

**UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE ELECTRÓNICA E INFORMÁTICA
CONCEPCIÓN - CHILE**



**“Gestión y diseño estratégico para la aplicación
Visión Security”**

SCARLETT DAHYANA PAVEZ VÁSQUEZ

**MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO EN INFORMÁTICA**

Profesor Guía: Javier Maldonado Carmona

Diciembre – 2024



CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): ☒ Memoria o trabajo de título; ☐ Tesis de Postgrado;

Título del trabajo: Gestión y diseño estratégico para la aplicación Visión Security

Nombre del candidato(a): Scarlett Dahyana Pavez Vásquez

Carrera / Grado: Ingeniería Informática

Campus: Concepcion ; Departamento: Electrónica e Informática

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Javier Maldonado, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL

☒ El trabajo **NO** contiene información que amerite confidencialidad y puede ser publicado de inmediato en repositorio con acceso abierto.

☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (embargo) por:

☐ 6 meses; ☐ 12 meses; ☐ 2 años; ☐ 3 años; ☐ 5 años; ☐ 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):
Proyecto que está en desarrollo para posibles inversores.

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha:

03/11/2025

Firma:

Estudiante o Candidato(a):

Fecha: 03-11-2025

Firma:

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

DEDICATORIA

Mi proyecto de título va dedicado a mi Rita.

AGRADECIMIENTOS

Quiero agradecer a mi querida Rita que, aunque ya no esté conmigo para verme terminar esta etapa, me acompañó durante el proceso de ella, yo sé que ella me acompaña y me verá en el más allá terminarla.

A mis papás por siempre estar ahí conmigo, entregarme las herramientas para tener educación, educarme, por sus esfuerzos para siempre darme lo mejor y su amor incondicional.

Y, por último, a mi amor Bastian, quien me ha acompañado y apoyado en esta etapa final.

RESUMEN

Visión Security tiene como objetivo mejorar la seguridad ciudadana en la región del Biobío, Chile, mediante una aplicación móvil que permite a los usuarios realizar reportes de incidentes como accidentes, asaltos y acoso. Esta herramienta tecnológica promueve la comunicación entre los usuarios, promoviendo una respuesta rápida y efectiva ante situaciones de emergencia.

Además, cuenta con un sitio web enfocado en clientes institucionales, como municipalidades y colegios, para proporcionar análisis de datos e informes personalizados. En este trabajo se detalla mi rol de coordinar el desarrollo del proyecto, garantizando que tanto la aplicación móvil como el sitio web cumplan con los requerimientos funcionales y técnicos.

Palabras Clave— Seguridad; Aplicación móvil; Reportes de incidentes; *Product Owner*; Región del Biobío.

ABSTRACT

Visión Security aims to improve citizen security in the Biobío region of Chile through a mobile application that allows users to report incidents such as accidents, assaults and harassment. This technological tool promotes communication between users, promoting a fast and effective response to emergency situations.

In addition, it has a website focused on institutional clients, such as municipalities and schools, to provide data analysis and customized reports. This work details my role in coordinating the development of the project, ensuring that both the mobile application and the website meet the functional and technical requirements.

Keywords— *Public safety; Mobile application; Incident reports; Product Owner; Biobío region.*

GLOSARIO

Frontend: Desarrollo de una arquitectura visual para un sitio web y aplicación móvil.

Carta Gantt: herramienta de gestión de proyectos que visualiza las tareas y sus plazos a lo largo del tiempo

framework: Estructura base o marco de trabajo que se utiliza para desarrollar un proyecto con objetivos específicos.

Sprints: Es un período breve de tiempo fijo en el que un equipo de scrum trabaja para completar una cantidad de trabajo establecida.

stakeholders: personas u organizaciones que se ven afectadas o tienen un interés en las actividades y decisiones de una empresa.

Flutter: Es un software *development kit* para interfaces de usuario de código abierto creado por Google.

Google Maps API: Las *APIs* de *Google Maps* brindan acceso a los datos y funciones de *Google Maps*.

AWS: Es una colección de servicios de computación en la nube pública

PHP, CSS Y HTML: Son lenguajes de programación que se utilizan para crear sitios web.

Bullying: Es un término que se refiere al acoso escolar.

Stakeholders: Personas u organizaciones que se ven afectadas por sus decisiones y actividades.

Workbench: Plataforma de gestión y distribución de datos de ubicación.

INDICE DE CONTENIDOS

Contenido

PORTADA	11
DEDICATORIA	3
AGRADECIMIENTOS	4
RESUMEN	5
ABSTRACT	5
GLOSARIO	6
INDICE DE CONTENIDOS	7
INDICE DE FIGURAS	9
INDICE DE TABLAS	10
INTRODUCCIÓN	11
CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA	13
1.1 CONTEXTO DEL PROBLEMA	13
1.2 LIMITACIONES DE LOS MÉTODOS ACTUALES	13
1.3 SOLUCIÓN PROPUESTA	14
1.4 ENFOQUE DEL PROYECTO	14
1.3 JUSTIFICACIÓN DEL PROYECTO	15
CAPÍTULO 2: MARCO CONCEPTUAL	17
2.1 CONCEPTOS TÉCNICOS	17
2.1.2 FRONTEND WEB	17
2.1.3 GEOLOCALIZACIÓN	17
2.1.4 BASE DE DATOS	17
2.1.5 SEGURIDAD Y PRIVACIDAD	17
2.2 METODOLOGÍAS	18
2.2.1 METODOLOGÍA AGIL	18
Fases y actividades del proyecto:	18
2.2.2 PROTOTIPADO RÁPIDO	19
2.3 RELACIONES CON OTROS TRABAJOS	19
2.3.2 APLICACIONES SIMILARES Y SU IMPACTO	20

Comentado [1]: Revisa la portada, esto no debería salir aquí.

Comentado [E82R1]:

2.4 OBJETIVOS DEL PROYECTO23

CAPÍTULO 3: PROPUESTA DE SOLUCIÓN24

3.1 DESARROLLO DE LA PROPUESTA25

3.1.1 RESULTADOS DE LA ENCUESTA27

3.2 METODOLOGÍA UTILIZADA¡Error! Marcador no definido.

3.2.1 PLANIFICACIÓN DE LOS SPRINTS32

3.3 SOLUCIÓN TECNOLÓGICA36

3.4 39

CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN40

CAPÍTULO 5: CONCLUSIONES46

REFERENCIAS BIBLIOGRÁFICAS48

INDICE DE FIGURAS

Figura 1 Planificación de Sprints	25
Figura 2 Pregunta 1	28
Figura 3 Pregunta 2	29
Figura 4 Pregunta 3	29
Figura 5 Pregunta 4	30
Figura 6 Pregunta 5	31
Figura 7 Carta Gantt	33
Figura 8 Planificación de Sprints	32
Figura 9 Costos estimados	33
Figura 10 Figura 2	37
Figura 11 Figura 3	37
Figura 12 Creación de cuenta	39
Figura 13 Inicio de sesión	39
Figura 14 Botón de emergencia	40
Figura 15 Contacto de emergencia	40
Figura 16 Correo de emergencia	41
Figura 17 Acceso denegado	41
Figura 18 Crear reporte	42
Figura 19 Menú 1	42
Figura 20 Menú 2	43
Figura 21 Descripción de Reporte	43
Figura 22 Reportes	44

INDICE DE TABLAS

Tabla 1: Comparación de tecnologías relacionadas con la seguridad	22
--	-----------

INTRODUCCIÓN

En un contexto global donde la seguridad personal es una preocupación creciente, la región del Biobío, en Chile, no es la excepción. Según el informe de la Encuesta Nacional de Seguridad Pública 2024, más del 75% de los chilenos siente temor de ser víctimas de delitos, lo que subraya la necesidad urgente de implementar soluciones tecnológicas que fortalezcan la seguridad comunitaria (Ministerio del Interior, 2024). En este escenario, el desarrollo de aplicaciones móviles para el reporte de incidentes se posiciona como una alternativa innovadora y necesaria.

La aplicación *Vision Security* tiene como objetivo principal empoderar a los usuarios de la región del Biobío, permitiéndoles reportar incidentes de seguridad a través de un mapa interactivo. Este enfoque facilita la comunicación en tiempo real con otros usuarios y las autoridades, fomentando una colaboración activa para mejorar la seguridad pública. Investigaciones recientes han demostrado que el uso de plataformas digitales puede incrementar la percepción de seguridad y optimizar la respuesta de las autoridades (Pérez & González, 2024).

A lo largo del desarrollo del proyecto *Vision Security*, se implementaron estrategias que facilitaron el cumplimiento de los objetivos establecidos y respondiera adecuadamente a las necesidades de los usuarios finales. Este trabajo incluyó la planificación, priorización de funcionalidades y supervisión constante del progreso del proyecto, todo ello bajo un enfoque iterativo basado en *sprints*, los cuales son ciclos cortos de trabajo de la metodología ágil *Scrum*, en los que se desarrolla e integran funcionalidades concretas del sistema.

El desarrollo del proyecto se dividió en tres *sprints*, ciclos de trabajo cortos y planificados que forman parte de la metodología *Scrum*, utilizada para gestionar proyectos de desarrollo de software de manera iterativa e incremental (Schwaber & Sutherland, 2020). Esta metodología permitió estructurar el trabajo en etapas manejables, con entregas funcionales al final de cada sprint, cada uno con una duración de un mes. Durante estos ciclos de trabajo, se llevaron a cabo reuniones que guiaron el progreso del equipo. Estas incluyeron:

- Reunión de planificación del sprint: En estas sesiones, se definieron y priorizaron las historias de usuario y las tareas necesarias para cumplir los objetivos del sprint.
- Reuniones diarias: Encuentros breves y diarios que permitieron revisar el progreso del equipo, identificar bloqueos y ajustar las tareas según las necesidades del día.
- Revisión del sprint: Al final de cada ciclo, se evaluó el trabajo completado y se realizó una demostración de las funcionalidades desarrolladas.
- Retrospectiva del sprint: Se reflexionó sobre el desempeño del equipo, identificando áreas de mejora y fortaleciendo los procesos que resultaron efectivos.

Reunión de refinamiento: Estas reuniones se realizaron de manera periódica para revisar y ajustar el producto, asegurando que las historias de usuario estuvieran bien definidas y alineadas con las prioridades del proyecto.

La presente memoria se estructura en 5 capítulos que abordan desde el análisis del contexto teórico de la seguridad urbana en Chile hasta la gestión y desarrollo del proyecto. Se describe la aplicación de metodologías ágiles y prototipado para la planificación, seguimiento y validación de funcionalidades a través de *sprints*. También se presentan los resultados y los aprendizajes derivados de esta experiencia.

CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA

1.1 CONTEXTO DEL PROBLEMA

En los últimos años, la región del Biobío al igual que otras regiones del país ha experimentado un aumento sostenido en delitos de alta connotación social, especialmente robos con violencia, homicidios y delitos vinculados al crimen organizado. Según el informe de la Biblioteca del Congreso Nacional de Chile (2023), entre 2019 y 2022 se observó un aumento sostenido en la tasa de delitos, destacando un alza del 47,2% en los robos con violencia y del 35,8% en los delitos de lesiones durante ese periodo.

Además del aumento en la frecuencia de estos hechos, la percepción de inseguridad también se ha intensificado. De acuerdo con la misma fuente, el 84% de la población encuestada cree que la delincuencia ha aumentado en su comuna, lo que evidencia una creciente desconfianza en los mecanismos tradicionales de denuncia y protección ciudadana (Biblioteca del Congreso Nacional, 2023, p. 7).

Actualmente, las formas más comunes de reporte de incidentes —como las llamadas telefónicas a las autoridades, publicaciones en redes sociales o denuncias presenciales— son fragmentadas y poco eficientes. Esta situación dificulta una respuesta rápida y organizada por parte de las instituciones encargadas de la seguridad, y refuerza la sensación de abandono entre la ciudadanía.

En respuesta a este contexto, mi rol en el desarrollo del proyecto *Vision Security* consistió en coordinar las etapas del trabajo a través de metodologías ágiles y prototipado rápido. Desde la definición de requerimientos hasta la supervisión de entregas, el enfoque estuvo en construir una solución digital que facilitara la comunicación entre ciudadanos y permitiera una reacción más oportuna frente a incidentes de seguridad.

1.2 LIMITACIONES DE LOS MÉTODOS ACTUALES

La situación actual presenta varias dificultades que complican la gestión de los incidentes de seguridad:

- **Dispersión de la Información:** Los reportes son dispersos y llegan a través de diferentes canales, lo que retrasa las respuestas. Además, no existe un sistema que centralice la información de manera accesible para las autoridades.
- **Accesibilidad Limitada:** Las plataformas disponibles no son fáciles de usar para todos los ciudadanos, especialmente aquellos que no están familiarizados con las redes sociales o plataformas digitales.
- **Coordinación Ineficiente:** Las autoridades carecen de herramientas para gestionar

y priorizar los reportes de manera efectiva, lo que afecta la capacidad de respuesta ante incidentes urgentes.

Estas limitaciones crearon la necesidad de desarrollar una solución que centralizara la información de manera rápida y accesible, permitiendo una respuesta más eficaz ante los incidentes.

1.3 SOLUCIÓN PROPUESTA

La labor desempeñada en la gestión de "*Vision Security*" se enfocó en la planificación, administración y aseguramiento de que la plataforma incorporara funcionalidades diseñadas para resolver los problemas identificados en los métodos tradicionales de reporte. Entre las funcionalidades integradas al sistema, se destacan las siguientes:

- **Botón de Emergencia:** En colaboración con el equipo de desarrollo, se definió la incorporación de un botón de emergencia que al ser activado se envía automáticamente la ubicación del usuario a un contacto de emergencia. Esta herramienta está diseñada para mejorar los tiempos de respuesta ante situaciones críticas.
- **Reportes:** Coordiné con el equipo para desarrollar una estructura de sistema que permitiera a los diferentes tipos de usuarios (común y estudiante) realizar reportes de manera eficiente y adaptada a sus necesidades específicas. El usuario común tiene la posibilidad de reportar incidentes como robos, acosos o accidentes de tráfico, mientras que el estudiante está limitado a realizar reportes dentro del entorno escolar, como situaciones de *bullying*, accesibles solo para miembros de la comunidad educativa.
- **Accesibilidad para Todos:** Me aseguré de que la plataforma fuera fácil de usar, incluso para aquellos que no estaban familiarizados con el uso de herramientas digitales, permitiendo una participación inclusiva de todos los ciudadanos.

1.4 ENFOQUE DEL PROYECTO

Mi labor en el proyecto incluyó la planificación y la coordinación del equipo que trabajó en el desarrollo de la plataforma. Algunas de mis responsabilidades fueron:

- **Definir Prioridades:** Identifiqué las funcionalidades clave que eran necesarias para mejorar la gestión de la seguridad, priorizando aquellas que respondieran a las dificultades más urgentes.
- **Supervisar el Desarrollo:** Coordiné con el equipo para asegurar que la implementación de las funcionalidades se alinea con las necesidades de los usuarios y se desarrollara de forma eficiente.

- **Seguimiento del Progreso:** Durante el desarrollo del proyecto, realicé un seguimiento continuo para observar cómo cada fase abordaba las problemáticas de seguridad y fomentaba una mejor comunicación entre los ciudadanos.

1.3 JUSTIFICACIÓN DEL PROYECTO

La percepción de inseguridad en la región del Biobío ha crecido considerablemente en los últimos años, influida por un aumento en los delitos de mayor connotación social. En este escenario, las limitaciones de los métodos tradicionales de reporte, como las llamadas telefónicas o publicaciones en redes sociales, subrayan la necesidad de una solución tecnológica que permita centralizar y agilizar la comunicación de incidentes de seguridad.

Mi trabajo se enfocó en coordinar y priorizar las funcionalidades necesarias para que la aplicación respondiera a las necesidades específicas de sus usuarios. Esto implicó liderar el proceso de definición de objetivos claros y establecer prioridades en el desarrollo, asegurando que el equipo técnico pudiera trabajar de manera alineada con el propósito del proyecto.

Entre las responsabilidades asignadas como *product owner* rol encargado de definir y priorizar los requerimientos del sistema para asegurar su alineación con los objetivos del proyecto se incluyó:

- Coordinar y participar en *sprints*, asegurando que los objetivos de cada iteración estuvieran alineados con las prioridades identificadas en el proyecto.
- Facilitar la comunicación entre el equipo, asegurando que las decisiones tomadas durante el desarrollo se mantuvieran en línea con los objetivos planteados.
- Supervisar la implementación de características específicas, como el botón de emergencia, velando por su alineación con los criterios establecidos y su utilidad en situaciones críticas.

El rol que se me fue asignado fue clave para la gestión en el equipo técnico para que estos contaran con un enfoque claro y que las funcionalidades se implementaran de forma estratégica, maximizando el impacto del proyecto.

El impacto esperado de *Vision Security* no solo radica en su capacidad para centralizar y agilizar los reportes de incidentes, sino también en la mejora de la percepción de seguridad.

Tecnologías Empleadas

Vision Security emplea una combinación de tecnologías avanzadas diseñadas para maximizar su efectividad en la prevención y gestión de incidentes de seguridad. Estas tecnologías permiten una experiencia intuitiva y adaptable para usuarios individuales,

comunidades, colegios y municipalidades:

- **Geolocalización:** La capacidad de identificar y compartir la ubicación en tiempo real es fundamental para las alertas de emergencia, permitiendo a los usuarios compartir su ubicación precisa con servicios de emergencia o contactos de emergencia.
- **Plataformas Móviles y Web:** *Vision Security* ofrece interfaces tanto móviles como *web*, adaptadas a los diferentes usuarios (particulares, colegios y municipalidades), garantizando un uso eficiente y adecuado a las necesidades específicas de cada uno.
- **Botón de Emergencia:** Este botón al ser presionado envía automáticamente la ubicación actual al contacto de emergencia. Esta funcionalidad es especialmente relevante en situaciones críticas en las que el usuario no puede hacer un reporte.
- **Autenticación y Control de Acceso:** Con un sistema de autenticación, *Vision Security* asegura que solo los usuarios autorizados tengan acceso a funcionalidades pertinentes, como el reporte de incidentes de acoso escolar en colegios o los módulos administrativos exclusivos para estos y las municipalidades.
- **Notificaciones y Alertas Automatizadas:** El sistema cuenta con un mecanismo que envía alertas y notificaciones en tiempo real a los usuarios registrados sobre eventos o situaciones de riesgo, mejorando la coordinación y colaboración de la comunidad en dichas situaciones.

CAPÍTULO 2: MARCO CONCEPTUAL

2.1 CONCEPTOS TÉCNICOS

2.1.1 ENTORNO DE DESARROLLO

Para el desarrollo móvil se utiliza Flutter, un *framework* de UI Google que permite crear aplicaciones nativas multiplataforma de alta eficiencia (Google Developers, 2023). Esta elección se basa en su capacidad para mantener una única base de código que funciona tanto en *Android* como en *iOS*, reduciendo los tiempos y costos de desarrollo. El lenguaje de programación empleado es *Dart*, conocido por su velocidad y facilidad para construir interfaces fluidas, lo que asegura un rendimiento óptimo en dispositivos móviles.

Comentado [3]: Referencia

2.1.2 FRONTEND WEB

El desarrollo del *frontend* del sitio web, es decir, la parte visual e interactiva con la que los usuarios interactúan directamente se realizó utilizando combinado con tecnologías estándar como *HTML*, *CSS* y *JavaScript*. *PHP* fue seleccionado por su capacidad para gestionar interacciones dinámicas con el servidor mientras que *HTML*, *CSS* y *JavaScript* permitieron construir una interfaz intuitiva, adaptable y eficiente).

2.1.3 GEOLOCALIZACIÓN

Dentro de la aplicación móvil, se implementa un sistema de geolocalización utilizando *Google Maps API*, que permite a los usuarios realizar reportes de incidentes en ubicaciones específicas. Este sistema incluye funcionalidades como marcadores personalizados para eventos y la posibilidad de trazar rutas hacia puntos de interés. La integración asegura una experiencia de usuario sencilla y confiable, cuidando la privacidad mediante el anonimato en los reportes.

Gant

2.1.4 BASE DE DATOS

Para la gestión de datos, se utiliza *MySQL* un sistema de gestión de bases de datos relacional reconocido por su estabilidad, rendimiento y escalabilidad. La administración y modelado de las bases de datos se realiza mediante con *MySQL* una herramienta oficial que facilita el diseño, consultas y mantenimiento de bases de datos de forma visual (Oracle, 2023). Además, el sistema se implementa en *AWS (Amazon Web Services)*, lo que garantiza alta disponibilidad, tolerancia a fallos y escalabilidad, aspectos clave para aplicaciones que manejan datos sensibles en tiempo real.

2.1.5 SEGURIDAD Y PRIVACIDAD

La seguridad de la aplicación se refuerza mediante el uso de *HTTPS*, que cifra las comunicaciones entre el cliente y el servidor, reduciendo el riesgo de interceptación de datos. También, se implementan mecanismos de encriptación para proteger la información sensible almacenada, contribuyendo así a la integridad y confidencialidad de los datos del usuario. Además, se emplea autenticación de usuarios a través de contraseñas cifradas y *tokens* de sesión, minimizando riesgos de acceso no autorizado. Estas medidas aseguran la

protección de la información de todos los usuarios y refuerzan la confianza en el sistema.

2.2 METODOLOGÍAS

El desarrollo de *Vision Security* se llevó a cabo mediante la implementación de metodologías complementarias: la metodología ágil y el prototipado rápido. Estas permitieron una gestión eficiente del proyecto, priorizando las necesidades de los usuarios y cumpliendo con las entregas funcionales a través de un enfoque iterativo y adaptativo.

2.2.1 METODOLOGÍA ÁGIL

La metodología ágil es un enfoque de gestión de proyectos que permite una respuesta flexible a los cambios mediante iteraciones cortas, retroalimentación constante y entregas incrementales. En *Vision Security*, las tareas se organizaron en sprints ciclos breves de trabajo planificado que permitieron cumplir objetivos específicos dentro de tiempos definidos. Esta estrategia facilitó la planificación, el seguimiento del trabajo y la adaptación continua según las necesidades del equipo y las observaciones de los usuarios finales.

Según el Scrum Guide (Schwaber & Sutherland, 2020), el enfoque ágil busca *“entregar valor de forma continua a través de equipos autogestionados que trabajan en ciclos de desarrollo iterativos, denominados sprints”*.

Además, se utilizaron artefactos para documentar avances y validar funcionalidades durante el proceso.

Fases y actividades del proyecto:

Reunión de planificación del sprint:

- Durante estas reuniones, se definieron y priorizaron las historias de usuario basadas en las necesidades que requiere cada usuario, de las municipalidades, colegios y comunidades.
- Se establecieron objetivos claros para cada sprint, incluyendo el desarrollo de funcionalidades como el botón de emergencia, el mapa de reportes y la creación de comunidades.
- Mi rol consistió en seleccionar las tareas para el sprint que se alinearon con los objetivos generales del proyecto y estuvieran claramente detalladas para el equipo de desarrollo.

Reuniones diarias:

- Se llevaron a cabo encuentros cortos para revisar el avance de las tareas asignadas, identificar bloqueos y ajustar el trabajo según fuera necesario.
- A través de estas reuniones, se mantuvo un flujo constante de comunicación entre los miembros del equipo, permitiendo resolver dudas sobre los requisitos de las

Comentado [4]: Recien aquí las nombras!! Este es el centro de tu trabajo, aplicar estas metodologías. Referencias para ambas.

Comentado [5]: REferencia

funcionalidades y ajustando prioridades según los comentarios recibidos.

Revisión del sprint:

- Al finalizar cada sprint, se realizaron pruebas de demostración el cual emplea las funcionalidades desarrolladas para validar que cumplieran con los requisitos iniciales y las expectativas de los usuarios clave.
- En esta etapa, se revisaron y discutieron puntos específicos de cada funcionalidad, como la precisión de la geolocalización o la usabilidad del mapa de reportes.

Retrospectiva del sprint:

- Se reflexionó sobre lo que funcionó bien y las áreas que requerían mejoras en el proceso de trabajo.
- Se propusieron ajustes para optimizar la definición de las historias de cada usuario emplear una mejor alineación con los objetivos establecidos, además de proponer mejoras para evitar bloqueos recurrentes identificados en el equipo.

Reunión de refinamiento:

- Estas reuniones se llevaron a cabo de forma periódica para revisar y ajustar las historias de usuario, asegurando que estuvieran correctamente priorizadas y detalladas.
- Se realizaron organizaciones en las tareas, de manera que se enfocaran primero en las funcionalidades más críticas (como punto principal), tales como; el botón de emergencia, para maximizar el impacto en la experiencia del usuario final.

2.2.2 **PROTOTIPADO RÁPIDO**

El prototipado rápido es una técnica de desarrollo que permite generar versiones preliminares de un producto digital, con el objetivo de evaluar su funcionamiento, diseño y usabilidad en etapas tempranas. En el caso de *Vision Security*, esta metodología facilitó la creación de maquetas funcionales de la aplicación y el sitio web, que fueron presentadas a los profesores para recoger retroalimentación y detectar posibles mejoras antes de su implementación final. Esta práctica favoreció iteraciones ágiles y una mejora continua del producto, alineada con las necesidades reales de los usuarios.

"El prototipado rápido permite a los desarrolladores detectar errores en etapas iniciales del proyecto y ajustar características de diseño o funcionalidad en función del feedback recibido, reduciendo tiempos y costos en fases posteriores" (Preece, Rogers & Sharp, 2015).

2.3 RELACIONES CON OTROS TRABAJOS

2.3.1 ESTUDIOS SOBRE TECNOLOGÍA Y SEGURIDAD

Investigaciones recientes han explorado cómo las herramientas tecnológicas contribuyen a mejorar la seguridad pública, evidenciando su impacto positivo en la prevención y gestión

de incidentes. Según *News America Digital*, la integración de tecnología avanzada en plataformas de seguridad permite a los profesionales y fuerzas del orden obtener “un mayor conocimiento de la situación”. Además, se destaca que los sistemas de seguridad física modernos facilitan la colaboración en investigaciones mediante la recopilación, almacenamiento y revisión de pruebas digitales desde una única aplicación. Este enfoque no solo mejora la eficiencia operativa, sino que también fomenta una respuesta coordinada y basada en datos ante los desafíos de seguridad (“Cómo las nuevas tecnologías están transformando la seguridad ciudadana en Chile”, 2023).

El uso de plataformas tecnológicas centradas en la seguridad de las ciudades ha sido particularmente relevante en contextos urbanos, donde la densidad de población y la complejidad de los problemas sociales exigen soluciones integrales. Por ejemplo, proyectos como las ciudades inteligentes incorporan sensores, cámaras de vigilancia y análisis de datos en tiempo real para mejorar la seguridad ciudadana. Estas innovaciones no solo han demostrado reducir los índices de criminalidad en zonas críticas, sino que también han fortalecido la percepción de seguridad entre los habitantes.

En Chile, el desarrollo de tecnologías adaptadas a las necesidades locales, como aplicaciones de reporte ciudadano y sistemas de monitoreo en tiempo real, responde a un creciente interés por implementar soluciones que complementen las políticas públicas en seguridad. *Vision Security* se alinea con esta tendencia al ofrecer una plataforma que combina funcionalidades de geolocalización, notificaciones en tiempo real y herramientas específicas para distintos usuarios, tales como municipalidades, colegios y ciudadanos.

2.3.2 APLICACIONES SIMILARES Y SU IMPACTO

En el panorama actual, diversas aplicaciones tecnológicas han surgido como respuesta a la necesidad de mejorar la seguridad ciudadana mediante herramientas digitales. Estas aplicaciones destacan por su capacidad de centralizar información, facilitar la comunicación y optimizar la gestión de incidentes. No obstante, su efectividad varía según el enfoque, el público objetivo y las funcionalidades que ofrecen.

Una de las aplicaciones más reconocidas en el ámbito de la seguridad familiar es *Life360*. Su principal característica es el rastreo en tiempo real, que permite a los miembros de una familia conocer la ubicación de sus seres queridos en todo momento. Además, incluye funciones como alertas de llegada a destinos predefinidos, detección de accidentes automovilísticos y llamadas de emergencia. Estas funcionalidades hacen que *Life360* sea altamente valorada en el entorno familiar, ya que proporciona tranquilidad y conexión constante entre sus usuarios. Sin embargo, su diseño está limitado a las necesidades de pequeños grupos, lo que deja fuera a comunidades más amplias, como barrios, colegios o municipalidades, donde los requerimientos de seguridad son más complejos y diversos.

Otra aplicación relevante en este ámbito es *SOSafe*, ampliamente utilizada en Chile y otros países de América Latina. *SOSafe* permite a los ciudadanos reportar incidentes como robos,

accidentes, incendios y otras emergencias directamente desde sus dispositivos móviles. Los reportes son geolocalizados y se comparten con otros usuarios de la aplicación, fomentando la colaboración comunitaria. Además, *SOSafe* cuenta con alianzas estratégicas con municipalidades, lo que permite que algunos de los reportes sean derivados a las autoridades correspondientes. A pesar de su efectividad en la generación de redes de colaboración ciudadana, *SOSafe* carece de herramientas específicas para entornos educativos o módulos especializados para autoridades municipales, lo que puede limitar su capacidad de respuesta en escenarios que demandan un enfoque más personalizado.

En el ámbito educativo, las aplicaciones de reporte suelen estar menos desarrolladas. Aunque existen plataformas diseñadas para la comunicación interna en colegios, estas suelen centrarse en aspectos administrativos o pedagógicos, dejando de lado la seguridad. *Vision Security* se diferencia al ofrecer un espacio seguro para reportar incidentes de *bullying* y acoso, con acceso restringido solo a estudiantes y personal del colegio. Esto no solo fomenta un entorno más seguro, sino que también promueve la confianza entre los usuarios al garantizar la confidencialidad de los reportes.

Vision Security busca combinar lo mejor de estas aproximaciones, integrando funcionalidades avanzadas como:

- **Geolocalización en tiempo real**, que permite a los usuarios reportar emergencias desde su ubicación exacta.
- **Alertas personalizadas y notificaciones en tiempo real**, adaptadas a las necesidades específicas de los usuarios.
- **Módulos especializados para municipalidades**, que incluyen herramientas para visualizar, analizar y gestionar reportes de incidentes en sus áreas de responsabilidad.
- **Espacios seguros para colegios**, que abordan problemáticas específicas como el *bullying*, garantizando un entorno controlado y de acceso limitado.

La diferenciación de *Vision Security* no solo radica en la amplitud de sus funcionalidades, sino también en su diseño adaptable, que le permite responder a las necesidades de comunidades diversas, desde usuarios individuales hasta instituciones educativas y gubernamentales. Este enfoque integral asegura que la aplicación no solo sea útil para la gestión de emergencias, sino también una herramienta clave para fomentar la seguridad colaborativa y la participación ciudadana.

A continuación, se presenta una tabla que compara *Vision Security* con otras tecnologías o sistemas orientados a la seguridad ciudadana, destacando sus principales funcionalidades, ventajas, limitaciones y el público objetivo al que están dirigidas:

Tabla 1: Comparación de tecnologías relacionadas con la seguridad

PRODUCT OWNER VISION SECURITY

Fuente: Elaboración Propia.

Tecnología/ Sistema	Funcionalidades principales	Ventajas	Limitaciones	Público objetivo
<i>Life360</i>	Seguimiento en tiempo real de familiares y alertas de ubicación.	Muy popular, fácil de usar, ideal para familias.	Enfocado principalmente en grupos familiares.	Familias.
<i>SOSafe</i>	Reportes comunitarios, alertas de emergencia, y mapas de incidentes.	Comunidad activa, notificaciones en tiempo real.	Algunas funciones avanzadas son de pago.	Ciudadanos y comunidades vecinales.
<i>Vision Security</i>	Reportes de seguridad (robos, acosos, accidentes), geolocalización, alertas automatizadas, integración con colegios y municipalidades.	Centraliza información, promueve la colaboración ciudadana.	En desarrollo; requiere adopción masiva para máximo impacto.	Ciudadanos, municipalidades, colegios

Vision Security no solo ofrece funcionalidades comparables con las tecnologías existentes, sino que también introduce un enfoque adaptado a diversos públicos, como comunidades educativas y gobiernos locales. Esto permite que su impacto sea más amplio y especializado, posicionándola como una herramienta clave en la mejora de la seguridad ciudadana.

2.4 OBJETIVOS DEL PROYECTO

Objetivo General:

Coordinar y supervisar el desarrollo de una solución tecnológica que centralice la comunicación y gestión de incidentes de seguridad, mejorando la percepción y efectividad de las medidas de seguridad ciudadana en la región del Biobío. Esto incluye la integración de tecnologías como geolocalización y el botón de emergencia, manteniendo un enfoque centrado en las necesidades de los usuarios finales.

Objetivos Específicos:

Organizar el proyecto mediante metodologías ágiles:

- Dividir el desarrollo en *sprints* mensuales para estructurar el trabajo en ciclos iterativos.
- Participar en reuniones clave como la planificación, revisión y retrospectiva de los *sprints* para garantizar la correcta ejecución de las tareas.

Definir y priorizar funcionalidades del proyecto:

- Establecer prioridades para asegurar que las funcionalidades sean desarrolladas oportunamente.
- Gestionar las historias de los usuario y tareas del proyecto de acuerdo con los avances y necesidades identificadas en cada sprint.

Asegurar el cumplimiento de los objetivos del proyecto:

- Supervisar la implementación de las funcionalidades principales, asegurando que estas se alineen con los objetivos del proyecto.
- Garantizar que el equipo cuente con los recursos y la información necesarios para avanzar en el desarrollo del producto.

CAPÍTULO 3: PROPUESTA DE SOLUCIÓN

En el desarrollo de la aplicación *Vision Security*, se empleó una metodología ágil basada en *sprints* de un mes cada uno, para asegurar la entrega de funcionalidades de forma iterativa y continua. A lo largo de tres *sprints*, se abordaron las principales características de la plataforma, cumpliendo con las necesidades de los usuarios y asegurando la calidad del producto final. Cada sprint se centró en un conjunto de funcionalidades clave, que fueron validadas y refinadas mediante pruebas de usabilidad y retroalimentación constante.

- **Sprint 1:** Implementación de la funcionalidad de reportes de incidentes, donde los usuarios pueden seleccionar la ubicación de los incidentes en el mapa y generar reportes en tiempo real. Además, se implementó el botón de emergencia, que permite a los usuarios enviar alertas de emergencia rápidamente a sus contactos registrados en caso de un incidente.
- **Sprint 2:** Desarrollo del historial de incidentes, permitiendo a los usuarios ver y analizar eventos pasados en un área determinada. Además, se incorporó la funcionalidad de crear comunidades y ver la ubicación de amigos en el mapa, lo que permitió fortalecer la interacción social entre los usuarios, aumentando la sensación de seguridad al poder monitorizar en tiempo real la ubicación de sus seres cercanos.
- **Sprint 3:** Evaluación de la efectividad de la aplicación a través de reportes personalizados dirigidos a clientes premium, incluyendo la prueba beta del servicio.

3.1 DESARROLLO DE LA PROPUESTA

El proceso de desarrollo de la solución se llevó a cabo utilizando una metodología ágil, estructurada en tres *sprints* de un mes cada uno. Un *sprint* es un ciclo de trabajo breve y enfocado, característico de la metodología *Scrum*, en el cual se desarrollan funcionalidades incrementales del producto.

Cada *sprint* tuvo un enfoque específico, permitiendo entregar versiones funcionales y parciales de la aplicación que facilitarían la interacción progresiva de los usuarios con el sistema.

En este caso, los *sprints* fueron organizados de manera práctica en el marco de las asignaturas del año académico, lo que permitió avanzar de forma iterativa y recibir retroalimentación constante de los profesores, quienes actuaron como evaluadores y validadores de los avances del proyecto.

Figura 1 Planificación de Sprints

PLANIFICACIÓN DE CADA SPRINTS						
Sprint	Duración	Historia de Usuario	Condiciones de Aceptación	Reuniones	Experimentos	Prioridad
Numero 1	1mes	Como Usuario/Alumno, quiero poder reportar incidentes como acosos, robos, bullying, accidentes, entre otros, en el mapa, para que otros Usuarios/Alumnos puedan ser alertados y tomar medidas preventivas.	1. Selección de Ubicación: El usuario debe poder seleccionar y marcar la ubicación del incidente en el mapa. 2. Detalles del Reporte: El reporte debe incluir una descripción detallada y una categoría específica del incidente (por ejemplo, acoso, robo, bullying, accidente). 3. Visibilidad en Tiempo Real: El reporte debe ser visible para otros usuarios en tiempo real y actualizable automáticamente en el mapa. 6. Botón de emergencia: El usuario podrá enviar una alerta a través del botón de emergencia, enviando su ubicación actual a su contacto de emergencia.	1. Reunión de planificación del sprint: Definición y priorización de las historias de usuario y tareas para el próximo sprint. 2. Reunión diaria: Encuentros cortos y diarios para revisar el progreso, identificar bloqueos y ajustar el trabajo del día. 3. Revisión del sprint: Evaluación del trabajo completado durante el sprint y demostración de las funcionalidades desarrolladas. 4. Retrospectiva del sprint: Reflexión sobre el sprint para identificar lo que funcionó bien, lo que no. 5. Reunión de Refinamiento del Backlog: Sesiones periódicas para revisar y ajustar el backlog del producto, asegurando que las historias de usuario estén bien definidas y priorizadas.	1. Validación de Reportes de Incidentes. 2. Pruebas de Usabilidad 3. Pruebas de Rendimiento 4. Evaluación de Visibilidad en Tiempo Real 5. Recolección de Feedback de Usuarios	Alta
Numero 2	1mes	Como usuario, quiero poder ver un historial de incidentes en el área donde me encuentro, para analizar la seguridad de diferentes ubicaciones y tomar decisiones informadas sobre por dónde transitar.	1. Inicio de Sesión: El usuario debe iniciar sesión y acceder a la aplicación móvil correctamente. 2. Interacción con la App: El usuario debe poder interactuar con el menú y el mapa de la aplicación móvil sin problemas. 3. Acceso al Historial: El usuario debe poder acceder a un historial de incidentes. 4. Visualización del Historial: El historial debe mostrar un mapa con los incidentes marcados y la posibilidad de ver detalles de cada incidente al hacer clic en ellos. 5. Creación de comunidades: El usuario debe poder crear comunidades y ver la ubicación de sus amigos.	1. Reunión de planificación del sprint: Definición y priorización de las historias de usuario y tareas para el próximo sprint. 2. Reunión diaria: Encuentros cortos y diarios para revisar el progreso, identificar bloqueos y ajustar el trabajo del día. 3. Revisión del sprint: Evaluación del trabajo completado durante el sprint y demostración de las funcionalidades desarrolladas. 4. Retrospectiva del sprint: Reflexión sobre el sprint para identificar lo que funcionó bien, lo que no. 5. Reunión de Refinamiento del Backlog: Sesiones periódicas para revisar y ajustar el backlog del producto, asegurando que las historias de usuario estén bien definidas y priorizadas.	1. Validación del Historial de Incidentes 2. Pruebas de Usabilidad 3. Evaluación de Actualización de Datos 4. Recopilación de Feedback	Alta
Numero 3	1mes	Como cliente/usuario, quiero poder analizar la efectividad de la app móvil a través de reportes personalizados, para evaluar si es conveniente contratar el servicio premium.	1. Acceso a la Prueba Beta: El cliente/usuario debe ingresar a la prueba beta utilizando una cuenta proporcionada por Vision Security. 2. Interacción con la App: El cliente/usuario debe poder visualizar el mapa y realizar reportes personalizados, específicos para el área administrativa de su municipio. 3. Evaluación de Efectividad: A través de la prueba, el cliente/usuario debe poder evaluar la efectividad de los reportes.	1. Reunión de planificación del sprint: Definición y priorización de las historias de usuario y tareas para el próximo sprint. 2. Reunión diaria: Encuentros cortos y diarios para revisar el progreso, identificar bloqueos y ajustar el trabajo del día. 3. Revisión del sprint: Evaluación del trabajo completado durante el sprint y demostración de las funcionalidades desarrolladas. 4. Retrospectiva del sprint: Reflexión sobre el sprint para identificar lo que funcionó bien, lo que no. 5. Reunión de Refinamiento del Backlog: Sesiones periódicas para revisar y ajustar el backlog del producto, asegurando que las historias de usuario estén bien definidas y priorizadas.	1. Validación de Reportes Personalizados 2. Evaluación de Usabilidad 3. Verificación de Precisión y Relevancia 4. Recolección de Feedback de Clientes	Alta

Comentado [7]: Necesitas agregar evidencias de todo lo que estas contando. Ese fue tu trabajo, ahí esta donde debes mostrar todo lo que hiciste.

Sprint 1: Reporte de Incidentes en el Mapa

Comentado [8]: Evidencias de todo lo que comentastes.
Reuniones, pruebas, reportes y demás...

En el primer *sprint*, el objetivo fue crear la funcionalidad básica de la aplicación, permitiendo a los usuarios reportar incidentes en el mapa. Durante este *sprint*, se trabajó en las siguientes características clave:

- **Selección de Ubicación:** El usuario podía seleccionar la ubicación del incidente en el mapa, permitiendo una visualización precisa de los eventos.
- **Detalles del Reporte:** Los usuarios podían ingresar una descripción detallada y seleccionar una categoría para cada incidente (por ejemplo, acoso, robo, accidente).
- **Visibilidad en Tiempo Real:** Los reportes creados serán visibles en tiempo real en el mapa para otros usuarios, permitiendo alertar sobre incidentes ocurridos en la zona.
- **Botón de Emergencia:** Se implementó un botón de emergencia para que los usuarios pudieran enviar alertas rápidamente a sus contactos registrados en caso de un incidente.

Sprint 2: Historial de Incidentes

El segundo sprint se centró en desarrollar dos funcionalidades clave: el historial de incidentes y la opción de crear comunidades y ver la ubicación de amigos en el mapa. Ambas características fueron fundamentales para proporcionar a los usuarios una experiencia más completa y mejorar la interacción social dentro de la plataforma.

Las características desarrolladas durante este *sprint* fueron:

- **Acceso al Historial:** Los usuarios podrán ver el historial de incidentes ocurridos en su zona, lo que les permitirá tener una visión clara de la seguridad en el área y tomar decisiones informadas.
- **Interacción con el Mapa:** Se mejoró la visualización de los incidentes en el mapa, permitiendo a los usuarios hacer clic en los reportes para obtener más detalles, tales como la descripción del incidente y la categoría.
- **Crear Comunidades:** Los usuarios ahora pueden formar comunidades dentro de la aplicación, lo que refuerza la cooperación y el apoyo mutuo en temas de seguridad.
- **Ver Ubicación de Amigos:** Además, los usuarios pueden visualizar la ubicación de sus amigos en el mapa, lo que incrementa la sensación de seguridad al poder estar al tanto de su ubicación en tiempo real.

Sprint 3: Reportes y Prueba Beta

El último sprint estuvo enfocado en la implementación de funcionalidades. También se llevó a cabo una prueba beta para evaluar la efectividad de estas funcionalidades y obtener retroalimentación. Las características clave fueron:

- **Acceso a Reportes:** Los primeros usuarios podían realizar reportes e interactuar con la aplicación.
- **Prueba Beta:** Los *tester* tuvieron la oportunidad de probar estas funcionalidades y evaluar su efectividad, lo que permitió verificar el uso su correcto de este y a su vez el funcionamiento.

Este sprint también incluyó un proceso de validación y retroalimentación para garantizar que las funcionalidades implementadas cumplieran con las expectativas de los *tester*.

Encuestas a Usuarios Beta:

Durante la prueba beta, se aplicó una encuesta a aproximadamente 15 personas que interactuaron con la aplicación en el contexto de la Feria de Software. La participación fue completamente voluntaria, por lo que no se exigió un número mínimo de respuestas. Los encuestados fueron en su mayoría jóvenes, aunque también participaron personas de distintas edades, tanto de sexo femenino como masculino. Esta instancia permitió recopilar percepciones directas de usuarios reales sobre el uso de la aplicación, aportando información relevante para el análisis de la experiencia de uso y la usabilidad del sistema. Los resultados destacaron que un 94.1% de los usuarios consideraron que las funcionalidades cubrían sus necesidades, y un 64.7% mencionó que la aplicación mejoró su percepción de seguridad.

Comentado [9]: Detalles

3.1.1 RESULTADOS DE LA ENCUESTA

Como parte del proceso de validación del proyecto, se aplicó una encuesta para evaluar la experiencia de usuario durante la fase beta de *Vision Security*. Esta acción tuvo como propósito verificar si las funcionalidades implementadas cumplían con los objetivos definidos y respondían a las necesidades de los usuarios. La encuesta, aplicada de manera online, incluyó preguntas orientadas a medir la facilidad de uso, la claridad del diseño, la disponibilidad de funcionalidades clave (como el botón de emergencia y el reporte de incidentes), así como la percepción de seguridad al utilizar la aplicación.

Los resultados obtenidos entregan información valiosa para entender cómo es percibido por el usuario el sistema.

Gráfico 1: Facilidad de Uso de la Aplicación

La mayoría de los usuarios calificaron la facilidad de uso de la aplicación como muy fácil (47,1%) o fácil (41,2%). Solo una pequeña proporción (5,9%) encontró la aplicación difícil de usar, lo que sugiere que la mayoría de los usuarios tiene una experiencia positiva con la

interfaz de la app.

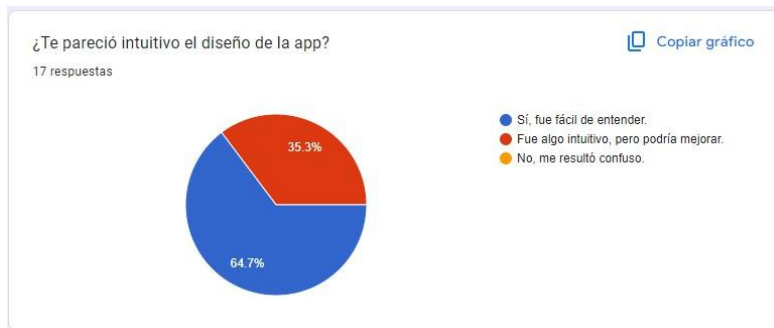


Figura 2 Pregunta 1

Fuente: Elaboración propia.

Gráfico 2: Intuición del Diseño

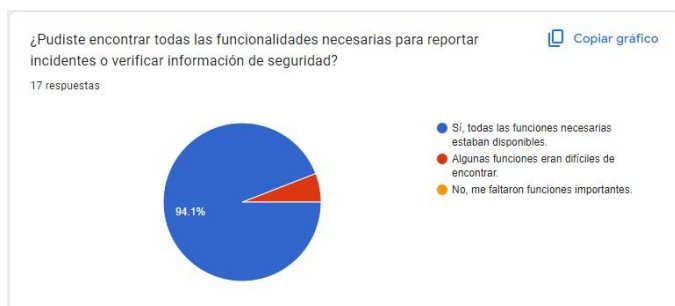
El 64,7% de los usuarios consideró que el diseño de la app fue fácil de entender, mientras que el 35,3% opinó que, aunque fue algo intuitivo, podría mejorar. Estos resultados indican que, aunque la app es generalmente comprensible, se pueden realizar ajustes para mejorar aún más la experiencia de usuario.

*Figura 3 Pregunta 2*

Fuente: Elaboración propia.

Gráfico 3: Disponibilidad de Funcionalidades

Un alto porcentaje de usuarios (94,1%) indicó que todas las funcionalidades necesarias estaban disponibles, lo que valida la implementación de las funcionalidades clave. Sin embargo, el 5,9% de los usuarios mencionó que algunas funciones fueron difíciles de encontrar, lo que sugiere que es necesario mejorar la accesibilidad o la organización de estas funcionalidades.

*Figura 4 Pregunta 3*

Fuente: Elaboración propia.

Gráfico 4: Impacto en la Sensación de Seguridad

El 64,7% de los usuarios expresó que la app les ayuda a sentirse más seguros, mientras que el 35,3% indicó que la aplicación podría mejorar en algunos aspectos para aumentar la sensación de seguridad. Esto resalta la efectividad general de la app en términos de proporcionar seguridad, pero también sugiere que algunas mejoras podrían hacerla aún más eficaz.

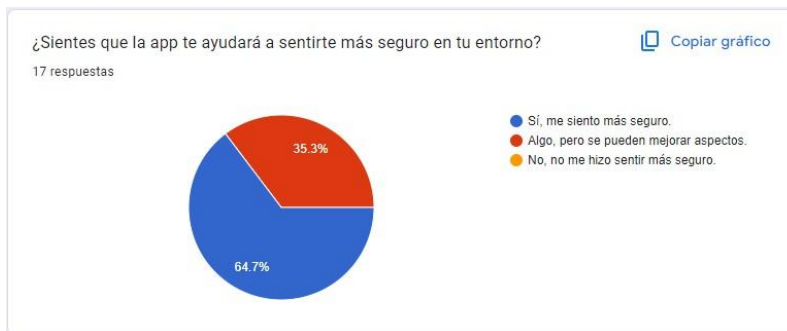


Figura 5 Pregunta 4

Fuente: Elaboración propia.

Gráfico 5: Funcionalidades Más Útiles

Los usuarios destacaron diferentes funcionalidades como las más útiles. El 23,5% mencionó la validación de reportes de incidentes, el 35,3% la creación de comunidades y la vista de ubicación de amigos, y otro 23,5% destacó el botón de emergencia. Estos resultados indican que la capacidad de interactuar con otros usuarios y tener acceso inmediato a la información de seguridad son características clave valoradas por los usuarios.



Figura 6 Pregunta 5

Fuente: Elaboración propia

3.2 APLICACIÓN DE LA METODOLOGÍA SCRUM EN EL PROYECTO

En este proyecto, se adoptó la metodología *Scrum* como marco de trabajo ágil para coordinar el desarrollo de la solución propuesta. Esta metodología se estructuró en tres *sprints* de un mes, donde se planificaron, desarrollaron y validaron funcionalidades clave de la aplicación. A continuación, se describe cómo se aplicó Scrum en cada etapa del desarrollo, considerando las tareas prioritizadas, la dinámica de trabajo y la entrega de artefactos funcionales.

Planificación temporal del proyecto

Como parte de la planificación y ejecución del proyecto, se utilizó una Carta Gantt, herramienta de gestión de proyecto que permitió organizar las tareas de desarrollo, distribuidas en tres *sprints*. Esta planificación visual facilitó el seguimiento del avance, la priorización de funcionalidades y la coordinación entre los integrantes del equipo.

La siguiente figura muestra la planificación por semanas, considerando actividades como la configuración inicial del proyecto, la implementación de funcionalidades, y la validación de los módulos desarrollados:

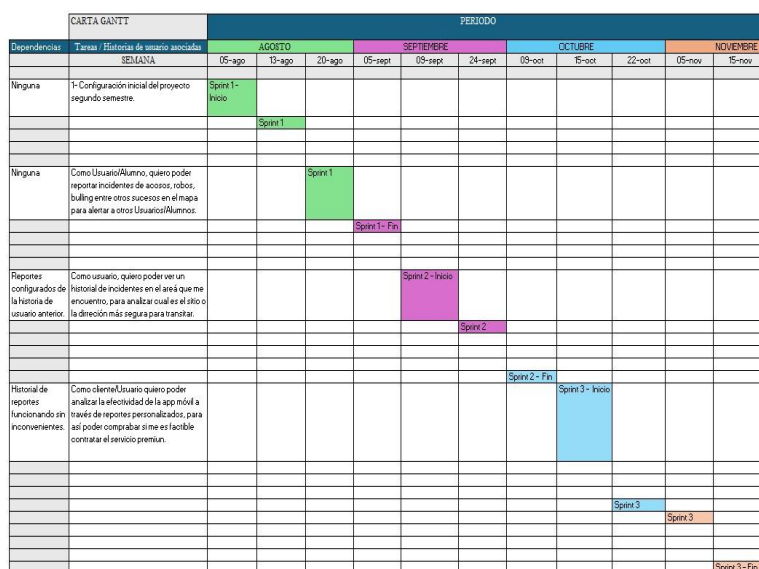


Figura 7 Carta Gantt

Fuente: Elaboración propia.

La planificación temporal del proyecto fue representada mediante una carta Gantt, la cual resume las tareas principales y su distribución en los tres sprints de desarrollo.

PLANIFICACIÓN DE LOS SPRINTS:

Cada sprint fue iniciado con una instancia de coordinación entre los integrantes del equipo, enmarcada en las actividades académicas vinculadas al proyecto. En estas sesiones de trabajo, se definieron los objetivos específicos de cada iteración, priorizando las historias de usuario conforme a los requerimientos funcionales establecidos.

Esta dinámica permitió mantener un enfoque ágil y flexible, asegurando que las funcionalidades desarrolladas en cada sprint respondieran a las necesidades identificadas y aportaran valor progresivo al producto.

PLANIFICACIÓN DE CADA SPRINTS						
Sprint	Duración	Historia de Usuario	Condiciones de Aceptación	Reuniones	Experimentos	Prioridad
Numero 1	1 mes	Como Usuario/Alumno, quiero poder reportar incidentes de acosos, robos, bullying entre otros sucesos en el mapa para alertar a otros Usuarios/Alumnos.	1- El usuario debe poder seleccionar la ubicación en El mapa. 2- El reporte debe incluir una descripción y categoría del incidente. 3- El reporte debe ser visible para otros usuarios en tiempo real.	1- Reunión de planificación del sprint. 2- Reunión diaria (Daily Standup). 3- Revisión del sprint. 4- Retrospectiva del sprint.	Validación de la funcionalidad de reporte de incidentes con un grupo de usuarios.	Alta
Numero 2	1 mes	Como usuario, quiero poder ver un historial de incidentes en el área que me encuentro, para analizar cual es el sitio o la dirección más segura para transitar.	1- El usuario inicia sesión y hace ingreso a la app móvil. 2- El usuario interactúa con el menú y el mapa de la app móvil. 3- El usuario debe poder acceder a un historial de incidentes filtrado por fecha y tipo de incidente. 4- El historial debe mostrar un mapa con los incidentes marcados.	1- Reunión de planificación del sprint. 2- Reunión diaria (Daily Standup). 3- Revisión del sprint. 4- Retrospectiva del sprint.	Validación del historial de incidentes con usuarios para asegurar su utilidad y usabilidad.	Alta
Numero 3	1 mes	Como cliente/Usuario quiero poder analizar la efectividad de la app móvil a través de reportes.	1- El cliente / usuario ingresa a una prueba beta a través de una cuenta creada por Vision Security. 2- El cliente / usuario puede visualizar el mapa y realizar reportes personalizados enfocados al área administrativa de su municipio. 3- Gracias a la prueba realizada el cliente / usuario puede comprobar la efectividad de los reportes.	1- Reunión de planificación del sprint. 2- Reunión diaria (Daily Standup). 3- Revisión del sprint. 4- Retrospectiva del sprint.	Validación de la funcionalidad de los reportes.	Alta

Figura 8 Planificación de Sprints

Fuente: Elaboración propia

Estimación de Costos por Sprint (Simulación):

Dado que *Vision Security* fue desarrollado en el marco de un proyecto académico, no se incurrió en costos reales de producción. Sin embargo, para fines ilustrativos y con el objetivo de demostrar la proyección de recursos necesaria en un entorno real, se presenta a continuación una simulación de costos estimados por *sprint*. Esta tabla contempla recursos humanos, infraestructura y herramientas comúnmente requeridas en un proyecto de desarrollo de software ágil.

Es importante señalar que el alojamiento en la nube se realizó utilizando una cuenta de *Amazon Web Services (AWS)* facilitada por el profesor guía, con vigencia solo durante el periodo académico, por lo que no generó costos asociados durante el desarrollo.

Categoría	Descripción	Costo estimado por Sprint	Total 3 sprint
Desarrollo Frontend	Implementación de interfaz y funcionalidades	250.000	750000
Desarrollo Backend	Lógica del sistema y conexión con base de datos	250.000	750000
Gestión del Proyecto	Planificación, seguimiento y priorización de tareas	150.000	450000
Diseño UX/UI	Prototipado y diseño visual	100.000	300000
Pruebas y validaciones	Evaluación de funcionalidades por sprint	100.000	300000
Infraestructura	Servicios como hosting o base de datos en la nube	100.000	300000
Total estimado			2700000

Figura 9 Costos estimados

Fuente: Elaboración propia

Desarrollo Frontend / Backend:

Basado en trabajo a media jornada (10–15 h/semana) durante 4 semanas por parte de desarrolladores.

Gestión del Proyecto (rol de coordinación y planificación):

Representa el tiempo dedicado a priorización de funcionalidades, planificación de entregas y seguimiento del equipo.

Diseño UX/UI:

Considera diseño de prototipos, pruebas con usuarios y ajustes visuales.

Pruebas y Validaciones:

Incluye revisión funcional y detección de errores.

Infraestructura (simulada):

Estimado para uso de servicios como AWS, base de datos remota (MySQL) y herramientas de despliegue.

Nota: En este proyecto académico se utilizó una cuenta AWS otorgada por el profesor, sin costos asociados.

Historias de Usuario Prioritarias:

Historia de Usuario 1: Reportar Incidentes en el Mapa

- **Descripción:** "Como usuario, quiero poder reportar incidentes como acosos, robos, *bullying*, accidentes, entre otros, en el mapa, para que otros usuarios puedan ser alertados y tomar medidas preventivas."
- **Objetivo:** Asegurarse de que la funcionalidad de reportar incidentes estuviera completamente operativa en el primer sprint. Esto incluyó la capacidad de seleccionar la ubicación del incidente en el mapa, proporcionar detalles completos del reporte y garantizar que la visibilidad en tiempo real fuera eficiente.
- **Prioridad:** Alta

Sprint Asignado: *Sprint 1*

Historia de Usuario 2: Acceder al Historial de Reportes

- **Descripción:** "Como usuario, quiero poder ver un historial de incidentes en el área donde me encuentro, para analizar la seguridad de diferentes ubicaciones y tomar decisiones informadas sobre por dónde transitar."
- **Objetivo:** Permitir que los usuarios pudieran acceder al historial de reportes. Esta funcionalidad fue para que los usuarios pudieran tener acceso a los diferentes tipos de reportes a su alrededor.
- **Prioridad:** Alta

Sprint Asignado: *Sprint 2*

Historia de Usuario 3: Crear Comunidades y Ver Ubicación de Amigos

- **Descripción:** "Como usuario, quiero poder crear comunidades y ver la ubicación de mis amigos en el mapa, para estar más tranquilo sabiendo que están seguros."
- **Objetivo:** Desarrollar una funcionalidad que permitiera a los usuarios formar comunidades dentro de la aplicación, lo que fortalecería la interacción social y la cooperación en la seguridad. Además, poder visualizar la ubicación de los amigos en el mapa fue clave para que los usuarios pudieran mantenerse conectados en situaciones de riesgo.
- **Prioridad:** Alta
- **Sprint Asignado:** *Sprint 2*

Historia de Usuario 4: Implementación del Botón de Emergencia

- **Descripción:** "Como usuario, quiero un botón de emergencia que me permita enviar una alerta a mis contactos registrados en caso de un incidente."
- **Objetivo:** Garantizar que los usuarios pudieran enviar alertas de emergencia rápidamente mediante un botón de fácil acceso en la interfaz. Esta funcionalidad se consideró esencial desde el inicio del proyecto y se priorizó para el primer sprint.
- **Prioridad:** Alta
- **Sprint Asignado:** *Sprint 1*

3.3 SOLUCIÓN TECNOLÓGICA

Para el desarrollo de esta aplicación, se diseñó una solución tecnológica integral que combina múltiples herramientas, lenguajes de programación y servicios en la nube, asegurando escalabilidad, seguridad y un rendimiento adecuado para los objetivos planteados.

1. Aplicación Móvil

Tecnología Utilizada: Se empleó *Flutter* como marco de desarrollo para crear la aplicación móvil. *Flutter* fue seleccionado por su capacidad para desarrollar aplicaciones multiplataforma, lo que permite que la app sea compatible tanto con Android como con *iOS* utilizando una sola base de código.

2. Servidor Backend

Lenguaje y Entorno: Se utilizó PHP para implementar la lógica del servidor. *PHP* fue elegido por su flexibilidad y su integración nativa con bases de datos *MySQL*.

3. Base de Datos

Tecnología Utilizada: Se utilizó una base de datos relacional *MySQL* alojada en *AWS RDS* (*Relational Database Service*). *MySQL* fue seleccionado por su rendimiento y su capacidad para manejar consultas complejas.

4. Infraestructura en la Nube**Servicios Utilizados:**

- **AWS EC2:** Se utilizó una instancia en la nube para alojar el servidor *backend*. Esto asegura disponibilidad y escalabilidad para manejar diferentes niveles de tráfico.
- **AWS RDS:** Servicio para alojar la base de datos, proporcionando un entorno optimizado para la gestión de datos relacionales.

5. Integración de Componentes

Comunicación Cliente-Servidor: La aplicación móvil interactúa con el servidor *backend* a través de solicitudes *HTTP* y *APIs RESTful*. Estas solicitudes son procesadas en el *backend*, donde se validan, se ejecutan las operaciones necesarias y se devuelven las respuestas correspondientes.

Sincronización: Los tres componentes (aplicación móvil, *backend* y base de datos) trabajan de manera sincronizada, garantizando que las operaciones realizadas por los usuarios se reflejen en tiempo real y sin inconsistencias.

6. Pruebas y Validaciones

Pruebas Funcionales: Las funcionalidades principales de la aplicación, como el registro de usuarios, el inicio de sesión y el envío de reportes, fueron evaluadas directamente en la versión funcional del sistema. Estas pruebas se realizaron utilizando la propia interfaz de la aplicación móvil durante su desarrollo.

Cada acción como registrar un nuevo usuario o generar un reporte fue ejecutada varias veces para verificar que la respuesta del sistema fuera la esperada. Posteriormente, se revisaron los datos almacenados en la base de datos mediante *MySQL Workbench*, confirmando que los registros se guardaran correctamente y sin errores.

Pruebas de Integración: Para asegurar el correcto funcionamiento del sistema, se realizaron pruebas de integración observando el comportamiento de la aplicación móvil al interactuar con el *backend* mediante solicitudes *HTTP*. Estas pruebas permitieron verificar que los datos enviados como el registro de usuarios o el reporte de incidentes fueran correctamente procesados y almacenados en la base de datos. La validación se complementó mediante la revisión de los registros en *MySQL Workbench* y la consola del entorno de desarrollo.

Comentado [10]: Cómo se hizo esto? Evidencia de las pruebas

Comentado [E811R10]: No tengo mas evidencias de las ya puestas en la memoria

Comentado [E812R10]:

Figura 1:

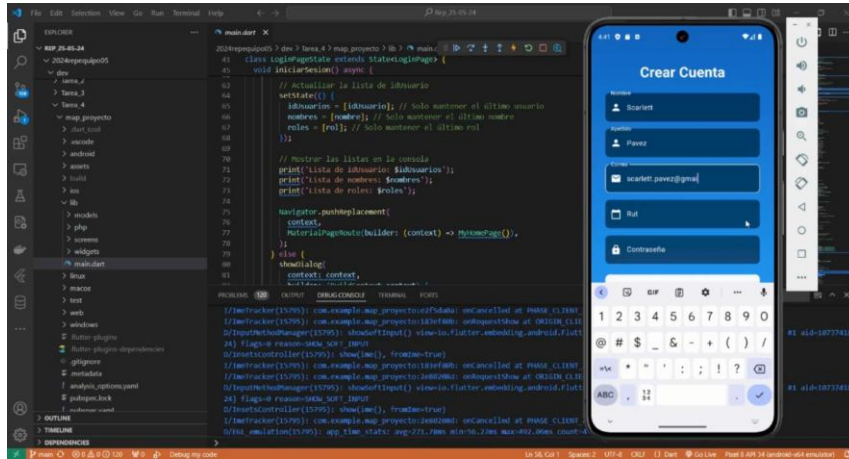


Figura 10 Figura 2

Fuente: Elaboración propia

La Figura 1 muestra parte del código en *Flutter* y una simulación en un emulador móvil *Android*, donde se validó la creación de usuarios a través del formulario de registro.

Figura 2:

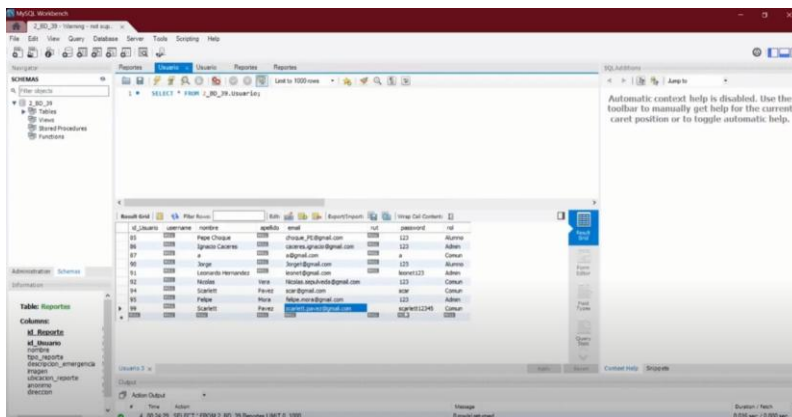


Figura 11 Figura 3

Fuente: Elaboración propia

La Figura 2 corresponde a una visualización en *MySQL Workbench*, donde se confirmó que los datos ingresados desde la aplicación móvil eran almacenados correctamente en la base de datos, lo cual evidencia que la conexión entre *frontend*, *backend* y base de datos se realizó sin errores.

3.4 ELEMENTOS TECNOLÓGICOS CLAVE

1. **Botón de Emergencia en la Aplicación Móvil**
 - Permite que los usuarios activen una alerta en caso de emergencia.
 - Envía un correo electrónico al contacto de emergencia registrado con los datos de ubicación en tiempo real.
2. **Mapa Interactivo de Incidentes**
 - Visualiza reportes de incidentes en tiempo real.
3. **Módulos Administrativos para Instituciones**
 - **Municipalidades:** Herramientas para visualizar reportes, generar estadísticas y tomar decisiones informadas sobre la asignación de recursos.
 - **Colegios:** Espacio seguro para gestionar casos de acoso y *bullying*, con acceso limitado al personal autorizado.
4. **Comunidades de Usuarios**
 - Posibilidad de crear grupos, como vecinos, compañeros de trabajo o familia, para compartir información relevante sobre la seguridad en tiempo real.
5. **Sistema de Geolocalización Avanzada**
 - Integración con servicios de mapas para obtener ubicaciones precisas.
 - Funcionalidad de rastreo en tiempo real, útil en casos de búsqueda de personas o monitoreo durante emergencias.
6. **Compatibilidad Multidispositivo**
 - Aplicación móvil disponible para sistemas Android.
 - Versión web optimizada para dispositivos de escritorio, tabletas y móviles.

CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN

La validación de la solución propuesta se realizó mediante una combinación de pruebas funcionales, evaluaciones de usabilidad y retroalimentación de usuarios finales durante la fase de prueba beta. Estas pruebas buscaban garantizar que la aplicación *Vision Security* respondiera efectivamente a las necesidades identificadas y demostrara su validez en el contexto de seguridad ciudadana de la región del Biobío.

Pruebas Funcionales:

Se evaluaron todas las funcionalidades clave, como el reporte de incidentes, el botón de emergencia y la generación de reportes como estudiante. Las pruebas se realizaron mediante interacción directa con la aplicación durante la fase de prueba beta, donde los usuarios ejecutaron estas funciones en distintos escenarios.

Durante estas evaluaciones, no se identificaron errores en la ejecución ni interrupciones en el funcionamiento, lo que permitió constatar que las funcionalidades operaban correctamente en condiciones normales de uso.

Creación de cuenta e inicio de sesión:

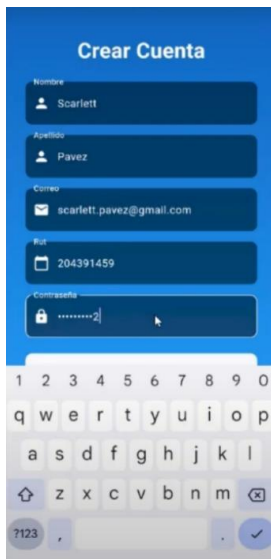


Figura 12 Creación de cuenta

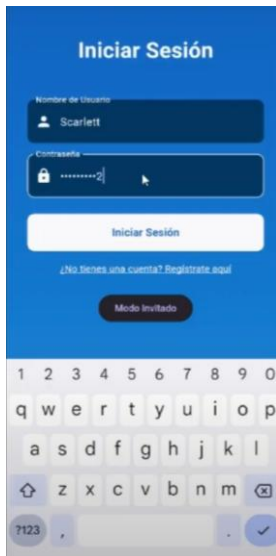


Figura 13 Inicio de sesión.

Comentado [13]: Cómo? Evidencias... pruebas de carga, estadísticas de uso,...

Sigo con esta duda...

Comentado [14]: Título e identificación de las imágenes.

Botón de emergencia:

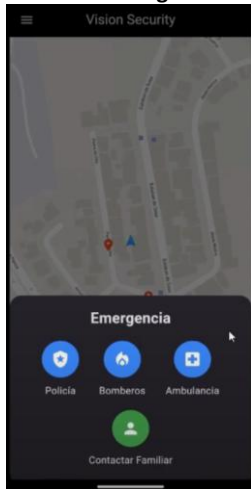


Figura 14 Botón de emergencia

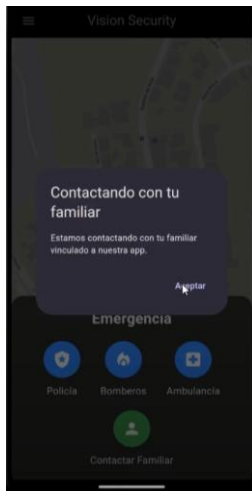


Figura 15 Contacto de emergencia

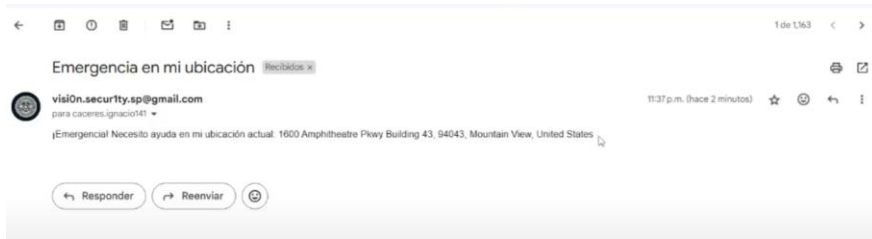


Figura 16 Correo de emergencia

Acceso denegado a los usuarios que no tengan cuenta de estudiante:

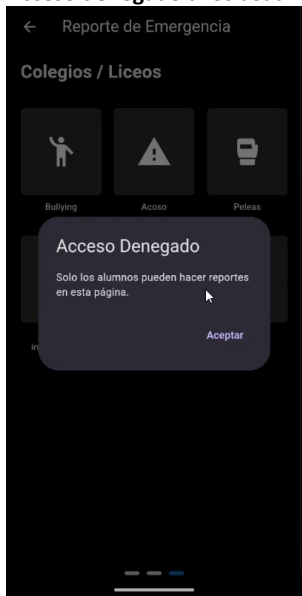


Figura 17 Acceso denegado.

Navegación de menú y creación de reporte:

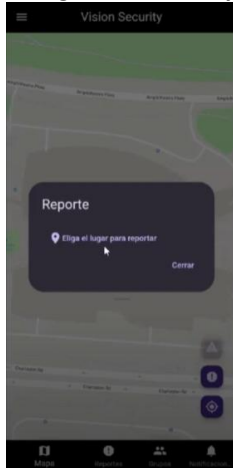


Figura 18 Crear reporte

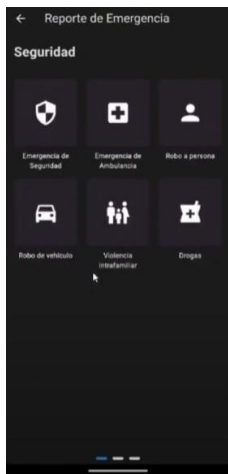


Figura 19 Menú 1

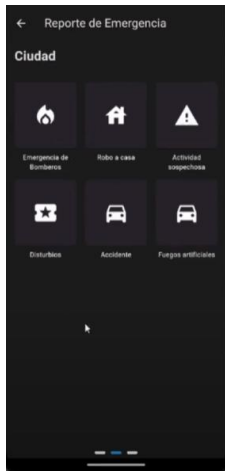


Figura 20 Menú 2

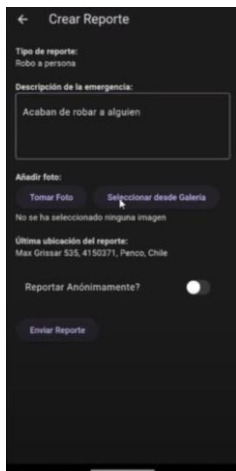


Figura 21 Descripción de reporte

Reportes:



Figura 22 Reportes

Comentado [15]: No se ve nada

CAPÍTULO 5: CONCLUSIONES

La implementación de *Vision Security* demostró ser una solución válida para centralizar y mejorar la gestión de incidentes de seguridad en la región del Biobío. A continuación, se presentan las principales conclusiones:

Alcances y Resultados:

El proyecto logró cumplir con varios de los objetivos definidos al inicio, entre ellos:

Implementación de un sistema funcional para el reporte de incidentes georreferenciados:

Durante el Sprint 1 se desarrolló la funcionalidad que permite a los usuarios registrar incidentes directamente en un mapa interactivo, seleccionando manualmente la ubicación exacta. Cada reporte incluye una categoría robo, acoso, accidente, entre otros y una descripción del hecho, que queda visible en la aplicación.

Incorporación de un botón de emergencia vinculado a contactos de confianza:

En el mismo sprint, se integró un botón de emergencia en la interfaz móvil, el cual, al ser presionado, envía automáticamente un correo electrónico al contacto registrado por el usuario, incluyendo su ubicación actual.

Desarrollo de un historial accesible de reportes anteriores:

En el Sprint 2 se trabajó la funcionalidad del historial, que permite a los usuarios visualizar eventos pasados registrados en su zona o áreas de interés.

Estas funcionalidades fueron priorizadas durante la planificación de los *sprints* y validadas en las fases de revisión durante las presentaciones en clases. Además, se obtuvieron comentarios de usuarios durante la prueba beta, realizada en la feria de software, lo que permitió ajustar detalles menores y confirmar su utilidad práctica.

Limitaciones Identificadas:

Aunque la prueba beta evidenció un alto nivel de aceptación, algunos usuarios reportaron dificultades menores para encontrar ciertas funcionalidades, lo que sugiere oportunidades de mejora en el diseño de la interfaz.

Además, debido a que la validación se realizó en un entorno controlado y con una muestra limitada de usuarios, no fue posible evaluar completamente el comportamiento de la aplicación en escenarios reales de alta demanda o con una comunidad extensa.

Aportes del Trabajo:

A través del trabajo iterativo, se logró dar forma a una solución tecnológica que permite centralizar reportes de incidentes y facilitar la comunicación en situaciones de riesgo, integrando

Comentado [16]: Reescribir... las conclusiones van en función de los objetivos y sus resultados.

funcionalidades como el botón de emergencia, el mapa interactivo de incidentes.

Uno de los principales aportes fue el diseño y planificación de funcionalidades orientadas a distintos perfiles de usuarios ciudadanos, colegios y municipalidades priorizando sus necesidades y traduciendo esas prioridades en entregas funcionales durante cada sprint. El proceso de planificación, definición de historias de usuario y seguimiento constante permitió avanzar hacia un producto funcional que, aunque aún en versión inicial, establece una base para continuar desarrollando herramientas de seguridad comunitaria.

REFERENCIAS BIBLIOGRÁFICAS

- Gobierno de Chile, Subsecretaría de Prevención del Delito. (2024). Por primera vez, ENUSC entrega cifras de victimización y percepción a nivel comunal. <https://www.seguridadpublica.cl/noticia/por-primera-vez-enusc-entrega-cifras-de-victimizacion-y-percepcion-a-nivel-comunal>
- Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. <https://scrumguides.org/scrum-guide.html>
- Oracle. (2023). MySQL Workbench. <https://www.mysql.com/products/workbench/>
- Amazon Web Services. (2023). What is AWS?. <https://aws.amazon.com/what-is-aws/>
- Mozilla Developer Network. (2024). Maps Platform Documentation. <https://developer.mozilla.org/>
- Google Developers. (2023). Flutter: Build apps for any screen. <https://flutter.dev/>
- Google Maps API Google Developers. (2024). Maps Platform Documentation. <https://developers.google.com/maps>
- Gobierno de Chile, Subsecretaría de Prevención del Delito. (2024). *Encuesta Nacional Urbana de Seguridad Ciudadana (ENUSC)*. <https://www.seguridadpublica.cl>
- Biblioteca del Congreso Nacional de Chile. (2023). *Evolución del delito en Chile. Periodo 2019 - 2023* (Informe de Fiscalización). https://obtienearchivo.bcn.cl/obtienearchivo?id=repositorio/10221/35843/2/BCN_GF_Evolucion_del_delito_en_Chile.pdf
- Preece, J., Rogers, Y., & Sharp, H. (2015). Interaction design: Beyond human-computer interaction (4th ed.). John Wiley & Sons.
- Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*. Scrum.org. Recuperado de <https://scrumguides.org/scrum-guide.html>

Comentado [17]: Revisar referencias. muchas de ellas no apuntan al contenido citado. Además no están presentes en el texto. Si están aquí, es que se usaron en el documento. Usa APA.

Pudieras buscar documentos científicos.