ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA EN SISTEMAS

DESARROLLO DE UNA APLICACIÓN MÓVIL PARA EL APOYO A LA BÚSQUEDA DE PERSONAS DESAPARECIDAS EN EL DISTRITO METROPOLITANO DE QUITO

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

DAVID ENRIQUE MONCAYO BALSECA
davidmoncayo4@gmail.com

DIRECTOR: TANIA ELIZABETH CALLE JIMÉNEZ
tania.calle@epn.edu.ec

Quito, Diciembre 2018

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por David Enrique Moncayo Balseca, bajo mi supervisión.

Tania Elizabeth Calle Jiménez

DIRECTOR DE PROYECTO

DECLARACIÓN

Yo, David Enrique Moncayo Balseca, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

David Enrique Moncayo Balseca

DEDICATORIA

A mi Señor, por cuya misericordia estoy aquí, Él sonríe al vernos. A papi Jorge, que ha sido una bendición tan grande para mi vida, por su amor y corrección cuando más lo necesito. A mi madre bella, Lolita, gracias por estar siempre dispuesta a ayudarme, gracias por sus oraciones e ímpetu, esto me ha traído hasta aquí. A cada uno de mis hermanos y hermanas incluyendo mis cuñados que los considero mis hermanos también: César, Belén, Isabel, Jorge, Mayrita, Sebastián, Jenny y Pato, son lo máximo, estoy agradecido con cada uno, los amo. A mi familia, que me ha dado apoyo, amor y consejos sabios, no se que sería de mí sin ellos, los quiero mucho. A mis amigos, que se han dado modos de ayudarme en este proceso, en especial a Paúl y Ronny.

AGRADECIMIENTO

A Dios por sobre todas las cosas, que me ha dado todo lo que necesito y me ha rodeado de personas que han sido de gran bendición a mi vida. A mi familia por su amor incondicional, por sus oraciones y apoyarme en todo este proceso universitario, por cada corrección y afecto brindado que me ha hecho crecer como persona. A mi líder, que ha sido un apoyo incondicional, por sus consejos y oraciones, gracias porque he sido librado de muchas cosas a través de ti. A cada amigo, que me ha dado su apoyo, ánimo o puesto las pizzas. Dios me ha regalado hermosos momentos junto a ustedes. A cada persona que conforma nuestra querida Politécnica Nacional que ha sido un impulso para crecer y superarnos como estudiantes, en especial a mi tutora que ha sido una tremenda ayuda y facilitadora en la consecución de este trabajo. Dios les recompense.

Contenido

RESUMEN ix

ABSTRACT ix

1. INTRODUCCIÓN 10

1.1. Justificación 10

1.2. Estado del arte 11

Proyecto Emilia/ Alerta Emilia 12

Tecnología para localización de personas accidentadas en lugares sin cobertura 12

App Finder 13

Alpify o Safe 365 14

2. METODOLOGÍA 15

2.1. Equipo Scrum 15

2.2. Eventos Scrum 15

El Sprint 16

Planeación de Sprint 16

Revisión del Sprint 17

2.3. Artefactos de Scrum 17

Lista de Producto (Product Backlog) 18

Lista de Pendientes del Sprint (Sprint Backlog) 18

Incremento 18

2.4. Requerimientos de la aplicación 19

2.5. Herramientas por utilizarse 20

Trello 20

Editor de texto 21

Android Studio 21

Google Maps Platform 21

Firebase Firestore 21

2.6. Implementación 23

2.7. Historias de usuario preliminares 23

2.8. Sprint 1 28

Durante el Sprint 28

Revisión del Sprint 35

2.9. Sprint 2 36

Durante el Sprint 38

Revisión del Sprint ... 46
2.10. Sprint 3 ... 47
Durante el Sprint ... 47
Revisión del Sprint ... 54
2.11. Sprint 4 ... 54
Durante el Sprint ... 55
Revisión del Sprint ... 58
2.12. Sprint 5 ... 59
Durante el Sprint ... 61
Revisión del Sprint ... 65
3. RESULTADOS Y DISCUSIÓN ... 66
3.1. Análisis de resultados ... 68
Sección 1: Registrarse ... 68
Sección 2: Seleccionar búsqueda ... 69
Sección 3: Entrar al mapa y comenzar a rastrear su posición ... 70
Sección 4: Explore las alertas ... 72
Sección 5: Visualice las alertas en el mapa ... 73
4. CONCLUSIONES ... 78
5. RECOMENDACIONES ... 80
Bibliografía ... 81
ANEXO 1: Distribución de sistemas operativos móviles en Ecuador a noviembre 2018 ... 85
ANEXO 2: Solicitud al Sr. Victor Manuel Araus Macias, General de Distrito y Director Nacional de la DINASED para realización de pruebas ... 86
ANEXO 3: Memorando 2018-3319-DINASED-PN del Despacho del Sr. Victor Manuel Araus Macias, General de Distrito y Director Nacional de la DINASED ... 88
ANEXO 4: Ejemplo de Prueba de Usabilidad de la aplicación tomada a un colaborador, miembro de la DINASED ... 89
ANEXO 5: Fotografías de la Exposición y Taller llevado a cabo el martes 4 de diciembre de 2018 ... 92

Índice de figuras

Figura 1 Menú Principal 29
Figura 2 Pantalla de Registro 30
Figura 3 Pantalla de Inicio de Sesión 31
Figura 4 Pantalla Mapa y Menú lateral 32
Figura 5 Código en el layout de los elementos de MapsActivity 34
Figura 6 Implementación del mapa 34
Figura 7 Llamada asíncrona de un mapa de Google Maps 34
Figura 8 Algoritmo para dibujar las líneas en el mapa según la distancia recorrida a partir del punto anterior 39
Figura 9 Aplicación rastreando la localización del dispositivo 41
Figura 10 Aplicación al dejar de rastrear la localización 42
Figura 11 Estructura de la base de datos NOSql 43
Figura 12 Función que obtiene los "tracks" de los usuarios 45
Figura 13 Camino obtenido de un usuario junto con última localización 45
Figura 14 Mensaje de error al ingresar una cédula inválida 49
Figura 15 Información de usuario en el módulo de administración Firebase Authentication 49
Figura 16 Toast con tiempo amigable al usuario 51
Figura 17 Estructura de una búsqueda activa en la base de datos 52
Figura 18 Búsquedas activas a las que un usuario está ligado 53
Figura 19 Implementación de búsquedas activas 54
Figura 20 Diseño de la pantalla de Crear Alerta 56
Figura 21 Lista de diferentes tipos de alertas generadas con sus estados 57
Figura 22 Pantalla Mis Alertas, mostrando alertas generadas por el usuario 58
Figura 23 Lista de la segunda pestaña mostrando las personas encontradas 62
Figura 24 Botón de acción flotante (fab) naranja para añadir nuevas búsquedas 63
Figura 25 Formulario de creación de búsquedas 64
Figura 26 Editor de traducciones de Android Studio 65
Figura 27 Movimiento del usuario de la aplicación 71
Figura 28 Consumo de datos de la app en 20min 75
Figura 29 Carga mínima y pico del CPU con la aplicación ejecutándose en segundo plano 76

RESUMEN

El presente proyecto integrador busca optimizar los recursos y unificar el esfuerzo de la DINASED, familiares y voluntarios que buscan a personas desaparecidas por medio de una aplicación móvil que muestre en tiempo real los lugares registrados y hallazgos, los cuales se visualizan desde que se empieza la búsqueda. Esta Aplicación información relevante como trayectorias recorridas, alertas sobre la búsqueda y ubicación de los buscadores en un mapa interactivo y fácil de usar.

La aplicación fue construida de forma iterativa utilizando la metodología Scrum y es un aporte a la tecnología actual en cuanto a la problemática de personas desaparecidas. Utiliza servicios de Google; Mapas para Android para mostrar un mapa personalizable donde se muestra la información de la búsqueda. Firebase Firestore que es una base de datos NoSQL en la nube con capacidades de sincronización en tiempo real que evita la necesidad de un servidor.

ABSTRACT

The present integrative project seeks to optimize resources and unify the effort of DINASED, family members and volunteers who are looking for missing people through a mobile application which shows, in real time, the registered places and the findings which are shown since the search begins. This App shows relevant information such as roads traveled, alerts about the search and the location of the seekers on an interactive and easy to use map.

The application was built in an iterative way using the Scrum methodology and it is a contribution to current technology boarding the problem of missing people. It uses Google services such as Maps for Android in order to show a customizable map with search information. Also, Firebase Firestore, a cloud, NoSQL database with real time synchronization capabilities which avoids the need of a server.

1. INTRODUCCIÓN

En el Ecuador los reportes de personas desaparecidas son notificados a la Policía Nacional o al Servicio Integrado de Seguridad ECU 911 (ECU 911), las autoridades recomiendan acudir al Unidad de Policía Comunitaria (UPC) más cercano a la última locación conocida del desaparecido para solicitar una búsqueda inmediata [1]. También sugieren presentar la denuncia correspondiente a Dirección Nacional de Delitos Contra la Vida, Muertes Violentas, Desapariciones, Extorsión y Secuestros (DINASED) [1].

La DINASED se encarga de las investigaciones de desaparecidos desde noviembre del 2013 [2]. En la gestión de su primer año se incrementó la proporción de casos resueltos de un 45 en 2013 a un 67 por ciento hasta lo registrado en el año 2014 [2]. En 2018 hasta el mes de junio se han localizado 97% de casos por desaparición [3]. Además de una efectividad mejorada, la DINASED se enfrentó a un aumento en las denuncias de personas desaparecidas en un 323% en comparación de enero 2013 a enero 2014 [2], lo cual indica que cualquier mejora en la eficacia del trabajo de esta institución, ya sea por medio de la gestión o herramientas que optimicen su labor, es bienvenida.

Normalmente este esfuerzo es organizado de manera independiente. Es decir que La Comandancia General de la Policía Nacional se organiza internamente al remitir a la DINASED, la cual asigna los casos de desapariciones a la Unidad de Desapariciones (UDPE) [4], mientras que los familiares realizan su propio esfuerzo por encontrar a la persona desaparecida.

1.1. Justificación

El presente documento del proyecto Integrador remarca la importancia del actuar prontamente y trabajar de manera coordinada cuando casos de desaparición se presentan, este proyecto integrador propone potenciar los esfuerzos de la policía, buscadores y familiares de una persona desaparecida por medio de una herramienta móvil.

Esta herramienta móvil fue creada tomando en cuenta ciertos aspectos de una desaparición marcados como importantes por trabajos anteriores [5] y publicaciones de organizaciones como el Departamento de Justicia de los Estados Unidos [6], aspectos como el que hacer cuando se descubre una posible desaparición, a donde acudir y que información es relevante para la investigación.

Al suceder una desaparición se debe determinar el lapso de tiempo desde el último avistamiento de la víctima y cuando se descubrió su desaparición, este “espacio de

oportunidad” es de suma importancia determinarla [6]. Este espacio de oportunidad sirve de referencia al recibir testimonios de posibles avistamientos y usarlos como información útil. El tiempo después de acudir con las autoridades es crucial para recopilar información de avistamientos y pistas que puedan ayudar a los buscadores a encontrar el paradero del desaparecido. Es una buena práctica que la nueva información a ser transmitida a los entes buscadores sea filtrada por la autoridad pertinente, en nuestro caso es la policía, puesto que los expertos indican “Durante esta investigación nada debe ser asumido y todo debe ser verificado” [6].

Existe un sistema de alerta de emergencia (Emergency Alert System) que se usa para diseminar de manera inmediata información sobre un desaparecido después de haber sido reportado a las autoridades competentes; se incorporó en 1996 después de la desaparición de Amber Hagerman, en su honor se denominó programa de Alerta AMBER [5]. Este programa consiste en que la policía notifique a los medios de comunicación el secuestro de un menor, el público televidente y radioyente recibirá una notificación a través de avisos emitidos con toda la información que esté disponible para que éste proporcione los ojos y oídos adicionales que aumentan la probabilidad de captura del secuestrador del niño antes de que hayan transcurrido las primeras tres horas críticas [5].

Este mecanismo de alerta ha tenido porcentajes de efectividad de hasta el 95% en diferentes estados de México [7], con un promedio de 68% en todo el país [8]. La difusión de información es de vital importancia para ayudar al personal asignado de la policía y los buscadores (familiares, amigos y voluntarios) a conocer el estado de la búsqueda y saber dónde deben buscar, esto se puede aplicar tanto en la búsqueda de menores edad como de otros desaparecidos. Usando la estrategia de la alerta AMBER al poner más ojos y oídos pendientes de la búsqueda de una persona desaparecida y publicando en un canal común las pistas del desaparecido se puede mejorar la eficacia de la búsqueda. Además, se puede potenciar el valor de la información al implementar la geolocalización de estas pistas en el canal común de comunicación.

La geolocalización [9] en la aplicación propuesta, ayudará a geolocalizar las personas que están buscando a una persona desaparecida e información pertinente a la búsqueda, al conocer donde están ubicadas los entes que buscan, es más fácil trazar rutas de búsqueda y dividir el esfuerzo de la manera más eficaz.

1.2. Estado del arte

En cuanto a aplicaciones y mecanismos de reacción cuando existe una persona desaparecida se encuentran desde aplicaciones que transmiten en un radio de alcance la

notificación de un menor desaparecido, hasta aplicaciones que realizan operaciones de rastreo celular. A continuación, se detalla estos proyectos:

Proyecto Emilia/ Alerta Emilia

Toma su nombre en memoria de Emilia Benavides, cuya muerte marcó una pauta para que el Estado ecuatoriano desarrolle acciones concretas sobre desapariciones de menores.

La finalidad del llamado “Proyecto Emilia” es el de activar una alerta a nivel local cuya duración se limita a 72 horas [10], cuya prioridad es la de salvaguardar la integridad física y psicológica de la persona desaparecida. Esta herramienta se basa en la búsqueda inmediata de menores de edad que se encuentren en alto riesgo de sufrir algún daño ya sea físico o mental [11].

La activación de esta alerta se divide en las siguientes fases:

Recepción de la denuncia, evaluación del riesgo y descripción de posibles sospechosos y otra información en el Sistema de Atención Integral o en la Policía Nacional.

La información se valida por el Fiscal Provincial y jefe zonal de la Policía y de ser necesario se aprueba la activación de la Alerta Emilia.

Se hace uso de Facebook para emitir una notificación con la información del menor desaparecido en todas las cuentas de los usuarios que se encuentren en un radio de 160 km, de igual manera los medios de comunicación difunden los datos del desaparecido [11].

Las denuncias en las que no se aprueba el uso de esta herramienta son atendidas bajo los protocolos de búsqueda regulares. Esta aplicación puede trabajar en equipo con la propuesta en este proyecto integrador, ya que después de validada la alerta o verificada por los protocolos regulares, se puede crear la entrada a esta búsqueda en la aplicación para gestionar la búsqueda entre los policías, familiares y voluntarios autorizados.

Tecnología para localización de personas accidentadas en lugares sin cobertura

Desarrollada por la Universidad de Alicante esta tecnología que permite localizar a personas en lugares remotos donde no existe cobertura celular ya se trate de personas accidentadas o extraviadas, donde la rapidez en el rescate es esencial. Útil también en situaciones de emergencias derivadas de terremotos, inundaciones e incendios forestales, donde las infraestructuras de telefonía móvil en muchas ocasiones quedan inutilizadas [12].

Emite una señal Wifi que actúa como baliza de socorro y permite abarcar varios kilómetros de distancia, puede incorporarse a cualquier Smartphone y no necesita cobertura móvil. La

señal contiene la ubicación (coordenadas) de la persona accidentada o desaparecida que usa el Smartphone emisor y un mensaje breve como: “necesito ayuda”, “estoy herido”. Para detectar la señal de auxilio, el investigador ha creado también un dispositivo receptor portátil, que emplearían los equipos de rescate o los refugios de montaña. Este dispositivo cuenta con una antena y se conecta al Smartphone de los efectivos de búsqueda [12].

Cuando se produce una incidencia, la persona accidentada solo tiene que activar la app del teléfono móvil que emitirá la baliza de auxilio periódicamente -puede ser durante horas o incluso días, aunque se encuentre inconsciente, indicando las coordenadas de su posición. Durante los ensayos llevados a cabo en tierra y en mar se demostró que el dispositivo puede captar la señal de socorro del Smartphone emisor hasta una distancia de tres y dos kilómetros, respectivamente, aunque hay posibilidades de aumentar su capacidad de cobertura [13]. El objetivo de esta app es optimizar la búsqueda y reducir el tiempo de rastreo.

App Finder

Esta aplicación fue creada por alumnos del “Instituto Politécnico Nacional” de México cuya finalidad es encontrar personas extraviadas por medio de referencias geográficas tomadas a través de las coordenadas de sus celulares. Se creó con la finalidad de encontrar personas desaparecidas a causa de desastres naturales, puntualmente en caso de un evento sísmico [14].

Funciona sin internet, al conectarse a un teléfono cercano que disponga de Wi-Fi Direct y enviar los datos para subirlos a internet con el fin de que cualquier persona pueda localizarlo, dispone de un radio de medio kilómetro en el cual esta app puede detectar cualquier teléfono celular cercano [15]. Además, con los datos recibidos permite informar a la familia del desaparecido y a los usuarios hacer búsquedas y consultar reportes de incidentes que hayan ocurrido en un radio de 500 metros en tiempo real por medio de una API Web que publica los datos en una página de disponible para consulta [14].

Esta aplicación es de rastreo y es sumamente útil en casos de desastres naturales o como paso preliminar para inspeccionar un área cercana a la última localización del desaparecido. Se podría utilizar como un complemento a la aplicación FindMe, en el caso de cuando se genera una alerta y esta se muestra en el mapa, los buscadores pueden ir a dicho lugar y usar Finder para hacer un barrido de los celulares en el alcance de 500m para localizar al desaparecido.

Alpify o Safe 365

Es una aplicación de geolocalización móvil cuyo objetivo es facilitar y agilizar las labores de rescate de personas accidentadas, heridas o perdidas. En cuanto a su funcionamiento, esta cuenta con un botón “de pánico" que una vez pulsado envía una alerta al número de emergencia local con la ubicación exacta de la persona e igualmente se puede conocer el recorrido que ha realizado. Se puede utilizar sin tener activados los datos móviles y permite enviar la ubicación a través de una llamada telefónica a emergencias [16].

Sobre las prestaciones básicas, se ha destacado el “localizable en todo momento" de tal forma, que, si una persona sale de paseo y no regresa, su familia puede "dar denuncia" de la desaparición y el servicio de emergencia introduciendo su número de teléfono en el sistema puede ver el recorrido que ha hecho esta persona y el lugar donde se encuentra, incluso si ha perdido la cobertura o está sin batería [16].

Safe365, antes conocido como Alpify, es un localizador GPS para familias diseñado para conocer el estado de seres queridos a la distancia. Su localizador de seguimiento GPS permite conocer la ubicación de personas a través de su teléfono móvil en tiempo real [17].

La diferencia con la aplicación propuesta es que esta es una aplicación de comunicación para el afectado y de rastreo como se mencionó anteriormente, mientras que FindMe es una herramienta para los buscadores con el objetivo de organizar la búsqueda. Comparando el alcance de personas que tienen estas aplicaciones, FindMe tiene uso en más tipos de desaparecidos como personas que no poseen celular; personas de la tercera edad y niños.

2. METODOLOGÍA

Para el desarrollo de este proyecto se utilizó parte de la metodología de desarrollo de software ágil iterativa “Scrum”. Esta metodología permitió añadir de forma incremental las funcionalidades de la solución de software móvil para buscar personas desaparecidas y fue escogida como la más adecuada puesto que la construcción de la solución podría tener cambios en los requerimientos, así como la adición de los mismos. Metodologías como Extreme Programming (XP) y Disciplined Agile Delivery (DAD) soportan cambios en los requerimientos durante la iteración lo cual puede generar problemas en el desarrollo puesto que algunos requerimientos tendrían que ser movidos a las iteraciones siguientes. Esto no pasa en Scrum ya que deja fijos los requerimientos durante el sprint, cualquier cambio a los requerimientos es tratado como un nuevo requerimiento para sprints posteriores [18].

La aplicación se construyó de forma iterativa con software funcionando al finalizar cada ciclo de desarrollo o Sprint, término que fue definido de una forma adaptativa que hizo frente a las necesidades de búsquedas que se encontraron a medida que se indagó en el tema y se desarrolló la aplicación. Las partes de Scrum que se utilizaron principalmente fueron eventos y artefactos detallados a continuación:

2.1. Equipo Scrum

El desarrollo de esta solución de software fue realizado por una sola persona, las pruebas de usabilidad se dieron lugar con el apoyo del personal de la DINASED. Por esta razón en el equipo de desarrollo solo constó el investigador, tomando en cuenta la opinión del personal de la Policía Nacional asignado.

Los equipos Scrum se organizan a sí mismos, son funcionales en diferentes ámbitos y eligen la forma más eficiente de cumplir su trabajo en lugar de ser direccionados por personas externas al equipo. Equipos de Scrum entrega productos iterativamente y de manera incremental, maximizando las oportunidades de retroalimentación. Entregas incrementales de producto “terminado” (done) asegura una versión potencialmente útil del producto funcionando siempre disponible [19].

2.2. Eventos Scrum

Los eventos Scrum que se implementaron en el desarrollo de la aplicación son:

- El Sprint
- Planeación del Sprint
- Revisión del Sprint

Los eventos prescritos son usados en Scrum para crear regularidad y minimizar la necesidad de reuniones no definidas en Scrum. Todos los eventos tienen una duración máxima. Una vez que un Sprint comienza, su duración está definida y no puede ser acortada o alargada. El resto de los eventos pueden terminar cuando se logra cumplir su propósito, asegurando que una cantidad de tiempo es usada sin permitir el desperdicio de tiempo en el proceso [19].

Fuera del Sprint, el cual es un contenedor para otros eventos, cada acto en Scrum es una oportunidad formal para inspeccionar o adaptar algo y están diseñados específicamente para permitir transparencia crítica e inspección. Al no incluir estas acciones se tiene como resultado una transparencia reducida y se pierde una oportunidad para adaptar e inspeccionar.

El Sprint

El Sprint, es el corazón de Scrum, tiene duración desde una semana hasta un mes en el cual el producto "Terminado", usable y potencialmente listo para producción es creado. Un nuevo Sprint comienza inmediatamente después de la terminación del Sprint previo [19]. En el caso de estudio el sprint durará un mes y tendrá en consideración los eventos de:

- Planeación del Sprint
- Trabajo de desarrollo.
- Revisión.

Durante el desarrollo del proyecto, el investigador del equipo de desarrollo es quien tomará la decisión sobre el alcance del sprint y del proyecto en sí dependiendo de la necesidad y los datos obtenidos a medida que se construye la aplicación. Cabe recalcar que: "durante el sprint no se hacen cambios que pongan en peligro la meta del mismo, los objetivos de calidad no deben decrecer y el alcance puede ser clarificado y renegociado entre el dueño del producto y el equipo de desarrollo mientras se aprende más" [19].

Cuando el horizonte de un sprint es muy largo, la definición que se está construyendo puede cambiar, crecer en complejidad e incrementar el riesgo. Los Sprints permiten predictibilidad al asegurar la inspección y adaptación del progreso encaminado a la meta del sprint al menos una vez cada mes [19].

Planeación de Sprint

Se planifican todas las tareas, el tiempo que necesitará cada una y cómo se lograrán. Todo esto en el tiempo designado para el sprint. Para lograrlo se necesita el Product Backlog, el último incremento del producto, la capacidad proyectada del equipo de desarrollo y el

desempeño pasado del mismo. El número de elementos seleccionados del Product Backlog para el sprint depende únicamente del equipo de desarrollo y solo el mismo puede estimar lo que pueden lograr en el sprint [19].

En el caso de estudio, el investigador calcula cuánto de la aplicación se puede desarrollar tomando aquellas historias de usuarios que podrá cumplir dentro del tiempo del sprint, tomando en cuenta los elementos descritos anteriormente como el último incremento y la capacidad proyectada de sí mismo, en el caso de estudio es de 40 horas por sprint.

Al tener determinada la meta del sprint y seleccionado los ítems del Product Backlog, el equipo de desarrollo decide cómo se construirá la funcionalidad hasta que se convierta en un incremento de producto "terminado" (done). Los elementos seleccionados del Product Backlog junto con el plan para lograrlos se llama "Sprint Backlog" [19].

Revisión del Sprint

Toma lugar al final del sprint para inspeccionar el incremento y adaptar el Product Backlog si es necesario. Durante la revisión el equipo scrum y las partes interesadas colaboran sobre lo que se hizo durante el sprint. Basados en esto y en cualquier cambio al Product Backlog durante el sprint, los asistentes cooperan en los pasos siguientes a realizar para optimizar valor [19]. El resultado es un Product Backlog que define los posibles elementos de este para el próximo sprint. El Product Backlog puede ser ajustado para aprovechar nuevas oportunidades [19].

Los eventos de Scrum que no van a ser utilizados son el Scrum Diario (Daily Scrum) y la Retrospectiva de Sprint (Sprint Restrospective). El Scrum Diario no será utilizado porque el investigador es el único miembro del Equipo Scrum, por lo cual no tendría sentido las reuniones. La Retrospectiva del Sprint no será usada porque la parte de inspeccionar el equipo y hacer un plan de mejoras que es parte de la retrospectiva, se realizará en la Revisión del Sprint.

2.3. Artefactos de Scrum

Los artefactos de Scrum representan trabajo o valor para proveer transparencia y oportunidades de inspeccionar y adaptar. Los artefactos definidos por Scrum son específicamente diseñados para maximizar transparencia de información clave de tal manera que todos tengan el mismo entendimiento del artefacto [19].

En el caso de estudio se van a utilizar los 3 artefactos propuestos en la guía de Scrum, estos artefactos ayudarán a organizar el trabajo y proporcionar información útil de cuanto se ha avanzado en el desarrollo de la aplicación móvil y las actividades siguientes para

conseguir finalizar exitosamente el proyecto integrador, los artefactos serán detallados a continuación:

Lista de Producto (Product Backlog)

Es una lista ordenada de todo lo que podría ser necesario en el producto, y es la única fuente de requisitos para cualquier cambio a realizarse en el producto. El dueño de producto (Product Owner) es el responsable de la lista de producto, incluyendo su contenido, disponibilidad y ordenación [20].

La lista de Producto inicial será el resultado del inquirir por primera vez los requerimientos de la aplicación propuesta para apoyar a la búsqueda de personas desaparecidas en el Distrito Metropolitano de Quito. Después de indagar los requerimientos, estos serán esquematizados en historias de usuario, elementos útiles para la planeación de los Sprints.

La Lista de Producto evoluciona a medida que el producto y el entorno en el que se usará también lo hacen. La Lista de Producto cambia constantemente para identificar lo que el producto necesita para ser adecuado, competitivo y útil. Mientras el producto exista, su Lista de Producto también existe [20].

Lista de Pendientes del Sprint (Sprint Backlog)

Es el conjunto de elementos de la “Lista del Producto” seleccionados para el sprint más un plan para entregar el “Incremento del producto” y conseguir el “Objetivo del sprint”. La “Lista de pendientes del Sprint” es una predicción hecha por el equipo de desarrollo sobre qué funcionalidad formará parte del próximo “Incremento” y el trabajo necesario para entregar esa funcionalidad en un Incremento “Terminado” [8]. Además, es un plan con un nivel de detalle suficiente permitiendo que los cambios en el progreso se puedan entender en el Scrum Diario y hace visible todo el trabajo que el equipo de desarrollo identifica como necesario para alcanzar el “Objetivo del Sprint” [20].

El equipo de desarrollo modifica la Lista de Pendientes del Sprint durante el sprint y la lista emerge a lo largo del mismo. Esto sucede a medida que se trabaja sobre el plan y aprende más acerca del trabajo necesario para conseguir el “Objetivo del Sprint”. Solo el equipo puede cambiar la lista de pendientes durante el sprint [20].

Incremento

Es la suma de todos los elementos de la “Lista de Producto” completados durante un sprint y el valor de los incrementos de todos los sprints anteriores. Al finalizar un sprint, el nuevo incremento debe cumplir la definición de “terminado”, lo cual significa que está en

condiciones de ser utilizado. El incremento debe estar en condiciones de utilizarse sin importar si el dueño del producto decide liberarlo o no [20].

2.4. Requerimientos de la aplicación

Para hacer frente a la problemática de las acciones que se deben tomar cuando una persona es determinada como desaparecida y por consiguiente, la policía (en este caso la DINASED) comienza con la búsqueda del desaparecido. Aquí es donde toma lugar la aplicación móvil fruto de este Proyecto Integrador.

Los requerimientos de la búsqueda que la aplicación puede cubrir son los siguientes:

- Mostrar los casos de desaparecidos que están en estado de búsqueda activa.
- Mostrar a las personas que están conectadas actualmente buscando.
- Mantener corriendo la aplicación aún con el teléfono bloqueado, si así lo desea el usuario.
- Marcar los caminos que han recorrido los buscadores y el tiempo desde que lo recorrieron.
- Mostrar información relevante, ya sea pistas o avistamientos.
- Lograr que la aplicación reconozca 2 tipos de usuarios: Agente Policial y Buscador (que puede ser un familiar o voluntario para la búsqueda) ambos tipos de usuarios tendrán permisos en común, pero el usuario de tipo Agente Policial, tendrá más permisos mostrados más adelante.
- Crear alertas, ya sea de tipo pista o avistamiento.
- Para el tipo de usuario Buscador:
    - Mostrar alertas creadas por el mismo usuario.
    - Visualizar las alertas que hayan sido aprobadas por la autoridad competente (en este caso La Policía, este requerimiento viene de la premisa de los expertos del Departamento de Justicia de E.E.U.U. “Durante esta investigación nada debe ser asumido y todo debe ser verificado” [6].)
- Para el tipo de usuario Agente Policial tiene la posibilidad de realizar las mismas acciones que el buscador con las siguientes diferencias:
    - Este usuario visualiza todas las alertas de las búsquedas en las que se encuentre, aprobadas y sin aprobar.
    - Este usuario puede aprobar o rechazar las alertas (sean pistas o avistamientos) publicadas por los usuarios de la aplicación que estén rastreando en esa búsqueda activa.

- Crear pruebas de usabilidad y carga.

Estos son los requerimientos iniciales que servirán para construir el Product Backlog de la aplicación.

2.5. Herramientas por utilizarse

Para el desarrollo de este proyecto integrador se necesitan las siguientes herramientas.

- Herramienta para tener tableros Kanban que contengan las historias de los Sprints de desarrollo (Trello)
- Herramienta para escritura de las historias de usuario (Editor de Texto)
- Herramienta para desarrollo en Android para Windows (Android Studio)
- Interfaz de Programación de Aplicaciones, (API) del inglés para acceso a mapas e información geográfica (Google Maps Platform)
- Base de datos en la nube con actualización en tiempo real (Firebase FIrestore)

Trello

Es una herramienta gratuita, que otorga el manejo de muchas características de desarrollo ágil. Por lo que el equipo de desarrollo está conformado por una persona, Trello es más que suficiente para gestionar las historias de usuario, el Sprint Backlog y las historias seleccionadas para cada sprint y es una solución gratis en equipos de hasta cinco personas. Trello permite la creación de tableros Kanban, en los que se puede personalizar columnas, donde se almacena una lista de Pendientes del Producto (Sprint Backlog) como:

- Historias en progreso.
- Historias por hacer.
- Control de calidad.
- Estado de “terminado”.

A su vez, permite diferentes acciones y adiciones a las historias como:

- Asignar tareas a los miembros.
- Añadir etiquetas.
- Desglosar en checklists actividades necesarias para completar la historia.
- Establecer fechas de vencimiento.
- Adjuntar archivos pertinentes a dicha historia.

Editor de texto

Para escribir el trabajo, realizar los cuadros en los que se detallarán las historias de usuario que conforman la Lista de Producto (Product Backlog) y demás partes del documento del proyecto integrador se utilizará el editor de texto más conocido, Microsoft Word.

Android Studio

La aplicación propuesta en este proyecto integrador será desarrollada para celulares con sistema operativo Android, por lo que se necesita un Ambiente de Desarrollo Integrado (IDE) del inglés que cumpla con los requerimientos y facilidades para configurar el proyecto de desarrollo para este sistema operativo móvil. Hace más de 5 años se utilizaba el IDE Eclipse con el plug-in de Google, Android Development Tools [21]. Ahora se utiliza Android Studio que es un IDE especializado para el desarrollo en Android, entre sus principales ventajas se tiene que su rendimiento es mejor que Eclipse, es más intuitivo, tiene integrado un compilador Gradle y es mejor para diseñar interfaces [22].

Se escogió desarrollar la aplicación para el sistema operativo Android puesto que desde que se inició este proyecto integrador (febrero 2018) hasta 8 de mayo del 2018 en Ecuador el sistema operativo Android tiene el 86,18% del mercado de celulares y iOS el 10,55% [23]. Por lo que es el sistema operativo más idóneo para desarrollar la aplicación para el Ecuador.

Google Maps Platform

La Plataforma de Google Maps tiene 3 Productos principales: Mapas, Rutas y Lugares. La aplicación usa la opción Mapas, que permite obtener un mapa personalizable para geolocalizar los buscadores y dibujar sus trayectorias. La aplicación usa el API de Mapas estáticos [24].

Se escogió Google Maps por sobre opciones abiertas como Open Street Maps y Maps.Me debido a su disponibilidad respaldada por Google y por todas las opciones que su API proporciona al proyecto. Los mapas de Google tienen años de madurez y han sido utilizados en gran cantidad de proyectos, garantizan la fiabilidad, rendimiento y más importante la familiaridad de los usuarios con los mapas. El Kit de Desarrollo de Software (SDK) de Maps para Android permite agregar un mapa de Google a una aplicación nativa de Android. El plan gratuito permite un número ilimitado de cargas móviles de mapas estáticos [25] lo cual suple completamente la necesidad de la aplicación propuesta.

Firebase Firestore

La aplicación necesitará de un servidor que recepte información sobre las rutas que siguen los buscadores. Los datos sobre rutas y tiempo recorrido serán expuestos a los demás

usuarios después de un lapso de tiempo tomando en cuenta la última revisión. En un inicio se pensó crear una aplicación servidor que haría el proceso antes descrito, pero afortunadamente se encontró un servicio de bases de datos en tiempo real que realizan estas operaciones sin necesidad de tener una aplicación servidor.

Los datos actualizados pueden verse en tiempo real, la información es guardada por la base de datos en una nube permitiendo que sea disponible a través de internet y los usuarios pueden acceder a través de una aplicación que ha sido registrada con los permisos de lectura y escritura. La aplicación se registra en la consola de Firebase y se genera una llave que se añade en las dependencias del proyecto de Android.

Firebase posee dos productos Firebase Firestore y Firebase Realtime Database, ambos guardan información en una base de datos no relacional y proporcionan actualizaciones de datos en tiempo real. Firebase Firestone "es una base de datos de documentos NoSQL que permite fácilmente almacenar, sincronizar y acceder a datos desde aplicaciones móviles y web a una escala global" [26] . Es una solución escalable que actualmente se encuentra en fase beta aunque es arriesgado utilizarla, fue escogida como la mejor opción para almacenar y sincronizar datos para los usuarios de la aplicación.

Además, esta solución de almacenamiento y back-end ofrece un plan básico sin pago que contiene:

- 1GB de Almacenamiento de datos total.
- 10GB al mes de ancho de banda.
- 20000 escrituras de documentos al día.
- 50000 lecturas de documentos al día.
- 20000 borradas de documentos al día [26].

2.6. Implementación

Como se describió anteriormente, se utilizará partes de la Metodología de Desarrollo iterativa Scrum. El desarrollo del proyecto se dividió en Sprints de 1 mes, al fin del cual se entregará un software funcionando según las especificaciones y criterios de aceptación detalladas en las historias de usuario.

Las historias de usuario salen de las necesidades de los buscadores de personas desaparecidas. Estas necesidades son traducidas en tareas de desarrollo, las cuales son organizadas y también desglosadas en historias más pequeñas. Tomando en cuenta la estrategia que tiene el software Jira (el cuál es un software para manejo de proyectos de software con facilidades para mostrar las historias de usuario e incidencias [27] de la aplicación en tableros Kanban personalizables) las historias grandes que están compuestas de otras historias más pequeñas se denominan “Epic”.

Esta es la primera versión de las historias de usuario, a lo largo de los sprints y principalmente al final de cada uno se revisarán las historias de usuario, se modificarán, añadirán o eliminarán según convenga el investigador para:

- Cumplir con los objetivos planteados
- Cubrir partes del problema a resolver (apoyar a la búsqueda de personas desaparecidas) que no se habían determinado en el primer abordaje del mismo.
- Desechar o modificar características de la aplicación que no aportan a la resolución del problema o sean obsoletas a medida que se conoce más el dominio de la aplicación

2.7. Historias de usuario preliminares

Id:	**1**
Tema:	Epic - Tener un menú principal
Descripción	• Botón Registrarse • Botón entrar como invitado (Entraría y vería el menú lateral deshabilitado los botones) • Botón, iniciar sesión
Parámetros de aceptación	
Parte de Historia:	

Id:	**2**
Tema:	Tener una pantalla de registro

Descripción	La aplicación debe registrar los siguientes datos:
	• Email
	• Num. de cédula
	• Nombres
	• Apellidos
	• Contraseña
	• Repetir Contraseña
	• Tipo Usuario
	  ○ DINASED
	  ○ Buscador
	• (Identificación única de DINASED [en caso de ser usuario DINASED])

Parámetros de aceptación	
Parte de Historia:	1

Id:	**3**
Tema:	Epic – Pantalla mapa
Descripción	• Menu lateral • Pantalla principal con mapa • Botón circular iniciar tracking
Parámetros de aceptación	
Parte de Historia:	

Id:	**4**
Tema:	Epic - Menú lateral
Descripción	La Pantalla mapa debe tener un menú lateral que aparece al deslizar el dedo desde la izquierda. El menú debe mostrar las siguientes opciones: • Nombre de la persona Registrada • Botón Alertas: Botón que despliega una lista de alertas (que han sido aprobadas). • Botón Mis Alertas: Muestra las alertas generadas por el usuario y su estado (aprobadas o pendientes)
Parámetros de aceptación	
Parte de Historia:	3

Id:	**5**
Tema:	Item Alertas

Descripción	La alerta debe tener la siguiente información:
	• Tipo (menú desplegable)
	  ○ Pista
	  ○ Avistamiento

	• Título • Descripción (150 caracteres) • Ubicación.
Parámetros de aceptación	El ítem de menú debe mostrar los datos antes mencionados dentro de una lista.
Parte de Historia:	4

Id:	**6**
Tema:	Item Mis Alertas

Descripción	La alerta debe tener la siguiente información:
	• Tipo (menú desplegable)
	  ○ Pista
	  ○ Avistamiento
	• Título
	• Descripción (150 caracteres)
	• Ubicación.
	• Estado (aprobadas o pendiente o rechazado) Usuarios de la DINAPEN ponen alertas aprobadas automáticamente

Parámetros de aceptación	El ítem de menú debe mostrar los datos antes mencionados dentro de una lista.
Parte de Historia:	4

Id:	**7**
Tema:	Item Crear Alertas (primero)

Descripción	La alerta debe tener la siguiente información para ingresar:
	• Tipo (menú desplegable)
	  ○ Pista
	  ○ Avistamiento
	• Título
	• Descripción (150 caracteres)
	• Ubicación.
	• Botón guardar

Parámetros de aceptación	El ítem de menú debe mostrar los datos antes mencionados dentro de una lista.
Parte de Historia:	4

Id:	**8**
Tema:	Botón Iniciar tracking
Descripción	• Debe mostrar con un color cuando con el GPS activado guarde los lugares revisados por medio de geolocalizar el dispositivo y el movimiento de la persona
Parámetros de aceptación	Debe funcionar mientras la persona esté recorriendo los lugares y mostrar las calles coloreadas en el mapa.

	Debe pedir que se reactive la localización si es que esta se desactiva. Debe mostrar solo el mapa si la localización esta desactivada y se ha cancelado el aviso anterior.
Parte de Historia:	3

Id:	**9**
Tema:	Botón detener tracking
Descripción	• Debe dejar de pintar las zonas transitadas si se ha detenido el tracking
Parámetros de aceptación	
Parte de Historia:	3

Id:	**10**
Tema:	Conectar con la API de Google Maps y visualizar mapas
Descripción	
Parámetros de aceptación	Mapas interactivos que permitan navegar
Parte de Historia:	3

Id:	**11**
Tema:	Visualizar buscadores cercanos y lugar buscado por ellos
Descripción	• Poder ver buscadores que han estado buscando cerca al hacer “pan” o zoom out de los mapas ☺
Parámetros de aceptación	Debe mostrarse con un ícono las personas que están buscando
Parte de Historia:	3

Id:	**12**
Tema:	Guardar en Firebase y mostrar datos de Firebase
Descripción	• Poder ver buscadores que han estado buscando cerca fetching de Firebase
Parámetros de aceptación	Debe mostrarse con un ícono las personas que están buscando
Parte de Historia:	3

Id:	**13**
Tema:	Pantalla que muestre las búsquedas activas para trackear por los desaparecidos seleccionados

Descripción	• Ver las búsquedas activas en una pantalla previa al mapa para que el buscador pueda seleccionar que búsquedas va a realizar
	• Esto servirá para que en el mapa se muestren los recorridos sólo de otros usuarios que esten en la misma búsqueda
	  ○ Name
	  ○ Age
	  ○ LastSeen
	  ○ Gender
	  ○ Descripción

Parámetros de aceptación	Pantalla que muestre en una lista las búsquedas activas y deje seleccionar con un visto. Boton que lleve a la pantalla principal de mapa con los ids de las búsquedas para mostrar solo los recorridos de usuarios con las búsquedas seleccionadas.
Parte de Historia:	3

Id:	**14**
Tema:	Diseño de las pantallas de Menú principal, Registro, Inicio de Sesión, Mapa, Menú lateral
Descripción	Diseñar las pantallas: • Menú principal • Pantalla Registro • Pantalla Inicio de Sesión • Mapa • Menú lateral
Parámetros de aceptación	El Menú principal debe tener 2 botones, uno para registrarse y para iniciar Sesión y un tercero más pequeño para ingresar como invitado. Una Pantalla de Registro con los campos descritos en la historia Nº 2. Una Pantalla de Iniciar Sesión con los campos descritos en la historia Nº 15 Una Pantalla principal que muestre un mapa y tenga un menú lateral.
Parte de Historia:	

Id:	**15**
Tema:	Tener una pantalla de iniciar sesión
Descripción	La aplicación debe registrar los siguientes datos:

	• Email • Contraseña Después debe llevar a la pantalla mapa al hacer click en el botón iniciar sesión
Parámetros de aceptación	La pantalla debe permitir ingresar el correo electrónico y contraseña de usuarios previamente registrados.
Parte de Historia:	1

2.8. Sprint 1

En el primer Sprint de desarrollo de este proyecto se decidieron partes del mismo que no se reflejan en las historias de usuario preliminares. Estás partes son:

- Definición de los requerimientos de la aplicación
- Creación de las historias de usuario

Estas actividades son fundamentales para el desarrollo de este proyecto ya que direccionan el trabajo futuro siendo un fundamento sobre el cuál construir. La definición de los requerimientos de la aplicación fue descrita en el primer capítulo del presente documento. Mientras que las historias de usuario preliminares fueron desglosadas en la sección anterior de este capítulo.

Al cimentar las dos primeras actividades del Sprint 1, las siguientes historias que fueron tomadas para el Sprint 1 son las siguientes:

- Nº 14 Diseño de las pantallas de Menú principal, Registro, Inicio de Sesión, Mapa, Menú lateral.
- Nº 10 Conectar con la API de Google Maps y visualizar mapas

Durante el Sprint

Se comenzó creando un proyecto en Android Studio, el nombre de la aplicación es "FindMe" y la opción escogida fue un proyecto con una actividad vacía. Se añadió este repositorio a GIthub para el control de cambios.

Nº 14 Diseño de las pantallas de Menú principal, Registro, Inicio de Sesión, Mapa, Menú lateral.

Las pantallas están diseñadas según las necesidades encontradas en la historia. A continuación, se detallará el desarrollo de las cinco pantallas englobadas en la historia.

El menú principal consta de dos botones: "REGISTRARSE" e "INICIAR SESIÓN" y un botón extra de "Iniciar sesión como invitado" el cuál será utilizado para ingresar directamente al mapa sin autenticarse puesto que esa parte de la funcionalidad no ha sido

desarrollada aún y se implementará en el Sprint 3, hasta la consecución de dicho sprint se utilizará este botón para ingresar a la pantalla Mapa. Esta pantalla se ve de la siguiente forma:

Figura 1 Menú Principal

La Pantalla de registro fue construida utilizando una plantilla de un formulario de inicio de sesión proporcionada por el IDE. La plantilla se llama “Login Activity”. Esta plantilla fue personalizada con los campos detallados en la historia Nº 2 los cuales son los siguientes:

- Email
- Num. de cédula
- Nombres
- Apellidos
- Contraseña
- Repetir Contraseña
- Tipo Usuario
    - DINASED
    - Buscador

- (Identificación única de DINASED [en caso de ser usuario DINASED])

La apariencia de la pantalla es la siguiente:

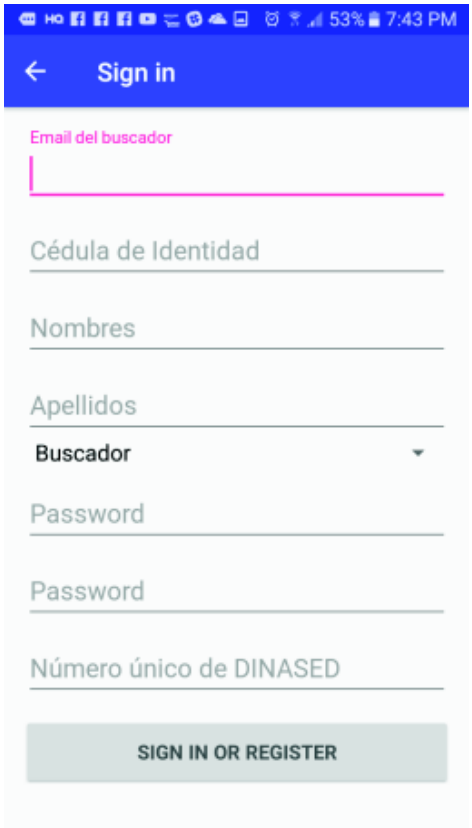


Figura 2 Pantalla de Registro

La Pantalla de Inicio de Sesión fue diseñada usando la misma plantilla detallada en la pantalla anterior, la diferencia es que no se añadieron campos a esta plantilla. La pantalla consta de los siguientes campos:

- Email
- Contraseña

Y la pantalla tiene la siguiente apariencia con el teclado desplegado:

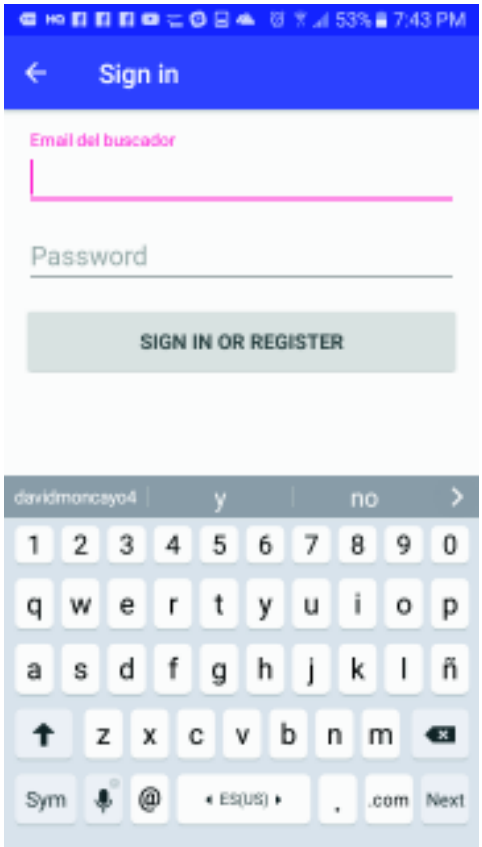


Figura 3 Pantalla de Inicio de Sesión

La pantalla del Mapa y el Menú lateral fueron creados con la plantilla “Navigation Drawer Activity” la cuál proporciona una actividad vacía con un botón flotante (Floating Action Button) y un menú lateral (Drawer) que se despliega al tocar el botón con tres líneas en la barra principal superior o al realizar el gesto de deslizar el dedo desde la izquierda hacia la derecha.

El diseño de esta pantalla se realizó sin mostrar el mapa de Google maps, esto fue realizado más adelante en la historia Nº 10, no obstante, el diseño fue exitoso y esta es la apariencia de esta pantalla:

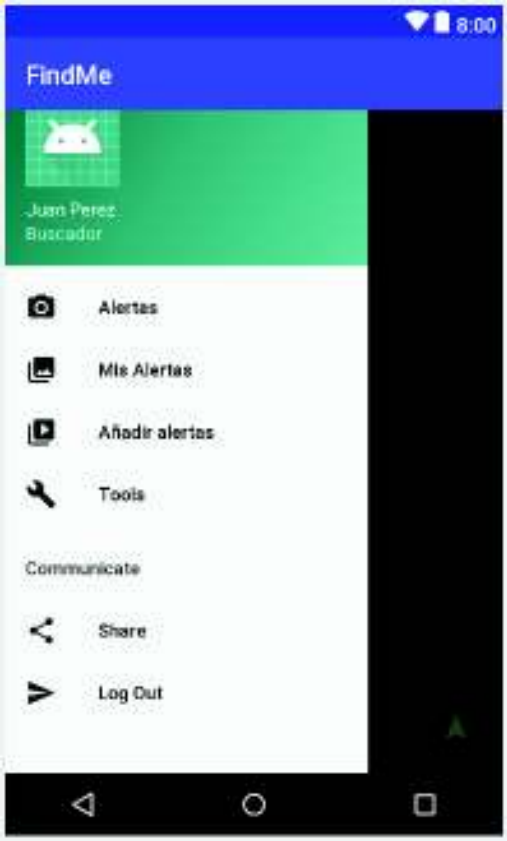


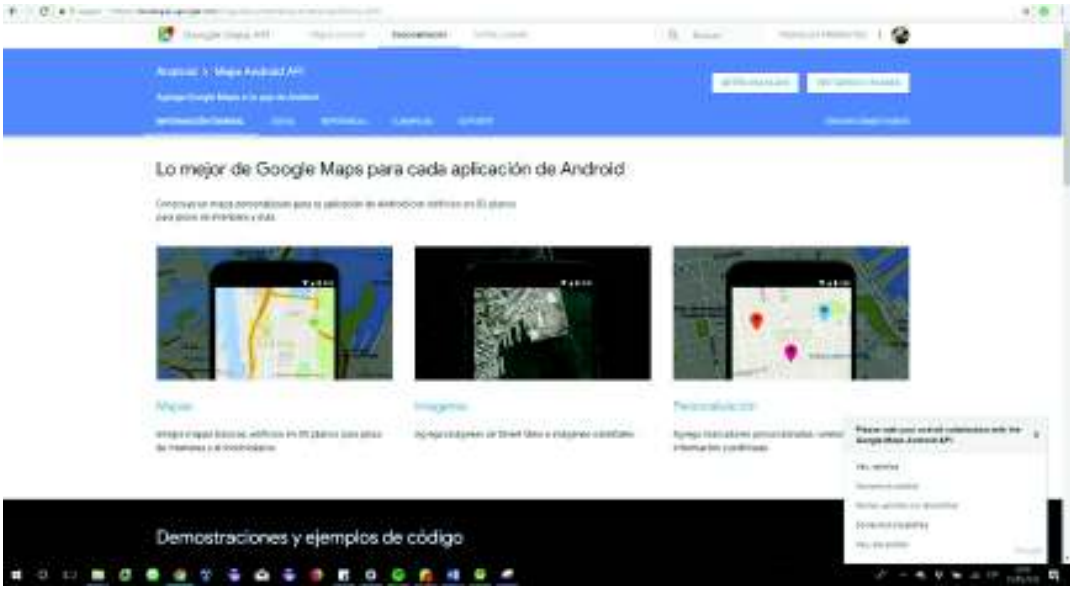
Figura 4 Pantalla Mapa y Menú lateral

Nº 10 Conectar con la API de Google Maps y visualizar mapas

Esta historia fue designada para el primer sprint al cual se le podría denominar un sprint de diseño. El propósito de esta historia es conectar con la API de Google maps para usar las funciones permitidas por este servicio más adelante en el Sprint 2. Por este Sprint se gestionará el acceso a la API de Google registrando la cuenta del investigador y la aplicación para su posterior implementación en la aplicación.

Para el registro de la aplicación se necesita tener una cuenta de Google, y acceder a https://developers.google.com/maps/?hl=es-419. Luego se selecciona la opción de "API de Google Maps para Android".

En la API para Android se tienen dos opciones: Google Maps Android API y Google Places API for Android. Para el propósito de este Sprint aún no se necesita convertir coordenadas geográficas a lugares significativos o conocer las direcciones para ir de un lugar a otro. Sino simplemente tener un mapa dinámico en el cual se puedan representar datos de la localización de los buscadores y marcadores donde sea útil y necesario. Es por esta razón que se seleccionó el API de Google Maps para Android.



Por razones de seguridad no se mostrará cual es la API Key que se obtuvo del sitio. En las guías de Google Maps API [28] se encuentran las instrucciones para añadir al proyecto de Android las dependencias y la clave de la API.

En la implementación existieron conflictos para combinar la funcionalidad de un fragmento de Google Map con una actividad con un menú lateral (Drawer) y un botón de acción flotante (Floating Action Button). El problema se originó por intentar implementar la clase Fragment Activity dentro de una clase que extiende AppCompatActivity. Esta última es la que tiene el menú lateral y botón de acción mientras que la primera abarca el mapa.

La solución a este problema se dio al llamar al layout del fragmento que contiene el mapa directamente desde la clase que extiende AppCompatActivity y llamar al fragmento en el layout del menú lateral y botones de dicha clase, la clase tiene como contenedor a un “Navigation View” donde se ubicará los elementos visibles e interactivos de esta clase. Todas las operaciones en el mapa vinieron de esta clase. El nombre de esta clase es “MapsActivity”, a continuación, se muestra el código para su implementación.

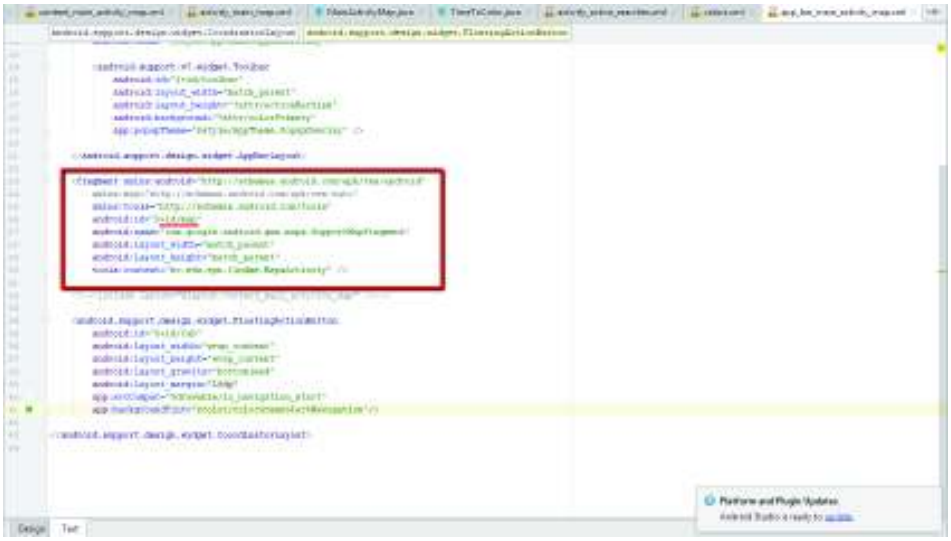


Figura 5 Código en el layout de los elementos de MapsActivity

Para implementar el fragmento del mapa en la clase se debe implementar la interfaz “OnMapReadyCallback” que requiere la función “onMapReady” que es llamada cuando la función “getMapAsync” del objeto SupportMapFragment es ejecutada.

A continuación, se tiene la clase Maps Activity que implementa la interfaz pública “OnMapReadyCallback” y la función requerida para su implementación “onMapReady”:

```
public class MapsActivity extends AppCompatActivity implements OnMapReadyCallback

    @Override
    public void onMapReady(GoogleMap googleMap) {

        mMap = googleMap;
        Log.d(TAG,  msg: "Entro por aqui!");
        Intent intent =  getIntent();
```

Figura 6 Implementación del mapa

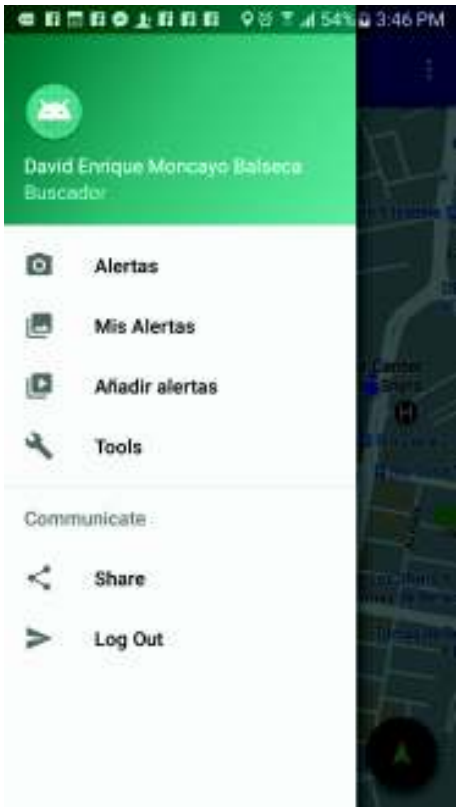
Después de esto se tiene la llamada asíncrona del mapa en el método onCreate de la actividad/clase MapsActivity, en este caso se colocó al final de este método la llamada para que todo el resto de la interfaz esté lista e inicializada.

```
        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
                .findFragmentById(R.id.map);
        mapFragment.getMapAsync( onMapReadyCallback: this);

    }
```

Figura 7 Llamada asíncrona de un mapa de Google Maps

Como resultado de esta historia se tiene una actividad con un mapa interactivo que permite mover la cámara del mismo y visualizar lugares con la información que ofrece Google en sus mapas:



Revisión del Sprint

Al finalizar este Sprint se aprecia el cumplimiento de los objetivos del mismo. Se definió los requerimientos iniciales de la aplicación tomados desde el primer acercamiento al problema a resolver, es decir el ayudar a la búsqueda de personas desaparecidas. El investigar el problema en cuestión en diferentes fuentes dio como resultado el apoyar a familiares, amigos, voluntarios y autoridades en las primeras horas críticas, y si lo amerita, días posteriores a una desaparición confirmada. El apoyo a la búsqueda se logrará con una aplicación que permita localizar las personas que están buscando y los lugares y tiempo desde que se buscó a la persona desaparecida.

Las necesidades que serán cubiertas por la aplicación de software fueron plasmadas en historias de usuario preliminares definidas al comienzo de este Sprint. De estas, dos historias fueron seleccionadas y terminadas. La historia número 14, diseñar las pantallas de Menú principal, Registro, Inicio de Sesión, Mapa y Menú lateral fue terminada satisfactoriamente.

En esta historia fue de gran ayuda usar los diseños predeterminados que el IDE Android Studio proporciona, esto agilizó la construcción de la pantalla de Registro y de Inicio de Sesión. De la misma forma la construcción de la pantalla Mapa y Menú lateral fue potenciado por el uso de la plantilla “Navigation Drawer Activity” que fue ajustado a medida.

Se logró concluir la historia número 10 que consiste en conectar con el API de Google Maps y visualizar el mapa en la actividad “MapsActivity” de la aplicación. Existieron contratiempos en la realización de esta historia por confusión y falta de claridad en las funcionalidades de las clases que extienden FragmentActivity y Fragment. La primera es una clase que soporta el uso de fragmentos, mas no un Fragmento que pueda ser embedido en otra actividad. El segundo, como su nombre lo indica, es una parte de layout reutilizable y que puede ser llamado desde una actividad que extiende AppCompatActivity o FragmentActivity.

Después de investigar sobre el tema, se solucionó el contratiempo al implementar la clase MapsActivity que extiende de AppCompatActivity y permite que se llame el fragmento del mapa de Google Maps, a esta actividad se configuró el resto de elementos gráficos necesarios para completar el diseño de la pantalla y dar apertura a la posterior construcción de funcionalidades en los siguientes sprints.

Se eligió extender de la clase AppCompatActivity en lugar de FragmentActivity porque la primera permite la implementación de un botón de acción flotante (Floating Action Button) que será necesario para los siguientes sprints.

Como oportunidades de mejora para los siguientes sprints se investigará más sobre cuáles son las mejores prácticas para implementar las funcionalidades que restan por construir. Y de esta forma tener un panorama más claro para dar la mejor solución a los requerimientos del sprint y cumplir los objetivos del mismo.

2.9. Sprint 2

Al comenzar este sprint se notó que existía funcionalidades que no habían sido descritas en las historias de usuario y que necesitaban ser creadas y formen parte del Product Backlog, cabe recalcar que por definición de los autores de Scrum, el Product Backlog nunca está terminado y siempre evoluciona [19], es por esto que a medida que el desarrollo del proyecto avanza las historias de usuario son añadidas, refinadas y hasta eliminadas si es que así se requiere para afrontar de mejor manera el problema, lograr los objetivos planteados o afrontar nuevos retos.

La historia añadida trata sobre la capacidad de la aplicación para mostrar el tiempo desde que cierto tramo fue recorrido por un buscador. La historia detallada es la siguiente:

Id:	**16**
Tema:	Mostrar el tiempo desde que cierto tramo fue recorrido
Descripción	En la aplicación se debe mostrar el tiempo desde que los tramos recorridos dibujados en el mapa fueron recorridos. Ya sean días, horas o minutos
Parámetros de aceptación	Mostrar de una forma amigable el tiempo en que cierto tramo pintado fue recorrido
Parte de Historia:	3

La historia anterior pasa a formar parte de la Lista de Pendientes del Producto más no de la Lista de Pendientes del Sprint aún. Esto sucede porque se necesita implementar primeramente la funcionalidad de Iniciar el registro de lugares recorridos por los buscadores que es parte de la historia Nº 8. También es necesario implementar la historia Nº 12 para subir los datos de los lugares recorridos por el usuario de la aplicación y también descargar de Firebase las rutas recorridas por otros usuarios.

Como se mencionó anteriormente, se necesita construir la funcionalidad de la historia Nº 8 para marcar las zonas recorridas por el usuario por medio de un botón en la pantalla principal de mapa. También se tiene la funcionalidad de la historia Nº 9 que es dejar de marcar el recorrido del usuario. Los objetivos de ambas historias se llevarán a cabo con el mismo botón por motivos de practicidad y diseño, de esta manera la interfaz de la pantalla mapa se verá menos poblada.

También una necesidad a priori es guardar estos datos en la base de datos no relacional Firestore (de Firebase) y traer los datos que han sido almacenados ahí de otros usuarios tanto la información de donde han recorrido y sus últimas ubicaciones. Es por esta razón que se va a implementar la historia Nº 12 y Nº11 refinando la primera y borrando la última, puesto que la Nº 11 está contenida en la Nº12.

La historia Nº 12 queda estructurada de la siguiente forma:

Id:	**12**
Tema:	Guardar en Firebase y mostrar datos de Firebase sobre buscadores cercanos y lugares buscados por ellos
Descripción	• Poder ver buscadores que han estado buscando cerca al hacer “pan” o zoom out de los mapas ☺

Parámetros de aceptación	Debe mostrarse con un ícono las personas que están buscando Debe mostrarse con líneas conectadas las calles por la que los buscadores han recorrido
Parte de Historia:	3

Por consiguiente, las historias tomadas para este Sprint son las siguientes:

- Nº 8 Botón Iniciar tracking
- Nº 9 Botón detener tracking
- Nº 12 Guardar en Firebase y mostrar datos de Firebase sobre buscadores cercanos y lugares buscados por ellos
- Nº 11 Visualizar buscadores cercanos y lugar buscado por ellos (que es absorbida por la historia Nº 12)

Durante el Sprint

A continuación, se destacan detalles importantes sobre la implementación de las historias antes mencionadas.

Nº 8 Botón Iniciar tracking

La primera implementación del segundo sprint comienza con la funcionalidad detrás del botón “Iniciar Tracking”, esta es una de las funcionalidades centrales de la aplicación, junto con la funcionalidad de guardar estos datos de la base de datos y recibir datos sobre buscadores cercanos.

En breves rasgos lo que hace este botón es invocar el servicio de actualizaciones de localización el cual es configurado por medio de parámetros en el código y envía la localización del dispositivo en coordenadas de latitud y longitud. Estas a su vez son procesadas para determinar cuando el dispositivo se ha movido una distancia mayor o igual a X metros, si esta condición se cumple se dibuja una polilinea entre el punto inicial y el punto en el que se cumplió la condición. Después de esto si el dispositivo se mueve otros X metros de la última posición dibujada, entonces se añade otro punto a la polilínea dibujada anteriormente.

Figura 8 Algoritmo para dibujar las líneas en el mapa según la distancia recorrida a partir del punto anterior

El servicio de actualizaciones de localización es una función del cliente del Proveedor de Localización (Fused Location Provider Client) que es parte del paquete de localización de Android (com.google.android.gms.location). El cliente puede ser configurado con un objeto de pedido de localización, en inglés “LocationRequest”, cuyos parámetros son:

- El intervalo normal del pedido de localización: Este es configurado en milisegundos e indica cuan frecuentemente se pedirá la localización geográfica del dispositivo.
- Máximo intervalo de pedido de localización: También configurado en milisegundos, determina el intervalo más rápido permitido para las actualizaciones de localización, es decir el intervalo que toma menos segundos en proveer la localización del dispositivo.
- Prioridad del pedido: Establece cuán importante es la petición para que se pueda asignar la prioridad de dar una respuesta con la localización del dispositivo.

Después, se define aquello que se hará con las actualizaciones de localización, esta clase, llamada “LocationCallback” posee una función llamada “onLocationResult”, esta es la función en la que se determina como se manejará la información recibida por el proveedor.

En esta función se configura el algoritmo que acepta el primer valor de localización del servicio y posteriormente compara con los siguientes puntos hasta que un punto cumpla con la distancia mínima establecida, al cumplirse la condición anterior, este punto pasa a

ser el nuevo punto de referencia actual y se vuelve a recibir nuevos puntos de localización del dispositivo del buscador hasta que se cumpla con la condición nuevamente.

De esta manera se construye la trayectoria del buscador de una manera eficaz, puesto que, si el buscador se quedara en un mismo lugar por un periodo relativamente largo de tiempo, la aplicación no tomaría en cuenta sino solo 1 de los puntos recorridos en esa área puesto que el resto de las localizaciones no cumpliría con la distancia mínima que deben tener los puntos.

Con este algoritmo se puede llamar a las funciones de la API de Google Maps para que dibuje un nuevo punto en la ruta del buscador cada vez que se cumpla la condición mínima de distancia. Esta es la lógica incorporada en el LocationCallback.

Al tener configurado los parámetros de actualización de la información de posición del dispositivo (Actualizaciones de Localización) y además definir la función de lo que se hará con las actualizaciones de localización (Callback de Localización). Se llama al servicio que "pide" dichas actualizaciones conforme a los parámetros de estos dos objetos, esta es una función directa de la clase Cliente de proveedor de localización, "FusedLocationProviderClient" en inglés.

Toda esta funcionalidad es invocada por una función en el código denominada "Comenzar actualizaciones de ubicación". A su vez también se determinó una función con la lógica para dejar de recibir las actualizaciones de ubicación y guardar los datos en memoria y posteriormente subirlos a Firestore, esta fue denominada "Detener actualizaciones de ubicación".

Las funciones tanto de comenzar y detener las actualizaciones de ubicación son llamadas desde el botón principal de la aplicación "Floating Action Button" o fab. Por medio de un Booleano se determina si se comienza a recibir las actualizaciones de ubicación, dibujar la trayectoria o si se detienen las actualizaciones y se guardan los puntos geográficos de este tramo.

Nº 9 Botón detener tracking

Esta historia fue implementada con el mismo botón que inicia el rastreo y las actualizaciones de localización por medio de un Booleano. Cuando la aplicación está recibiendo la posición del buscador el botón es de color rojo con un ícono de detener. Mientras que cuando la aplicación no está escribiendo la ubicación del dispositivo el botón de acción flotante tiene un color verde con un ícono de navegación.

En la lógica de este Booleano, cuando se está rastreando la posición del dispositivo del usuario, se detiene las actualizaciones de localización y se cambia el color del botón y su ícono para mostrar al usuario que se ha dejado de marcar su localización, también se guarda en memoria las polilíneas recorridas.

A continuación, un gráfico con los resultados de esta historia, al presionar el botón de comenzar tracking este se vuelve morado con un ícono que muestra que el usuario puede detener el rastreo:

Figura 9 Aplicación rastreando la localización del dispositivo

Al presionar el botón, se deja de rastrear la localización

Figura 10 Aplicación al dejar de rastrear la localización

Nº 12 Guardar en Firebase y mostrar datos de Firebase sobre buscadores cercanos y lugares buscados por ellos

Esta es la otra historia central de la aplicación en el que se persiste los recorridos del usuario, se muestran los tramos recorrido de otros usuarios y se muestra el último lugar recorrido por otros usuarios en el mapa. Esto requiere las siguientes subtareas:

- Diseño preliminar de la estructura de la base de datos NoSQL: Esta denotará como se guardan los datos de la posición recorrida por los buscadores, esta estará clasificada por usuario.

- Escribir el código que guardará los datos de los buscadores cercanos y los lugares recorridos en la estructura definida en el punto anterior.

La estructura de la base de datos representado en un árbol JSON es la siguiente:

Figura 11 Estructura de la base de datos NOSql

En la Figura vemos que los datos están divididos datos de Localización y Datos del usuario por ciudad, se comienza por las búsquedas activas que representan los datos de las personas desaparecidas que se van a buscar. Seguido están los datos de la última localización de los usuarios (LastLocations) de la aplicación, definidos por su id único (Uid) en la base de datos. Por último, por cada usuario se guarda información sobre las trayectorias que ha realizado en la búsqueda y las alertas que ha generado, esto se guarda bajo la colección “tracks” y alertas respectivamente. Además, hay ciertos campos que son útiles para saber cuántos recorridos y alertas tiene cada usuario y las búsquedas activas en las que está involucrado.

Finalmente, en la colección de Datos del usuario se tiene cierta información importante, pero no sensible, sobre el buscador. Esto será útil para mostrar a los otros usuarios información relevante sobre los recorridos de otros usuarios y donde fue su último lugar de búsqueda.

En base a esta estructura definida y, que puede cambiar en un futuro, se codifica la inserción de datos. Esto se logra añadiendo una funcionalidad para armar un objeto con los datos definidos en la estructura “track” de la base de datos. Ésta se llama cada vez que el usuario pulsa el botón de “Detener tracking”, o cada vez que el usuario se ha movido una cantidad de 5 puntos que cumplen con la condición de distancia.

Esta función se llama aún antes de dibujarla en el mapa puesto que es más importante persistirla, ya que, si por alguna razón fallara la aplicación del usuario, los datos de su recorrido ya están en la nube y son accesibles al resto de usuarios que se encuentren en la misma búsqueda.

Teniendo ya la funcionalidad que guarda las rutas recorridas, se pueden obtener estas rutas y de otros usuarios en la aplicación cuando esta se abre. Esto se logró al hacer un barrido por los usuarios existentes y obtener los puntos por los que pasaron en los últimos 5 días, se tomó como referencia las 72h en las que está activa la alerta Emilia [11] añadiendo 2 días más en los que se considera relevante saber por donde se ha buscado.

En la implementación por código se busca en la colección “tracks” de cada usuario los recorridos cuyas fechas de creación sean no menores a 5 días de la fecha y hora actual:

```
private void getTrackInformation(final String id) {
    //long timeDiff = 432000000;//5 days
    long longtimeToSearch = System.currentTimeMillis() - timeDiff;
    //usuarios.document(id).collection("tracks").get().addOnCompleteListener(new OnCompleteListener<QuerySnapshot>()){


    usuarios.document(id).collection( R "tracks").whereGreaterThanOrEqualTo( R "lastTraveledMillis", longtimeToSearch).get().addOnCompleteListener((task) -> {
            if (task.isSuccessful()) {
                for (DocumentSnapshot document : task.getResult()) {
                    Log.d(TAG,  msg: id + " and " + username);
                    Log.d(TAG,  msg: document.getId() + " => " + document.getDate( R "lastTraveled"));
                    drawLineToMap(document.toObject(TrackObject.class));
                }
            }
    });
```

Figura 12 Función que obtiene los "tracks" de los usuarios

Como se observa en la figura anterior se obtiene este dato comparando las fechas en milisegundos a la fecha y hora actual. Después de obtener los recorridos correspondientes se procede a dibujar dichas líneas.

Para representar la ultima locación de un buscador se hace una consulta a la colección LastLocations en la base datos, los datos obtenidos se dibujan con marcadores con un ícono personalizado.

Figura 13 Camino obtenido de un usuario junto con última localización

Por limitaciones de Firebase Firestore no se puede obtener datos con condiciones en diferentes campos. Esto sucedió en la Actividad Alertas al obtener las alertas de otros usuarios con un tiempo determinado de caducidad y también que sean alertas en estado "Checked"

Revisión del Sprint

En este Sprint se implementaron funcionalidades centrales de la aplicación en cuanto a los caminos recorridos y últimas localizaciones de los buscadores. Para los próximos Sprints se planea implementar historias de apoyo a estas funcionalidades y a la de reportar alertas. Las funcionalidades de apoyo brindarán seguridad y facilidad de visualización de la información refiriéndose a la implementación de registro e inicio de sesión de usuarios junto a la visualización de alertas e información importante respectivamente.

Se encontró una posible brecha de seguridad en una funcionalidad propuesta en la historia Nº 1 de Tener un menú principal, la cual será completada el siguiente Sprint. Tener un botón "Entrar como invitado" propone entrar al mapa con el menú lateral y el botón Iniciar tracking deshabilitado. Su objetivo era que un usuario pueda ver las actualizaciones en tiempo real sin estar registrado aún. El problema encontrado fue que podría ser usado maliciosamente por personas que hayan causado la desaparición de alguien para espiar donde están buscando los familiares y voluntarios autorizados y personal de la DINASED, lo cual sería perjudicial y entorpecería completamente la búsqueda.

Por esta razón es que se decidió quitar este punto de la historia 1 que será completada el siguiente sprint, quedando la historia Nº 1 de la siguiente forma:

Id:	**1**
Tema:	Epic - Tener un menú principal
Descripción	• Botón Registrarse • Botón, iniciar sesión
Parámetros de aceptación	
Parte de Historia:	

Al comenzar a usar la tecnología de Firebase es necesario revisar con detenimiento la documentación, donde hay puntos clave que no se explican de manera textual, al tiempo de este sprint, la tecnología estaba en fase beta y la mejor fuente de resolución de interrogantes es la página de referencia de Firebase FireStore [29] y la experiencia de

usuarios con conocimientos maduros en redes de intercambio de conocimiento (como StackOverflow).

Antes de la finalización del Sprint se cambió el código para que los usuarios puedan ver la última localización de los usuarios de la búsqueda cada vez que se mueven 5 puntos de su última posición. Refiriendose a 5 puntos en Firebase, que están configurados con un mínimo de 15 metros de distancia, por lo que las actualizaciones de la última localización tendrían un lapso de actualización mínimo de 75 metros. De esta forma hacemos más preciso el dato de la última localización de los buscadores sin afectar de gran manera el consumo de datos.

2.10. Sprint 3

Al tomar los diseños realizados en el primer sprint, se procederá a crear la funcionalidad detrás de las pantallas de registro e inicio de sesión. Para este sprint se hará uso de ciertas directrices y patrones de diseño de validación de datos para el registro y de los métodos de autentificación que ofrece Firebase.

En este sprint se pretende enriquecer la funcionalidad y utilidad de la aplicación al permitir el registro usuarios y habilitarlos a ligar a su cuenta los recorridos realizados en la búsqueda de un desaparecido. También se busca presentar información útil sobre los recorridos de otros usuarios por medio de mostrar en el mapa el tiempo en el que se realizaron los mismos.

Otro objetivo de este sprint es que el usuario pueda elegir que personas está buscando y de esta forma visualizar solamente información relevante a dichas búsquedas. Esto implica ligar los datos del usuario a las búsquedas que se encuentren activas.

Por consiguiente, las historias que serán implementadas en este Sprint son:

- Nº 2 Tener una pantalla de registro
- Nº 15 Tener una pantalla de iniciar sesión (se completaría la historia Nº 1 Tener un menú principal)
- Nº 16 Mostrar tiempo desde que cierto tramo fue recorrido en el mapa
- Nº 13 Pantalla que muestre las búsquedas activas para hacer tracking de los desaparecidos

Durante el Sprint

Nº 2 Tener una pantalla de registro

Esta historia está compuesta por dos tareas principales:

- Validar los campos llenados por el usuario.
- Conectarse con el servicio de autentificación de Firebase para crear el usuario.

En la Figura 2 se aprecia los campos que deben ser validados para el registro de usuarios. Las validaciones para implementar precisan que:

- Los campos no estén vacíos excepto por el número único de DINASED para usuarios tipo buscador.
- La contraseña tenga más de 6 caracteres.
- La cédula provista sea válida.
- Las contraseñas coincidan.
- El Email sea válido
- El identificador de la DINASED sea correcto (esto se implementará al coordinar con la policía su uso)

La actividad valida los campos al oprimir el botón “Registrarse”; por medio de condiciones valida que cada campo cumpla los requisitos antes mencionados. Cabe recalcar que los algoritmos de validación de cédula y de email fueron adaptados de un algoritmo existente [30] y de una validación simple de carácter, respectivamente. Como resultado de oprimir el botón se tiene errores en los campos o la creación del usuario en Firebase.

En el caso de tener errores, el campo cuya validación falló obtiene el enfoque (focus) y se muestra un recuadro con el error que tuvo:

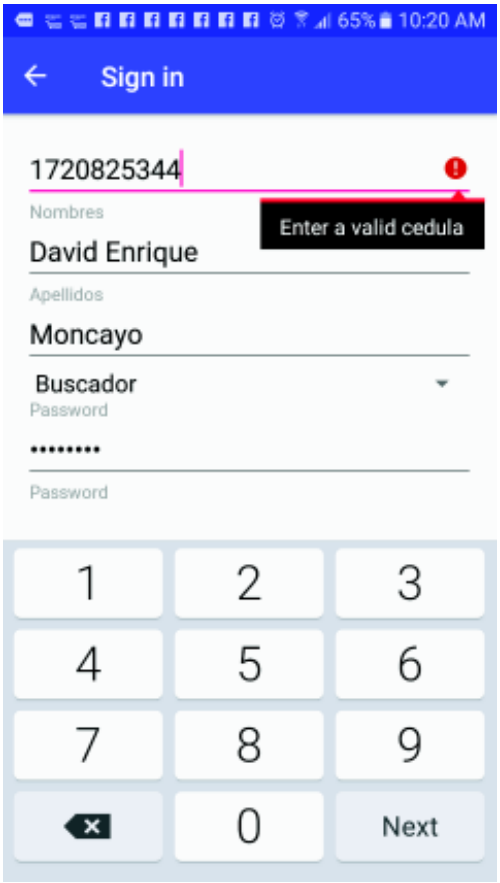


Figura 14 Mensaje de error al ingresar una cédula inválida

En caso de pasar todas las validaciones, se procede a crear un usuario con el email y la contraseña provista. Esta es una función de Firebase Authentication que genera un token de ingreso entre el cliente y el servidor como resultado de un registro exitoso. Y el usuario de Firebase pasa a estado de autenticado. Esto funciona por medio de una función asíncrona.

Con el usuario registrado exitosamente en Firebase Authentication, se genera un User Unique Id y guarda la información de forma interna de tal manera que solo se muestra al administrador el mail, el modo de autenticación y el Uid del usuario. También se tiene la opción de resetear la clave por medio de un mail de confirmación, inhabilitar o eliminar la cuenta.

Figura 15 Información de usuario en el módulo de administración Firebase Authentication

Después de registrado el usuario, se procede a guardar el resto de los datos en la base de Firebase Firestore en la colección de “UserData” en un documento que tiene por id el Uid del usuario como identificador único.

Los datos a guardar son:

- Nombres
- Apellidos
- Cédula
- Booleano de si es un usuario de la DINASED
- Número único de la DINASED (si aplica)

Nº 15 Tener una pantalla de iniciar sesión

Para iniciar sesión se utiliza la pantalla diseñada en el primer sprint, se hacen validaciones de que los campos correo y contraseña no estén vacíos y se llama al servicio de Firebase el cuál por debajo utiliza el API KEY para autenticarse como una aplicación registrada y así poder hacer llamadas al servicio de autenticación de usuarios.

Siendo exitoso el inicio de sesión la pantalla lleva directamente al mapa. Al implementar la historia Nº 13 la pantalla de inicio de sesión llevará a una pantalla con las búsquedas activas disponibles.

En la Pantalla Mapa se implementó la funcionalidad del ítem del menú lateral de Cerrar Sesión el cual lleva a la pantalla de Menú Principal.

Nº 16 Mostrar tiempo desde que cierto tramo fue recorrido en el mapa

Para implementar esta historia se requirió de dos actividades principales:

- Transformar a un lenguaje amigable el tiempo desde que se recorrió dicha trayectoria partiendo de su fecha en milisegundos
- Añadir esta cadena a la polilínea como una etiqueta (tag) que se muestra al momento de que el usuario hace un toque en la polilínea mostrada en el mapa.

Por medio de un algoritmo que, tomando la diferencia de tiempo en milisegundos, para cada unidad de tiempo saca el residuo en esa unidad de tiempo comenzando con milisegundos, segundos, minutos hasta llegar a una unidad de tiempo en la que no existan enteros (ej: 0 días). Al llegar a ese cero o terminar con las unidades de tiempo guarda un map con estas magnitudes para luego crear una cadena amigable al usuario.

Esta cadena es luego añadida como un tag a la polilínea que se va a dibujar y se muestra al usuario con un Toast que le indica el tiempo desde que ese tramo fue recorrido:

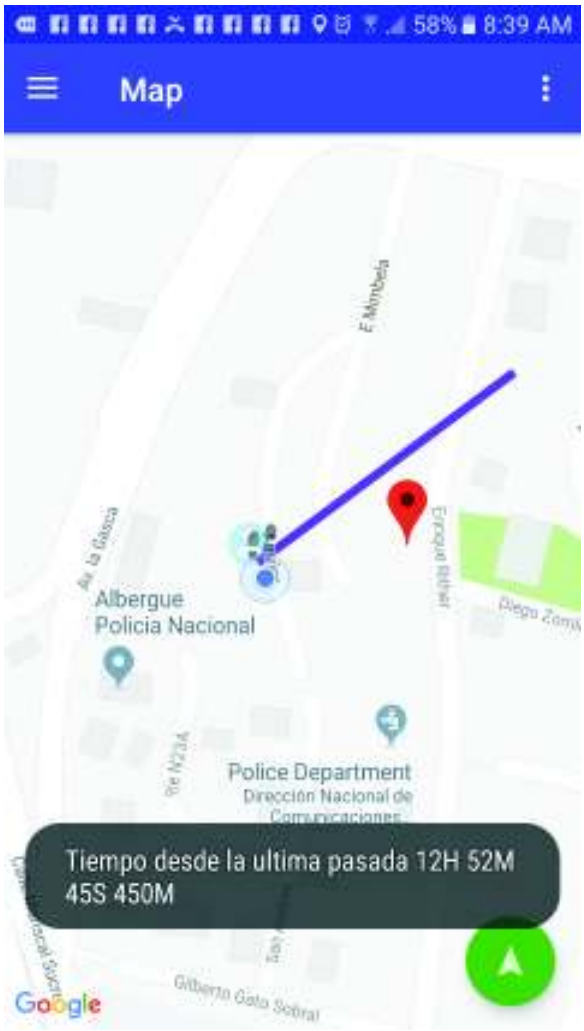


Figura 16 Toast con tiempo amigable al usuario

Nº 13 Pantalla que muestre las búsquedas activas para hacer tracking de los desaparecidos

Esta historia fue implementada al utilizar los campos de la base de datos de búsquedas activas (ActiveSearches) y ligándolas a los usuarios que estén involucrados en esa búsqueda. De esta manera todos los datos generados por el usuario como trayectorias y alertas se ligan a una o varias búsquedas activas.

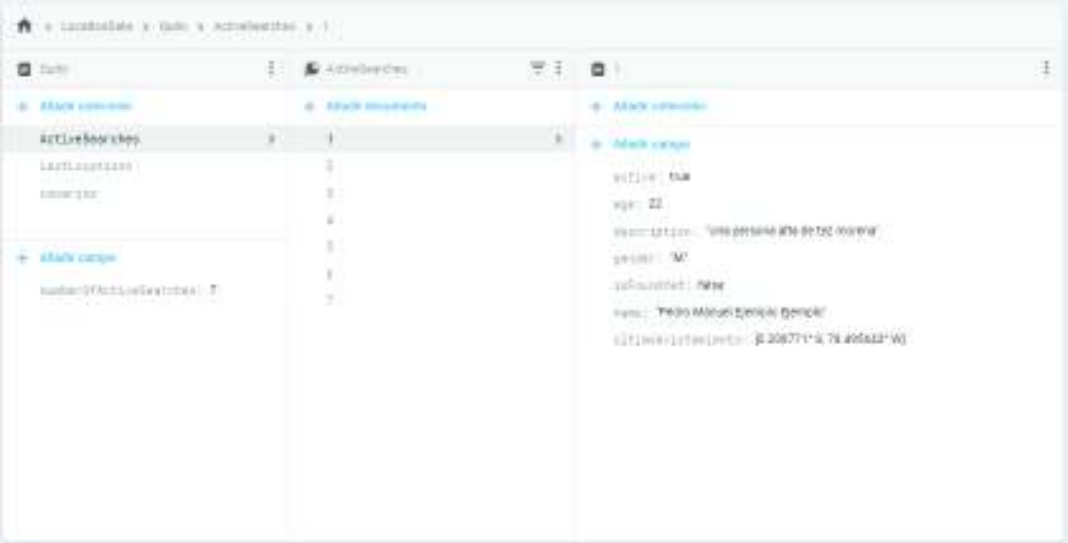


Figura 17 Estructura de una búsqueda activa en la base de datos

Aquí se aprecian los campos que se usarán para describir a una persona desaparecida, en caso de que se requiera más detalle se puede añadir en el campo description, los campos a usarse son:

- Nombre: Nombre Completo.
- Activo: Booleano que denota si esta búsqueda está activa o ha sido cancelada.
- Edad: Edad al momento del reporte.
- Descripción: Aquí se detallan rasgos característicos de la persona a buscar como estatura, color de cabello, ropa que vestía, tatuajes, etc.
- Sexo: Hombre o mujer.
- ¿Ha sido encontrado?: Boolean que denota si esta búsqueda activa ha finalizado con éxito.
- Último avistamiento: coordenadas de donde fue visto por última vez antes de su desaparición.

Se liga el id de la búsqueda activa a un array de búsquedas activas en el documento del usuario que buscará al desaparecido:

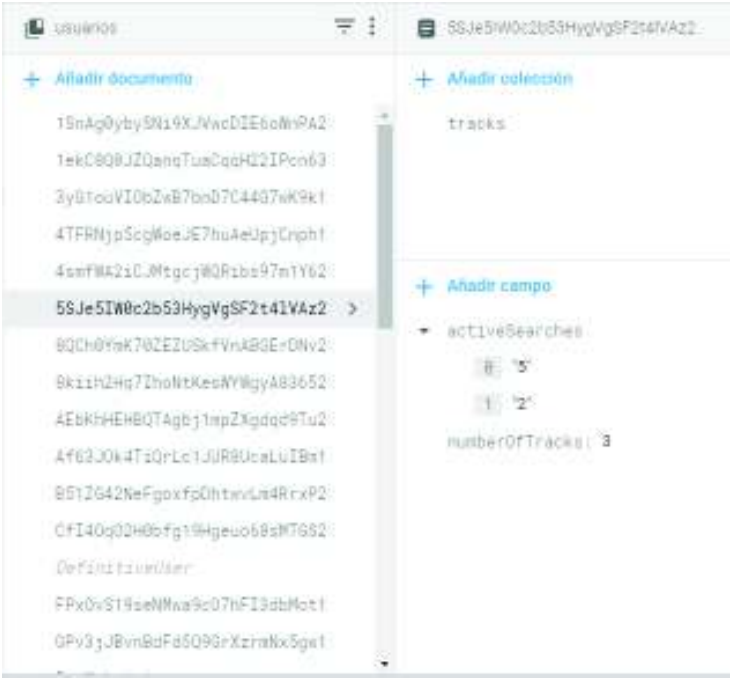


Figura 18 Búsquedas activas a las que un usuario está ligado

De esta forma se pueden obtener estos datos para ser mostrados en una pantalla previa al ingreso al mapa que sean seleccionables.

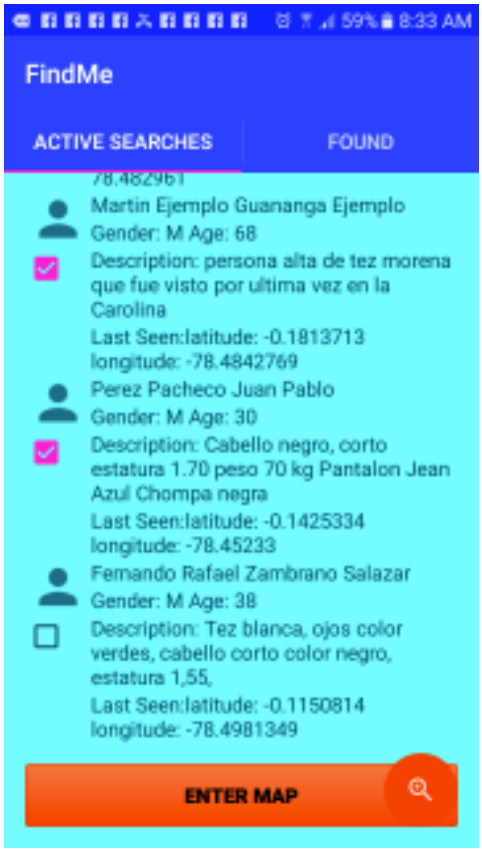


Figura 19 Implementación de búsquedas activas

Revisión del Sprint

El proyecto en este punto va tomando una estructura más sólida en la que los objetivos del proyecto comienzan a cumplirse, ahora se puede apreciar autenticación y control sobre las personas que usan la aplicación y se diferencia la información generada para una búsqueda activa u otra.

2.11. Sprint 4

Este Sprint está enfocado únicamente a las alertas en cómo estas son creadas, aprobadas y mostradas a los usuarios. Se requirió que se diseñe las pantallas para Crear Alertas y mostrar la Lista de alertas como paso previo a implementar las funcionalidades antes descritas.

Se añade la historia Nº 17 a la Lista del Producto y a la lista del presente Sprint:

Id:	17
Tema:	Diseñar las pantallas para Crear Alertas, Alertas y Mis Alertas

Descripción	• Crear alertas debe contener los siguientes datos de la alerta para ingresar:
	  ○ Título
	  ○ Tipo
	  ○ Descripción
	  ○ Ubicación
	• Alertas debe mostrar toda la información de la alerta y al final debe tener un botón de Añadir alerta para todos los usuarios. Además, debe tener 2 botones habilitados: Aprobar, Rechazar alerta para los usuarios DINASED
	• Mis Alertas debe mostrar toda la información de alertas pero solo del usuario actual.

Parámetros de aceptación	Ingreso de alertas, Visualización de alertas, Funcionalidad en los botones propuestos
Parte de Historia:	4

Por consiguiente, las historias que serán implementadas en este Sprint son:

- Nº 17 Diseñar las pantallas para Crear Alertas, Alertas y Mis Alertas
- Nº 7 Item de menú lateral Crear Alertas
- Nº 5 Item de menú lateral Alertas
- Nº 6 Item de menú lateral Mis Alertas (Termina Historia Nº4)

Durante el Sprint

Nº 17 Diseñar las pantallas para Crear Alertas, Alertas y Mis Alertas

Con la estructura definida en la base de datos en el segundo sprint se procedió a diseñar una pantalla para crear alertas con los Campos necesarios. Estos son los campos que se requirieron del usuario:

Figura 20 Diseño de la pantalla de Crear Alerta

Para mostrar las alertas en una lista tanto para “Mis Alertas” como “Alertas” se usará una misma pantalla, y por código en la historia 5 se filtrará para mostrar solo alertas del usuario o todas las alertas de la búsqueda activa.

Nº 7 Item de menú lateral Crear Alertas

Con la estructura definida en la base de datos en el segundo sprint Y la pantalla creada en la historia anterior se procedió a ligar los elementos de la alerta con un objeto entidad que sería posteriormente cargado a la base de datos.

Para entrar a esta actividad se pasa como parámetro la ubicación desde donde se está generando la alerta desde la actividad mapas (MainActivityMap) a la Actividad de crear alerta (NewAlert) esta se liga al objeto alerta a ser persistido y las coordenadas se muestran de forma legible para el usuario en el recuadro inferior.

Después de esto el proceso es más simple, se guarda la alerta en la colección alerts del documento con el Uid del usuario que está creando la alerta.

Nº 5 Item de menú lateral Alertas

Se creó un “viewHolder” usando clases que extienden de RecyclerView, clase que optimiza el uso de recursos de memoria para mostrar listas extensas de elementos. Este Holder fue diseñado de tal forma que muestre el ícono del tipo de alerta y muestre la información principal de cada alerta junto con un texto de colores que indica si es que la alerta ha sido aprobada o no por la DINASED

Para los usuarios de tipo DINASED se habilitan los botones “Aprobar alerta” o “Rechazar alerta” después de que el agente haya verificado los datos [6], principio que es muy importante en las investigaciones.

De esta manera se obtiene la siguiente lista de alertas generadas en la búsqueda por los usuarios:

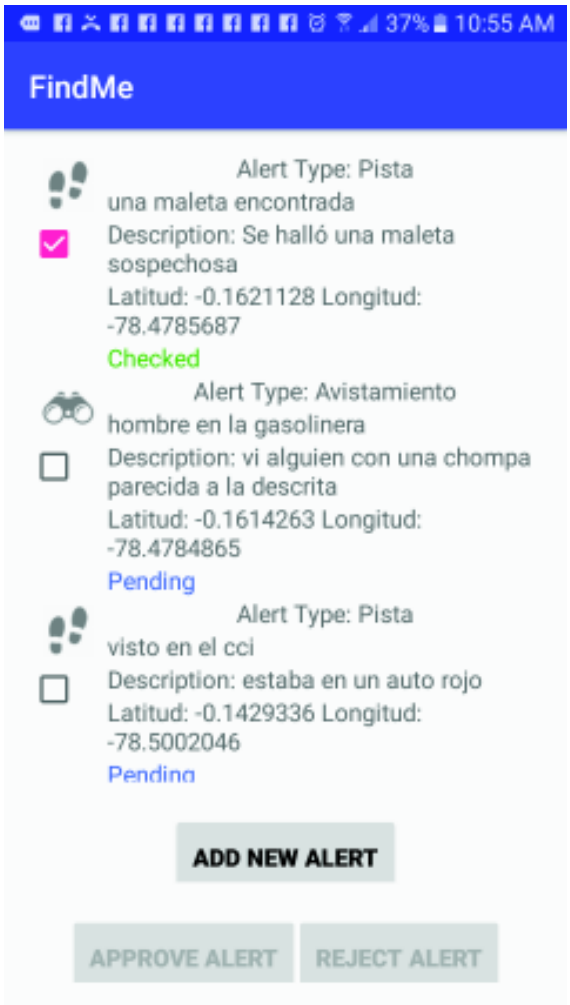


Figura 21 Lista de diferentes tipos de alertas generadas con sus estados

Se puede apreciar que la Figura anterior fue tomada de un usuario que no es del tipo DINASED por lo que no puede aprobar o rechazar alertas.

Nº 6 Item de menú lateral Mis Alertas

En esta historia se reutiliza todo lo anterior con la diferencia de que el título de la actividad cambia y se muestran solamente las alertas generadas por el usuario. Esto se logra haciendo una consulta diferente a la base de datos dependiendo de que opción del menú lateral del mapa se haya escogido:

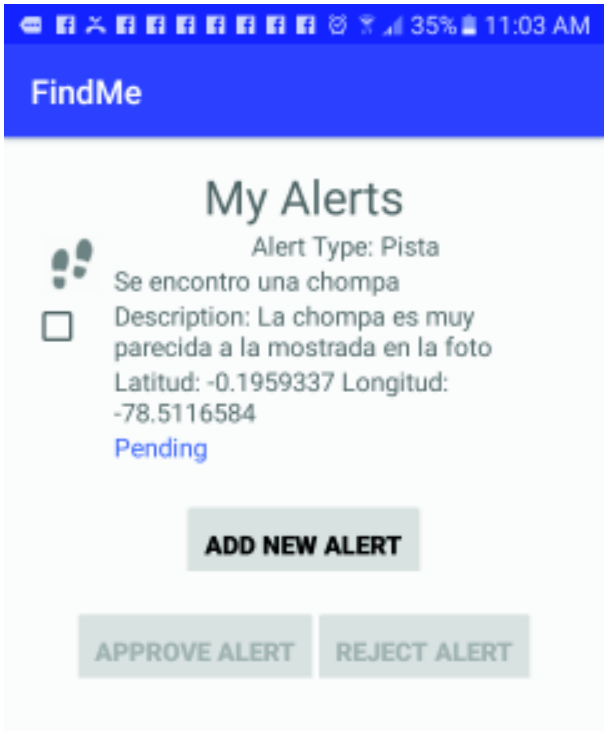


Figura 22 Pantalla Mis Alertas, mostrando alertas generadas por el usuario

Antes de la finalización del Sprint se cambió el código para que los usuarios puedan ver la última localización de los usuarios de la búsqueda cada vez que se mueven 5 puntos de su última posición. Refiriendose a 5 puntos en Firebase, que están configurados con un mínimo de 15 metros de distancia, por lo que las actualizaciones de la última localización tendrían un lapso de actualización mínimo de 75 metros. De esta forma hacemos más preciso el dato de la última localización de los buscadores sin afectar de gran manera el consumo de datos.

Revisión del Sprint

En el plan de este proyecto integrador se había contemplado la creación de un chat de comunicaciones para este Sprint, este no se llevó a cabo puesto que los buscadores civiles

se organizan en grupos de otros servicios de mensajería instantánea como WhatsApp, teniendo beneficios como una amplia gama de funcionalidades, fotografías, envío de documentos, audios, interfaz web, encriptación de punto a punto y más. Por lo que no se encontró de utilidad duplicar algunas de estas funcionalidades en una aplicación bastante madura.

Además de que en nuestro país es más fácil la comunicación por medio de este servicio ya que existen tarifas gratuitas para su uso. Mientras que, si se implementara en la aplicación propuesta, habría un costo adicional en datos por el envío de mensajes. Sin contar que pondría en peligro el objetivo del sprint y aumentaría la complejidad de este.

Al finalizar este Sprint se aprecia que ahora las alertas pueden ser:

- Creadas por parte de los buscadores para su posterior aprobación por usuarios de la DINASED.
- Creadas y automáticamente aprobadas para usuarios de la DINASED.
- Visualizadas en el mapa según los dos tipos definidos, pista (huellas) y avistamiento (largavistas).

2.12. Sprint 5

Este Sprint es un adicional a lo que estaba planificado cuyo objetivo es suplir o mejorar ciertas funcionalidades después de usar internamente la aplicación y de tener reuniones con el tutor del proyecto. Este sprint contiene también mejoras en la navegación y en la usabilidad de la misma en 2 idiomas. Estos requerimientos se desglosan de la siguiente forma:

Id:	**18**
Tema:	Modificar pantalla búsquedas activas para mostrar con tabs también personas encontradas
Descripción	• La pantalla de Búsquedas deberá mostrar dos pestañas, la primera mostrará la información actual de las búsquedas de personas que pronto serán encontradas y la segunda mostrará las personas que han sido encontradas
Parámetros de aceptación	Navegabilidad entre ambas pestañas y correcta visualización de contenido
Parte de Historia:	4

Id:	19
Tema:	Poner escuchadores para obtener en tiempo real actualizaciones de trayectorias, últimas localizaciones y alertas. No solo al ingresar a la aplicación
Descripción	Utilizar los métodos provistos por Firebase para escuchar adiciones en los campos antes mencionados y mostrarlas al usuario en tiempo real
Parámetros de aceptación	Actualización de contenidos de la aplicación sin necesidad de volver a entrar
Parte de Historia:	3

Id:	20
Tema:	Al hacer clic en una alerta, llevar a su ubicación en el mapa.
Descripción	Llevar al usuario a la ubicación de la alerta en el mapa al dar clic en el detalle de la alerta en la pantalla Alertas y Mis Alertas
Parámetros de aceptación	Redirección al mapa
Parte de Historia:	

Id:	21
Tema:	Pantalla para crear búsquedas activas.
Descripción	Diseñar y programar una pantalla que tenga los campos necesarios para crear una búsqueda activa. Se puede acceder a ella desde la pantalla “Búsquedas Activas”
Parámetros de aceptación	Búsqueda se crea correctamente
Parte de Historia:	

Id:	22
Tema:	Traducción de los textos, para tenerlos disponibles tanto en inglés como español
Descripción	Utilizar el editor de Android Studio para traducir los textos que actualmente están en ingles al Español, para las pruebas con la DINASED
Parámetros de aceptación	Textos se muestran en español al usuario
Parte de Historia:	

Por consiguiente, las historias que serán implementadas en este Sprint son:

- Nº 18 Modificar pantalla búsquedas activas para mostrar con tabs también personas encontradas.
- Nº 19 Poner escuchadores para obtener en tiempo real actualizaciones de trayectorias, últimas localizaciones y alertas. No solo al ingresar a la aplicación (Completa Historia Nº 3).
- Nº 20 Al hacer clic en una alerta, llevar a su ubicación en el mapa.
- Nº 21 Pantalla para crear búsquedas activas.
- Nº 22 Traducción de los textos, para tenerlos disponibles tanto en inglés como español.

Durante el Sprint

Nº 18 Modificar pantalla búsquedas activas para mostrar con tabs también personas encontradas.

Esta modificación tiene el propósito de motivar a los usuarios de la aplicación al ver también una lista de personas encontradas en la pantalla de “Busquedas Activas”. Entonces las personas que hayan sido encontradas y registrada su búsqueda activa en la aplicación, podrán luego pasar a ser parte de la lista de personas encontradas gracias al esfuerzo de los buscadores.

Para la implementación de esta funcionalidad se opto por usar una barra de soporte la cual albergará los tabs (Búsquedas Activas y Encontrados), se modificará el contenido principal de la pantalla por un paginador “ViewPager” para cambiar de un tab a otro con un gesto.

Dentro del paginador se albergará el RecyclerView que contiene las búsquedas activas. Para la implementación de mostrar los dos tipos de búsquedas activas, Activas y Encontrados, se traerá todas las búsquedas activas disponibles en la zona de Quito y se almacenará en memoria estos objetos y se filtrará dicha lista según el tipo de búsqueda activa que se necesite mostrar.

Figura 23 Lista de la segunda pestaña mostrando las personas encontradas

Nº 19 Poner escuchadores para obtener en tiempo real actualizaciones de trayectorias, últimas localizaciones y alertas. No solo al ingresar a la aplicación

Al revisar las diferentes opciones para obtener actualizaciones en tiempo real de los datos que se suben a Firebase mientras el usuario utiliza la aplicación, en la documentación oficial [31]. Se opto por escuchar cambios en los documentos de que los usuarios que están registrados en la búsqueda activa del usuario actual. De esta forma cuando se detecte alguna adición, solo ese documento será obtenido para que la aplicación la use.

En el caso de estudio, al obtener una actualización se va a dibujar una trayectoria, poner un marcador personalizado de última localización o representar una alerta en el mapa, dependiendo de lo que se obtenga.

Los escuchadores son puestos en marcha una vez que el usuario haya entrado al mapa y se haya obtenido todos los datos relevantes referentes a las búsquedas activas en las que se encuentra. Esto completa el objetivo de la aplicación en el que se obtenga actualizaciones en tiempo real.

Nº 20 Al hacer clic en una alerta, llevar a su ubicación en el mapa.

Esto se logró al llamar a las actividades Alertas, Mis Alertas y Nueva Alerta esperando un resultado de regreso. En caso de éxito en la tarea, es decir, cuando el usuario selecciona alguna alerta de la lista o crea una nueva alerta, la actividad devuelve la coordenada donde se encuentra registrada esa alerta de tal forma que el mapa lo reciba y mueva la cámara hasta ese punto.

Nº 22 Pantalla para crear búsquedas activas

Esta historia se abordó al tomar experiencias del uso de botones de acción flotante (Como en la pantalla mapa) y un formulario donde ingresar información relevante a la búsqueda activa (Como en pantalla Crear Alerta).

En primera instancia se incorporó el botón de acción flotante que llevaría a la actividad de crear una nueva búsqueda activa.

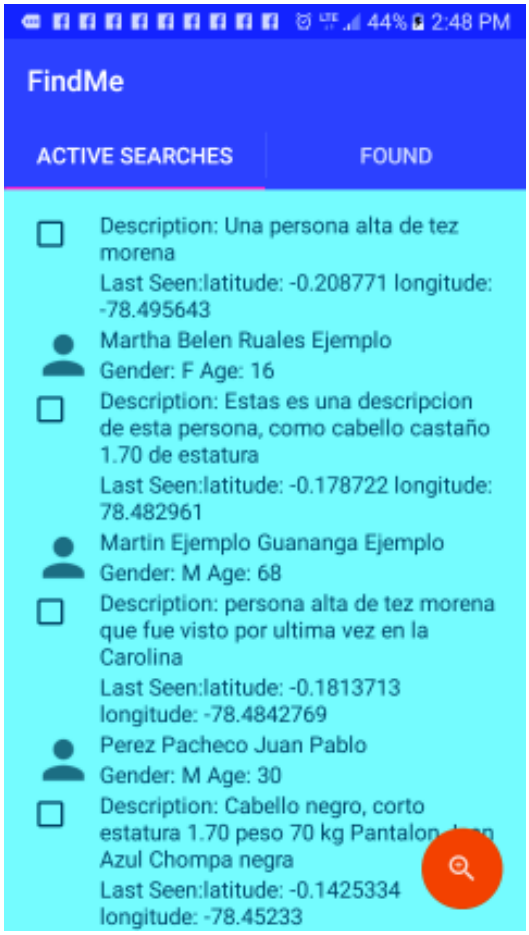


Figura 24 Botón de acción flotante (fab) naranja para añadir nuevas búsquedas

Basándose en la estructura de la base de datos se crearon campos de ingreso fácil de información sobre una búsqueda, los datos a ingresar son los siguientes:

- Nombre completo de la persona desaparecida
- Edad
- Sexo
- Descripción de la persona
- Última ubicación conocida
- Boolean de si ha sido encontrado aún o próximamente

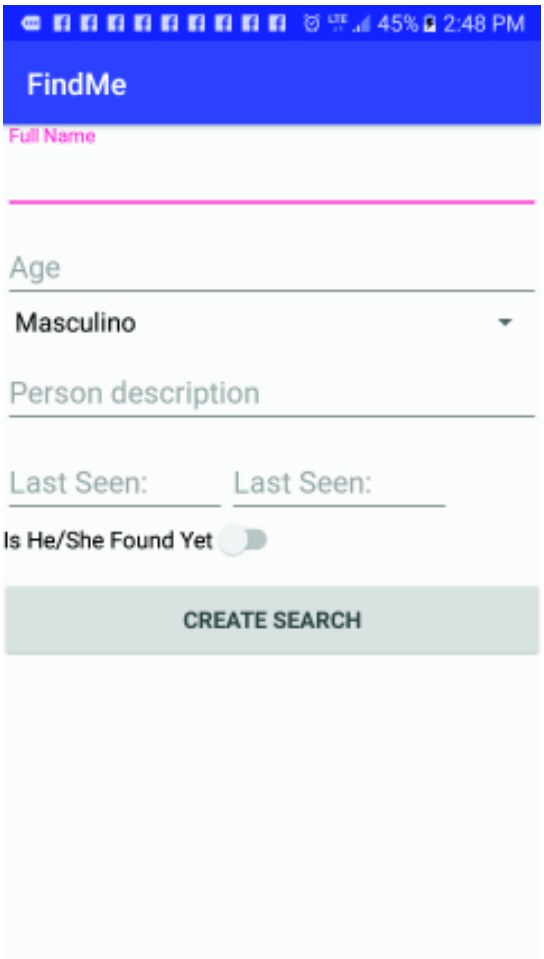


Figura 25 Formulario de creación de búsquedas

Nº 22 Traducción de los textos, para tenerlos disponibles tanto en inglés como español.

Para entregar una experiencia óptima de navegación se creó la traducción al español de la aplicación, esto se logra de manera óptima por medio del IDE Android Studio cuya interfaz

muestra los ids que han sido declarados en un archivo strings.xml y cuál es su valor actual en el idioma inglés, en otra columna se puede añadir una traducción a una gran variedad de idiomas.

MapsActivity.java | ActiveSearches.java | strings.xml | Translations Editor | app_bar_main_activity_map.xml | activity_main_map.xml | RegistroAc

Show All Keys | Show All Locales | ?

Key	Resource Fo...	Untranslata...	Default Value	Spanish (es)
app_name	app\src\main	☐	FindMe	FindMe
btn_Registrarse	app\src\main	☐	Registrarse	Registrarse
btn_iniciar_sesion	app\src\main	☐	Iniciar Sesión	Iniciar Sesión
btn_iniciar_sesion_como_invi	app\src\main	☐	Toca aquí para comenzar sesión como invitado	Toca aquí para comenzar sesión como invitado
title_activity_main_map	app\src\main	☐	MainActivityMap	
navigation_drawer_open	app\src\main	☐	Open navigation drawer	
navigation_drawer_close	app\src\main	☐	Close navigation drawer	
action_settings	app\src\main	☐	Settings	Configuración
title_activity_login	app\src\main	☐	Sign in	Iniciar Sesión
prompt_emailf	app\src\main	☐	Email del buscador	Email del buscador
prompt_ci	app\src\main	☐	Cédula de Identidad	Cédula de Identidad
prompt_password	app\src\main	☐	Password	Contraseña
prompt_names	app\src\main	☐	Nombres	
prompt_last_names	app\src\main	☐	Apellidos	
prompt_DINASED_number	app\src\main	☐	Número único de DINASED	
prompt_tipo_usuario	app\src\main	☐	Número único de DINASED	
action_sign_in	app\src\main	☐	Sign in or register	Iniciar Sesión o Registrarse
action_sign_in_short	app\src\main	☐	Sign in	
error_invalid_email	app\src\main	☐	This email address is invalid	Esta dirección de correo es inválida
error_invalid_password	app\src\main	☐	This password is too short	La contraseña es muy corta
error_incorrect_password	app\src\main	☐	This password is incorrect	Contraseña incorrecta
error_field_required	app\src\main	☐	This field is required	Este campo es obligatorio
permission_rationale	app\src\main	☐	Contacts permissions are needed for providing email[...]	Permisos a los contactos es necesario para sugerir emails
title_activity_registro_forum	app\src\main	☐	Sign in	Iniciar Sesión
active_searches_title	app\src\main	☐	Active Searches	Búsquedas Activas
found_people_title	app\src\main	☐	Found	Encontrados
active_searches_button_enterToMap	app\src\main	☐	Enter Map	Entrar al Mapa
latitude	app\src\main	☐	latitude	latitud
longitude	app\src\main	☐	longitude	longitud
my_alerts_title	app\src\main	☐	My Alerts	Mis alertas
all_aproved_alerts	app\src\main	☐	Aproved Alerts	Alertas aprobadas
all_alerts	app\src\main	☐	All Alerts	Todas las alertas

Key:
Default Value:
Translation:

Figura 26 Editor de traducciones de Android Studio

Revisión del Sprint

Este fue un sprint en el que se combinó el conocimiento obtenido de otros sprints para generar más valor al producto. Se pudieron divisar estos requerimientos al mostrar a otros tipos de usuarios su opinión sobre el sistema. Este es un principio del diseño de interacción. No todos los usuarios piensan como el diseñador [32] y es importante facilitar a la mayor cantidad de usuarios la realización de dichas tareas.

Mientras que funcionalidades como agregar nuevas búsquedas son esenciales, pero no se había contemplado hasta etapas postreras del proyecto.

3. RESULTADOS Y DISCUSIÓN

Al construir una aplicación es importante aplicar principios para el desarrollo tales como:

- Enfocarse en los usuarios y sus tareas.
- Observar, analizar y medir su desempeño con el sistema.
- Diseñar iterativamente [33].

Esta referencia tiene similitud con ciertos principios de Scrum, específicamente en que, al finalizar cada iteración, existe una revisión del sprint y del producto construido con miras a mejorar el proceso y el por lo tanto el producto final en el sprint consecuente.

Parte de este proceso de diseño es la evaluación de la experiencia del usuario, es decir, revisar que los usuarios puedan utilizar el software y les agrade hacerlo [32].

Con respecto a pruebas de carga de la aplicación, como se detalló en el capítulo 1 la solución de base de datos y lado del servidor que se está usando es Firebase Firestore la cual, en su plan gratis, tiene límites de hasta 100 usuarios simultáneos a la vez de ciertos parámetros límite en cuanto a ancho de banda, lectura y escritura de datos por mes.

Además, la infraestructura que se utiliza tanto en hardware, backend y hosting son provistos por Google. Por lo tanto, no haría falta realizar pruebas de carga ya que la disponibilidad y robustez del servicio dependería netamente de Google y de la conexión a internet que dispongan los dispositivos y no en algún software servidor o infraestructura provista por el desarrollador. Además de que se corre el riesgo de consumir la cuota gratuita de datos para la aplicación en el mes, lo cual detendría el desarrollo.

Las pruebas de rendimiento fueron realizadas al medir el tiempo que tomaban tareas básicas de la aplicación. También el uso del CPU y de los datos utilizados dado una carga específica, con parámetros descritos más adelante

Después de concluidos los sprints de desarrollo detallados en el capítulo anterior, se procedió a realizar un ensayo con la aplicación; este tomó lugar con un taller y exposición a miembros de la DINASED en la que se les mostró el propósito y motivación de la construcción de esta aplicación además de características de la misma. Al final realizaron una prueba de usabilidad de la aplicación que contenía actividades en la aplicación, luego de las cuales llenaron un cuestionario sobre su experiencia con la aplicación.

Las tareas que fueron asignadas a los miembros de la DINASED fueron divididas en secciones y desglosadas de la siguiente manera:

1. **Registrarse:** Registra una cuenta con tus datos reales, pasando las validaciones.

2. **Seleccionar búsqueda:** Identifica y entra a la búsqueda en la que vas a participar.
3. **Entrar al mapa y comenzar a rastrear su posición:** ¡Estás adentro! Explora las opciones de la aplicación, interactúa con las líneas de trayectoria generadas por ti y otros usuarios.
4. **Explore las alertas.**
5. **Visualice las alertas en el mapa:** Ubica las alertas generadas en el mapa.

Las preguntas que respondieron los colaboradores en este ensayo fueron las siguientes:

Nº de pregunta	Pregunta
Registrarse	
1	¿Le pareció intuitiva la pantalla de registro?
2	¿Qué dificultades tuvo al registrarse?
Seleccionar búsqueda	
3	¿Pudo encontrar en la lista a la persona que quería buscar?
Entrar al mapa y comenzar a rastrear su posición	
4	¿Pudo encontrar el botón de comenzar a rastrear su posición?
5	¿Al moverse, pudo ver en el mapa el registro de su trayectoria?
6	¿Pudo ver los movimientos de otros buscadores?
7	¿Pudo dejar de rastrear su posición?
8	¿Qué dificultades tuvo con la aplicación?
Explore las alertas	
9	¿Encontró la sección de alertas, le pareció intuitivo?
Visualice las alertas en el mapa	
10	¿Pudo visualizar las alertas en el mapa?
11	¿Pudo identificar los tipos de alerta: ¿Vista y avistamiento?
12	¿Qué dificultades tuvo?
13	¿Si fue necesario, pudo crear una alerta?

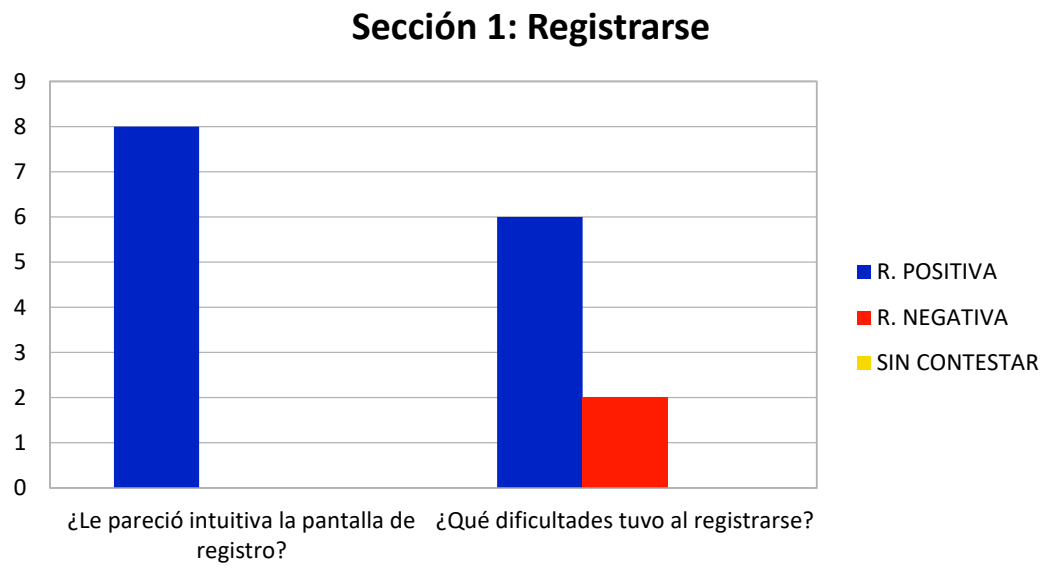
El análisis de los resultados fue del tipo univariado es decir que se determina para cada variable por separado una tabla de frecuencias [34]. Esta evaluación fue aplicada de manera aleatoria a 8 de los 11 colaboradores de la DINASED que participaron activamente en la prueba, es decir que, con su dispositivo móvil siguieron las actividades descritas anteriormente.

Estas preguntas son abiertas para medir no sólo la respuesta afirmativa o negativa de los policías, pero su criterio y observaciones a la aplicación. Esto revela necesidades de usuarios reales de la aplicación que pueden ser usadas como una oportunidad de mejora.

3.1. Análisis de resultados

A continuación, se detalla los resultados por pregunta junto con un análisis de los datos arrojados por la evaluación:

Sección 1: Registrarse



Pregunta 1: ¿Le pareció intuitiva la pantalla de registro?

Al ser una actividad con un flujo sumamente sencillo, al cual el usuario está familiarizado, fue calificada como intuitiva por todos los participantes, en cuanto a sus valoraciones verbales se encuentra como: útil, fácil de manejar, creativa y personalizada posiblemente por la apariencia de la pantalla y los errores amigables desplegados cuando algún dato no superaba las validaciones.

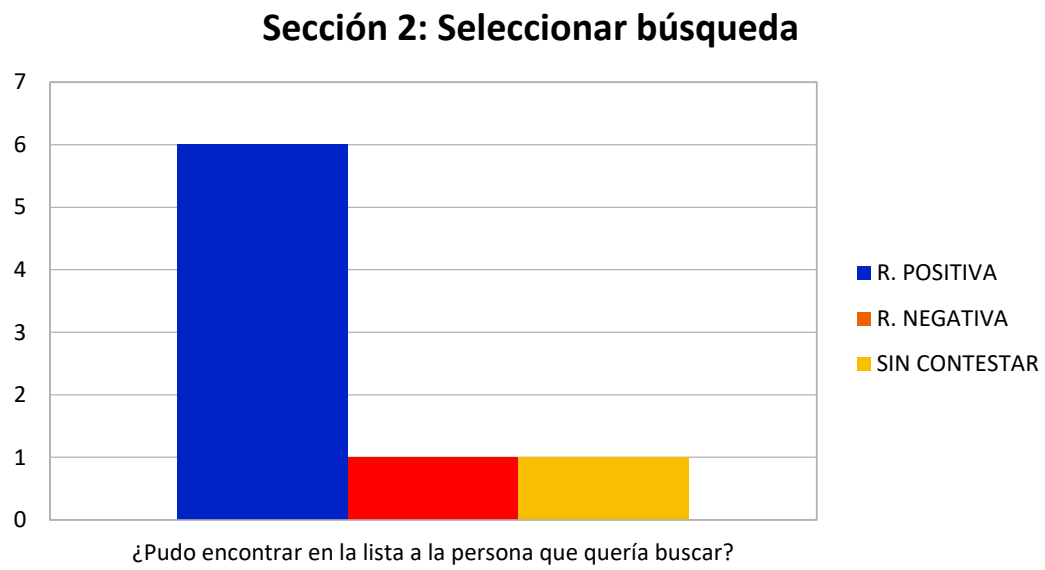
Pregunta 2: ¿Qué dificultades tuvo al registrarse?

En 6 de 8 participantes no se presentaron inconvenientes al momento de registrarse, los problemas que surgieron en las otras 2 personas fueron inconvenientes del teléfono y falta

de espacio para instalar la aplicación. Cabe recalcar que la aplicación pesa menos de 10mb.

En general se denota satisfacción y pertinencia entre esta parte de la aplicación y lo que el usuario necesita.

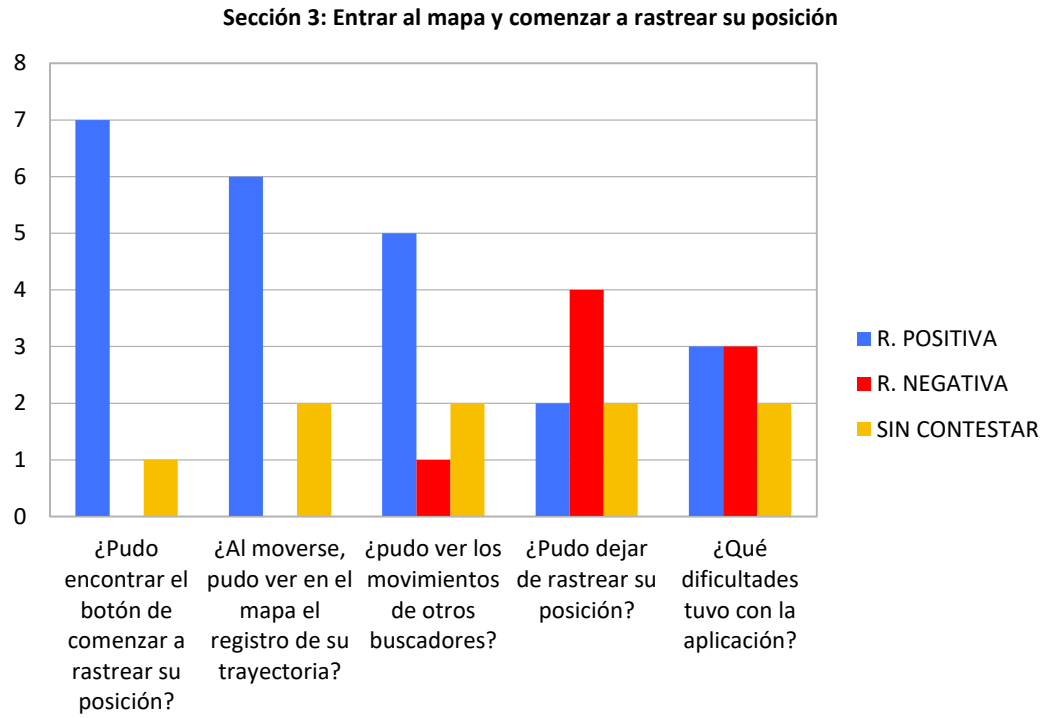
Sección 2: Seleccionar búsqueda



Pregunta 3: ¿Pudo encontrar en la lista a la persona que quería buscar?

En esta actividad se requirió que seleccionen para la búsqueda a un caso de prueba con el que se trabajaría el resto de actividades. 6 de 7 personas que realizaron la actividad pudieron completarla y entre sus opiniones resalta la facilidad para localizar lo que se requiere. Una persona tuvo dificultades en encontrar a la persona y otra no pudo completar este paso ni las siguientes actividades por problemas con su teléfono.

Sección 3: Entrar al mapa y comenzar a rastrear su posición



Pregunta 4: ¿Pudo encontrar el botón de comenzar a rastrear su posición?

Todos los voluntarios que continuaron con la prueba encontraron el botón para rastrear su posición, puede deberse a que es el botón más sobresaliente al momento de ingresar al mapa.

Pregunta 5: ¿Al moverse, pudo ver en el mapa el registro de su trayectoria?

En esta actividad, todos los colaboradores pudieron apreciar como se generaban líneas de trayectoria al momento de que ellos se movían voluntariamente o, simplemente la precisión del gps variaba en sus teléfonos y generaba pequeñas líneas de movimiento.

Figura 27 Movimiento del usuario de la aplicación

Pregunta 6: ¿Pudo ver los movimientos de otros buscadores?

Esto puede indicar que las líneas mostradas en el mapa denotan al usuario movimiento por parte de otro buscador o de él mismo, lo cuál es positivo. Incluso en las valoraciones verbales de la prueba de usabilidad de un usuario se denota que diferenció el código de colores de los recorridos que cambia con respecto a que tan nueva es la trayectoria de dicho buscador.

Pregunta 7: ¿Pudo dejar de rastrear su posición?

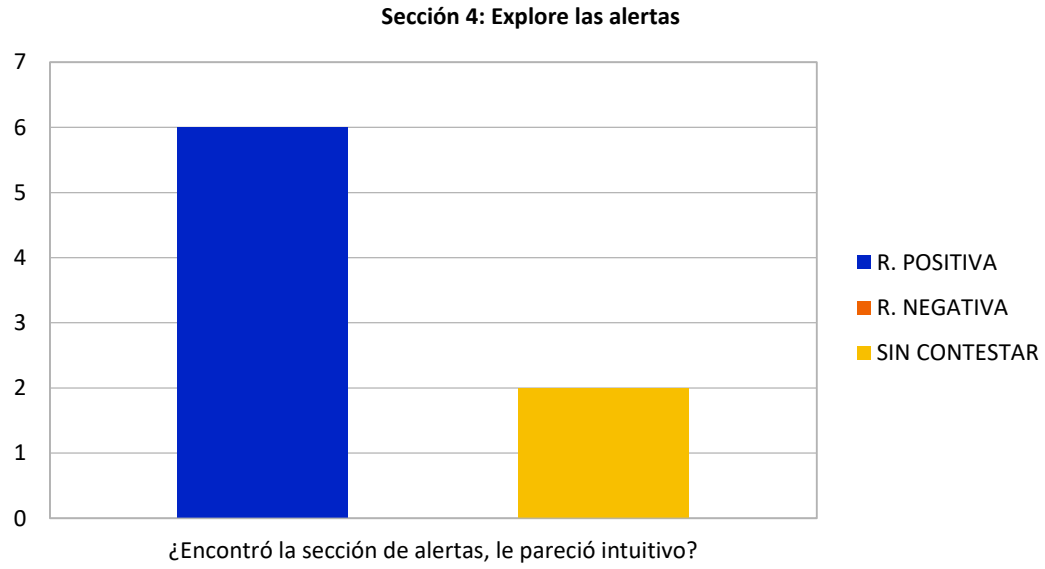
En esta pregunta existen ciertas personas que si pudieron dejar de rastrear su ubicación al pulsar el mismo botón de “Iniciar rastreo de posición”. Hubo una buena parte de personas que no lograron dejar de rastrear su posición. No existen razones escritas de dificultades

en esta tarea en las pruebas de usabilidad. Lo que podría indicar que la pregunta no fue expuesta de forma clara o se malentendió el mensaje de la misma.

Pregunta 8: ¿Qué dificultades tuvo con la aplicación?

En esta pregunta la mitad de los buscadores que participaron en esta tarea declaran no haber tenido problema con la aplicación, mientras que la otra mitad muestran que hubo problemas principalmente por el problema de adaptarse al uso de una aplicación nueva, lo cuál es comprensible. Si se indagase en más sesiones del uso del aplicativo se podría determinar una curva de aprendizaje en horas para familiarizarse con la aplicación.

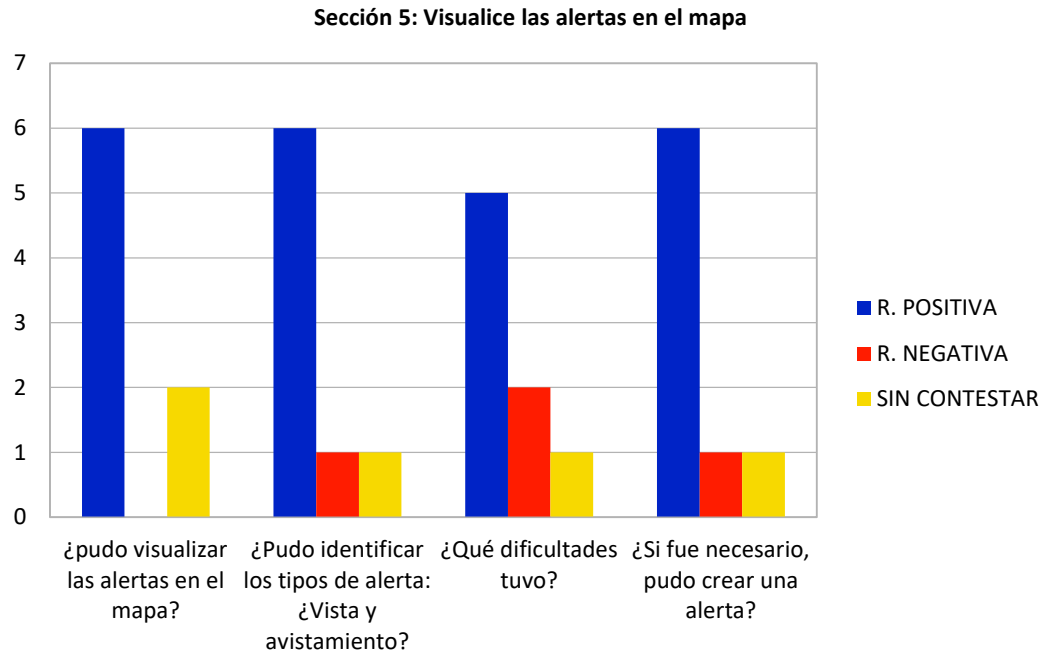
Sección 4: Explore las alertas



Pregunta 9: ¿Encontró la sección de alertas, le pareció intuitivo?

Todos los buscadores que participaron en este paso declaran haber encontrado la sección de alertas. Está sección se encuentra dentro de un “drawer” o panel lateral desplegable el cual se muestra cuando el usuario toca el ícono con 3 líneas. Quizás en este caso también influya de forma positiva la familiaridad con paneles desplegables en Android y el tiempo de adaptación a la aplicación.

Sección 5: Visualice las alertas en el mapa



Pregunta 10: ¿Pudo visualizar las alertas en el mapa?

El 100% de las personas que participaron en toda la prueba pudieron identificar los íconos de alertas en el mapa. Lo cual muestra que los íconos contrastan con el color de fondo de un mapa de Google.

Pregunta 11: ¿Pudo identificar los tipos de alerta: ¿Pista y avistamiento?

La gran mayoría de personas excepto una, pudieron diferenciar las alertas tipo Pista y Avistamiento. Se declara que por la escritura se facilitó la diferenciación de estas pistas, mientras que aún 2 personas confesaron que hubo dificultad para diferenciarlos. Se puede deber a que se necesita ir a la pantalla de Alertas o Mis Alertas para, como usuario, leer el tipo de alguna alerta y asociarlo a su ícono distintivo.

Pregunta 12: ¿Qué dificultades tuvo?

Cinco personas afirman no haber tenido dificultad, mientras que las otras dos personas comentan dificultades que fueron vistas anteriormente como el adaptarse al nuevo sistema e identificar las alertas.

Pregunta 13: ¿Si fue necesario, pudo crear una alerta?

Seis personas pudieron crear una alerta sin problemas, lo cual indica que el usuario tiene un mayor nivel de adaptación a la aplicación en este punto de la prueba. No se denotan

razones que abran el panorama de porque no se pudo continuar con la creación de alertas en este único caso.

En general se percibe una respuesta positiva por parte de los usuarios en cuanto a su nivel de satisfacción con la aplicación. En general los comentarios finales de los participantes califican a la aplicación propuesta como una buena herramienta de beneficio social. Se muestra la apertura a recibir capacitaciones sobre la herramienta. Estos resultados reflejan el entusiasmo y apertura que mostraron los participantes el día 4 de diciembre en el taller que se llevó a cabo en las instalaciones de la DINASED.

La respuesta a la aplicación propuesta fue favorable, puesto que una de las preocupaciones del personal de la DINASED era de que la aplicación no sea invasiva, es decir que no sea una herramienta netamente de control interno sobre la ubicación del policía, pero que sea una herramienta que ayude a la problemática de la búsqueda de personas desaparecidas sin atentar a la privacidad de sus usuarios. Esto se logró gracias a que el usuario tiene el control de cuando comienza a enviar datos sobre su ubicación y cuando detiene estas actualizaciones, por medio del botón Iniciar/detener rastreo de su posición.

También se realizaron pruebas de rendimiento en las cuales se determinó:

- La velocidad con la que las actualizaciones en tiempo real de Firebase se ejecutan en segundos.
- La cantidad de datos utilizados por la aplicación por un tiempo determinado en kilobytes.
- El porcentaje de CPU utilizado al rastrear la posición del buscador.

La velocidad con que las actualizaciones en tiempo real son transmitidas hacia los usuarios fue determinada al poner dos teléfonos a rastrear su posición de forma simultánea. También se simuló el movimiento del gps con una aplicación de terceros. De esta forma que cada teléfono subía datos de sus movimientos y descargaba datos de los movimientos.

Dada esta carga de datos con un teléfono se creó una alerta, cuando el botón de crear fue presionado. La actualización de una nueva alerta fue visualizada en menos de 1 segundo en el otro teléfono, aún antes de que la aplicación en el teléfono que envió la alerta se redirija a la pantalla del mapa. El experimento se repitió 2 veces.

Con respecto al uso de datos de la aplicación se le dio un uso intensivo a la aplicación por 20 minutos, en las condiciones de la prueba de velocidad anterior. En esta prueba se uso 1.9MB de bajada y 0.313MB de subida. Esto se debe a las actualizaciones del GPS que

están configuradas a ejecutarse cada 3 segundos como máximo, aumentando la precisión, pero también el consumo de datos. Lo que lo hace comparable al uso de datos de la app PlayStore en ese mismo tiempo (1.8MB).

Figura 28 Consumo de datos de la app en 20min

Finalmente se realizó una prueba para medir la utilización del CPU cuando el teléfono corrió la aplicación. A continuación, se listan las ciertas características del teléfono para la prueba, estas son las más determinantes en el desempeño de aplicaciones como son: la capacidad de procesamiento de CPU, la RAM el espacio disponible de memoria rápida fuera del caché para realizar operaciones, la renderización de gráficos y la cantidad de pixeles que se deberán llenar:

Las características son las siguientes:

Modelo	**Samsung Galaxy J7 (SMJ700M)**
Procesador	Exynos Octa 7580 (400 MHz – 1.5 GHz)
Memoria RAM	1370 MB
Resolución	720 x 1280 px

GPU	Mali-T720
Versión de Android	6.0.1

En las condiciones de las pruebas anteriores se midió los ciclos de reloj a los que tendió el CPU mientras se rastreaba la posición y se obtenían actualizaciones periódicas de localización de otro teléfono y alertas en la aplicación.

Se utilizó la App CPU-Z para medir los valores de utilización. Como resultado se obtuvo ciclos de reloj bajos (600 MHz) cuando la aplicación solo rastreaba la posición y periódicamente picos de utilización de 1100 MHz cuando actualizaciones de la posición se enviaban a Firebase. Esto se registraba en los 4 primeros hilos de ejecución, mientras que los otros 4 estaban detenidos, existieron muy escasos picos de 1500MHz. Adicionalmente la carga del CPU llegó al 30%.

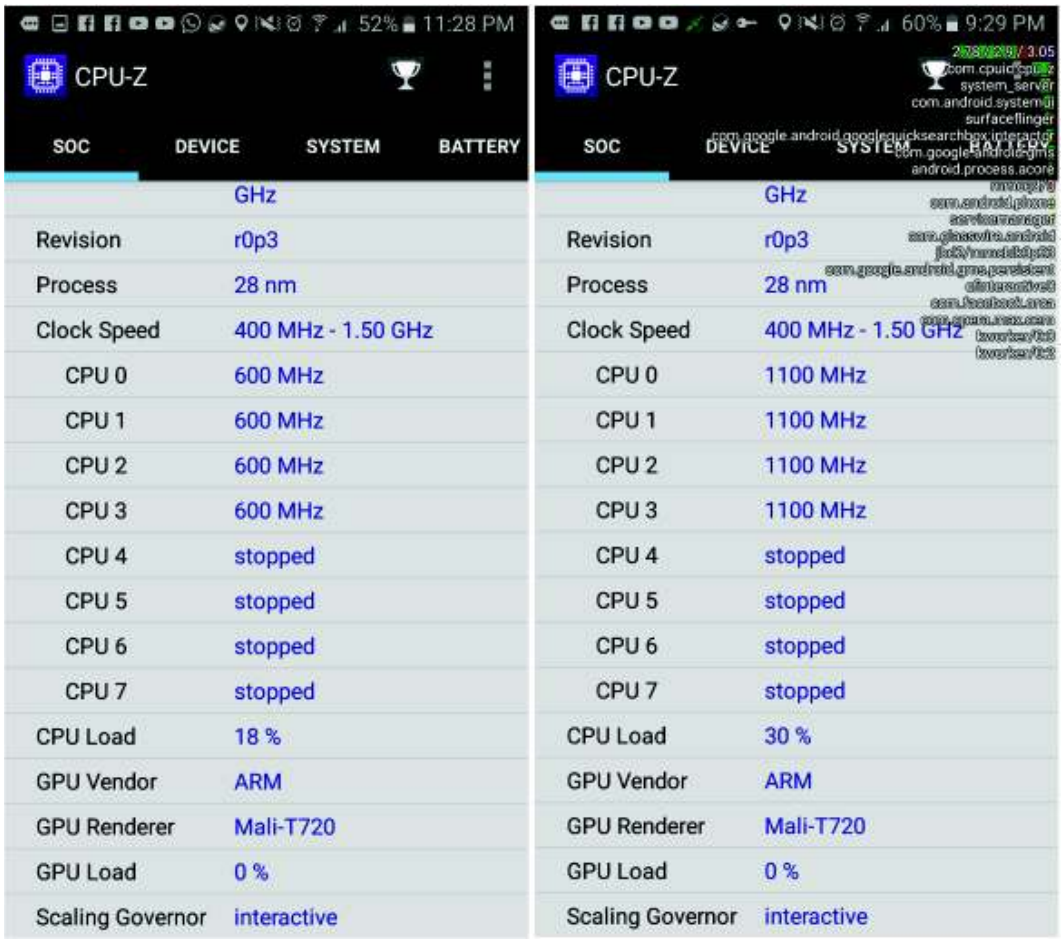


Figura 29 Carga mínima y pico del CPU con la aplicación ejecutándose en segundo plano

Los resultados arrojados por las pruebas de rendimiento denotan un desempeño muy bueno de la aplicación en cuanto a consumo de recursos de hardware. Adicionalmente se midió la temperatura del teléfono en las condiciones señaladas anteriormente las cuales no pasaron de los 36 ºC lo que muestra que ningún componente interno del teléfono alcanzo su límite. Denotando que las operaciones fueron realizadas de forma eficiente.

4. CONCLUSIONES

En este trabajo se desarrolló una aplicación móvil que puede utilizar servicios de GPS y servicios de Google como los productos de Firebase (Firestore, Authentication) y de Google Maps para Android, que fueron utilizados como servicio de base de datos NoSQL en la nube y como proveedor de mapas personalizables para mostrar información relevante a la búsqueda de personas desaparecidas. Fue construida usando tecnología nativa para tener mayor cobertura y compatibilidad con dichos servicios y funciones propias del dispositivo móvil, fortaleza de la cual carecen las aplicaciones híbridas que implementan tecnología web en su construcción.

La geolocalización de los buscadores y georreferenciación de sus trayectorias, ubicaciones y alertas generadas, fueron de suma importancia para apoyar a la búsqueda de personas desaparecidas, al enriquecer el valor de la información de los lugares que visitan o los hallazgos que comunican los buscadores.

La aplicación fue construida para dispositivos móviles con sistema operativo Android, lo cual es una fortaleza para su implementación en Ecuador puesto que, a noviembre de 2018 se registra que el 80% de la población de dispositivos móviles en nuestro país es Android, con un 15.77% de teléfonos con iOS [35]. Facilitando la difusión de la aplicación a más dispositivos.

El uso de la metodología ágil iterativa Scrum, permitió manejar los requerimientos que surgieron y que fueron modificados a lo largo del desarrollo, permitiendo la entrega de un software funcional que genere valor para el personal de la DINASED, familiares y voluntarios en la búsqueda de un desaparecido en el Distrito Metropolitano de Quito.

Basándose en las fuentes de consulta, se encontraron algunas aplicaciones en el ámbito de personas desaparecidas. Gran parte de estas se basan en rastrear los dispositivos móviles de un desaparecido. Sin embargo, el enfoque de esta aplicación es diferente ya que es una herramienta que apoya a la búsqueda y no al rastreo del celular de la persona. Independientemente de si la persona desaparecida tiene teléfono, o no, FindMe busca optimizar la búsqueda y reducir la duplicación de esfuerzos pues muestra en el mapa los lugares recorridos y el tiempo desde que se recorrió por un buscador

La aplicación fue modificada en cuanto a accesos para salvaguardar la privacidad y seguridad tanto de los usuarios como de la persona que está siendo buscada. Esto se logró al no permitir que personas no registradas puedan acceder a los datos de la búsqueda de un individuo puesto que, se podría usar malintencionadamente la aplicación por algún

agresor o cómplice en una desaparición para visualizar por donde se está buscando a la víctima, lo cual entorpecería la búsqueda.

Las pruebas de usabilidad y de un caso de estudio real que se realizó con la DINASED mostraron el uso que los agentes policiales darían a la aplicación y mostraron cuáles eran sus expectativas del software. Esto permitió medir que tan útil y amigable fue la interfaz de usuario y su experiencia con la aplicación en general. Hubo gran apoyo por parte de la institución lo que facilitó el desarrollo de la actividad.

Las pruebas de usabilidad arrojaron resultados positivos sobre la aplicación y sus funciones principales, reflejando satisfacción y una respuesta positiva con el software en general. Dando también un horizonte con oportunidades de mejora y adiciones que se pueden hacer para enriquecer la experiencia del usuario.

Según las fuentes consultadas, no existen aplicaciones cuyo fin sea el apoyar la búsqueda realizada por las autoridades competentes en casos de desapariciones, ni que este pensado en buscadores civiles como familiares, amigos o voluntarios. Lo que podría hacer de la app FindMe una de las primeras con este enfoque.

Otra fortaleza de la aplicación es que simplifica y organiza de forma visual en un mapa el progreso de una búsqueda trayendo un panorama más amplio que comunicaciones por radio o grupos de chat. Almacenando la información generada por muchas fuentes en un solo lugar, y que es útil para su análisis por parte de los usuarios, en especial para los expertos en el tópico de desapariciones.

Las pruebas de rendimiento de la aplicación denotaron un rápido tiempo de respuesta en cuanto a las actualizaciones en tiempo real de datos generados por otros usuarios como también un bajo consumo de los ciclos del CPU de la aplicación mientras realiza las tareas de rastreo y actualización de información en segundo plano. Sin embargo, el consumo de datos, aunque dentro de parámetros aceptables, consume alrededor de 1MB de bajada por cada 10 minutos de uso activo (con el botón de tracking activado). Algo que se podría optimizar al bajar la frecuencia de actualización del GPS y como se instancian en código los escuchadores de actualizaciones.

5. RECOMENDACIONES

Se recomienda implementar una interfaz que permita a la policía ingresar las cédulas o identificaciones de personas autorizadas para ayudar en la búsqueda de un desaparecido. Los denunciantes de una desaparición podrían llevar junto con la denuncia los números de cédula de familiares y amigos voluntarios para la búsqueda. El sistema validaría que las personas que se registren estén dentro de esa lista para permitirles el acceso, todo esto sería una medida de prevención al uso malicioso de la herramienta.

Para futuras versiones de la aplicación se recomienda añadir imágenes de los desaparecidos en la pantalla de Búsquedas. Para poder sincronizar las imágenes se puede usar un servicio en la nube que provea de una API para Android, existen soluciones gratuitas como Cloudinary [36] con las facilidades antes descritas.

Se recomienda dar más capacitaciones a todo el personal de la DINASED en cuanto al uso de la aplicación en caso de que la Escuela Politécnica Nacional decida trabajar en conjunto con la primera y este software sea una herramienta útil para ambas distinguidas instituciones.

Para la implementación práctica de esta aplicación, como medida de seguridad se debe coordinar con la DINASED la autorización de usuarios de la Policía que pueden registrarse. Por medio de una lista con números de cédula que pasaron por una función hash almacenados en la base, determinar que números de cédula tienen permiso para ser usuarios con el rol DINASED.

En cuanto a la creación de búsquedas activas en la aplicación por parte de los usuarios DINASED, se recomienda implementar un mapa pequeño para seleccionar el lugar del último paradero del desaparecido en lugar de los campos de latitud y longitud que se encuentran implementados en la versión actual.

Bibliografía

[1] U. d. I. d. P. Desaparecidas, «Policia Nacional del Ecuador,» [En línea]. Available: http://www.policiaecuador.gob.ec/unidad-de-investigacion-de-personas-desaparecidas/. [Último acceso: 5 Octubre 2017].

[2] Ministerio del Interior, «Dinased tiene efetividad del 67 por ciento de casos resueltos - Ministerio del Interior,» 2014. [En línea]. Available: http://www.ministeriointerior.gob.ec/dinased-tiene-efectividad-del-67-de-casos-resueltos/. [Último acceso: 5 Diciembre 2017].

[3] Ministerio del Interior, «Estadísticas de perrsonas desaparecidas año 2018,» [En línea]. Available: http://181.113.21.13:8080/registroinicial-war/desaparecidos2018.html. [Último acceso: 9 Diciembre 2018].

[4] Policia Nacional del Ecuador, «Policia Nacional del Ecuador » Dirección Nacional de Delitos Contra la Vida, Muertes Violentas, Desapariciones, Extorsión y Secuestros,» [En línea]. Available: http://www.policiaecuador.gob.ec/dinased/. [Último acceso: 18 Diciembre 2017].

[5] J. A. García Antonio y R. Mejía Guzmán, «Aplicación móvil para el apoyo en la búsqueda, rastreo y localización de personas menores a 18 años a través de boletines emitidos por Alerta AMBER - Repositorio Dspace,» 20 Febrero 2017. [En línea]. Available: http://tesis.ipn.mx/handle/123456789/21221. [Último acceso: 1 Septiembre 2017].

[6] U.S Department of Justice - Federal Bureau of Investigation, «National Criminal Justice Reference Service,» [En línea]. Available: https://www.ncjrs.gov/pdffiles1/Digitization/206248NCJRS.pdf. [Último acceso: 8 Enero 2018].

[7] Notimex, «Alerta AMBER tiene más de 95 por ciento de efectividad en Jalisco,» 26 Octubre 2016. [En línea]. Available: https://www.20minutos.com.mx/noticia/151259/0/alerta-amber-tiene-mas-de-95-por-ciento-de-efectividad-en-jalisco/. [Último acceso: 19 Enero 2018].

[8] Cimacnoticias, «Alerta AMBER: 68% de efectividad en 2012-16,» Forum en Línea, [En línea]. Available: http://forumenlinea.com/nuevo/index.php/96-revistas/forum-en-linea-347/3614-alerta-amber-68-de-efectividad-en-2012-16. [Último acceso: 19 Enero 2018].

[9] M. E. Chávez, «Introducción al concepto de geolocalización e instalación del software Google Earth,» educ.ar, [En línea]. Available: https://www.educ.ar/recursos/14673/introduccion-al-concepto-de-geolocalizacion-e-instalacion-del-software-google-earth. [Último acceso: 29 Agosto 2017].

[10] Gobierno de México, «Alerta Amber Mex: Preguntas frecuentes,» [En línea]. Available: http://www.alertaamber.gob.mx/swb/alertaamber/PreguntasFrecuentes. [Último acceso: 16 noviembre 2018].

[11] Crónica, «ALERTA EMILIA: ECUADOR, PRIMER PAÍS SUDAMERICANO EN CONTAR CON ALERTA PARA LOCALIZAR A MENORES DESAPARECIDOS EN ALTO RIESGO,» 23 Agosto 2018. [En línea]. Available: https://www.cronica.com.ec/informacion/item/22568-alerta-emilia-ecuador-primer-pais-sudamericano-en-contar-con-alerta-para-localizar-a-menores-desaparecidos-en-alto-riesgo. [Último acceso: 26 Noviembre 2018].

[12] Universidad de Alicante, «La UA diseña una app para localizar a personas desaparecidas en entornos sin cobertura de telefonía móvil,» 18 Diciembre 2017. [En línea]. Available: https://web.ua.es/es/actualidad-universitaria/2017/diciembre17/18-22/la-ua-disena-una-app-para-localizar-a-personas-desaparecidas-en-entornos-sin-cobertura-de-telefonia-movil.html. [Último acceso: 2018 Noviembre 26].

[13] Z. Viso, «Localizar a personas desaparecidas en sitios sin cobertura es posible con esta nueva app,» Nobbot: Tecnología para las personas, [En línea]. Available: https://www.nobbot.com/pantallas/app-localizar-personas-desaparecidas/. [Último acceso: 16 noviembre 2018].

[14] R. M. Robles, «IPN desarrolla aplicación para localizar personas extraviadas después de un sismo,» MVS Noticias, 26 Febrero 2018. [En línea]. Available: https://mvsnoticias.com/#!/noticias/ipn-desarrolla-aplicacion-para-localizar-personas-extraviadas-despues-de-un-sismo-139. [Último acceso: 26 Noviembre 2018].

[15] Entrepreneur Staff, «Esta aplicación te permite encontrar personas perdidas aun sin internet,» Entrepreneur, 28 febrero 2018. [En línea]. Available: https://www.entrepreneur.com/article/309752. [Último acceso: 16 noviembre 2018].

[16] Europa Press, «Nueva app de geolocalización para personas accidentadas o perdidas,» Europa Press, 03 12 2014. [En línea]. Available: https://www.europapress.es/extremadura/noticia-112-extremadura-incorpora-app-geolocalizacion-agilizar-rescate-personas-accidentadas-perdidas-20141203120813.html. [Último acceso: 27 Noviembre 2018].

[17] Safe365, «Localizador GPS familiar y de personas por móvil,» Google Play, [En línea]. Available: https://play.google.com/store/apps/details?id=app.alpify&hl=es. [Último acceso: 16 noviembre 2018].

[18] S. Ambler, «Agile Requirements Change Management,» [En línea]. Available: http://agilemodeling.com/essays/changeManagement.htm. [Último acceso: 4 Diciembre 2017].

[19] K. Schwaber y J. Sutherland, «The Scrum Guide,» Noviembre 2017. [En línea]. Available: https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf. [Último acceso: 30 Abril 2018].

[20] K. Schwaber y J. Sutherland, «Scrum Basics,» [En línea]. Available: http://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-es.pdf. [Último acceso: 7 Mayo 2018].

[21] F. Ableson, «Desarrolle aplicaciones Android con Eclipse,» IBM, 18 Febrero 2013. [En línea]. Available: https://www.ibm.com/developerworks/ssa/opensource/tutorials/os-eclipse-android/index.html. [Último acceso: 8 Mayo 2018].

[22] A. Ardíons, «Android Studio vs Eclipse,» Android Studio FAQs, 1 Febrero 2016. [En línea]. Available: https://androidstudiofaqs.com/conceptos/android-studio-vs-eclipse. [Último acceso: 8 Mayo 2018].

[23] StatCounter GlobalStats, «Mobile Operating System Share Ecuador,» StatCounter, [En línea]. Available: http://gs.statcounter.com/os-market-share/mobile/ecuador/#monthly-201802-201805. [Último acceso: 8 Mayo 2018].

[24] Google Maps Platform, «Google Maps Platform | Google Developers,» [En línea]. Available: https://developers.google.com/maps/documentation/. [Último acceso: 16 Mayo 2018].

[25] Google Maps Platform, «Precios y planes | Plataforma de Google Maps,» [En línea]. Available: https://cloud.google.com/maps-platform/pricing/. [Último acceso: 16 Mayo 2018].

[26] Firebase, «Comienza gratis, ajusta la escala y paga solo por lo que usas.,» [En línea]. Available: https://firebase.google.com/pricing/. [Último acceso: 16 Mayo 2018].

[27] Atlassian, «Jira | Software de seguimiento de proyectos e incidencias | Atlassian,» [En línea]. Available: https://es.atlassian.com/software/jira. [Último acceso: 8 Diciembre 2018].

[28] Google Maps API, «Obtener una clave de API,» [En línea]. Available: https://developers.google.com/maps/documentation/android-api/signup?hl=es-419. [Último acceso: 28 Mayo 2018].

[29] Firebase, «Estructura tu base de datos | Firebase Realtime Database | Firebase,» [En línea]. Available: https://firebase.google.com/docs/database/web/structure-data?hl=es-419. [Último acceso: 16 Mayo 2018].

[30] J. Pinzón, «Ingeniería en Sistemas: Validar Cédula Ecuatoriana (Java),» Blogspot, 5 Mayo 2013. [En línea]. Available: https://juanpinzon-1992.blogspot.com/2013/05/validar-cedula-ecuatoriana.html. [Último acceso: 13 Julio 2018].

[31] Google, «Obten actualizaciones ne tiempo real con Cloud Firestore,» [En línea]. Available: https://firebase.google.com/docs/firestore/query-data/listen?hl=es-419.

[32] Y. Rogers, H. Sharp y J. Preece, «Interaction Design,» John Wiley & Sons, Inc., United States of America, 2002.

[33] J. D. Gould y C. Lewis, «Designing for Usability: Key Principles and What Designers Think,» Communications of the ACM, 1985.

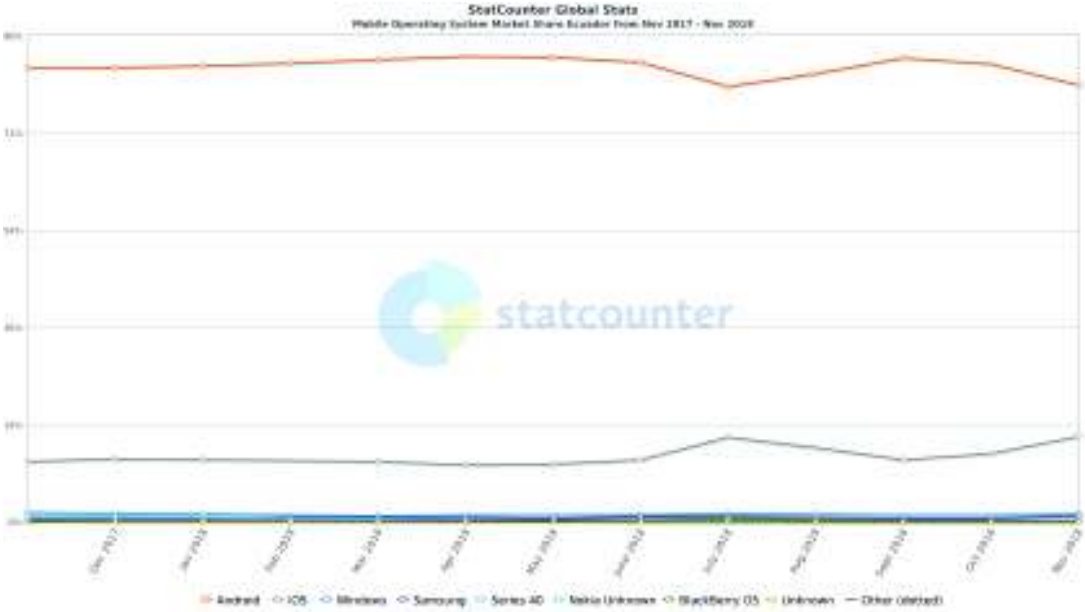
[34] G. S. Muñoz, «Análisis Univariado,» [En línea]. Available: https://www.dspace.espol.edu.ec/bitstream/123456789/19252/5/CAP%C3%8DTULO%203.-%20AN%C3%81LISIS%20UNIVARIADO.pdf. [Último acceso: 3 Diciembre 2018].

[35] Stat Counter, «Mobile Operating System Market Share Ecuador,» [En línea]. Available: http://gs.statcounter.com/os-market-share/mobile/ecuador. [Último acceso: 7 diciembre 2018].

[36] Cloudinary, «Image and Video Upload Optimized and CDN,» [En línea]. Available: https://cloudinary.com/. [Último acceso: 7 diciembre 2018].

[37] Firebase, «Firebase Realtime Database,» [En línea]. Available: https://firebase.google.com/products/database/. [Último acceso: 16 Mayo 2018].

ANEXO 1: Distribución de sistemas operativos móviles en Ecuador a noviembre 2018



En la gráfica se aprecia que el porcentaje de teléfonos con el sistema operativo Android es de casi el 81% de toda la población de sistemas operativos móviles en Ecuador superando ampliamente el 15.77% de los teléfonos Android [35]

ANEXO 2: Solicitud al Sr. Victor Manuel Araus Macias, General de Distrito y Director Nacional de la DINASED para realización de pruebas

Quito, 5 de Noviembre de 2018

General de Distrito Víctor Araus Macias

DIRECTOR NACIONAL DE LA DINASED

Presente.-

De mi consideración:

Después de saludarle atentamente, yo DAVID ENRIQUE MONCAYO BALSECA con C.C.: 1720825346 en calidad de estudiante de la ESCUELA POLITÉCNICA NACIONAL, me permito solicitarle su autorización para probar una aplicación móvil que fue revisada con la favorable atención de su equipo, el Coronel de Policia de Estado Mayor Julio César Miño, el Capitán Javier Lasso y en más profundidad con el Teniente Mario Padilla el día martes 30 de Octubre.

Como antecedente, conversé introductoriamente con el Capitán Javier Lasso que en su oficina me recibió. Después él me llevo al despacho del Coronel Julio César Miño, con quien conversamos sobre los detalles de lo que hace la aplicación y a que usuarios está dirigido. Además, conversamos sobre la posibilidad de probar el software con personal de la institución que acertadamente usted lidera en una fecha próxima. Para conversar de los detalles técnicos el Coronel asignó al Teniente Mario Padilla para conversar de los detalles de como fue implementado el software, de su funcionamiento

interno y los servicios que utiliza, al Teniente le pareció un buen acercamiento para el apoyo a la búsqueda de personas desaparecidas puesto que no es invasiva con el usuario o datos sensibles.

El software es una aplicación móvil, para el sistema operativo Android, con el propósito de apoyar a la búsqueda de personas desaparecidas llamado FindMe; este es mi proyecto de titulación en el que he trabajado 2 semestres. Él propósito es que sea una herramienta que ayude al personal de la DINASED, familiares y amigos autorizados de un desaparecido a registrar con un solo botón los lugares en los que han buscado al desaparecido y ver en tiempo real lugares recorridos por otros buscadores en un mapa interactivo. También los buscadores podrán visualizar alertas que hayan sido previamente aprobadas por la DINASED.

Con el Coronel Julio César Miño y el Teniente Mario Padilla, se estableció una fecha tentativa para estas pruebas, el sábado 10 y domingo 11, no sin antes tener su autorización.

Agradeciendo su favorable atención y respuesta a mi petición, me suscribo de usted.

Atentamente:

David Enrique Moncayo

C.C.:1720825346

Cel: 0984612938

Nota: Adjunto la certificación de mi tutora de proyecto de titulación, cédula y certificado de matrícula. Me disculpo el no adjuntar un carné estudiantil puesto que no dispongo de él por pérdida hace 4 años, favor solicito acepte como substituto el certificado de matricula.

ANEXO 3: Memorando 2018-3319-DINASED-PN del Despacho del Sr. Victor Manuel Araus Macias, General de Distrito y Director Nacional de la DINASED

Ministerio del Interior

POLICÍA NACIONAL DEL ECUADOR
DIRECCIÓN NACIONAL DE DELITOS CONTRA LA VIDA, MUERTES VIOLENTAS, DESAPARICIONES, EXTORSIÓN Y SECUESTROS

MEMORANDO No. 2018-3319-DINASED-PN
Quito, a 06 de noviembre del 2018

Señor Coronel de Policía de E.M.
Julio César Miño del Hierro
JEFE DE LA UNIDAD DE INVESTIGACIÓN DE PERSONAS DESAPARECIDAS DE LA DINASED

Para su conocimiento y fines pertinentes, remito el **Oficio N° S/N,** de fecha 05 de noviembre del 2018, suscrito por el señor David Enrique Moncayo, en el cual solicita se autorice probar una aplicación móvil el cual serviría como apoyo en la búsqueda de personas desaparecidas llamado FINDME, los días sábado 10 y domingo 11 de noviembre del 2018, herramienta tecnológica que ayudaría al personal policial de esta Dirección, familiares y amigos de un desaparecido a registrar con un solo botón los lugares en los que han buscado al desaparecido y ver en tiempo real lugares recorridos por otros buscadores en un mapa interactivo; al respecto me permito informar que esta Dirección AUTORIZA mencionada solicitud. Enterado de su contenido y dentro de las normas legales vigentes, sírvase realizar las coordinaciones pertinentes y efectúe las pruebas correspondientes e informe a este Despacho de manera oportuna los resultados obtenidos. **ESTRICTO CUMPLIMIENTO**.

Atentamente,
VALOR, DISCIPLINA Y LEALTAD

Victor Manuel Araus Macías
General de Distrito
DIRECTOR NACIONAL DE LA DINASED

DINASED-UD SECRETARIA
RECEPCIÓN DE DOCUMENTO

Dir: Av. José Herbono N° Oe7-75 y Mariscal Antonio José de Sucre, Telf: 02330157
dinased.secretaria@policiaecuador.gob.ec y dinasedecuador@gmail.com

que acertadamente usted lidera en una fecha próxima. Para conversar

ANEXO 4: Ejemplo de Prueba de Usabilidad de la aplicación tomada a un colaborador, miembro de la DINASED

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA EN SISTEMAS

PRUEBA DE USABILIDAD

Muchas gracias por ser parte de lo que Dios está haciendo en nuestra comunidad. Esta prueba tiene como propósito evaluar la experiencia de usuario de la aplicación para el apoyo de búsqueda de personas desaparecidas FindMe. Esta aplicación tiene como propósito ser un medio para coordinar la búsqueda de personas desaparecidas para que la DINASED, familiares y voluntarios autorizados puedan ayudar en la búsqueda de sus seres queridos.

A continuación, están las preguntas sobre las actividades propuestas en la aplicación:

1.- Registrarse

¿Le pareció intuitiva la pantalla de registro?

Muy creativa, debido a las necesidades, por la sociedad que se ve afectada con un familiar en alguna situación trágica.

¿Qué dificultades tuvo al registrarse?

Ninguna

2.- Seleccionar búsqueda

¿Pudo encontrar en la lista a la persona que quería buscar?

En el ejercicio, fue productivo

3.- Entrar al mapa y comenzar a rastrear su posición

¿Pudo encontrar el botón de comenzar a rastrear su posición?

Si

¿Al moverse, pudo ver en el mapa el registro de su trayectoria?

Si

¿pudo ver los movimientos de otros buscadores?

Se pudo visualizar.

¿Pudo dejar de rastrear su posición?

Al salir del programa

¿Qué dificultades tuvo con la aplicación?

Ninguna

4.- Explore las alertas

¿Encontró la sección de alertas, le pareció intuitivo?

Si, se las pude visualizar.

5.- Visualice las alertas en el mapa

¿pudo visualizar las alertas en el mapa?

Si

¿Pudo identificar los tipos de alerta: ¿Vista y avistamiento?

Si

¿Qué dificultades tuvo?

Ninguna

¿Si fue necesario, pudo crear una alerta?

Si se puede crear.

Déjenos su comentario sobre la aplicación

Muy buena, interesante

Muchas gracias por su colaboración!

ANEXO 5: Fotografías de la Exposición y Taller llevado a cabo el martes 4 de diciembre de 2018