciones provistas por el sistema para sus usuarios (Cevallos, 2018).

En la figura 40 se presenta el funcionamiento del sistema para el usuario y administrador a través de un modelo de casos de uso.

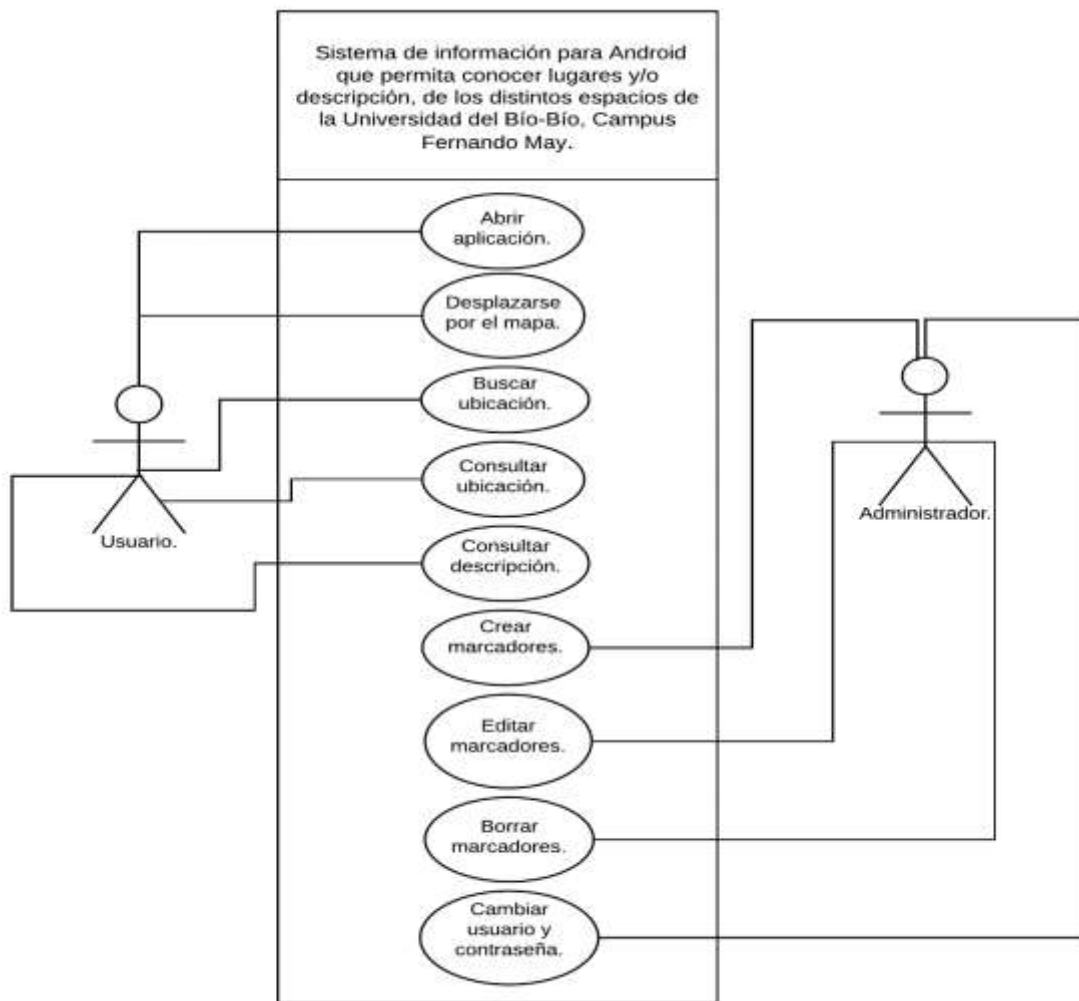


Figura 40. Diagrama de casos de uso (Inc. L. S., Lucidchart, 2018).

La tabla 10 muestra una explicación del caso de uso “desplazarse por el mapa” que presenta la figura 40.

Caso de uso: Desplazarse por el mapa.
ID: 2.
Breve Descripción: El usuario puede desplazarse por el mapa y encontrar la Universidad con los marcadores puestos en los lugares que posee.
Actores principales: Usuario.
Precondiciones: Usuario que no conoce el establecimiento, tiene la necesidad de encontrar éste establecimiento y sus espacios.
Flujo principal: El usuario con el dedo se desplaza por el mapa.
Postcondiciones: Ninguno.
Flujo alternativo: Ninguno.

Tabla 10. Caso de uso: Desplazarse por el mapa (Corporation, 2018).

La tabla 11 muestra una explicación del caso de uso “Consultar ubicación” que presenta la figura 40.

Caso de uso: Consultar ubicación.
ID: 4.
Breve Descripción: El usuario tiene la opción de consultar por una ubicación del establecimiento al cual desee llegar.
Actores principales: Usuario.
Precondiciones: Usuario que no conoce el establecimiento, tiene la necesidad de consultar por algún lugar.
Flujo principal: El usuario ingresa en el buscador el lugar que desea encontrar.
Postcondiciones: Ninguno.
Flujo alternativo: Ninguno.

Tabla 11. Caso de uso: Consultar ubicación (Corporation, 2018).

La tabla 12 muestra una explicación del caso de uso “Consultar Descripción” que presenta la figura 40.

Caso de uso: Consultar Descripción.
ID: 5.
Breve Descripción: El usuario tiene la opción de consultar la descripción de algún lugar del establecimiento.
Actores principales: Usuario.
Precondiciones: Usuario que no conoce el establecimiento y sus espacios, tiene la necesidad de consultar por la información de algún lugar.
Flujo principal: El usuario selecciona un marcador, en el cual se muestra un cuadro de texto con breve información de lugar. Posteriormente el usuario selecciona el cuadro para ver más información.
Postcondiciones: Ninguno.
Flujo alternativo: Ninguno.

Tabla 12. Caso de uso: Consultar Descripción (Corporation, 2018).

La tabla 13 muestra una explicación del caso de uso “Crear marcador” que presenta la figura 40.

Caso de uso: Crear marcador.
ID: 6.
Breve Descripción: El Administrador tiene la opción de crear un marcador en el mapa que representa un lugar del establecimiento.
Actores principales: Administrador.
Precondiciones: La Universidad cuenta con un nuevo lugar, por ende se necesita crear el marcador para dicho lugar.
Flujo principal: El administrador selecciona en el mapa y crea un marcador, después debe rellenar los campos con el título, subtítulo e información del lugar.
Postcondiciones: Ninguno.
Flujo alternativo: Ninguno.

Tabla 13. Caso de uso: Crear marcador (Corporation, 2018).

La tabla 14 muestra una explicación del caso de uso “Editar marcador” que presenta la figura 40.

Caso de uso: Editar marcador.
ID: 7.
Breve Descripción: El Administrador tiene la opción de editar un marcador en el mapa que representa un lugar del establecimiento.
Actores principales: Administrador.
Precondiciones: Existe algún error en los nombres de los lugares o en la información, por ende se necesita editar el marcador.
Flujo principal: El administrador selecciona el marcador y se llenan los campos automáticamente con la información correspondiente para después editarlos.
Postcondiciones: Ninguno.
Flujo alternativo: Ninguno.

Tabla 14. Caso de uso: Editar marcador (Corporation, 2018).

La tabla 15 muestra una explicación del caso de uso “Borrar marcador” que presenta la figura 40.

Caso de uso: Borrar marcador.
ID: 8.
Breve Descripción: El Administrador tiene la opción de borrar un marcador en el mapa que representa un lugar del establecimiento.
Actores principales: Administrador.
Precondiciones: se coloca mal un marcador en el mapa y no representa el verdadero lugar de la Universidad, por ende necesita ser eliminado.
Flujo principal: El administrador selecciona el marcador y luego el botón borrar, en el cual aparece un diálogo de confirmación y si lo acepta, finalmente se elimina.
Postcondiciones: Ninguno.
Flujo alternativo: Ninguno.

Tabla 15. Caso de uso: Borrar Marcador (Corporation, 2018).

La tabla 16 muestra una explicación del caso de uso “Cambiar usuario o contraseña” que presenta la figura 40.

Caso de uso: Cambiar usuario o contraseña.
ID: 9 .
Breve Descripción: El Administrador tiene la opción de cambiar su usuario o contraseña para entrar al perfil de administrador.
Actores principales: Administrador.
Precondiciones: El administrador encuentra muy difícil autenticarse o bien el usuario y contraseña son de fácil “hackeo”, por lo mismo la necesidad de cambiarlos.
Flujo principal: El administrador accede al menú de cambio, lo cual aparecen los valores actuales de acceso, más unos botones con el fin de cambiar estos valores. Posteriormente selecciona el botón confirmar y aparece un diálogo de confirmación. Finalmente se cambia cuando se confirma.
Postcondiciones: Ninguno.
Flujo alternativo: Ninguno.

Tabla 16. Caso de uso: Cambiar usuario o contraseña (Corporation, 2018).

7 Pruebas de Sistema.

Las pruebas de sistema son muy importantes, puesto que revelan la presencia de errores en el sistema. Probar reduce la probabilidad de que existan errores, pero nunca se puede asegurar que no quede ninguno (JoaquinOriente, 2018).

Por lo anterior se tiene como objetivo lograr encontrar estos errores, mencionarlos y posteriormente corregirlos, así como también probar si el software cumple con la especificación de requisitos, probar si existen efectos secundarios adversos, entre otros. También, el objetivo de estas pruebas es presentar información sobre la calidad del producto a las personas responsables de éste, es decir, encontrar fallas, aumentar la confianza en el nivel de calidad, facilitar la información para la toma de decisiones y evitar la aparición de defectos.

Para la realización de las pruebas se trabaja con pruebas de caja negra que es una técnica de pruebas de software en la cual la funcionalidad se verifica sin tomar en cuenta la estructura interna de código, detalles de implementación o escenarios de ejecución internos en el software. En las pruebas de caja negra, nos enfocamos solamente en las entradas y salidas del sistema, sin preocuparnos en tener conocimiento de la estructura interna del programa de software (pmoinformatica, 2018).

7.1 Elementos y Planificación de pruebas.

En este capítulo, se prueban los requisitos funcionales y no funcionales establecidos por el cliente.

7.1.1 Pruebas de los requisitos funcionales.

A continuación, se presenta los casos de pruebas y cumplimientos correspondiente a los requisitos funcionales especificados en el capítulo 4.4.1 de este informe.

Recordar que, se menciona al inicio de este capítulo que se utiliza la técnica de pruebas de caja negra que simplemente se enfoca en las entradas y salidas del sistema.

Visualización plano geográfico: La aplicación debe contener un plano geográfico que muestre la Universidad con sus instalaciones y alrededores, siempre y cuando haya conexión a internet.

Técnica de pruebas de caja negra: Requerimiento funcional / Caso de uso.

Caso 1: Datos de entrada: Conectarse a internet. Resultado esperado (Salida): El sistema captura los datos de la API de Google y muestra el mapa geográfico.

Caso 2: Datos de entrada: Desconexión a internet antes de entrar al mapa. Resultado esperado (Salida): Se envía un mensaje de “error de conexión”.

Caso 3: Datos de entrada: Desconexión a internet justo después de entrar al mapa. Resultado esperado (Salida): Se visualiza un mapa completamente plomo y no se visualiza los marcadores, más muestra un mensaje: “error de conexión, no se puede capturar los datos, reanude la aplicación”.

Sistema de posicionamiento global: La aplicación cuenta con los permisos de GPS, para mostrar la posición actual del usuario que esté usando la aplicación. En caso de estar desactivado, la aplicación permite al usuario poder activar la posición actual.

Técnica de pruebas de caja negra: Requerimiento funcional / Caso de uso.

Caso 1: Datos de entrada: GPS desactivado antes de entrar al mapa. Resultado esperado (Salida): Ventana emergente con opción para activar GPS.

Caso 2: Datos de entrada: GPS activado en el mapa. Resultado esperado (Salida): Se visualiza la posición actual del usuario en el mapa.

Caso 3: Datos de entrada: Desconexión GPS dentro del mapa. Resultado esperado (Salida): Se visualiza la última posición del usuario en el mapa.

Caso 4: Datos de entrada: No activación de GPS en la ventana emergente. Resultado esperado (Salida): se muestra el siguiente mensaje: “Camine para que aparezca el GPS o verifique que está activado”.

Visualización marcadores: Se muestra marcadores de los diferentes lugares e instalaciones que la Universidad del Bío-Bío tiene.

Técnica de pruebas de caja negra: Requerimiento funcional / Caso de uso.

Caso 1: Datos de entrada: Dispositivo conectado a internet. Resultado esperado (Salida): La aplicación captura los datos de la base de datos y se muestra los marcadores en el mapa.

Caso 2: Datos de entrada: Dispositivo desconectado a internet justo después de entrar al mapa. Resultado esperado (Salida): Mensaje que muestra: “Error de conexión, no se pueden mostrar los marcadores, vuelva a iniciar la aplicación.”

Visualización ruta: La aplicación genera las diferentes rutas que van desde la entrada del establecimiento hasta la ubicación por la cual se ha consultado.

Técnica de pruebas de caja negra: Requerimiento funcional / Caso de uso.

Caso 1: Datos de entrada: Seleccionar marcador. Resultado esperado (Salida): Se genera una ruta de acceso desde la entrada de la Universidad hasta el marcador seleccionado.

Caso 2: Datos de entrada: Seleccionar resultado generado por el autocompletar de la barra de búsqueda. Resultado esperado (Salida): Se genera una ruta de acceso desde la entrada de la Universidad hasta el lugar que se selecciona en la barra buscadora.

Caso 3: Datos de entrada: Seleccionar el botón buscar de un lugar que existe. Resultado esperado (Salida): Se genera una ruta de acceso desde la entrada de la Universidad hasta el lugar que se busca.

Visualización distancia: La aplicación debe mostrar la distancia en metros de las rutas generadas.

Técnica de pruebas de caja negra: Requerimiento funcional / Caso de uso.

Caso 1: Datos de entrada: Seleccionar marcador. Resultado esperado (Salida): Se genera una ruta de acceso y se muestra en pantalla la distancia de esta ruta en metros.

Caso 2: Datos de entrada: Seleccionar resultado generado por el autocompletar de la barra de búsqueda. Resultado esperado (Salida): Se genera una ruta de acceso y se muestra en pantalla la distancia de esta ruta en metros.

Caso 3: Datos de entrada: Seleccionar el botón buscar de un lugar que existe. Resultado esperado (Salida): Se genera una ruta de acceso y se muestra en pantalla la distancia de esta ruta en metros.

Visualización tiempo de tardanza: La aplicación debe mostrar el tiempo en minutos y segundos de tardanza con respecto a las rutas generadas, ya sea caminando o en bicicleta.

Técnica de pruebas de caja negra: Requerimiento funcional / Caso de uso.

Caso 1: Datos de entrada: Seleccionar marcador. Resultado esperado (Salida): Se genera una ruta de acceso y se muestra en pantalla el tiempo en minutos y segundos de lo que se tarda en llegar tanto caminando como en bicicleta.

Caso 2: Datos de entrada: Selecciona resultado generado por el autocompletar de la barra de búsqueda. Resultado esperado (Salida): Se genera una ruta de acceso y se muestra en pantalla el tiempo en minutos y segundos de lo que se tarda en llegar tanto caminando como en bicicleta.

Caso 3: Datos de entrada: Selecciona el botón buscar de un lugar que existe. Resultado esperado (Salida): Se genera una ruta de acceso y se muestra en pantalla el tiempo en minutos y segundos de lo que se tarda en llegar tanto caminando como en bicicleta.

Barra Buscadora: La aplicación debe contener un buscador para que el usuario pueda buscar más rápido los diferentes lugares e instalaciones que el establecimiento tiene. Además, el buscador debe poder autocompletarse para una mayor eficiencia.

Técnica de pruebas de caja negra: Requerimiento funcional / Caso de uso/ Partición de equivalencias.

Caso 1: Datos de entrada: El usuario ingresa 3 caracteres. Resultado esperado (Salida): La barra buscadora se autocompleta con los lugares existentes y que contengan los 3 caracteres ingresados.

Caso 2: Datos de entrada: El usuario ingresa caracteres incoherentes a lugares de la Universidad. Resultado esperado (Salida): La barra de búsqueda no se autocompleta.

Caso 3 Datos de entrada: El usuario ingresa un lugar que no existe y selecciona buscar. Resultado esperado (Salida): Se arroja el siguiente mensaje: "El lugar ingresado no existe, intente otro".

Descripción lugares: La aplicación cuando corresponda, debe mostrar información relevante e importante (descripción general, horarios de atención, etc.) del lugar por el cual se está consultando.

Técnica de pruebas de caja negra: Requerimiento funcional / Caso de uso.

Caso 1: Datos de entrada: El usuario selecciona el cuadro arrojado por un marcador que contiene información agregada a nivel de código. Resultado esperado (Salida): Se muestra información e imágenes del lugar seleccionado.

Caso 2: Datos de entrada: El usuario selecciona el cuadro arrojado por un marcador que contiene información agregada en una base de datos y el dispositivo está conectado a internet. Respuesta esperada (Salida): Se captura la información de la base de datos y se muestra al usuario.

Caso 3: Datos de entrada: El usuario selecciona el cuadro arrojado por un marcador que contiene información agregada en una base de datos y el dispositivo no está conectado a internet. Resultado esperado (Salida): No se muestra la información correspondiente, más solo el título del marcador o lugar seleccionado y un mensaje diciendo: "Conexión fallida, no se puede mostrar la información, vuelva a iniciar la aplicación".

Perfil administrador: La aplicación debe tener una opción para poder entrar en modo administrador (autenticado con su Usuario y contraseña), el cual sea capaz de poder editar, borrar y crear marcadores dentro del mapa.

Técnica de pruebas de caja negra: Requerimiento funcional / Caso de uso / Partición de equivalencias / Condiciones de Borde.

Caso 1: Datos de entrada: El administrador selecciona ingresar con los campos usuario y contraseña vacíos. Resultado esperado (Salida): Se muestra un mensaje diciendo que debe completar los campos para acceder al menú.

Caso 2: Datos de entrada: El administrador selecciona ingresar con el campo usuario o contraseña vacíos. Resultado esperado (Salida): Se envía un mensaje diciendo que complete el campo faltante para poder ingresar.

Caso 3 Datos de entrada: El administrador selecciona ingresar con los campos usuarios y contraseña completados, pero incorrectos. Resultado esperado (Salida): Se muestra un mensaje diciendo que ambos campos están incorrectos.

Caso 4: Datos de entrada: El administrador ingresa correctamente su clave, pero el usuario es incorrecto o el usuario es correcto y la clave incorrecta. Resultado esperado (Salida): Dependiendo del campo incorrecto, se muestra un mensaje diciendo que tal campo está incorrecto.

Caso 5: Datos de entrada: El administrador intenta ingresar más de 5 veces al perfil administrador, ingresando claves incorrectas. Resultado esperado (Salida): Se bloquean los campos y el administrador debe esperar para poder volver a ingresar datos.

Caso 6: Datos de entrada: En el menú crear, el administrador selecciona unos segundos en el mapa. Resultado esperado (Salida): Se genera un nuevo marcador y se habilitan los campos título, subtítulo, latitud, longitud e información.

Caso 6: Datos de entrada: En el menú crear, se ingresa más caracteres de lo establecido en el campo información. Resultado esperado (Salida): Se visualiza que el administrador ingresa caracteres, sin embargo, solo se acepta hasta el límite.

Caso 7: Datos de entrada: En el menú editar, el usuario selecciona un marcador en el mapa y posterior botón actualizar. Resultado esperado (Salida): Se llena los campos correspondientes al marcador seleccionado y posteriormente aparece un diálogo de confirmación para actualización.

Caso 8: Datos de entrada: En el menú borrar, el usuario selecciona un marcador en el mapa y posterior botón borrar. Resultado esperado (Salida): Aparece un diálogo de confirmación para poder borrar el marcador seleccionado.

Prueba de carga: La prueba de carga consiste en observar el comportamiento de una aplicación bajo una cantidad de peticiones esperadas. Esta carga puede ser un número esperado de usuarios concurrentes utilizando la aplicación y realizando transacciones.

Se ha usado la aplicación JMeter para poder realizar esta prueba de carga y se ha puesto el siguiente escenario:

Caso 1: Datos de entrada: 50 hilos (usuarios) realizando transacciones o peticiones al servidor y base de datos 10 veces cada 1 segundo. Resultado esperado(salida): El programa JMeter muestra que las peticiones se han realizado correctamente, pero con respecto a la latencia, tiempo de conexión y carga ha dado un número medianamente grande.

7.1.2 Pruebas de los requisitos no funcionales.

A continuación, se presenta los casos de pruebas y cumplimiento correspondiente a los requisitos no funcionales especificados en el capítulo 4.4.2 de este informe.

Se debe mencionar que, la aplicación ha sido utilizada por cuatro familiares, tres amigos y un guardia de la Universidad del Bío-Bío, en algunos con diferentes teléfonos celulares.

Eficiencia: En este punto, se mide el tiempo que el sistema tarda en responder a los usuarios con las funcionalidades del mismo, para posteriormente ser mostradas en el mapa a través de marcadores.

La segunda prueba consiste en acceder al perfil de administrador y el tiempo que tarda cuando éste selecciona el botón entrar después de autenticarse con su usuario y contraseña, por lo que se ve lo que el sistema demora en verificar que los datos son iguales y por lo tanto conceder el permiso para entrar.

La tercera prueba tiene que ver cuando el administrador del sistema crea, edita o actualiza marcadores, por lo que se ve el tiempo que tarda en enviar una petición y posteriormente recibirla para ser mostrada y actualizada en el mapa.

En general, las pruebas mencionadas funcionan bien en condiciones en que la red es de hogar, sin embargo, cuando se trata de la red de la Universidad y está saturada por la cantidad de usuarios que la ocupan, la aplicación tiende a ralentizarse en responder a las peticiones.

Por lo tanto, se puede decir que el caso de prueba con respecto a este requisito se cumple perfectamente en condiciones normales.

Seguridad: Este requisito tiene que ver que la administración (crear, editar, borrar) de los lugares del establecimiento solo puede ser tratados por el administrador del sistema, el cual debe estar autenticado para el acceso.

En una primera instancia, se le entrega el usuario y contraseña al administrador, en la cual si desea puede ser cambiada dentro de la aplicación y posteriormente se instancia la aplicación de Gmail por si quiere enviarse estos datos para recordar de mejor forma.

La seguridad funciona de tal forma que, si alguien ingresa 5 veces el usuario o contraseña incorrectos, se bloquea por unos segundos la posibilidad de escribir estos datos, posteriormente entre más errores se cometa, más tiempo. Por lo tanto, se dice que este requisito y caso de prueba se cumple.

Usabilidad: se dice que se cumple en cierta parte con el requisito y caso de prueba, ya que se tienen colores claros y llamativos, textos alineados y de tamaño moderado, mensajes de errores informativos, diseño responsivo, pero no se explica mucho sobre cómo usar la aplicación.

Además, mencionar que la aplicación ha sido catalogada como “bonita” por usuarios conocidos, dándole un valor importante a la aplicación.

Portabilidad: El sistema debe funcionar sin problemas en cualquier dispositivo móvil con sistema operativo Android con una versión igual o mayor a 5.0(Lollipop).

El requisito se cumple sin problemas, puesto que la aplicación ha funcionado correctamente tanto en la versiona más baja establecida en los requisitos como en la más alta(Nougat).

7.2 Conclusiones de las pruebas efectuadas.

La tabla 17 muestra el resultado obtenido por las pruebas efectuadas.

Prueba	Resultado
Visualización plano geográfico.	Satisfactorio.
Sistema de posicionamiento global.	Medio.
Visualización marcadores.	Satisfactorio.
Visualización ruta.	Medio.
Visualización distancia.	Satisfactorio.
Visualización tiempo de tardanza.	Satisfactorio.
Barra buscadora	Satisfactorio.
Descripción lugares.	Satisfactorio.
Perfil administrador.	Satisfactorio.
Eficiencia.	Satisfactorio en condiciones normales.
Seguridad.	Satisfactorio.
Usabilidad.	Medio.
Portabilidad.	Satisfactorio.

Tabla 17. Resultado pruebas efectuadas.

1.- En general, las pruebas realizadas anteriormente indican que el sistema fue desarrollado correctamente, puesto que con respecto a la opinión de las personas que han utilizado la aplicación, la han catalogado como “bonita” e interesante con respecto al propósito que tiene, sin embargo, han encontrado que las rutas no son muy atractivas visualmente y que, además, la aplicación no explica mucho sobre su utilización, pero finalmente, le han dado el visto bueno.

2.- Existen distintos eventos que podrían ocurrir a un usuario, por ejemplo, que el sistema se desconecte de internet o de la ubicación actual, y es por lo mismo que se intentan realizar las pruebas correspondientes y detectar todos los casos posibles; por lo mismo, es necesario tener un buen plan de pruebas para poder encontrar y corregir rápidamente los fallos que existan, fortaleciendo finalmente la calidad de la aplicación.

3.- Los fallos y errores que se generan y que han sido mencionados es debido a la poca experiencia que existe con la programación en Android, sin embargo, este problema se va solucionando a medida que se programa los requisitos y se investiga en internet el cómo realizar ciertas funciones.

4.- En general, se puede concluir que el uso de prueba junto a una buena planificación permite encontrar fallos en el sistema de una forma más rápida por ende más tiempo para poder resolverlas. Además, es necesario poder dar prioridad a todos los elementos del sistema (modelo, vista y controlador), puesto que cualquier fallo, error o bug hace que la aplicación pierda calidad.

8 Planes de implantación y puesta en marcha del sistema.

El proceso de implantación constituye a la última fase de la metodología de desarrollo de software y es posterior al proceso de prueba.

El objetivo de esta fase es realizar las actividades necesarias para poner a disposición de los usuarios el sistema de software (alfarosolis, 2018).

Para que el sistema funcione en la Universidad y en general en cualquier lugar, se sube la aplicación móvil a un servidor hosting.

Posteriormente de generada la implantación del sistema, los usuarios desde ya pueden comenzar a hacer uso del aplicativo y de sus correspondientes funcionalidades, y mientras tanto se hace supervisión de los resultados obtenidos.

También se hace una correspondiente capacitación a usuarios (abordado en el capítulo 9) para que estos puedan entender y hacer un uso correcto de la aplicación.

Terminando el periodo de pruebas, se inicia lo que es la puesta en marcha y se deja a todos los usuarios interactuar libremente con la aplicación móvil.

Finalmente, después de la puesta en marcha se debe realizar una mantención de la aplicación móvil y como se mencionada en el capítulo de la factibilidad económica, se debe realizar semestralmente, unos 4 días, para poder corregir errores si existiesen y agregar o modificar la información correspondiente a los lugares de la Universidad que día a día se expande más y más.

9 Plan de capacitación a usuarios.

Después de instalar y poner en marcha la aplicación móvil, es necesario y corresponde capacitar a éstos para que sepan utilizar de manera adecuada el nuevo sistema.

En general las capacitaciones a usuarios son de manera presencial, sin embargo, en este caso se capacita a través de un manual específico detallado en el anexo C, pues el sistema es fácil de utilizar como para realizar una charla o reunión.

Por otra parte, con respecto a la administración de la aplicación, es importante realizar una reunión con los dueños del producto, pues les corresponde entender completamente todas las funcionalidades del sistema y así hacer un buen uso de la aplicación, sin embargo, se decide que el administrador será el desarrollador de la aplicación, por lo que cualquier modificación al sistema, se debe contratar los servicios del mismo.

10 Conclusiones.

1.- En ocasiones los informáticos creen que solo es necesaria una sola reunión con el cliente para poder realizar un proyecto y luego no incluirlo más hasta el término de este. Es por lo mismo que muchos fracasan en el intento, no terminando el proyecto o bien desarrollando algo que no era lo que realmente el cliente quería. Incluir al cliente en el proyecto es clave ya que finalmente el entregable es para él y es necesario lograr satisfacer todas sus necesidades, por ello es importante en una primera instancia agendar las citas que sean necesarias hasta entender completamente lo que quiere y finalmente ir mostrando los avances para ver si el proyecto va bien encaminado o no, de esta manera se logra un buen proyecto y finalmente la satisfacción del cliente.

2.- Al no tener los conocimientos necesarios en Android, se tuvo que ser autodidacta, por lo mismo se debió hacer búsqueda de información para cada implementación que se quería realizar en la aplicación. Ser autodidacta en estos tiempos es muy importante, ya que en el mundo laboral se requieren de estas personas que puedan buscar información a través de sus propios medios.

3.- La importancia del cálculo del VAN en un proyecto es esencial, ya que indican si el negocio será o no rentable, viable o exitoso en el futuro. Lograr obtener un VAN positivo indica que en un plazo estimado (5 años, por ejemplo) se puede recuperar la inversión que ha puesto en marcha el negocio y posterior a esto obtener beneficios. Medir la viabilidad de un proyecto es fundamental para saber si una empresa o cliente debe apostar de lleno y arriesgarse por un proyecto, es por lo mismo que es importante hacer un buen análisis financiero y ver cómo está el mercado actual (si el proyecto existe o es similar).

4.- Con respecto a la metodología de desarrollo, la utilización de Kanban Personal, fue de mucha importancia, pues permitió optimizar los tiempos, lo cual se traduce en eficiencia. Además, de los múltiples cambios que podemos hacer durante el proceso de desarrollo, es otro de los factores que nos permite no tener que volver

a empezar prácticamente desde cero al detectar errores o querer implementar mejoras.

5.- De forma personal, la realización de este proyecto fue muy importante, ya que al ser el último año de carrera y próximo a ser un trabajador de esta área, sirvió para mejorar el nivel en programación, así como conocer de las nuevas tecnologías existentes, generando motivación para seguir estudiando y desarrollando proyectos de esta índole.

6.- Se puede decir que, el proyecto se ha realizado correctamente, pues se cumple con los objetivos propuestos y los requerimientos solicitados.

7.- Finalmente decir que se cumple con las expectativas personales, esto debido al desarrollo completo de la aplicación, la investigación constante y la motivación a seguir desarrollando.

11 Referencias bibliográficas.

- Fundación Wikimedia, I. (s.f.). WikipediA. Obtenido de <https://es.wikipedia.org/wiki/JMeter>
- AdMob. (2018). Obtenido de <https://www.google.es/admob/>
- ALEGSA. (2018). ALEGSA. Obtenido de <http://www.alegsa.com.ar/Dic/filezilla.php>
- alfarosolis. (2018). alfarosolis. Obtenido de <http://www.alfarosolis.com/content/PDFs/IF7100/Semana13/Implantacion.pdf>
- Antevenio, S. (2018). antevenio. Obtenido de <https://www.antevenio.com/blog/2016/09/como-colgar-una-aplicacion-en-google-play/>
- ATLASSIAN. (2018). Trello. Obtenido de <https://trello.com/logged-out>
- Bío-Bío, U. d. (2018). ubiobio. Obtenido de https://www.ubiobio.cl/w/#Vision_y_Mision
- Bío-Bío, U. d. (2018). ubiobio. Obtenido de http://www.ubiobio.cl/miweb/web2012.php?id_pagina=5152
- Bío-Bío, U. d. (2018). ubiobio. Obtenido de http://ubiobio.cl/w/#Tour_Virtual
- Bío-Bío, U. d. (2018). ubiobio. Obtenido de <http://ubiobio.cl/admision/>
- Cevallos, K. (2018). INGENIERÍA DEL SOFTWARE. Obtenido de <https://ingsoftwarekarlacevallos.wordpress.com/2015/06/04/uml-casos-de-uso/>
- Concept, C. (2018). JSON Formatter & Validator. Obtenido de <https://jsonformatter.curiousconcept.com/#>
- Corporation, L. (2018). SlideShare. Obtenido de <https://es.slideshare.net/varingenieria/descripcion-caso-de-uso>
- Economipedia. (2018). economipedia. Obtenido de <http://economipedia.com/definiciones/valor-actual-neto.html>
- Economipedia. (2018). economipedia. Obtenido de <http://economipedia.com/definiciones/tasa-interna-de-retorno-tir.html>
- EcuRed. (2018). Obtenido de https://www.ecured.cu/Metodologias_de_desarrollo_de_Software
- Fundación Wikimedia, I. (2018). WikipediA. Obtenido de https://es.wikipedia.org/wiki/Sublime_Text
- Fundación Wikimedia, I. (2018). WikipediA. Obtenido de <https://es.wikipedia.org/wiki/Notepad%2B%2B>

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de https://es.wikipedia.org/wiki/Android_Studio

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de https://es.wikipedia.org/wiki/Java_Development_Kit

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de <https://es.wikipedia.org/wiki/PhpMyAdmin>

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de [https://es.wikipedia.org/wiki/Java_\(lenguaje_de_programaci%C3%B3n\)](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n))

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de https://es.wikipedia.org/wiki/Extensible_Markup_Language

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de <https://es.wikipedia.org/wiki/SQL>

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de <https://es.wikipedia.org/wiki/PHP>

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de https://es.wikipedia.org/wiki/Business_Process_Model_and_Notation

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de https://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de https://es.wikipedia.org/wiki/Diagrama_de_casos_de_uso

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de https://es.wikipedia.org/wiki/Modelo_entidad-relaci%C3%B3n

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de <https://es.wikipedia.org/wiki/Singleton>

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de <https://es.wikipedia.org/wiki/JSON>

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de <https://es.wikipedia.org/wiki/Modelo%20%80%93vista%20%80%93controlador>

Fundación Wikimedia, I. (2018). WikipediA. Obtenido de [https://es.wikipedia.org/wiki/Kanban_\(desarrollo\)](https://es.wikipedia.org/wiki/Kanban_(desarrollo))

Inc., F. T. (2018). PdaNet. Obtenido de <http://pdanet.co/>

Inc., F. T. (2018). PdaNet. Obtenido de <http://pdanet.co/>

- Inc., L. S. (2018). Lucidchart. Obtenido de <https://www.lucidchart.com/pages/es>
- Inc., L. S. (2018). Lucidchart. Obtenido de <https://www.lucidchart.com/pages/es?noHomepageRedirect=true>
- Indeed. (2018). indeed. Obtenido de <https://www.indeed.cl/salaries/Ingeniero/a-inform%C3%A1tico/a-Salaries>
- ITNT. (2018). Soft32. Obtenido de <https://microsoft-project-professional.soft32.es/descarga-gratuita/?dm=0>
- JoaquinOriente. (2018). joaquinoriente. Obtenido de <http://joaquinoriente.com/2013/07/20/los-siete-principios-de-las-pruebas-software/>
- Mis Negocios En Internet. (2018). Obtenido de <https://www.misnegocios.co/2014/02/hosting-gratuito-000webhost.html>
- phpMyAdmin. (2018). Obtenido de <http://localhost/phpmyadmin/index.php>
- pmoinformatica. (2018). PMOinformatica. Obtenido de <http://www.pmoinformatica.com/2017/02/pruebas-de-caja-negra-ejemplos.html>
- Programación, H. (2018). hermosaprogramacion. Obtenido de <http://www.hermosaprogramacion.com/2015/02/android-volley-peticiones-http/>
- programafacil. (s.f.). programafacil. Obtenido de <https://programafacil.com/podcast/19-patron-modelo-vista-controlador/>
- S.A., P. F. (2018). PCFactory. Obtenido de <https://www.pcfactory.cl/producto/29599-smartphone-galaxy-j1-ace-lte-ve-quad-core-8gb-4-3-android-negro-claro-qr>
- SÁEZ, F. (2018). FacileThings. Obtenido de <https://facilethings.com/blog/es/personal-kanban>
- Sergio, G. y. (2018). Especificación de las pruebas. Chillán.
- SOLOTODO. (2018). Obtenido de <https://www.solotodo.com/products/23110-own-s4035-3g-negro>
- WikipediA. (2018). Obtenido de <https://es.wikipedia.org/wiki/Trello>
- Ziff Davs, L. (2018). PCMag. Obtenido de <https://latam.pcmag.com/>

12 anexos.

A continuación, se muestran los tiempos que se han invertido en este proyecto, así como también las especificaciones de las pruebas y el manual de usuario.

Anexo A. Planificación inicial vs. Esfuerzo real invertido.

En la figura 41 y figura 42, se muestra el proceso de inicio y fin del proyecto, así como el esfuerzo real que se ha utilizado.

Planificación inicial:
























		Task Mode	Task Name	Duration	Start	Finish
1			Inicio	50 days	Mon 24-09-18	Fri 30-11-18
2			Elaboración Informe	4 days	Mon 24-09-18	Thu 27-09-18
3			Investigación	5 days	Fri 28-09-18	Thu 04-10-18
4			Investigación de información relevante de los lugares de la Universidad	2 days	Fri 28-09-18	Mon 01-10-18
5			Estudio e investigación de funciones importantes para la realización de la aplicación	3 days	Tue 02-10-18	Thu 04-10-18
6			Diseño	10 days	Fri 05-10-18	Thu 18-10-18
7			Diseño del modelo de datos	3 days	Fri 05-10-18	Tue 09-10-18
8			Diseño de prototipo	3 days	Wed 10-10-18	Fri 12-10-18
9			Diseño de vistas o interfaz gráfica	3 days	Mon 15-10-18	Wed 17-10-18
10			Documentación diseño	1 day	Thu 18-10-18	Thu 18-10-18
11			Construcción del software	13 days	Fri 19-10-18	Tue 06-11-18
12			Programación del código	9 days	Fri 19-10-18	Wed 31-10-18
13			Pruebas del software	2 days	Thu 01-11-18	Fri 02-11-18
14			Documentación del código	2 days	Mon 05-11-18	Tue 06-11-18
15			Implementación	3 days	Thu 08-11-18	Mon 12-11-18
16			Pruebas de implementación	2 days	Thu 08-11-18	Fri 09-11-18
17			Entrega del software	1 day	Mon 12-11-18	Mon 12-11-18
18			Carta de cierre del	1 day	Fri 30-11-18	Fri 30-11-18

Figura 41. Planificación inicial y final en Carta Gantt (ITNT., 2018).

En la figura 41 se aprecia que, la duración del proyecto fue de 50 días, los cuales comienzan desde el 24 de septiembre hasta el 30 de noviembre, sin embargo, se estimaba poder terminar el proyecto antes de la entrega final; el 12 de noviembre y así tener días para hacer revisiones y correcciones al proyecto desarrollado.

Esfuerzo real invertido:
























		Task Mode	Task Name	Duration	Start	Finish
1			Inicio	50 days	Mon 24-09-18	Fri 30-11-18
2			Elaboración Informe	4 days	Mon 24-09-18	Thu 27-09-18
3			Investigación	8 days	Fri 28-09-18	Tue 09-10-18
4			Investigación de información relevante de los lugares de la Universidad	3 days	Fri 28-09-18	Tue 02-10-18
5			Estudio e investigación de funciones importantes para la realización de la aplicación	5 days	Wed 03-10-18	Tue 09-10-18
6			Diseño	14 days	Wed 10-10-18	Mon 29-10-18
7			Diseño del modelo de datos	4 days	Wed 10-10-18	Mon 15-10-18
8			Diseño de prototipo	4 days	Tue 16-10-18	Fri 19-10-18
9			Diseño de vistas o interfaz gráfica	5 days	Mon 22-10-18	Fri 26-10-18
10			Documentación diseño	1 day	Mon 29-10-18	Mon 29-10-18
11			Construcción del software	19 days	Tue 30-10-18	Fri 23-11-18
12			Programación del código	14 days	Tue 30-10-18	Fri 16-11-18
13			Pruebas del software	3 days	Mon 19-11-18	Wed 21-11-18
14			Documentación del código	2 days	Thu 22-11-18	Fri 23-11-18
15			Implementación	4 days	Mon 26-11-18	Thu 29-11-18
16			Pruebas de implementación	3 days	Mon 26-11-18	Wed 28-11-18
17			Entrega del software	1 day	Thu 29-11-18	Thu 29-11-18
18			Carta de cierre del proyecto	1 day	Fri 30-11-18	Fri 30-11-18

Figura 42. Esfuerzo real invertido en Carta Gantt (ITNT., 2018).

En la figura 42 se aprecia que, con respecto a la planificación inicial, la realización de las tareas termina justo el día de entrega, y en el cual comienza desde el 24 de septiembre hasta el 30 de noviembre; se ocuparon completamente los 50 días.

Anexo B. Especificación de pruebas de software.

La tabla 18 muestra la prueba de sistema: Desplazarse por el mapa.

ID prueba.	P02.	Fecha	20-11-2018.	
Caso de uso.	Desplazarse por el mapa (Tabla 10).			
Objetivo.	Verificar que el usuario pueda desplazarse correctamente por un mapa visible y que pueda ver su posición actual y marcadores en el mapa.			
Tipo de prueba.	Caja negra.			
Prueba.	Datos de entrada.	Resultado esperado.	Resultado obtenido.	Evaluación.
Datos válidos.	GPS activado y conectado a internet.	Visualización de un correcto mapa, de su posición actual y de los marcadores correspondientes.	IDEM Resultado esperado.	Resultado aprobado.
Datos no válidos.	GPS desactivado y no conectado a internet.	El sistema verifica que haya conexión a internet y esté activado el GPS.	Mensaje de alerta: "Conéctese a internet" y ventana emergente para la activación del GPS.	Resultado aprobado.

Tabla 18. Prueba de sistema: Desplazarse por el mapa (Sergio, 2018).

La tabla 19 muestra la prueba de sistema: Consultar ubicación.

ID prueba.	P04.	Fecha.	20-11-2018.	
Caso de uso.		Consultar ubicación (Tabla 11).		
Objetivo.		Verificar que el sistema muestre la correcta ubicación por el lugar que se consulta.		
Tipo de prueba.		Caja negra.		
Prueba.	Datos de entrada.	Resultado esperado.	Resultado obtenido.	Evaluación
Datos válidos.	“Tipear” en el buscador el lugar que desea encontrar.	Si el lugar existe, con solo 3 caracteres ingresados la barra se autocompleta con los lugares existentes.	Aparece la ruta de acceso, la distancia y el tiempo de llegada al lugar buscado.	Resultado aprobado.
Datos no válidos.	Caracteres incoherentes.	El sistema no muestra ningún lugar a través de la barra buscadora.	Mensaje diciendo: “El lugar no existe, intente otro”.	Resultado aprobado.

Tabla 19. Prueba de sistema: Consultar ubicación (Sergio, 2018).

La tabla 20 muestra la prueba de sistema: Consultar descripción.

ID prueba.	P05.	Fecha.	20-11-2018.	
Caso de uso.		Consultar descripción (Tabla 12).		
Objetivo.		Verificar que se entrega la información correcta del lugar que se ha consultado.		
Tipo de prueba.		Caja negra.		
Prueba.	Datos de entrada.	Resultado esperado.	Resultado obtenido.	Evaluación.
Datos válidos.	Seleccionar el cuadro de texto arrojado por un marcador buscado.	El sistema muestra en una nueva ventana la información correspondiente al lugar consultado.	IDEM Resultado esperado.	Resultado aprobado.
Datos no válidos.	Se consulta por sin internet por una ubicación la cual su información está contenida en una base de datos.	El sistema verifica que haya conexión a internet para poder ver la información.	Mensaje de alerta: "Conéctese a internet para poder ver la información del lugar".	Resultado aprobado.

Tabla 20. Prueba de sistema: Consultar descripción (Sergio, 2018).

La tabla 21 muestra la prueba de sistema: Crear marcador.

ID prueba.	P06.	Fecha.	20-11-2018.	
Caso de uso.		Crear marcador (Tabla 13).		
Objetivo.		Verificar que se crea correctamente un marcador.		
Tipo de prueba.		Caja negra.		
Prueba.	Datos de entrada.	Resultado esperado.	Resultado obtenido.	Evaluación.
Datos válidos.	Se selecciona el mapa para crear un marcador y se llenan los campos con información correspondiente al lugar.	Se crea correctamente el marcador en el mapa y se actualiza inmediatamente.	Se crea el marcador en el mapa, se muestra inmediatamente y se arroja un mensaje diciendo: "Marcador creado exitosamente".	Resultado aprobado.
Datos no válidos.	Se selecciona crear sin conexión a internet.	El sistema verifica que haya conexión a internet.	Mensaje de alerta: "Conéctese a internet para crear el marcador".	Resultado aprobado.
Datos no válidos.	Se ingresa un título que ya existe en la base de datos.	El sistema verifica si existe un marcador con título igual en la base de datos.	Mensaje informativo diciendo: "El marcador ya existe, intente otro".	Resultado aprobado.

Tabla 21. Prueba de sistema: Crear marcador (Sergio, 2018).

La tabla 22 muestra la prueba de sistema: Editar marcador.

ID prueba.	P07.	Fecha.	20-11-2018.	
Caso de uso.		Editar marcador (Tabla 14).		
Objetivo.		Verificar que se ha editado correctamente un marcador.		
Tipo de prueba.		Caja negra.		
Prueba.	Datos de entrada.	Resultado esperado.	Resultado obtenido.	Evaluación.
Datos válidos.	Se selecciona un marcador y se edita la información arrojada por éste.	Visualización del marcador con los datos que se han editado.	Visualización inmediata del marcador editado y un mensaje diciendo: "Marcador editado exitosamente".	Resultado aprobado.
Datos no válidos.	Se selecciona editar cuando no hay conexión a internet.	El sistema verifica que haya conexión a internet.	Mensaje de alerta: "Conéctese a internet si desea editar el marcador".	Resultado aprobado.
Datos no válidos.	Se selecciona editar cuando se ha cambiado el título del marcador con uno existente.	El sistema verifica si existe un marcador en la base de datos con el título que se ha ingresado.	Mensaje diciendo: "El título con este marcador ya existe, intente otro".	Resultado aprobado.

Tabla 22. Prueba de sistema: Editar marcador (Sergio, 2018).

La tabla 23 muestra la prueba de sistema: Borrar marcador.

ID prueba.	P08.	Fecha.	20-11-2018.	
Caso de uso.		Borrar marcador (Tabla 15).		
Objetivo.		Verificar que se ha borrado correctamente un marcador en el mapa.		
Tipo de prueba.		Caja negra.		
Prueba.	Datos de entrada.	Resultado esperado.	Resultado obtenido.	Evaluación.
Datos válidos.	Se selecciona un marcador y luego el botón borrar.	Se elimina el marcador y se actualiza el mapa.	Mensaje diciendo: "Marcador eliminado exitosamente" y se actualiza el mapa inmediatamente.	Resultado aprobado.
Datos no válidos.	Se confirma la eliminación del marcador cuando no hay conexión a internet.	El sistema verifica que haya conexión.	Mensaje de alerta: "Conéctese a internet si desea borrar el marcador."	Resultado aprobado.

Tabla 23. Prueba de sistema: Borrar marcador (Sergio, 2018).

La tabla 24 muestra la prueba de sistema: Cambiar usuario y contraseña.

ID prueba.	P09.	Fecha.	20-11-2018.	
Caso de uso.	Cambiar usuario y contraseña (Tabla 16).			
Objetivo.	Verificar que se ha cambiado correctamente el usuario y la clave del administrador.			
Tipo de prueba.	Caja negra.			
Prueba.	Datos de entrada.	Resultado esperado.	Resultado obtenido.	Evaluación.
Datos válidos.	Cambiar el usuario o contraseña y confirmar el diálogo que aparece cuando selecciona el botón cambiar.	Se cambia el usuario y contraseña y se muestra inmediatamente en la ventana actual .	Se cambia el usuario o contraseña, se visualiza inmediatamente y se muestra un mensaje diciendo: "Usuario o contraseña cambiado".	Resultado aprobado.
Datos no válidos.	Confirmar el diálogo para cambiar el usuario o contraseña sin conexión a internet.	El sistema verifica que haya conexión a internet.	Mensaje de alerta: "Conéctese a internet si desea cambiar el usuario o contraseña".	Resultado aprobado

Tabla 24. Prueba de sistema: Cambiar usuario y contraseña (Sergio, 2018).

Anexo C Manual usuario.

A continuación, se presenta el manual de usuario tanto para el administrador como para un usuario normal que utilice la aplicación.

Administrador aplicación.

Primero que todo mencionar que la aplicación se debe bajar desde Play Store o si se le ha sido proveída, instalar la aplicación a través del archivo con extensión “.apk”.

Paso 1: Abrir la aplicación y esperar a que avance automáticamente hasta el menú principal.

Paso 2: Seleccionar la opción “Login administrador” mostrada en el menú principal y ubicada en la esquina superior derecha de la pantalla.

Paso 3: Ingresar en los campos de textos habilitados, su usuario y contraseña proveídos por el desarrollador de la aplicación. Recuerda que estos datos inicialmente son “admin” y “admin” respectivamente y que, además, se puede cambiar dentro de la aplicación una vez ingresado al perfil de administrador.

Paso 4: Interactuar con alguna de las opciones desarrolladas en la aplicación y mostradas en el menú del perfil de administrador, las cuales son: Crear un marcador, editar o borrar un marcador y cambiar la configuración del administrador.

Paso 5: Es primordial entender que, para poder realizar estas actividades, es necesario tener una conexión a internet, sino no se puede realizar. En el caso de crear, editar y borrar un marcador, para una mejor experiencia de usuario, se recomienda tener activado el GPS para poder saber su ubicación exacta y de esa forma realizar una buena operación a estas actividades.

Paso 6: En el caso de escoger la opción de crear un marcador, se muestra por pantalla un mapa geográfico ubicado y enfocado en la Universidad, siempre que no tengas el GPS activado, en caso contrario se enfoca en la posición actual con un zoom bien alejado para que se logre ver ambos escenarios. Se menciona, además,

que, si no tiene el GPS activado en el momento de entrar al mapa, se muestra un diálogo de activación de la ubicación actual y en el caso de no tener internet no se deja iniciar.

Paso 6.1: Si se ha seguido correctamente las instrucciones, en el mapa se visualiza todos los marcadores que se han creado anteriormente y que están localizados en el interior de la Universidad.

Paso 6.2: Una vez conocido la posición de todos los marcadores, se puede crear uno presionando la pantalla unos segundos en el lugar que se quiera y automáticamente se crea un marcador de color rojo en el mapa. Posteriormente al aparecer este marcador, los demás marcadores puestos en el mapa desaparecen, sin embargo, se puede volver a colocarlos si se desea con uno de los botones que se muestra en ese momento.

Paso 6.3: Una vez puesto tu marcador en el mapa, es la hora de rellenar los campos correspondientes a este marcador con información correspondiente al lugar que se quiere mostrar. La información corresponde a Título, subtítulo, latitud, longitud y la información con una posibilidad de insertar máximo 500 caracteres. La longitud y la latitud corresponde a la posición del marcador puesta en ese momento y que puede ser cambiada seleccionado en el marcador y luego arrastrándolo por el mapa, por lo que automáticamente se actualiza estos dos campos.

Paso 6.4: Al tener todos los campos completos, es la hora de crear el marcador, y para ello se debe seleccionar el botón “crear” y la transacción se completa efectivamente siempre y cuando el título no existe con anterioridad con alguno de los marcadores puestos en el mapa; por lo mismo es necesario saber cuáles ya existen.

Paso 7: En el caso que se quiera “editar un marcador”, se selecciona cualquiera de los marcadores que estén dispuestos en ese momento en el mapa. Una vez hecha esta acción, automáticamente se coloca en los campos de texto la información correspondiente a ese lugar. En ese momento se puede cambiar la posición del

marcador, o bien cambiar solo el subtítulo de éste; los demás campos están deshabilitados.

Paso 8: Si se desea borrar un marcador, basta con seleccionar unos de los marcadores en el mapa, y luego una vez que se hayan completado automáticamente los campos de textos, se puede seleccionar el “botón borrar”. Tanto para borrar como actualizar un marcador es necesario confirmar la operación en un diálogo de texto que aparece una vez seleccionadas estas dos opciones.

Paso 9: La última opción en el menú del perfil de administrador es la configuración de administrador, la cual se puede hacer el cambio respectivo tanto de usuario como de contraseña para poder ingresar a este perfil, menú y opciones.

Paso 9.1: Al igual como en los otros casos, para hacer efectivo el cambio de usuario y contraseña es necesaria la conexión a internet y confirmar el diálogo que aparece una vez apretado el botón “cambiar”. La contraseña y usuario actual siempre aparecen en esta interfaz para recordarlas.

Usuario aplicación.

Primero que todo mencionar que la aplicación se debe bajar desde Play Store o si se le ha sido proveída, instalar la aplicación a través del archivo con extensión “.apk”.

Paso 1: Abrir la aplicación y esperar a que avance automáticamente hasta el menú principal.

Paso 2: Seleccionar el botón “Comenzar” para entrar a la interfaz que muestra el mapa geográfico posicionado en la Universidad o bien en su ubicación actual, siempre y cuando se tenga activado el GPS, en caso de no tenerlo se muestra una ventana emergente para activar dicha propiedad.

Paso 3: Una vez dentro del mapa, se visualiza todos los marcadores que se han creado y que están puestos en la Universidad, representado un espacio, departamento, instalación, etc.

Paso 4: En el mapa se muestra un “+” y “-”, que representan el zoom de la pantalla, para lo cual se recomienda utilizarlos para acercarse al mapa y ver con más detalles la posición de estos marcadores.

Paso 5: Usted como usuario, puede deslizarse por el mapa y encontrar algún lugar que busca, seleccionado en los diferentes marcadores, o bien puede utilizar la barra buscadora para el mismo propósito.

Paso 6: Ya sea seleccionando un marcador o bien buscado a un lugar a través de la barra buscadora, en ese instante se crea una ruta que parte desde la entrada de la Universidad y llega hasta el lugar que se ha buscado. También, se muestra en ese momento la distancia de dicha ruta y los tiempos tanto en bicicleta como a pie aproximados que se demora en llegar a ese lugar.

Paso 7: una vez encontrado el lugar, se menciona que, el marcador despliega un cuadro de texto mostrando el nombre del lugar del establecimiento y un “subtítulo” que identifica también el lugar.

Paso 8: Usted como usuario tiene la posibilidad de seleccionar dicho cuadro de texto, lo cual lo guía hacia una nueva ventana que muestra información relevante e importante de dicho lugar.

Paso 9: Finalmente, usted tiene la posibilidad de conocer la ubicación e información de más de 25 lugares repartidos el Campus Fernando May.