UNIVERSIDAD ESTATAL DE BOLÍVAR

FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN EMPRESARIAL E INFORMÁTICA

CARRERA DE SOFTWARE

TRABAJO DE INTEGRACIÓN CURRICULAR
PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO(A)(S) EN SOFTWARE

FORMA: PROYECTO TECNOLÓGICO

TEMA:

APLICACIÓN MÓVIL DE SEGURIDAD INTEGRAL PARA LA UNIDAD DE GESTIÓN DE RIESGOS EN LA UNIVERSIDAD ESTATAL DE BOLÍVAR

AUTOR(A):

PABLO DAVID HOLGUIN RIOS
NATALY DOMENICA SILVA VILLAGRAN

DIRECTOR(A):

ING. MARICELA ESPÍN

GUARANDA – ECUADOR

2026

TEMA DEL PROYECTO TECNOLÓGICO

APLICACIÓN MÓVIL DE SEGURIDAD INTEGRAL PARA LA UNIDAD DE GESTIÓN DE RIESGOS EN LA UNIVERSIDAD ESTATAL DE BOLÍVAR

DEDICATORIA

Dedico este proyecto, en primer lugar, a Dios por ser mi luz y mi guía inquebrantable a lo largo de este arduo camino, por otorgarme la sabiduría para aprender, la paciencia para resolver cada problema, la fortaleza y perseverancia necesaria para no rendirme en los momentos de mayor difícil y poder lograr superar cualquier obstáculo y culminar esta etapa tan importante en mi vida.

A mis padres, Juan Pablo Holguín Sánchez y Lorena Elizabet Ríos Agila los verdaderos forjadores de mi carácter, quienes, con su amor incondicional, sacrificio incansable y constante apoyo han sido el pilar fundamental en mi formación personal como profesional. Reconozco que detrás de cada logro académico que he obtenido están sus manos trabajadoras y sus corazones llenos de esperanza Gracias por ser mi mayor inspiración, por creer en mí y brindarme las herramientas para alcanzar este sueño, este logro es tanto suyo como mío,

A mi compañera de vida y futura esposa Katia Selena por ser mi mayor apoyo y el ancla que me sostuvo en este proceso. Gracias por iluminar mis días más pesados, por tu infinita comprensión en mis horas de ausencia y por sostener mi mano durante todo este trayecto tu fe ciega en mi capacidades han sido la pieza clave para completar este gran objetivo. Este logro lleva también tu nombre, porque sin tu aliento constante, el camino habría sido mucho más difícil. Te dedicó este triunfo con la promesa de seguir alcanzando juntos todos los sueños que nos depara el futuro.

Pablo David Holguín Ríos

Dedico este trabajo primordialmente a Dios, por ser mi guía, mi pilar fundamental y la fuente inagotable de mi inspiración. Pongo en sus manos este esfuerzo, agradeciendo la salud, la vida y la fortaleza que me brindo para transformar cada obstáculo en perseverancia y culminar con éxito esta etapa académica,

De manera especial, dedico este logro a mis padres, Medardo Ángel Silva y Elvia Roció Villagrán. Gracias por su amor incondicional y por se el refugio donde siempre halle consuelo, incluso en mis momentos de distancia o frustración. Su sacrificio y motivación constante fueron el motor que me permito levantarme ante el cansancio y lugar por mis sueños; gracias a su ejemplo, soy la persona que soy hoy.

Finalmente, dedico este triunfo a mis hermanos Susana y Álvaro, por su apoyo emocional y por ser mi aliento en lo días más difíciles. Este título no es solo mío, sino una muestra del amor y el esfuerzo que, como familia, hemos compartido y construido juntos.

Nataly Doménica Silva Villagrán

AGRADECIMIENTO

Expreso mi más sincero y profundo agradecimiento a la Universidad Estatal de Bolívar y de manera muy especial a la carrera de Software por haberme abierto sus puertas y brindarme una formación académica excelente durante estos años.

Un agradecimiento profundo a nuestra directora de proyecto, Ing. Maricela Espín, por su invaluable guía, paciencia, conocimientos y constante orientación a lo largo de este trabajo de integración curricular. Su dedicación y profesionalismo fueron piezas clave para el éxito de este proyecto.

A la Unidad de Gestión de Riesgos de la Universidad, por la apertura, la disposición de tiempo y la confianza depositada en nosotros para implementar esta solución tecnológica en beneficio y protección de toda nuestra comunidad universitaria.

De manera muy personal y llena de amor, quiero agradecer a mi padres y a mis hermanos. A mis padres, por el esfuerzo incansable, su respaldo incondicional y por darme el privilegio de educarme y alcanzar este gran sueño. A mis hermanos, por su compañía constante, su aliento en mis mi momentos de mayor estrés y por se mi apoyo incondicional en cada paso de esta etapa. Este triunfo también es de ustedes.

Pablo David Holguín Ríos

Con profunda gratitud reconozco la presencia constante de Dios Todopoderoso a lo largo de este camino. Su amor y bondad infinita me sostuvieron en cada etapa, especialmente en los momentos de duda, cansancio y dificultad. Cada experiencia vivida, tanto los logros como los desafíos, se transformó en una enseñanza que fortaleció mi fe, forjo mi carácter y amplio mi forma de comprender la vida. Su guía me dio la fuerza necesaria para seguir adelante cuando sentí que no podía más, recordándome que toda dificultad trae consigo una oportunidad para creer, aprender y volver a creer en mí.

Agradezco sinceramente la salud, la sabiduría y la perseverancia recibidas, dones que me permitieron culminar una etapa fundamental de mi vida y alcanzar una meta que hoy representa no solo un logro académico, sino también personal.

A mi alma mater, la Universidad Estatal de Bolívar, expreso mi más profundo agradecimiento, en especial a los docentes de la carrera de Software, quienes con vocación, ética y compromiso compartieron sus conocimientos y me guiaron con paciencia y profesionalismo, dejando una huella significativa en mi formación académica y humana.

Mi eterno agradecimiento a mis padres, quienes han sido el pilar más fuerte de mi vida. Gracias por su amor incondicional, por su apoyo emocional y económico, por su confianza inquebrantable y caminar conmigo en cada paso de este proceso, incluso cuando el camino parecía difícil.

A mis hermanos, gracias por su apoyo constate, su comprensión y sus palabras de aliento, que fueron un refugio en los momentos más exigentes. De igual manera, agradezco con profundo cariño a mi gato, quien, aunque ya no se encuentra conmigo, fue una compañía silenciosa pero invaluable. Su presencia me brindo calma, consuelo y fortaleza durante las largas jornadas de esfuerzo y desvelo. Fue un amor sincero e incondicional, un ser de cuatro patas que dejo una huella imborrable en mi corazón y cuyo recuerdo me acompañara siempre.

Finalmente, agradezco de corazón a todas las personas que, de manera directa o indirecta, formaron parte de este proceso y constituyeron con su apoyo, confianza y palabras de ánimo para hacer posible la culminación de esta etapa tan significativa de mi vida.

Nataly Doménica Silva Villagrán

CERTIFICADO DE VALIDACIÓN

Ing. Maricela Espín, Ing. Henry Vallejo e Ing. Paúl Sánchez, en su orden Director y Pares Académicos del Trabajo de Integración Curricular "APLICACIÓN MÓVIL DE SEGURIDAD INTEGRAL PARA LA UNIDAD DE GESTIÓN DE RIESGOS EN LA UNIVERSIDAD ESTATAL DE BOLÍVAR" desarrollado por el Sr. Pablo David Holguin Rios y la Srta. Nataly Domenica Silva Villagran

CERTIFICAN

Que, luego de revisado el Trabajo de Integración Curricular en su totalidad, cumple con las exigencias académicas de la carrera SOFTWARE, por lo tanto, autorizamos su presentación y defensa.

Guaranda, 27 de febrero del 2026

Firmado electrónicamente por: MARICELA ARACELI ESPIN MOREJON

Ing. Maricela Espín.
Directora

Firmado electrónicamente por: HENRY FERNANDO VALLEJO BALLESTEROS

Ing. Henry Vallejo
Par Académico

Firmado electrónicamente por: PAUL OSWALDO SANCHEZ FRANCO

Ing. Paúl Sánchez
Par Académico

DERECHOS DE AUTOR

Nosotros **HOLGUIN RIOS PABLO DAVID** y **SILVA VILLAGRAN NATALY DOMENICA** portadores de la Cédula de Identidad No **1752504926** y **1805666797** en calidad de autores y titulares de los derechos morales y patrimoniales del Trabajo de Titulación: **APLICACIÓN MÓVIL DE SEGURIDAD INTEGRAL PARA LA UNIDAD DE GESTIÓN DE RIESGOS EN LA UNIVERSIDAD ESTATAL DE BOLÍVAR**, modalidad **PROYECTO TECNOLÒGICO**, de conformidad con el Art. 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN, concedemos a favor de la Universidad Estatal de Bolívar, una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservamos a mi/nuestro favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizo/autorizamos a la Universidad Estatal de Bolívar, para que realice la digitalización y publicación de este trabajo de titulación en el Repositorio Digital, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Los autores declaran que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.

Holguin Rios Pablo David

Silva Villagran Nataly Domenica

Escaneado con CamScanner

INDICE DE CONTENIDOS

TEMA DEL PROYECTO TECNOLÓGICO i
DEDICATORIA ii
AGRADECIMIENTO iv
CERTIFICADO DE VALIDACIÓN vi
DERECHOS DE AUTOR (Acorde al modelo emitido por biblioteca general) vii
INDICE DE CONTENIDOS viii
INDICE DE TABLAS xiii
INDICE DE FIGURAS xiv
INTRODUCCIÓN 1
RESUMEN 2
ABSTRACT 3
PISI RIMAY 4
CAPÍTULO I 5
FORMULACIÓN GENERAL DEL PROYECTO 5
1.1. Tema 5
1.2. Descripción del Problema 5
1.3. Justificación 6
1.4. Objetivos 7
1.4.1. Objetivo General 7
1.4.2. Objetivos Específicos 7
CAPÍTULO II 8
MARCO TEÓRICO 8
2.1. Antecedentes 8
2.2. Científico 9

2.2.1. Metodología de Desarrollo de Software 9
2.2.2. Metodologías Agiles 9
2.2.3. Diferencias Metodologías Agiles y Metodologías Tradicionales 9
2.2.4. Metodología Scrum 10
2.2.5. Organización o Roles de Scrum 10
2.2.6. Flujo o fases de un proyecto Scrum 11
2.3. Conceptual 13
2.4. Legal 16
2.5. Georeferencial 17
Metodología de Desarrollo de Software 18
3.1. Metodología de Desarrollo de Software 18
Técnicas e Instrumentos de Recopilación de Datos 19
3.1.1. Entrevista 19
3.1.2. Observación 19
CAPÍTULO IV 20
INGENIERÍA DEL PROYECTO 20
4.1. Análisis y alcance 20
4.1.1. Identificación de los Stakeholders 20
4.1.2. Equipo Scrum 21
4.1.3. Creación del Product Backlog 21
4.1.4. Especificación de Requerimientos 21
4.1.5. Requerimientos No Funcionales 28
4.1.6. Características de Usuarios 30
4.1.7. Alcance del Producto 31
4.1.8. Límites del Sistema 32
SPRINT 1: DISEÑO Y FUNDAMENTOS DEL SISTEMA 32
4.2. Planificación del Sprint (Sprint Planning) 32

4.2.1. Objetivo del Sprint 33
4.2.2. Actividades Principales 33
4.2.3. Sprint Backlog 33
4.2.4. Desarrollo diario (Daily Scrum) 34
4.2.5. Revisión del Sprint (Sprint Review) 34
4.2.6. Retrospectiva del Sprint 35
4.2.7. Resultados del Sprint 1 36
4.3. Diagramas Vistas Arquitectónicas 4+1 40
4.3.1. Vista de escenarios 41
4.3.2. Vista de desarrollo 45
4.3.3. Vista Lógica 46
4.3.4. Vista de Procesos 47
4.3.5. Vista Física 49
4.4. Diagramas base de datos 49
4.4.1. Modelo Entidad Relación 49
4.4.2. Modelo Relacional 50
SPRINT 2 DESARROLLO DEL SISTEMA ADMINISTRATIVO 51
4.5. Planificación Sprint 51
4.5.1. Objetivo del Sprint 51
4.5.2. Actividades Principales 51
4.5.3. Sprint Backlog 52
4.5.4. Desarrollo Diario (Daily Scrum) 53
4.5.5. Revisión del Sprint (Sprint Review) 53
4.5.6. Retrospectiva del Sprint (Sprint Retrospective) 55
4.5.7. Resultados del Sprint 2 56
SPRINT 3 DESARROLLO DEL APLICATIVO MOVIL 68
4.6. Planificación del Sprint 68

4.6.1. Objetivo del Sprint 68
4.6.2. Actividades Principales 69
4.6.3. Sprint Backlog 70
4.6.4. Desarrollo diario (Daily Scrum) 71
4.6.5. Revisión del Sprint (Sprint Review) 71
4.6.6. Retrospectiva del Sprint (Sprint Retrospective) 72
4.6.7. Resultados del Sprint 3 73
SPRINT 4 IMPLEMENTACIÒN, PRUEBAS Y ENTREGAS 82
4.7. Planificación Sprint 82
4.7.1. Objetivo del Sprint 82
4.7.2. Actividades Principales 82
4.7.3. Sprint Backlog 83
4.7.4. Desarrollo diario (Daily Scrum) 84
4.7.5. Revisión del Sprint (Sprint Review) 84
4.7.6. Retrospectiva del Sprint (Sprint Retrospective) 84
4.7.7. Resultados del Sprint 4 84
4.8. Pruebas 84
4.8.1. Pruebas de Rendimiento 84
4.8.2. Pruebas de Usabilidad 85
4.8.3. Prueba de Funcionalidad 86
4.8.4. Pruebas de Seguridad 87
CONCLUSIONES 89
RECOMENDACIONES 90
BIBLIOGRAFÍA 91
ANEXOS 94
Anexo 1 94
Anexo 2 94

Anexo 3 95

Anexo 5 97

Anexo 6 138

INDICE DE TABLAS

Tabla 1 Diferencias metodologías ágiles y metodologías tradicionales 9
Tabla 2 Involucrados 20
Tabla 3 Autentificación de usuarios 22
Tabla 4 Gestión de usuarios 22
Tabla 5 Visualización de estadísticas de reportes 23
Tabla 6 Atención y respuesta a incidentes y alertas de pánico 23
Tabla 7 Gestión de contenidos 24
Tabla 8 Configuración y preferencias de perfil 25
Tabla 9 Autentificación de usuario 25
Tabla 10 Creación de reporte 26
Tabla 11 Alerta de pánico 27
Tabla 12 Acceso a la información 27
Tabla 13 *Usuarios y acciones* 30
Tabla 14 Tareas Sprint 1 33
Tabla 15 Ejecución Sprint 1 34
Tabla 16 Historia de usuario seleccionadas 52
Tabla 17 Ejecución del Sprint 2 54
Tabla 18 Historia de Usuario Móvil 70
Tabla 19 Ejecución del Sprint 3 71
Tabla 20 Backlog Sprint Cierre 83
Tabla 21 Pruebas de Rendimiento 85
Tabla 22 Pruebas de Usabilidad 86
Tabla 23 Pruebas de Funcionalidad 86
Tabla 24 Prueba de Funcionalidad Sprint 3 87
Tabla 25 Pruebas de Seguridad 88

INDICE DE FIGURAS

Figura 1 Mapa Satelital de la Universidad Estatal de Bolívar..... 17
Figura 2 Interfaz de Inicio de Sesión..... 36
Figura 3 Panel Administrativo 37
Figura 4 Interfaz Móvil Login..... 38
Figura 5 Interfaz Móvil Registrarse - Crear Cuenta..... 38
Figura 6 Interfaz Móvil Principal..... 39
Figura 7 Diagramas de Arquitectura de Alto Nivel..... 40
Figura 8 Diagrama Caso de Uso General..... 41
Figura 9 Diagrama Caso de Uso Director 42
Figura 10 Diagrama Caso de Uso Administrador 43
Figura 11 Diagrama Caso de Uso Comunidad Universitaria 44
Figura 12 Vista de Desarrollo..... 45
Figura 13 Vista Lógica..... 46
Figura 14 Vista de Procesos Panel Administrativo..... 47
Figura 15 Vista de Procesos App Móvil 48
Figura 16 Vista Física..... 49
Figura 17 Modelo Entidad Relación 50
Figura 18 Modelo Relacional..... 51
Figura 19 Administración de Usuarios..... 56
Figura 20 Gestión de Usuarios 57
Figura 21 Crear Usuario..... 57
Figura 22 Editar Usuario..... 58
Figura 23 Acción Eliminar Usuario 58
Figura 24 Editar usuario..... 59
Figura 25 Estadísticas y Reportes..... 59
Figura 26 Gestión de Incidentes..... 60
Figura 27 Cambiar Estado Gestión de Incidentes 60
Figura 28 Detalle de Incidente 61
Figura 29 Ubicación del Incidente 61
Figura 30 Monitoreo de Pánico..... 62
Figura 31 Atender Emergencia..... 62
Figura 32 Ubicación de Emergencia de Pánico..... 63

Figura 33 Gestión de Contenidos 63
Figura 34 Sección de Publicaciones 64
Figura 35 Sección de Publicaciones Editar 64
Figura 36 Sección de Publicaciones Eliminar 65
Figura 37 Sección Puntos de Encuentro y Zonas de Peligro 65
Figura 38 Editar Puntos de Encuentro y Zonas de Peligro 66
Figura 39 Eliminar Puntos de Encuentro y Zonas de Peligro 66
Figura 40 Configuración y Personalización de Cuenta 67
Figura 41 Actualización de Datos 67
Figura 42 Configuración Apariencia de Sistema 68
Figura 43 Ventana para Cerrar Sesión 68
Figura 44 Modulo de Alerta de Pánico 74
Figura 45 Permisos de Ubicación 74
Figura 46 Notificación de Confirmación de Ayuda 75
Figura 47 Confirmación de Ayuda en Camino 75
Figura 48 Modulo de Reporte de Incidente 76
Figura 49 Botón Activar Cámara 76
Figura 50 Estado del Reporte 77
Figura 51 Detalle del Reporte 77
Figura 52 Protocolos de Seguridad 78
Figura 53 Detalle del Protocolo de Seguridad 78
Figura 54 OpenStreetMap Zonas de Peligro y Puntos de Encuentro 79
Figura 55 Configuración General 80
Figura 56 Visualización de Perfil 80
Figura 57 Historial de mis Reportes 81
Figura 58 Historial de Alertas 81
Figura 59 Notificaciones y Ajustes 81
Figura 60 Ayuda y Soporte 82

INTRODUCCIÓN

La seguridad y la gestión oportuna de riesgos son pilares fundamentales para garantizar el bienestar físico y emocional dentro de cualquier institución de educación superior. En la Universidad Estatal de Bolívar, la gestión de emergencias ha presentado limitaciones considerables al depender de reportes verbales y canales informales. Esta ausencia de un sistema oficial digitalizado dificulta la notificación, control y seguimiento de eventualidades como emergencias médicas, problemas de infraestructura, incendios o actos delictivos, lo que incrementa la vulnerabilidad de estudiantes, docentes y personal administrativo.

Ante este escenario, surge la necesidad de modernizar y digitalizar diversos protocolos por esa razón el desarrollo de la Aplicación Móvil de Seguridad Integral para la Unidad de Gestión de Riesgos en la Universidad Estatal de Bolívar. El propósito central es proporcionar a la Unidad de Gestión de Riesgos un canal oficial, ágil y confiable que permite a la comunidad universitaria emitir alertas de pánico inmediatas, reportar incidentes con precisión geográfica y mantenerse informada sobre protocolos de prevención y puntos de encuentro. De tal forma el trabajo tecnológico de integración curricular se ha estructurado de la siguiente manera:

CAPÌTULO I: Aborda de diferentes puntos principales para la elaboración del proyecto detallando la descripción del problema, la justificación de la solución propuesta y los objetivos generales y específicos que guía el trabajo.

CAPÌTULO II: Se expone el marco teórico, abarcando los antecedentes de proyectos similares y las bases conceptuales, tecnológicas y legales que sustentan la arquitectura del software.

CAPÌTULO III: Describe la metodología a utilizar detallando el uso del marco de trabajo ágil SCRUM y las técnicas de recopilación de datos.

CAPÌTULO IV: Presenta la ingeniería del proyecto en su totalidad. Incluye el análisis de requerimientos, la definición arquitectónica (modelo 4+1), el diseño de la base de datos y la ejecución secuencias de los cuatro Sprints, los cuales culminan con la implementación, pruebas y entrega del sistema operativo.

En definitiva, el sistema web y aplicativo móvil cumple con los objetivos planteados y fortalece a las necesidades en la seguridad integral. Se recomienda en adaptar la plataforma frente a futuros cambios en los protocolos de actuación o en la infraestructura del campus.

RESUMEN

La Unidad de Gestión de Riesgos de la Universidad Estatal de Bolívar, enfrentaba la falta de un sistema oficial digitalizado para el reporte y seguimiento de incidentes dentro de la institución; lo cual generaba retrasos en la atención, perdida de información y una débil cultura de prevención. Para resolver esta problemática, se construyó una plataforma tecnológica bajo un arquitectura cliente-servidor desacoplada.

El desarrollo se rigió por la metodología ágil Scrum, estructurando el trabajo en iteraciones o Sprints. A nivel tecnológico, el sistema administrativo web se construyó con la librería React, mientras que el backend y la base de datos se implementaron usando el framework Laravel (API REST) y PostgreSQL.

La aplicación móvil orientada a la comunidad universitaria, fue desarrollada de forma nativa para el sistema operativo Android utilizando el lenguaje Kotlin y el patrón arquitectónico MVVM. Entre sus funcionalidades principales destacan la activación de una alerta de pánico con geolocalización inmediata, el reporte de incidentes con evidencia fotográfica y la consulta de protocolos de seguridad interactivos integrados con OpenStreetMap.

Como resultado, el sistema logro centralizar exitosamente el monitoreo de emergencias, optimiza la coordinación de respuestas ante alertas institucionales y dota a la comunidad universitaria de una herramienta tecnológica robusta, escalable e intuitiva.

Palabras clave: Laravel, Scrum, Android, Seguridad

ABSTRACT

This Risk Management Unit of the Universidad Estatal de Bolivar faced a lack of an official digitized system for reporting and tracking incidents within the institution, which caused delays in response times, information loss, and a weak culture of prevention. To address this issue, a technological platform was developed based on a decoupled client-server architecture. The development process followed the Scrum agile methodology, structuring the workflow into iterations or Sprints. Technologically, the web administrative system was built using the React library, while the backend and database were implemented using the Laravel framework (RESTful API) and PostgreSQL.

The mobile application, aimed at the university community, was developed natively for the Android operating system using Kotlin and the MVVM architectural pattern. Its main features include the activation of a panic alert with real-time geolocation, incident reporting with photographic evidence, and access to interactive safety protocols integrated with OpenStreetMap.

As a result, the system successfully centralized emergency monitoring, optimized the coordination of responses to institutional alerts, and provided the university community with a robust, scalable, and intuitive technological tool.

Keywords: Laravel, Scrum, Android, Security.

PISI RIMAY

Universidad Estatal de Bolívar hatun yachana wasimanta Llakikunata Rikuk Ukuka (Unidad de Gestión de Riesgos), mana shuk antanikik llikata (sistema oficial digitalizado) charirkachu llakikunata willankapak, chaykunata rikunkapakpash; chaimantami yanapankapak unayarka, willaykuna chinkarirka, shinallatak mana llakikunata harkankapak yachay tiyarkachu.

Kay llakita allichinkapakmi, shuk mushuk tecnología llikata rurashka cliente-servidor arquitectura nishkapi. Kay rurayka Scrum metodología ágil nishkawanmi purirka, llamkayta Sprints nishkapi rakishpa.

Tecnología nishkapika, web llika kamachikkataka React librería-wanmi rurashka, kutin backend shinallatak base de datos nishkataka Laravel (API REST) shinallatak PostgreSQL ukupimi rurashka.

Hatun yachana wasipi kawsakkunapakmi shuk aplicación móvil rurashka, Android sistema operativo-pak nativa nishkapi, Kotlin shimipi shinallatak MVVM patrón arquitectónico nishkapi rurashka. Paypak ashtawan mutsurishka ruraykunapika kaykunami tiyan: shuk manchaypi utka willankapak (alerta de pánico) maypi kashkata utka rikuchishpa (geolocalización inmediata), llakikunata willankapak fotokunawan, shinallatak ima rurana kamachiykunata rikunkapak OpenStreetMap llikawan.

Tukupika, kay llikaka tukuy llakikunata shukllapi rikunkapakmi yanapan, utka yanapayta ruran llakikuna tiyakpi, shinallatak hatun yachana wasipi kawsakkunaman shuk sinchi, wiñarik (escalable), alli intindina herramienta tecnológica nishkatami kun.

Sapi shimikuna: Laravel, Scrum, Android, Harkay.

CAPÍTULO I

FORMULACIÓN GENERAL DEL PROYECTO

1.1. Tema

APLICACIÓN MÓVIL DE SEGURIDAD INTEGRAL PARA LA UNIDAD DE GESTIÓN DE RIESGOS EN LA UNIVERSIDAD ESTATAL DE BOLÍVAR.

1.2. Descripción del Problema

En la Universidad Estatal de Bolívar, la gestión de riesgos y la seguridad institucional presenta limitaciones considerables, esto se debe, principalmente, a que no disponen de un sistema oficial digitalizado en el cual puedan notificar, controlar y dar seguimiento a los incidentes en diferentes circunstancias, tanto los eventos relacionados a la seguridad, las emergencias médicas, infraestructura, acoso, incendios o robos se dan a conocer de manera verbal y se documentan en reportes informales, generando retrasos en la atención, perdida de información ,ausencia de registros estadísticos y una débil cultura de prevención.

Esta deficiencia en el sistema de reportes impide que situaciones de riesgo, como la falta de equipos de emergencia (extintores), daños estructurales, emergencias médicas o actos delictivos, sean gestionado de manera eficiente, a su vez incrementando la vulnerabilidad de estudiantes, docentes y personal administrativo dentro de la comunidad universitaria.

La unidad de gestión de riesgos de la Universidad Estatal de Bolívar no cuenta con las herramientas tecnológicas necesarias para una gestión adecuada, que permitan emitir alertas inmediatas con niveles de riesgo, en la difusión de rutas de evacuación, la orientación a la comunidad sobre el contenido de una mochila de emergencia y la emisión de información preventiva en tiempo real. Esto limita su capacidad de respuesta y disminuye la cultura de prevención.

En consecuencia, la ausencia de un sistema digital integral de seguridad afecta la eficiencia operativa, dificulta la planificación preventiva y compromete la seguridad física y emocional de los miembros de la Universidad Estatal de Bolívar. Surge, por tanto, la necesidad de desarrollar una aplicación móvil de seguridad integral, con acceso exclusivo mediante credenciales institucionales, que permita a los usuarios reportar incidentes, recibir alertas, consultar información de emergencia y fortalecer los procesos de prevención y respuesta ante riesgos dentro de la universidad.

1.3. Justificación

La implementación de una aplicación móvil de seguridad integral para la unidad de gestión de riesgos en la Universidad Estatal de Bolívar representa una solución innovadora y necesaria frente a las limitaciones actuales en materia de prevención y atención de emergencias. En la institución no existe un sistema digital oficial Para el reporte, control y seguimiento de incidentes, lo que restringe de manera significativa la capacidad respuesta institucional.

Ante esta situación, se plantea el desarrollo de una aplicación móvil que fortalezca la labor de la unidad de gestión de riesgo mediante la centralización de reportes con credenciales institucionales, la emisión de alertas inmediatas clasificadas por niveles de riesgo, la disfunción de rutas de evacuación y la entrega de recomendaciones preventivas como el contenido de una mochila de emergencia. Esta herramienta proporcionará a la comunidad universitaria un canal oficial, ágil y confiable para interactuar con el sistema de seguridad institucional.

La pertinencia del proyecto radica en que responde a una necesidad prioritaria de la Universidad Estatal de Bolívar: contar con un sistema integral que reduzca la vulnerabilidad de la comunidad y potencie la capacidad de reacción frente a diversos escenarios de riesgo. En el ámbito social, la aplicación contribuirá al fortalecimiento de una cultura de seguridad más participativa, donde estudiantes, docentes y personal administrativo desempeñen un rol activo en la prevención y el reporte de incidentes. Desde la perspectiva tecnológica, constituye una solución factible e innovadora, alineada las tendencias de transformación digital en la educación superior y al uso extendido de dispositivos móviles.

Finalmente, el impacto esperado de este proyecto se traduce en mejoras sustanciales en la seguridad universitaria, optimización de los tiempos de respuesta, disponibilidad de datos confiables para la toma de decisiones y consolidación del rol de la unidad de gestión de riesgo como garante del bienestar institucional. Por lo tanto, la propuesta no sólo es viable y pertinente sino también imprescindible para salvaguardar la integridad física y emocional de la comunidad universitaria, consolidando a la Universidad Estatal de Bolívar como una institución comprometida con la seguridad, la previsión y la protección de sus miembros.

1.4. Objetivos

1.4.1. Objetivo General

Desarrollar una aplicación móvil de seguridad integral para las Unidad de Gestión de Riesgos en la Universidad Estatal de Bolívar.

1.4.2. Objetivos Específicos

- Analizar los requerimientos de la Unidad de Gestión de Riesgos para garantizar que la aplicación cumpla con sus expectativas y necesidades.
- Diseñar las funcionalidades principales de la aplicación móvil utilizando tecnologías adecuadas para plataformas Android.
- Implementar la aplicación móvil asegurando su fiabilidad, calidad y aceptación por parte de la comunidad universitaria

CAPÍTULO II

MARCO TEÓRICO

2.1. Antecedentes

Diversos estudios y proyectos han evidenciado la importancia y la factibilidad de implementar aplicaciones móviles orientadas a la seguridad y gestión de emergencias en instituciones educativas y sociales. Estas herramientas permiten optimizar los tiempos de respuesta, centralizar la información y fomentar una cultura preventiva en las comunidades a las que se aplican.

En la Escuela Superior Politécnica del Litoral (ESPOL) se desarrolló la aplicación ESPOL Alert, diseñada para que la comunidad universitaria pueda reportar incidentes mediante un botón de alerta, llamada o mensajería instantánea. El sistema notifica a los brigadistas con la ubicación geográfica del evento, lo que agiliza la respuesta y fortalece la comunicación interna. Este proyecto, construido con Flutter y bajo la metodología ágil SCRUM, demostró su efectividad reduciendo los tiempos de atención a menos de cuatro minutos (Loja & Márquez, 2023).

De manera similar, en la Universidad de las Américas, (Cachipuendo, 2019) implementó una aplicación para personas en situación de vulnerabilidad, que permite emitir alertas de emergencia, registrar la ubicación mediante GPS y almacenar evidencia en tiempo real en la nube. Este proyecto validó la pertinencia de las aplicaciones móviles como medio confiable para la atención inmediata de incidentes.

Por su parte, en la Universidad Católica de Colombia, (Mora.Melo, 2022) diseñó una aplicación con botón de pánico para notificar emergencias médicas y de seguridad. El sistema incluyó pruebas de usabilidad para garantizar la funcionalidad y aceptación del software por parte de los usuarios, mostrando que estas herramientas fortalecen la gestión de riesgos institucionales.

En el contexto ecuatoriano, (Ayala & Sigcha, 2022) desarrollaron una aplicación móvil para la seguridad comunitaria en Latacunga, incorporando un panel de administración web, alarmas físicas y virtuales, así como gestión de zonas y usuarios mediante Firebase. Este antecedente evidencia que el uso de tecnologías móviles no solo es factible en instituciones académicas, sino también en comunidades locales. A nivel, que permiten la emisión de notificaciones masivas, la geolocalización de incidentes, el reporte estudiantil y funciones preventivas como “caminata segura”.

Estos ejemplos demuestran que el uso de la tecnología móvil en seguridad universitaria constituye una tendencia consolidada y efectiva. Luego del análisis de los trabajos relacionados a mi tema de desarrollo tecnológico se establece que es adecuado implementar la aplicación móvil de seguridad integral para la unidad de gestión de riesgos en la Universidad Estatal de Bolívar.

2.2. Científico

2.2.1. Metodología de Desarrollo de Software

Según (Maida & Pacienzia, 2015):

La metodología para el desarrollo del software es un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con altas posibilidades de éxito. Una metodología para el desarrollo de software comprende los procesos a seguir sistemáticamente para idear, implementar y mantener un producto software desde que surge la necesidad del producto hasta que cumplimos el objetivo por él que fue creado.

2.2.2. Metodologías Agiles

(Loayan, 2025) manifiesta "que es una metodología iterativa, es decir se realizan entregas cíclicas y en cada entrega se realizan todas las fases del ciclo: desde toma de requerimientos, diseño, verificación y entrega…se entrega valor y recibe feedback constantemente durante todo el proyecto".

Las metodologías agiles son excelente opción para proyectos desarrollo de software complejos o con requisitos cambiantes. Permiten que el equipo se adapte a las necesidades del cliente y entregar un producto de alta calidad de forma rápida y eficiente.

2.2.3. Diferencias Metodologías Agiles y Metodologías Tradicionales

Tabla 1 *Diferencias metodologías ágiles y metodologías tradicionales*

Características	Metodologías tradicionales	Metodologías ágiles
Tipos	Cascada, Cascada, Modelo en V y Modelo Incremental	Scrum, Kanban, Lean y XP (Extreme Programming)
Estructura organizativa	Lineal	Iterativa
Escala de proyectos	Grandes (pero también aplicable a medianos)	Pequeños, medianos y escalables a proyectos grandes
Requisitos	Bien definidos antes de empezar, no flexibles	Dinámicos, susceptibles a cambios durante el proyecto

Implicación del cliente	Baja	Alta
Modelo de desarrollo	Ciclo de vida secuencial	Entrega evolutiva e incremental
Planificación	Se planifica todo con gran detalle al inicio	Planificación ajustada de Sprint en Sprint
Duración	Ciclos largos, con entregas al final de cada fase.	Ciclos cortos (2-4 semanas por Sprint)

Elaborado por: (Loarte, 2024)

Una vez examinadas las diferencias entre las metodologías, se prefirió en utilizar "SCRUM" como enfoque principal. Siendo su principal característica es la flexibilidad y adaptación a diferentes cambios.

2.2.4. Metodología Scrum

Según (Schwaber & Sutherland, 2020):

La metodología scrum es un marco de trabajo iterativo e incremental que permite que los equipos de desarrollo puedan entregar productos de calidad en ciclos cortos que se denominan como sprints. Cada uno de ellos normalmente tienen una duración aproximada de 2 a 4 semanas, a cuál se finaliza mediante la entrega de una funcionalidad del sistema.

2.2.5. Organización o Roles de Scrum

(Rodríguez, 2023) indica lo siguiente:

- **Product Owner**

Su función es garantizar que el equipo de desarrollo proporcione valor al trabajo, suministrando la orientación clara sobre el producto y los tiempos.

- **Scrum Master**

Su responsabilidad es velar por la efectividad del equipo a partir de la metodología Scrum, formando a los equipos de desarrollo y propietarios del producto.

- **Developer Team (Equipo de desarrollo)**

Su competencia consiste en trabajar colaborativamente para el logro de los objetivos y se caracterizan por tener habilidades diferentes que contribuyen al éxito del proceso y del producto.

2.2.6. Flujo o fases de un proyecto Scrum

Para (Zemsania, 2018) la metodología scrum cuenta con cinco fases y cada una comprende diversas características:

2.2.6.1. Inicio

- **Creación de la visión del proyecto.** En esta primera fase se revisa el caso de negocio del proyecto (o Project Business Case) con el objetivo de crear una Declaración de la visión del proyecto, que servirá como punto de partida y proporcionará un enfoque para el conjunto del proyecto. Durante este proceso se identifica el Product Owner.
- **Identificación del Scrum Master y los Stakeholders.** Se utilizan criterios de selección específicos para identificar al Scrum Master y a los stakeholders.
- **Formación de equipos.** Normalmente, el Product Owner es la persona responsable de

identificar a los miembros del equipo.

- **Desarrollo de épica(s).** La Declaración de visión del proyecto se utiliza como base para el desarrollo de épicas o conjuntos de historias de usuario. Las historias de usuario son las funcionalidades que debe incorporar un sistema de software, y cuya implementación aporta valor al cliente.
- **Creación del Backlog Priorizado del Producto.** Se crean las épicas y se priorizan para crear un Backlog Priorizado del Producto para el proyecto.
- **Realización de la planificación del lanzamiento.** El equipo Scrum revisa la historia de usuario en el Backlog Priorizado del Producto para desarrollar un cronograma del lanzamiento, que es un programa de implementación por fases. Se determina también la duración del sprint.

2.2.6.2. Planificación y Estimación

- **Creación de historias de usuario.** Se crean las historias de usuario y los criterios de aceptación de las historias de usuario. Normalmente, las historias de usuario son escritas por el Product Owner y se diseñan para asegurar que los requisitos del cliente estén claramente representados y sean comprensibles para todos los stakeholders. Se pueden crean las historias de usuario y se incorporan al Backlog Priorizado del Producto.
- **Estimación de historias de usuario.** Se estima el esfuerzo y tiempo necesario para desarrollar la funcionalidad de cada historia de usuario.
- **Compromiso de historias de usuario.** El equipo Scrum se compromete a entregar al Product Owner las historias de usuario aprobadas para un sprint.

- **Identificación de tareas.** Las historias de usuario se dividen en tareas específicas.
- **Estimación de tareas.** Se estima el esfuerzo y tiempo necesario para cumplir cada una de las tareas.
- **Creación del Sprint Backlog.** El equipo Scrum elabora un Sprint Backlog que contiene todas las tareas que se deben llevar a cabo en un sprint como parte de la Reunión de Planificación del Sprint.

2.2.6.3. Implementación

- **Creación de entregables.** El Equipo Scrum trabaja en el Sprint Backlog para crear los entregables del sprint. Los problemas o retos que puede ir encontrando el equipo Scrum pueden actualizarse en un registro de impedimentos o Impediment Log.
- **Realización del Daily Standup.** Se lleva a cabo una reunión diaria, también llamada Daily Standup, donde los miembros del equipo Scrum informan al resto sobre sus progresos y los problemas que le han surgido.
- **Actualización del Backlog Priorizado del Producto.** Se analiza cualquier cambio o actualización al backlog.

2.2.6.4. Revisión y Retrospectiva

- **Validación del sprint.** El equipo Scrum muestra los entregables del sprint al Product Owner y a los stakeholders relevantes en una Reunión de Revisión del Sprint para que sean aprobados.
- **Reunión retrospectiva del sprint**. El Scrum Master y el equipo Scrum se reúnen para hacer una retrospectiva de todo el sprint. En este punto, es importante documentar todo lo aprendido para aplicarlo a futuros sprints y se pueden aprobar algunas mejoras o recomendaciones (Agreed Actionable Improvements).

2.2.6.5. Lanzamiento

- **Envío de entregables.** Los entregables aceptados se envían a los stakeholders y se documenta la conclusión del sprint a través del acuerdo de entregables funcionales (Working Deliverables Agreement).
- **Realización de reunión retrospectiva.** En este punto, en el cual el proyecto se da por finalizado, los stakeholders y los miembros del equipo Scrum se reúnen para hacer una retrospectiva del proyecto y documentar lo aprendido.

2.3. Conceptual

Gestión de riesgos.

Según Soler et al (2018) es un conjunto de actividades concatenadas que a partir de un contexto dado se identifican, analizan y evalúan los riesgos para determinar los principales y mitigarlos con las acciones pertinentes (p.56).

Seguridad integral.

Para (Chancas, 2021) es un enfoque holístico de la seguridad que tiene en cuenta todos los aspectos del bienestar de una persona, organización o comunidad. Va más allá de las nociones tradicionales de seguridad, que se centran en amenazas físicas, para incluir también amenazas económicas, ambientales y sociales.

Software.

(Hashemi, 2024) manifiesta;

Que es conjunto de instrucciones, datos o programas que se utilizan para operar computadoras y ejecutar tareas específicas. Es lo opuesto al hardware, que describe los aspectos físicos de una computadora. Software es un término genérico utilizado para referirse a aplicaciones, scripts y programas que se ejecutan en un dispositivo. Puede considerarse como la parte variable de una computadora, mientras que el hardware es la parte invariable.

Android.

Es el nombre de un sistema operativo que se emplea en dispositivos móviles, por lo general con pantalla táctil. De este modo, es posible encontrar tabletas (tablets), teléfonos móviles (celulares) y relojes equipados con Android, aunque el software también se usa en automóviles, televisores y otras máquinas. (Pérez, 2021)

Firebase.

Es una plataforma en la nube para el desarrollo de aplicaciones web y móvil. Está disponible para distintas plataformas (iOS, Android y web), con lo que es más rápido trabajar en el desarrollo. (López, 2020)

Flutter.

(Cristancho, 2022) indico:

Flutter es un kit de desarrollo de software (SDK) de código abierto creado por Google que permite desarrollar aplicaciones multiplataforma (móvil, web y escritorio) desde una única base de código. Fue lanzado en 2017 y utiliza el lenguaje de programación Dart para crear aplicaciones nativas compiladas para iOS, Android y otras plataformas. Una de sus principales ventajas es que permite desarrollar aplicaciones para diferentes

sistemas operativos desde un solo código fuente, lo que agiliza el desarrollo y mantenimiento. El componente básico en Flutter son los widgets, que definen la interfaz y lógica de la app, y cuenta con conjuntos de widgets para diseño Material (Android) y Cupertino (iOS).

Stakeholders.

(Nava, 2024) señala:

Son todas las personas, grupos o entidades que tienen un interés o influencia en una organización y que se ven afectados por sus decisiones y actividades. Esto incluye tanto a los que forman parte de la empresa como a aquellos que están fuera de ella. En términos sencillos, los stakeholders abarcan cualquier persona que pueda afectar o verse afectada por los resultados de una empresa.

Base de datos.

Para (Astera, 2025):

Una base de datos es una colección estructurada de datos almacenados electrónicamente, diseñada para facilitar el acceso, gestión, modificación y recuperación eficiente de información. Está organizada en tablas, filas y columnas que facilitan la consulta y manipulación de los datos. Normalmente, su administración se realiza mediante un Sistema de Gestión de Bases de Datos (DBMS), que permite controlar y operar la base de datos con seguridad e integridad. En ámbitos empresariales, las bases de datos se utilizan para almacenar información relevante como registros de clientes, productos, transacciones o cualquier dato necesario para la gestión organizacional.

Sprint.

Es un enfoque que forma parte de las prácticas ágiles de gestión de proyectos. Fue popularizada por Jake Knapp y sus colegas de Google Ventures en su libro "Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days". El objetivo de un Sprint es validar ideas y resolver problemas complejos a través de la creación de prototipos y pruebas con usuarios en un periodo muy corto, típicamente una semana. (Montenegro, 2024)

Laravel.

(GoDaddy, 2023):

Es uno de los frameworks más populares y ampliamente utilizados de PHP. Proporciona una estructura y un conjunto de herramientas para desarrollar aplicaciones web de manera rápida, eficiente y con un código organizado. Al usar Laravel, los desarrolladores pueden

centrarse en la lógica de la aplicación en lugar de preocuparse por problemas comunes de desarrollo web, como la gestión de bases de datos, el enrutamiento y la autenticación.

MVC.

El MVC o Modelo-Vista-Controlador es un patrón de arquitectura de software que, utilizando 3 componentes (Vistas, Models y Controladores) separa la lógica de la aplicación de la lógica de la vista en una aplicación. Es una arquitectura importante puesto que se utiliza tanto en componentes gráficos básicos hasta sistemas empresariales. (Hernández, 2015)

PHP.

(Gomila, 2023) manifestó que;

Cuyas siglas significan "Hypertext Preprocessor," es un lenguaje de programación ampliamente utilizado en el desarrollo web. Su principal función es procesar el código en el lado del servidor para generar contenido dinámico en las páginas web. Esto significa que PHP es la fuerza motriz detrás de la interactividad que encuentras en línea, desde formularios de registro hasta foros de discusión en tiempo real.

API.

Una API, o interfaz de programación de aplicaciones, es un conjunto de reglas o protocolos que permite a las aplicaciones informáticas comunicarse entre sí para intercambiar datos, características y funcionalidades. (Goodwin, 2024)

Frameworks.

Es un marco o esquema de trabajo utilizado en el desarrollo de software que proporciona una estructura básica y un conjunto de herramientas reutilizables para facilitar y agilizar el proceso de creación de aplicaciones. Su objetivo principal es reducir la repetición de código, asegurar buenas prácticas y mantener la consistencia del software. Los frameworks suelen implementar patrones de diseño conocidos, como el Modelo-Vista-Controlador (MVC), para organizar el código de forma modular y mantenible. Las ventajas de usar un framework incluyen ahorro de tiempo, eficiencia en el desarrollo colaborativo, mayor seguridad, disponibilidad de herramientas y soporte comunitario. Algunos ejemplos populares de frameworks son .NET Framework, Laravel (PHP), Django (Python), Ruby on Rails (Ruby), Angular (JavaScript), Symphony y Zend Framework (PHP). (Lucena, 2023)

Feedback.

(Zendesk, 2024):

Proceso de retroalimentación que consiste en proporcionar información a una persona o equipo sobre su desempeño, comportamiento o resultados con el objetivo de mejorar, corregir errores y fomentar el aprendizaje continuo. En el desarrollo de software, el feedback es fundamental para alinear objetivos, identificar áreas de mejora, reforzar comportamientos positivos y tomar decisiones informadas. Puede ser positivo o negativo, y es un elemento clave tanto en la gestión de equipos como en la satisfacción del cliente y la mejora continua de productos o servicios.

2.4. Legal

El presente proyecto se sustenta en el marco normativo vigente en el Ecuador, que regula la seguridad ciudadana, la gestión de riesgos y la protección de la comunidad universitaria.

La Constitución de la República del Ecuador (2008) en su artículo 389 establece que el Estado es responsable de proteger a las personas, las colectividades y la naturaleza frente a los efectos negativos de desastres de origen natural o antrópico, mediante la prevención, mitigación, respuesta, rehabilitación y reconstrucción. Asimismo, ordena que todas las instituciones públicas y privadas implementen mecanismos de gestión de riesgos.

La Ley de Seguridad Pública y del Estado (2009) determina la responsabilidad de las instituciones del sistema de seguridad en diseñar y aplicar políticas, planes, programas y proyectos orientados a la protección de la ciudadanía. En este sentido, las universidades, como entes de formación y concentración poblacional, están obligadas a adoptar medidas que fortalezcan la seguridad integral de su comunidad.

El Reglamento del Sistema Nacional Descentralizado de Gestión de Riesgos (2014) establece las directrices para la implementación de estrategias y protocolos de prevención, preparación, respuesta y recuperación frente a riesgos y emergencias. Dicho reglamento define también el rol de las instituciones educativas como actores clave en la reducción de la vulnerabilidad de la población.

En el ámbito institucional, la Unidad de Gestión de Riesgos de Desastres (2018) tiene como misión principal “reducir riesgos de desastres en la Universidad Estatal de Bolívar, con un equipo humano comprometido y capacitado, apoyado en tecnología y en las expectativas de la comunidad universitaria y generando niveles óptimos de seguridad ante emergencias y desastres”. Cuenta con normativas internas sobre seguridad y bienestar universitario, en las que se contempla la responsabilidad de la unidad de gestión de riesgos

de implementar medidas preventivas y correctivas para salvaguardar la integridad física y emocional de los estudiantes, docentes y personal administrativo.

Este marco legal respalda la pertinencia y obligatoriedad de desarrollar herramientas tecnológicas, como la aplicación móvil de seguridad integral propuesta, para garantizar la protección de la comunidad universitaria y dar cumplimiento a la legislación nacional e institucional vigente.

2.5. Georeferencial

El proyecto se llevó a cabo en la Universidad Estatal de Bolívar, ubicado en la ciudad de Guaranda, Provincia Bolívar, Ecuador.

Latitud: −1.6028

Altitud: 2.668 msnm

Longitud: −79.0069

Direccion: CXHV+H3 Guanujo, Av. Ernesto Che Guevara s/n y Av. Gabriel Secaira,

Figura 1

Mapa Satelital de la Universidad Estatal de Bolívar



Nota: Google Maps

CAPÍTULO III

METODOLOGÍA

Metodología de Desarrollo de Software

3.1. Metodología de Desarrollo de Software

Para el desarrollo de la aplicación móvil y el sistema web administrativo, se empleó el marco de trabajo SCRUM, dado que facilita la adaptación a cambios a lo largo del proyecto propuesto, estructurando actividades en ciclo de trabajo de duración fija a los que se denominan como Sprints. Por la cual se organizó un cronograma que se dividió en 16 semanas identificando y estableciendo tareas dentro de cada fase, asegurándonos de que todas las fases del proyecto tengan un objetivo claro y unos entregables tangibles.

El proceso se dividió en diferentes fases:

Fase 1: Inicio

En esta etapa se determinó el alcance, así como los objetivos generales y específicos de nuestro proyecto. Además, asignamos los roles del equipo de desarrollo y de los interesados de. nominado como (stakeholders) de la unidad de gestión de riesgos.

Fase 2: Planificación

En esta etapa se creó el *Product Backlog*, el cual represento el listado priorizado de funcionalidades, requisitos y tareas para el desarrollo del proyecto.

Fase 3: Desarrollo

Durante esta fase, se llevó a cabo la implementación de las funcionalidades que, a su vez, prioriza cada uno los sprints. Realizando reuniones diarias (*Diarias Scrums*) para seguir el avance y resolver cualquier obstáculo que pudiera surgir.

Fase 4: Pruebas

Se realizarán pruebas unitarias, de integración y de usuario dentro de cada sprint para validar que el sistema cumple con los requisitos establecidos. Permitiendo la detección temprana de problemas, lo cual es crucial para la seguridad.

Fase 5: Implementación

Una vez que el sistema ha sido probado y validado, se despliega en un entorno real para su uso por parte de los usuarios finales. Esta fase incluye la capacitación de los administradores y usuarios sobre cómo utilizar la plataforma.

Para garantizar la ejecución y el éxito del proyecto, se asignó roles claros a los miembros del equipo como es Product Owner, Scrum Master y equipo de Desarrollo.

Técnicas e Instrumentos de Recopilación de Datos

Se recopilo información mediante una entrevista al personal seleccionado y observación a los procesos. Estas técnicas permitieron obtener información organizada y oportuna

3.1.1. Entrevista

La aplicación de la entrevista se la realizo al director encargado de monitorear y controlar la Unidad de Gestión de Riesgos permitiendo recabar información específica para la obtención de los requerimientos del sistema

3.1.2. Observación

La observación nos permitió conocer cómo se realizan procesos que se utilizan dentro de la Unidad de Gestión de Riesgos, incluyendo flujos de comunicación, procesos de reporte de incidentes, protocolos de actuación y gestión de información de seguridad.

CAPÍTULO IV

INGENIERÍA DEL PROYECTO

4.1. Análisis y alcance

El desarrollo de la plataforma de seguridad integral universitaria se fundamenta en una arquitectura de Cliente-Servidor, basada en servicios (API REST), lo que separa eficazmente las responsabilidades entre el procesamiento de datos y las aplicaciones de uso final.

Para el sistema de gestión administrativa se adoptó una arquitectura desacoplada. El backend fue implementado utilizando el framework Laravel, el cual funciona como una API REST encargada de la lógica de negocio. La seguridad y la gestión de la base de datos. Por su parte, el fronted administrativo se desarrolló utilizando la Liberia React, operando como una aplicación de página única.

Esta elección se justifica por la necesidad de ofrecer una interfaz web dinámica, veloz y altamente interactiva, cualidades que proporciona React, mientras se mantiene la robustez, escalabilidad y seguridad que ofrece Laravel en el servidor. Esta separación no solo facilita el mantenimiento del código, permitiendo actualizaciones independientes en el fronted o backend, sino que también mejora la experiencia del usuario al reducir los tiempos de carga mediante el consumo asíncrono de datos.

En el desarrollo del aplicativo móvil, se implementó de forma nativa para sistemas operativos Android utilizando el lenguaje Kotlin y el patrón arquitectónico Modelo-Vista-Vista-Modelo. Esta combinación asegura que el sistema sea robusto en el servidor y estable en los clientes, ofreciendo una solución integral y efectiva que responda a las necesidades de respuesta inmediata de la comunidad universitaria.

4.1.1. Identificación de los Stakeholders

Tabla 2 *Involucrados*

Stakeholder interno	**Definición breve**
Director de la Unidad de Gestión de Riesgos	Valida y aprueba el sistema
Administrador	Gestión de contenidos
Stakeholder externo	**Definición**
Comunidad universitaria	Personal que se encarga de reportar incidentes y de consumir la información emitida

Nota: Elaboración propia

4.1.2. Equipo Scrum

- **Scrum Master:** Directora académica encargada de facilitar y asegurar que el equipo de desarrollo siga el proceso scrum su rol incluye:
    1. Facilitar el proceso Scrum.
    2. Fomentar un ambiente de colaboración, trasparencia y compromiso dentro del equipo.
    3. Ayuda a eliminar impedimentos que puedan afectar al progreso del equipo.
- **Product Owner:** Pablo Holguin miembro clave del equipo de desarrollo, comunicador fundamental entre el stakeholder interno de la Universidad Estatal de Bolívar y los desarrolladores. Sus responsabilidades incluyen:
    1. Definir y priorizar el backlog del sistema.
    2. Clarificar los requisitos funcionales.
    3. Poseer una comunicación efectiva con el stakeholder y los desarrolladores.
    4. Verificar que cada entrega cumpla con las necesidades de la Unidad de Gestión de Riesgos.
- **Developer Team (equipo de trabajo):** Integrado por Nataly Silva y Pablo Holguin encargados de diseñar, desarrollar, probar e implementar el sistema a desarrollarse. Asegurando que se cumplan los estándares de calidad y funcionalidades requeridas.

4.1.3. Creación del Product Backlog

El Product backlog se estableció como la lista maestra de las funcionalidades requeridas para la aplicación móvil de seguridad integral y su sistema de gestión administrativa. La arquitectura, fue seleccionada para garantizar que el sistema se escalable y ofrezca un alto rendimiento y estabilidad.

Por ello, el marco de trabajo scrum permitió un desarrollo ágil y adecuado, identificando la importancia crítica de los requerimientos de seguridad en base a las reuniones mantenidas con los stakeholders.

La priorización se centró en aquellas historias de usuarios que aportan el mayor valor funcional y de seguridad a la comunidad universitaria.

4.1.4. Especificación de Requerimientos

Para la definición de los requerimientos funcionales del sistema, se adoptó el formato de Historias de Usuario, alineado a la metodología ágil Scrum. Estas historias describen las interacciones y funcionalidades del software desde la perspectiva de los usuarios finales,

y fueron establecidas a partir del levantamiento de información realizado mediante las entrevistas y observación directa en la Unidad de Gestión de Riesgos.

Historias de Usuario Gestión Administrativa

Tabla 3 *Autentificación de usuarios*

Historia de Usuario			
Numero: 1	**Usuario:** Director de la Unidad, Administradores		
Nombre de Historia:	Autentificación de usuarios		
Prioridad de Negocio:	Alta	**Riesgo en desarrollo:**	Bajo
Programador responsable:	Nataly Silva y Pablo Holguin		
Descripción: El usuario inicia sesión en el sistema mediante el uso de credenciales (usuario y contraseña) para acceder a las funcionalidades correspondientes a su rol.			
Criterios de Validación: 1. El sistema muestra un formulario con campos para usuario y contraseña. 2. Valida credenciales. 3. El sistema dirige a la interfaz con las funcionalidades autorizadas a su rol. 4. Se aplican mensajes de error ante credenciales invalidas.			

Nota: Elaboración propia

Tabla 4 *Gestión de usuarios*

Historia de Usuario			
Numero: 2	**Usuario:** Director de la Unidad		
Nombre de Historia:	Administración de usuarios y asignación de roles		
Prioridad de Negocio:	Alta	**Riesgo en desarrollo:**	Bajo
Programador responsable:	Nataly Silva y Pablo Holguin		
Descripción: El director de la unidad crea, edita y elimina las cuentas administrativas además de asignar roles específicos para garantizar el control de acceso.			
Criterios de Validación:			

1. Permite registrar nuevos usuarios con roles específicos.
2. Edita información de usuarios existentes.
3. Activa o desactiva cuentas de usuarios.

Nota: Elaboración propia

Tabla 5 *Visualización de estadísticas de reportes*

Historia de Usuario			
Numero:	3	**Usuario:**	Director de la Unidad
Nombre de Historia:	Visualización de estadísticas de reportes		
Prioridad de Negocio:	Alta	**Riesgo en desarrollo:**	Bajo
Programador responsable:	Nataly Silva y Pablo Holguin		

Descripción: El director de la unidad visualiza estadísticas del reporte de incidentes y alertas de pánico, para la toma de decisiones y evaluación la seguridad.

Criterios de Validación:

1. Incluir gráficos que muestren la distribución de reportes por tipo de incidente.
2. Se deberá visualizar un indicador con el número total de reportes recibidos en un periodo y el porcentaje de reportes resueltos.
3. Se debe poder filtrar las estadísticas por rango de fechas.

Nota: Elaboración propia

Tabla 6 *Atención y respuesta a incidentes y alertas de pánico*

Historia de Usuario			
Numero:	4	**Usuario:**	Director de la Unidad, Administradores
Nombre de Historia:	Atención y respuesta a incidentes y alertas de pánico		
Prioridad de Negocio:	Alta	**Riesgo en desarrollo:**	Alto
Programador responsable:	Nataly Silva y Pablo Holguin		

Descripción: El director y los administradores recibirán todos los reportes de incidentes y alertas de pánico de la app móvil para poder darles seguimiento, cambiar su estado y coordinar la respuesta de emergencia de forma prioritaria.

Criterios de Validación:

1. Los reportes se visualizan en una tabla con información clave: tipo, fecha, ubicación y estado
2. Debe haber filtros para buscar por estado (Pendiente, En Curso, Resuelto), tipo de incidente y rango de fechas.
3. Se debe poder cambiar el estado del reporte sobre la gestión del incidente.
4. El sistema debe notificar al usuario que reportó el incidente sobre el cambio de estado.
5. Las alertas de pánico deben notificar con un aviso visual y sonoro destacado en el dashboard.
6. La alerta debe mostrar la ubicación en tiempo real o la última conocida del usuario.
7. Se debe registrar la hora de activación y la identidad del usuario que activó la alerta.
8. Debe haber un registro histórico de todas las alertas de pánico gestionadas.

Nota: Elaboración propia

Tabla 7 *Gestión de contenidos*

Historia de Usuario			
Numero: 5		**Usuario:**	Administradores
Nombre de Historia:	Publicación de contenidos		
Prioridad de Negocio:	Alta	**Riesgo en desarrollo:**	Alta
Programador responsable:	Nataly Silva y Pablo Holguin		

Descripción: El administrador crea, publica, edita y elimina mochila de emergencia, alertas, plan estudiantil, recomendaciones, protocolos de seguridad, zonas de peligro y puntos de encuentro de manera centralizada para que la información se distribuya inmediatamente a la comunidad universitaria a través de la aplicación móvil.

Criterios de Validación:

1. Formularios de creación con campos para títulos, cuerpo y adjuntos
2. Debe existir una tabla para gestionar, editar y eliminar el contenido publicado.
3. Mapa para agregar puntos de encuentro y zonas de peligro.

Nota: Elaboración propia

Tabla 8 *Configuración y preferencias de perfil*

Historia de Usuario			
Numero: 6	**Usuario:** Director de la Unidad, Administradores		
Nombre de Historia:	Personalización de cuenta		
Prioridad de Negocio:	Baja	**Riesgo en desarrollo:**	Bajo
Programador responsable:	Nataly Silva y Pablo Holguin		

Descripción: El director de la unidad y los administradores pueden configurar su perfil, y personalizar apariencia

Criterios de Validación:

1. Sección de configuración perfil accesible desde el menú principal.
2. El usuario puede cambiar sus credenciales como datos y contraseña
3. Se debe personalizar apariencia de la ventana del sistema para mejor comodidad.

Nota: Elaboración propia

Historias de Usuario Aplicación Móvil.

Tabla 9 *Autentificación de usuario*

Historia de Usuario			
Numero: 7	**Usuario:** Comunidad universitaria		
Nombre de Historia:	Autenticación Segura		
Prioridad de Negocio:	Alta	**Riesgo en desarrollo:**	Medio
Programador responsable:	Nataly Silva y Pablo Holguin		

Descripción: La comunidad universitaria, inicia sesión de manera segura utilizando las credenciales institucionales para acceder a las funcionalidades de la aplicación.

Criterios de Validación:

1. La aplicación muestra una pantalla de login con campos para usuario y contraseña. Además, permite la creación de cuentas.
2. El sistema valida las credenciales.
3. Inicia sesión exitosamente, el usuario es redirigido a la pantalla principal del sistema.
4. Se deben mostrar mensajes de error claros y específicos para credenciales incorrectas.
5. Debe existir una opción para recuperar/restablecer la contraseña
6. Se realiza una sola vez para establecer la Sesión Persistente, a menos que el usuario cierre sesión explícitamente.

Nota: Elaboración propia

Tabla 10 *Creación de reporte*

Historia de Usuario			
Numero: 8		**Usuario:** Comunidad Universitaria	
Nombre de Historia:	Reporte de incidentes		
Prioridad de Negocio:	Alta	**Riesgo en desarrollo:**	Alta
Programador responsable:	Nataly Silva y Pablo Holguin		

Descripción: La comunidad universitaria enviara un reporte detallado de un incidente o condición de riesgo que este observando, incluyendo la ubicación, para que la Unidad de Gestión de Riesgos pueda actuar.

Criterios de Validación:

1. Formulario para crear un nuevo reporte debe incluir campos obligatorios para el tipo de incidente y una descripción.
2. Se debe permitir ingresar la ubicación GPS del dispositivo al enviar el reporte.
3. Se Debe permitir adjuntar una imagen o foto como evidencia (opcional).
4. El sistema debe mostrar una confirmación al usuario de que el reporte ha sido enviado con éxito.

Nota: Elaboración propia

Tabla 11 *Alerta de pánico*

Historia de Usuario			
Numero: 9	**Usuario:** Comunidad Universitaria		
Nombre de Historia:	Activación de alerta de pánico		
Prioridad de Negocio:	Alta	**Riesgo en desarrollo:**	Alto
Programador responsable:	Nataly Silva y Pablo Holguin		

Descripción: La comunidad universitaria activara una alerta de pánico de manera rápida, preferiblemente en un solo toque, para solicitar ayuda de emergencia de forma inmediata a la Unidad de Gestión de Riesgos.

Criterios de Validación:

1. Debe haber un botón de pánico prominente y accesible en la interfaz.
2. Al activarse, la aplicación debe emitir un sonido además debe enviar una señal inmediata de alerta de máxima prioridad al sistema administrativo.
3. La aplicación debe mostrar en pantalla la confirmación que indique que la alerta ha sido enviada y que la ayuda está en camino.

Nota: Elaboración propia

Tabla 12 *Acceso a la información*

Historia de Usuario			
Numero: 10	**Usuario:** Comunidad universitaria		
Nombre de Historia:	Consulta de material de Seguridad y notificaciones		
Prioridad de Negocio:	Alta	**Riesgo en desarrollo:**	Alta
Programador responsable:	Nataly Silva y Pablo Holguin		

Descripción: La Comunidad universitaria accederá fácilmente a las recomendaciones, mochila de emergencia, protocolos de seguridad, alertas, plan estudiantil, puntos de encuentros y zonas de peligro emitidos por la administración en una sección dedicada para mantener informado y saber cómo actuar en diferentes situaciones.

Criterios de Validación:

1. La aplicación debe tener secciones claramente etiquetadas para la gestión de cada uno de los contenidos.
2. El contenido debe cargarse dinámicamente desde el servidor de gestión.
3. El usuario debe poder seleccionar un ítem para ver el contenido completo.
4. Cada notificación debe mostrar un título, un breve resumen y la fecha/hora.

Nota: Elaboración propia

Tabla 13 *Configuración general*

Historia de Usuario			
Numero: 11		**Usuario:** Comunidad universitaria	
Nombre de Historia:	Configuración general		
Prioridad de Negocio:	Media	**Riesgo en desarrollo:**	Alta
Programador responsable:	Nataly Silva y Pablo Holguin		

Descripción: La Comunidad universitaria accederá fácilmente a las configuraciones generales y cierre de sesión.

Criterios de Validación:

1. Tendrá acceso a secciones claramente etiquetadas como ver perfil, historial de reportes, historial de alertas, ajustes y notificaciones, ayuda y soporte.
2. Contará con un botón para el cierre de sesión.

Nota: Elaboración propia

4.1.5. Requerimientos No Funcionales

Para la puesta en marcha del sistema de seguridad universitaria, es impredecible determinar los requerimientos no funcionales para garantiza la eficiencia de sus funcionalidades, la seguridad en el manejo de los datos de usuarios, la calidad y la usabilidad, buscando la mejor experiencia para el usuario final y el administrador.

Seguridad

- Se implementa un estricto control de acceso basado en roles (Director, Administrador) utilizando las funciones de autenticación y autorización nativas de Laravel para restringir el acceso a funcionalidades sensibles y datos críticos.
- Se garantiza la prevención de ataques comunes como CSRF y SQL Injection mediante el uso de tokens de seguridad de Laravel y consultas parametrizadas.
- Toda comunicación entre la App Móvil y el servidor debe realizarse a través de HTTPS/SSL/TLS para proteger la integridad y confidencialidad de datos.
- El proceso de autenticación debe utilizar métodos de hashing seguros. La aplicación debe evitar el almacenamiento de contraseñas y utilizar tokens de sesión cifrados para mantener la sesión del usuario.
- La aplicación móvil debe permitir la activación de la Alerta de Pánico sin requerir la autenticación del usuario en el momento de la activación.

Rendimiento

- El tiempo máximo para la carga y visualización de un nuevo reporte de incidente o alerta de pánico en el Dashboard no debe exceder los 5 segundos.
- La activación de la Alerta de Pánico y el envío del registro al servidor deben completarse en un tiempo máximo de 5 segundos para garantizar la inmediatez.
- La aplicación debe limitar el uso de la funcionalidad GPS para la geolocalización solo al momento de enviar un reporte o una alerta, evitando un consumo excesivo de la batería en segundo plano.

Disponibilidad

- El servicio API del servidor debe garantizar una disponibilidad mínima del 99.5% para asegurar la recepción continua de reportes y alertas de pánico, incluso en periodos de alta demanda.
- Se debe implementar un mecanismo de backup (respaldo) de la base de datos de manera periódica y automática para asegurar la integridad de la información y permitir una rápida recuperación del sistema ante fallos críticos.

Usabilidad

- La interfaz del Sistema de Gestión Administrativa (Web) debe ser totalmente responsiva, garantizando su correcta operación y visualización en cualquier navegador de escritorio.
- El sistema de gestión emite notificaciones visuales claras y mensajes de feedback concisos para confirmar el éxito de acciones críticas.

- La aplicación móvil debe mantener la sesión activa del usuario de la Comunidad Universitaria de forma persistente hasta que este realice un cierre de sesión explícito.
- Interfaz de usuario sumamente intuitiva, permitiendo que las funciones críticas (Reporte y Alerta de Pánico) se localicen y operen en un máximo de dos interacciones, incluso en situaciones de estrés.
- Navegación a través de secciones informativas (Noticias, Protocolos) sencilla, permitiendo al usuario acceder al contenido completo en menos de tres taps desde la pantalla principal.

Mantenimiento

- Código fuente estructurado bajo principios de desacoplamiento: el backend sigue los estándares API RESTful de Laravel (Controladores y Recursos), el panel web utiliza una arquitectura basada en componentes reutilizables de React, y la aplicación móvil aplica el patrón MVVM de Android. Esta separación facilita la detección de errores (debugging) y el mantenimiento modular de cada plataforma.
- Documentación técnica enfocada en la especificación de los endpoints de la API, la estructura de componentes del frontend y la lógica de negocio móvil, asegurando la transferencia de conocimiento para futuros desarrolladores.

Escalabilidad

- La arquitectura desacoplada (Cliente-Servidor vía API) permite la expansión de funcionalidades en el panel web sin alterar la lógica central del backend, garantizando un crecimiento sostenible del sistema con un impacto mínimo en la operación actual.

4.1.6. Características de Usuarios

Tabla 13 *Usuarios y acciones*

Rol	Funciones Principales
Director de la Unidad de Gestión de Riesgos	• Supervisa el uso general del sistema de gestión y apoya la toma de decisiones administrativas y de seguridad. • Gestiona usuarios internos: crea, edita y asigna roles a los administradores. • Recibe, visualiza y coordina la respuesta prioritaria a las alertas de pánico.

	• Consultas estadísticas de reportes de incidentes por tipo, periodo y estado
Administrador	• Registra entradas y da seguimiento a los reportes de incidentes, actualizando su estado (Pendiente, En curso, Resuelto). • Publica, edita y elimina contenidos de seguridad: noticias, recomendaciones y protocolos. • Recibe y gestiona la notificación visual y sonora de las alertas de pánico. • Configura su perfil y preferencias básicas del sistema de gestión.
Comunidad Universitaria	• Activa la Alerta de Pánico de manera inmediata para solicitar ayuda urgente. • Reporta incidentes o condiciones de riesgo, incluyendo la geolocalización y descripción. • Consume y consulta la información de seguridad emitida (noticias, protocolos). • Visualiza el estado de sus reportes de incidentes enviados y recibe notificaciones.

Nota: Elaboración propia

4.1.7. Alcance del Producto

El objetivo principal del sistema es proveer una solución integral de seguridad que garantice la comunicación eficiente y la coordinación de respuesta ante incidentes y situaciones de pánico dentro de la comunidad universitaria.

Funciones principales

Aplicación Móvil

- Permitir a la comunidad universitaria iniciar sesión de manera segura utilizando credenciales institucionales.
- Habilitar la función de Alerta de Pánico de un solo toque, enviando la ubicación geográfica en tiempo real a la plataforma de gestión.
- Permitir el registro detallado de incidentes o condiciones de riesgo, incluyendo la geolocalización y la descripción del evento.

- Visualización de Noticias, Recomendaciones y Protocolos de seguridad emitidos por la administración.

Sistema de Gestión Administrativa

- Implementación del control de acceso por roles específicos para garantizar la seguridad de los datos sensibles y las funcionalidades administrativas.
- Recepción centralizada y gestión prioritaria de los reportes de incidentes y las alertas de pánico. Incluye el seguimiento, cambio de estado y registro de acciones tomadas.
- Administración completa para la creación, edición y publicación de contenido (noticias, protocolos, recomendaciones) que se distribuye a la aplicación móvil.
- Generación de métricas y gráficos estadísticos para el Director sobre la incidencia de reportes (por tipo y periodo) y la gestión de alertas, apoyando la toma de decisiones.

4.1.8. Límites del Sistema

- El desarrollo de la aplicación móvil se limitará a la plataforma Android. La implementación para el sistema operativo iOS queda fuera del alcance inicial del proyecto.
- El sistema no incluirá integración directa con sistemas de terceros externos a la universidad, como servicios de emergencia nacionales (ej. 911) o sistemas de gestión académica existentes. La alerta se maneja internamente por la Unidad de Gestión de Riesgos.
- El sistema no implementará un módulo de chat en tiempo real o comunicación de voz entre el usuario y el administrador para la gestión de incidentes. La comunicación se limita a notificaciones de estado y contenido informativo.

SPRINT 1: DISEÑO Y FUNDAMENTOS DEL SISTEMA

4.2. Planificación del Sprint (Sprint Planning)

Se definió los objetivos y el alcance del Sprint 1, enfocado en establecer los fundamentos técnicos del sistema de seguridad universitaria. En este sprint se priorizaron tareas clave con baja dependencia externa, permitiendo al equipo concentrarse en los componentes iniciales que servirían como base arquitectónica para el desarrollo.

4.2.1. Objetivo del Sprint

Establecer la base arquitectónica y funcional del sistema, utilizando el modelo de vistas 4+1 para asegurar una cobertura completa del diseño. Esto incluye el diseño de la interfaz principal, la definición de las vistas arquitectónicas y el diseño de la base de datos.

4.2.2. Actividades Principales

Se priorizaron cuatro tareas claves con alta prioridad y baja dependencia externa para garantizar el fundamento arquitectónico del sistema. Las cuales fueron:

- **Diseño de interfaces.**
- **Vistas arquitectónicas.**
- **Diseño base de datos.**
- **Arquitectura de software.**

Las tareas sus responsables su estimación se detallan a continuación.

4.2.3. Sprint Backlog

La siguiente tabla detalla las tareas programadas para el periodo de octubre de 2025, donde estableceremos los fundamentos del sistema:

Tabla 14 *Tareas Sprint 1*

Código	Tarea	Detalle	Prioridad	Estimación	Responsable	Estado
1	Diseño de interfaz	Interfaz web y móvil	Alta	(5 días) 06 al 08 de Octubre	Desarrolladores	Pendiente
2	Vistas arquitectónicas	Elaboración de las 5 vistas arquitectónicas: vista de escenarios, vista de desarrollo, vista lógica, vista de procesos, vista física.	Alta	(5 días) 09 al 13 de Octubre	Desarrolladores	Pendiente

3	Base de datos	Diseño del esquema relacional y modelo entidad-relación	Alta	(3 días) 14 al 16 de Octubre	Desarrolladores	Pendiente
4	Arquitectura de software	Inicialización de repositorios: Laravel React (Vite) Proyecto Android	Alta	(2 días) 17 al 20 de Octubre	Desarrolladores	Pendiente

Nota: Elaboración propia

4.2.4. Desarrollo diario (Daily Scrum)

Durante la ejecución del sprint 1 el equipo de desarrollo llevo a cabo reuniones diarias de 15 minutos. Estas sesiones permitieron un seguimiento continuo del avance con el objetivo de cumplir con las metas establecidas en el sprint Backlog.

4.2.5. Revisión del Sprint (Sprint Review)

Se trabajaron actividades claves enfocadas en la planificación, análisis, y diseño del sistema de seguridad integral conforme a lo definido en el sprint backlog.

Presentación de los entregables y avances obtenidos del sprint 1

Tabla 15 *Ejecución Sprint 1*

ID	Tarea	Resultados	Responsable	Estado
1PS	Diseño de interfaz	Se definieron las interfaces para la app móvil, y el Dashboard con mapa interactivo para el panel web.	Desarrolladores	Completado
2PS	Definición Arquitectónica (4+1)	Se documentó la arquitectura usando el modelo 4+1, detallando la interacción desacoplada entre la API REST y los clientes.	Desarrolladores	Completado

3PS	Base de datos	Se desarrolló el Modelo Entidad-Relación, el Modelo Relacional con claridad estructural	Desarrolladores	Completado
4PS	Arquitectura de software	Se configuró el entorno de desarrollo: Backend en Laravel Frontend Administrativo en React App Móvil con Kotlin/Jetpack Compose.	Desarrolladores	Completado

Nota: Elaboración propia

4.2.6. Retrospectiva del Sprint

Dentro del sprint 1 se desarrolló de sitio web, el equipo scrum se llevó a cabo un trabajo enfocado en el diseño del sistema, interfaz de usuario, la definición arquitectónica 4+1, base de datos y la arquitectura de software.

Las fortalezas y obstáculos que se encontraron durante el sprint se detallaran a continuación:

Fortalezas:

- El sistema fue aprobado y entendido por todos los involucrados gracias a la claridad en el diseño.
- La estructuración del interfaz de usuario fue acogida de excelente manera, siendo de manera intuitiva facilitando la navegación y la interacción de usuario.
- El modelo de base de datos permite que el diseño sea escalable y bien estructurada.
- Arquitectura de software efectiva la cual permite buena separación de responsabilidades y escalabilidades.
- Buena comunicación y colaboración dentro del equipo manteniendo el enfoque y la resolución de problemas de manera eficiente.
- Los lapsos y distribución de tiempos establecidos fueron apropiados para cada actividad a realizarse.

Obstáculos:

- Inconvenientes en el modelamiento de diagramas vista 4+1.
- Disponibilidad de recursos técnicos.

4.2.7. Resultados del Sprint 1

Para el desarrollo del Sistema de Seguridad Integral, se priorizó una solución escalable, intuitiva y modular, buscando una respuesta eficiente ante emergencias y ajustándose a las necesidades de la comunidad universitaria.

Interfaz Web.

Diseño Interfaz de Inicio de Sesión

La imagen a continuación muestra el formulario de inicio de sesión al sistema de Seguridad Integral de la Universidad Estatal de Bolívar. Donde los usuarios administrativos deben ingresar sus credenciales como el correo y contraseña, el sistema lo validara y permite un acceso seguro, en caso de error muestra un mensaje detallando el problema.

Figura 2 *Interfaz de Inicio de Sesión*

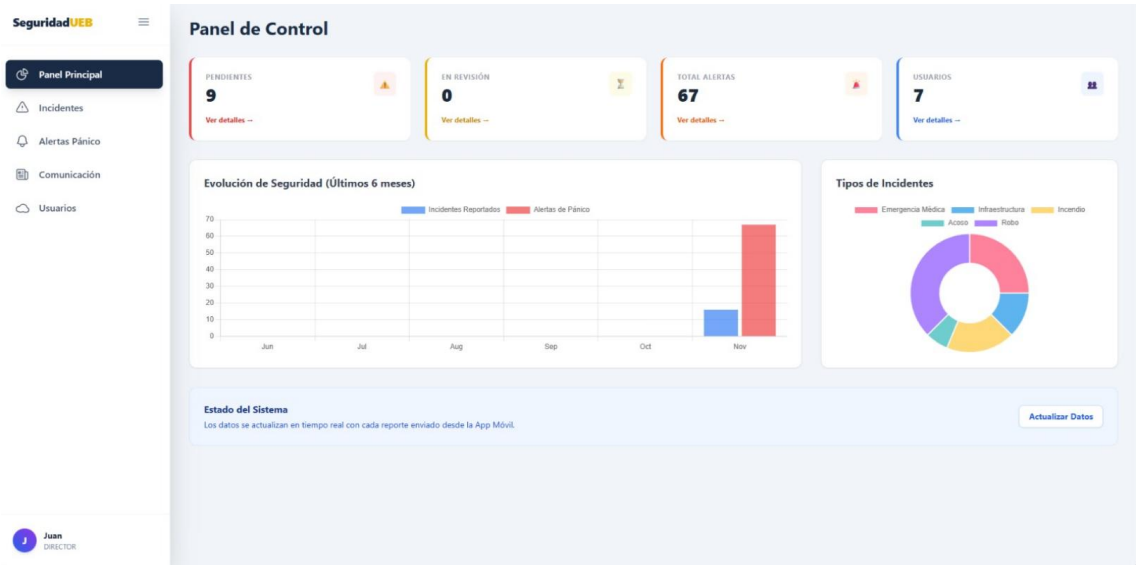


Nota: Elaboración propia

Diseño Interfaz Página Principal

La imagen indica el dashboard del personal administrativo, donde se visualizan tarjetas indicadoras, en la parte inferior se observan graficas donde se muestra la evolución de eventos y los tipos de incidentes. Además, cuenta con un menú lateral de navegación para sus diferentes secciones; para el director se incluye la sección gestión de usuario.

Figura 3 *Panel Administrativo*



Nota: Elaboración propia

Interfaz Móvil.

Diseño Interfaz de Bienvenida e Inicio de Sesión

La imagen muestra la interfaz del aplicativo móvil del sistema de Seguridad Integral de la Universidad Estatal de Bolívar. Donde la comunidad universitaria ingresa sus credenciales como el correo y contraseña, el sistema lo validara y permite un acceso seguro, en caso de error muestra una alerta detallando el problema. Además, incluye una sección de registro en caso de no tener cuenta.

Figura 4 *Interfaz Móvil Login*



Nota: Elaboración propia

Figura 5 *Interfaz Móvil Registrarse - Crear Cuenta*



Nota: Elaboración propia

Diseño de la Interfaz Principal

La imagen revela el panel principal, cuenta con diferentes tarjetas con iconos representativo y colores distintivos; en la parte inferior se encuentra un menú con diferentes secciones.

Figura 6 *Interfaz Móvil Principal*



Nota: Elaboración propia

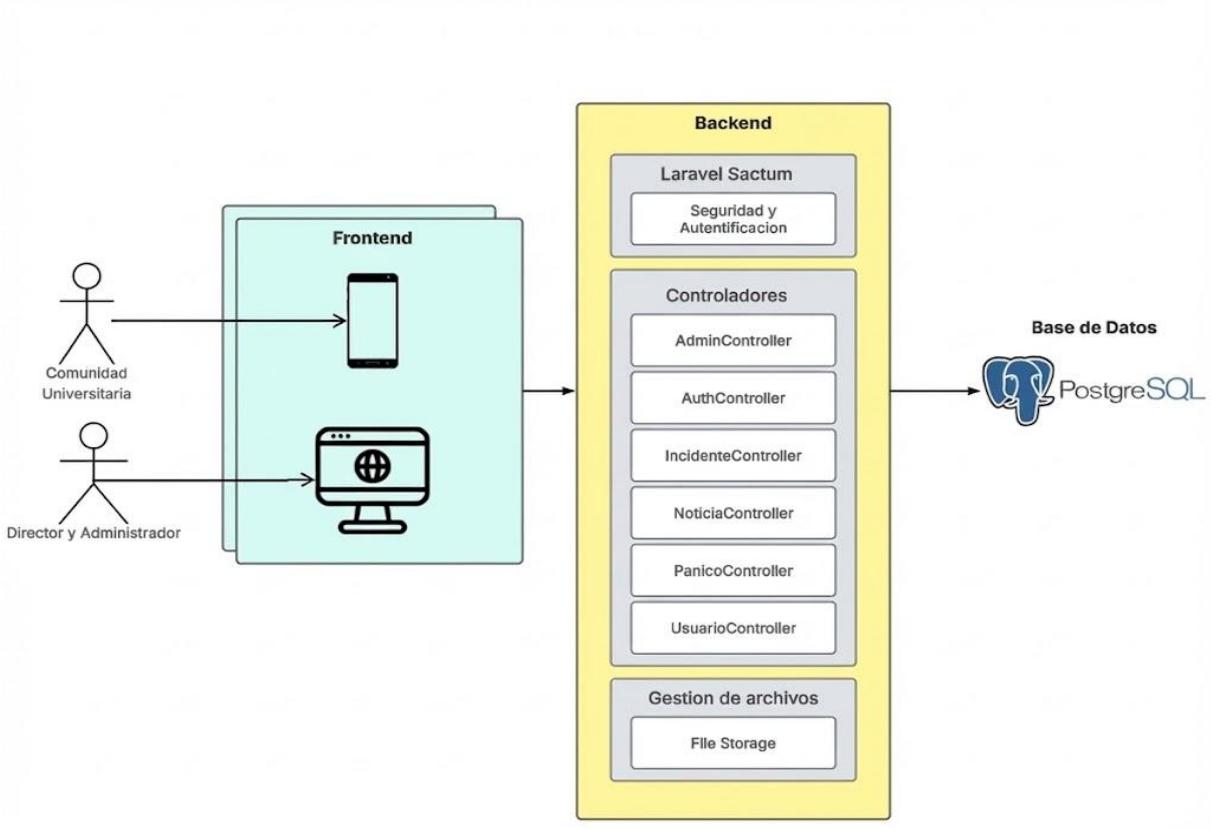
4.3. Diagramas Vistas Arquitectónicas 4+1

"Para garantizar la escalabilidad, la modularidad y el mantenimiento eficiente de la plataforma, el proyecto se fundamenta en una Arquitectura de Software Distribuida. Este enfoque arquitectónico permite desacoplar la lógica de presentación de la lógica de negocio, facilitando la integración de múltiples plataformas cliente sobre una misma infraestructura de datos.

En este contexto, la arquitectura del sistema sigue un patrón de diseño Cliente-Servidor desacoplado. En la capa Frontend, se dispone de dos interfaces: una aplicación. móvil nativa desarrollada en Android (con Kotlin) para la comunidad universitaria y un panel web administrativo construido con React para la Unidad de Gestión de Riesgos.

Ambas interfaces se comunican con el Backend a través de una API RESTful segura desarrollada sobre el framework Laravel (PHP). El servidor gestiona la lógica de negocio, la autenticación mediante Laravel Sanctum y el procesamiento de alertas en tiempo real. Finalmente, la persistencia de la información se realiza en una base de datos relacional PostgreSQL, garantizando la integridad y escalabilidad de los datos institucionales."

Figura 7 *Diagramas de Arquitectura de Alto Nivel*

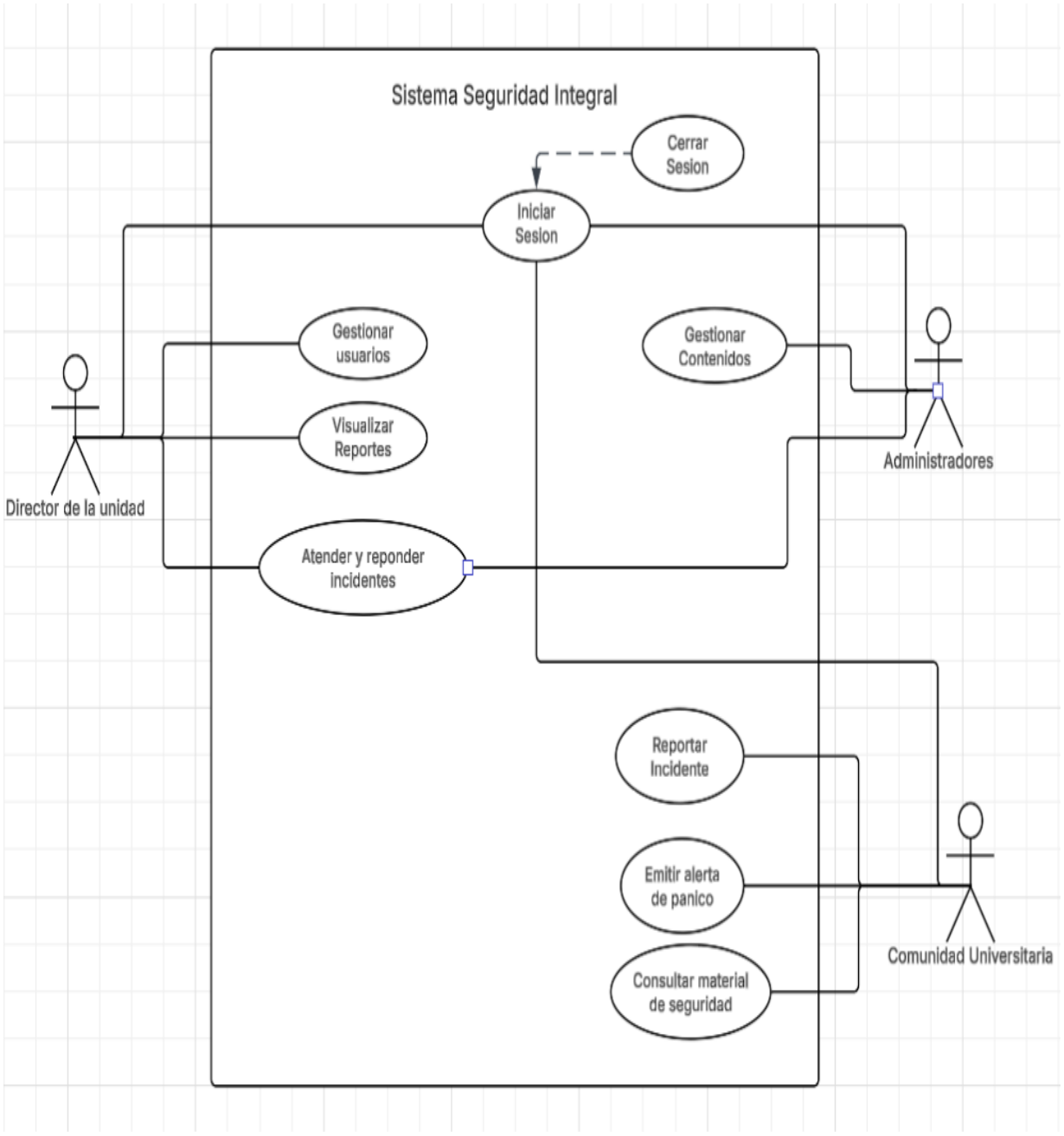


Nota: Elaboración propia

4.3.1. Vista de escenarios

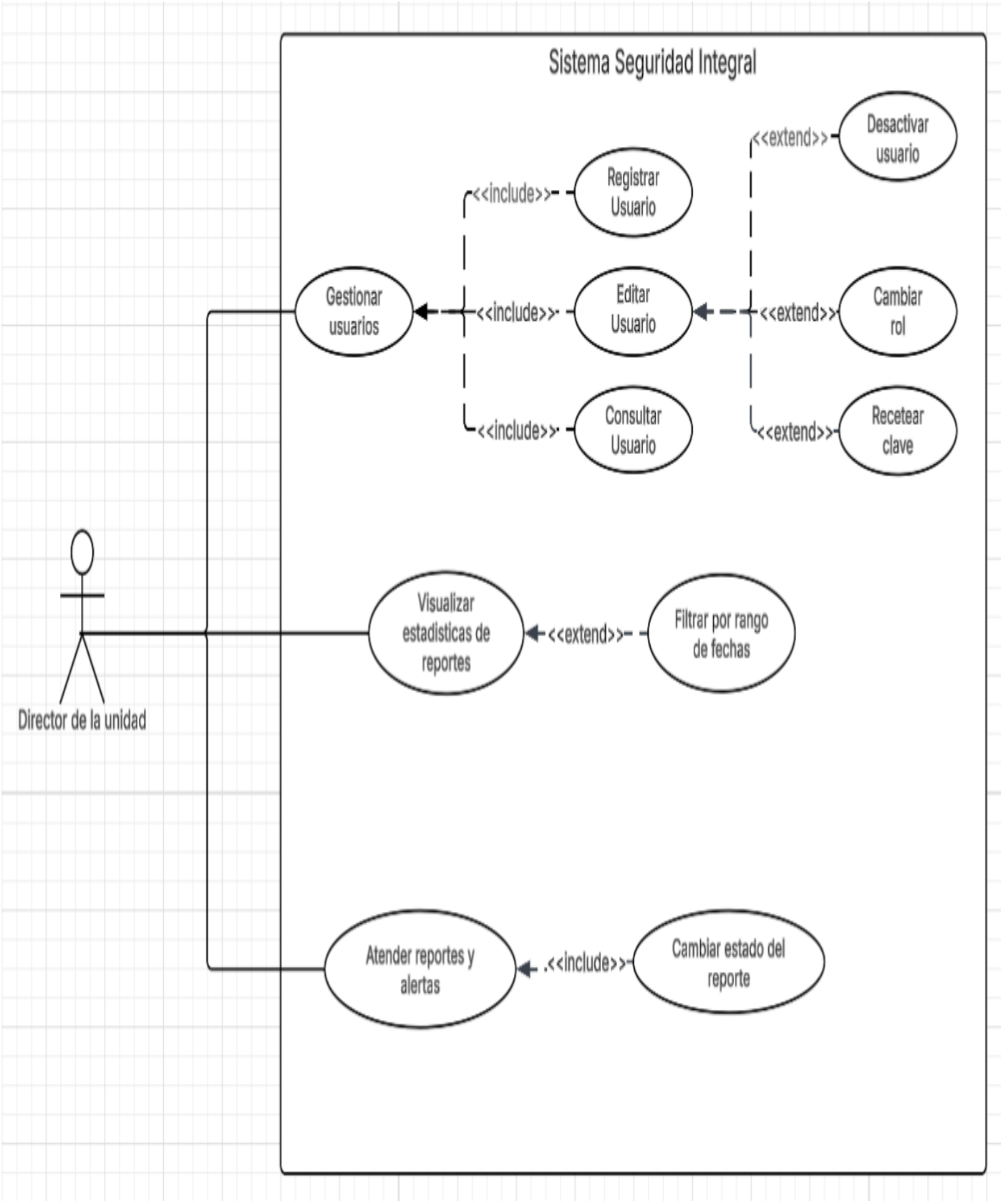
Los escenarios están representados mediante diagramas de casos de uso, los cuales permiten visualizar cómo interactúan los diferentes actores con el Sistema de Seguridad Integral. Para una comprensión estructurada, se presenta en primer lugar la visión general de la plataforma y posteriormente el desglose funcional por cada perfil de usuario.

Figura 8 *Diagrama Caso de Uso General*



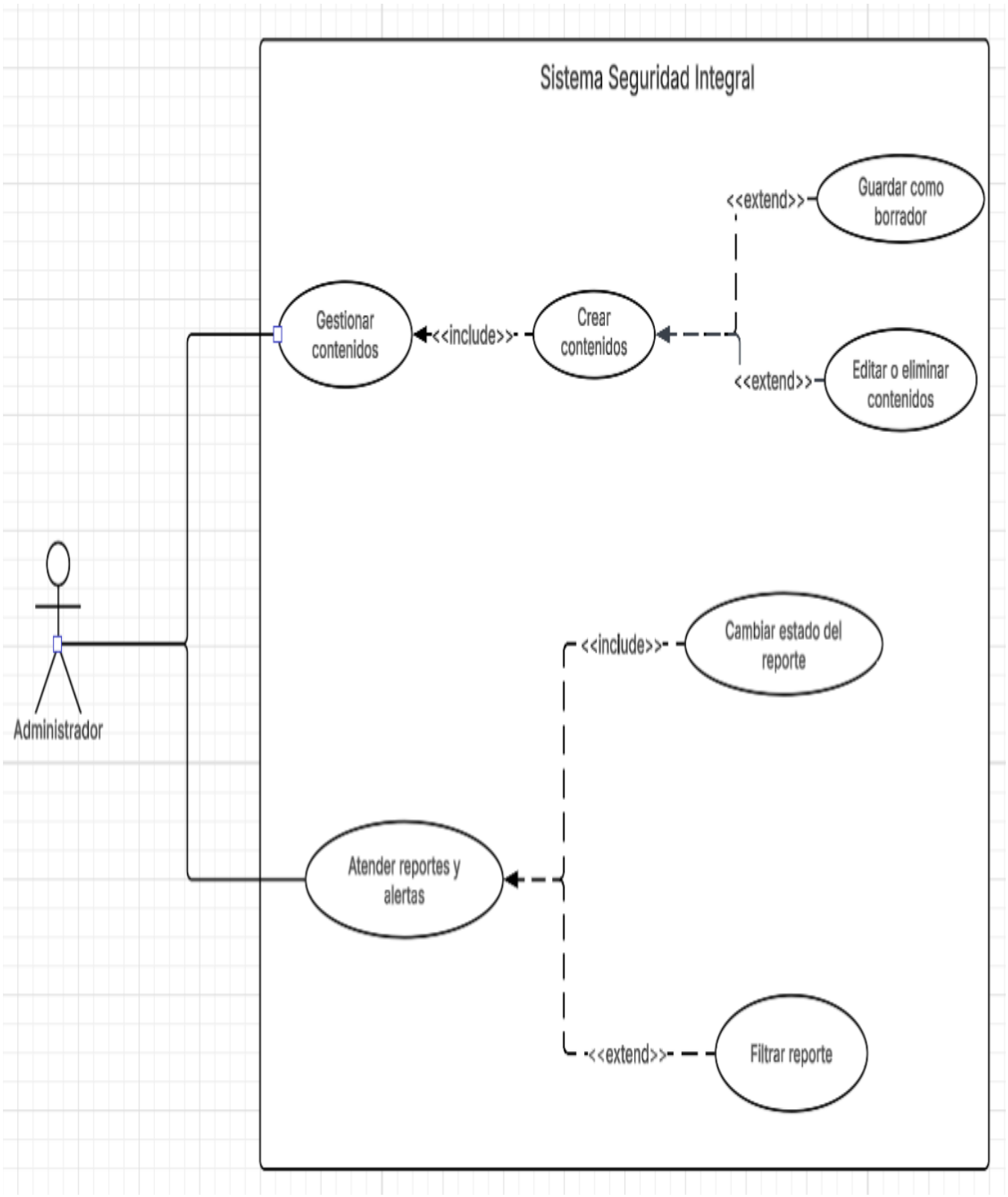
Nota: Elaboración propia

Figura 9 *Diagrama Caso de Uso Director*



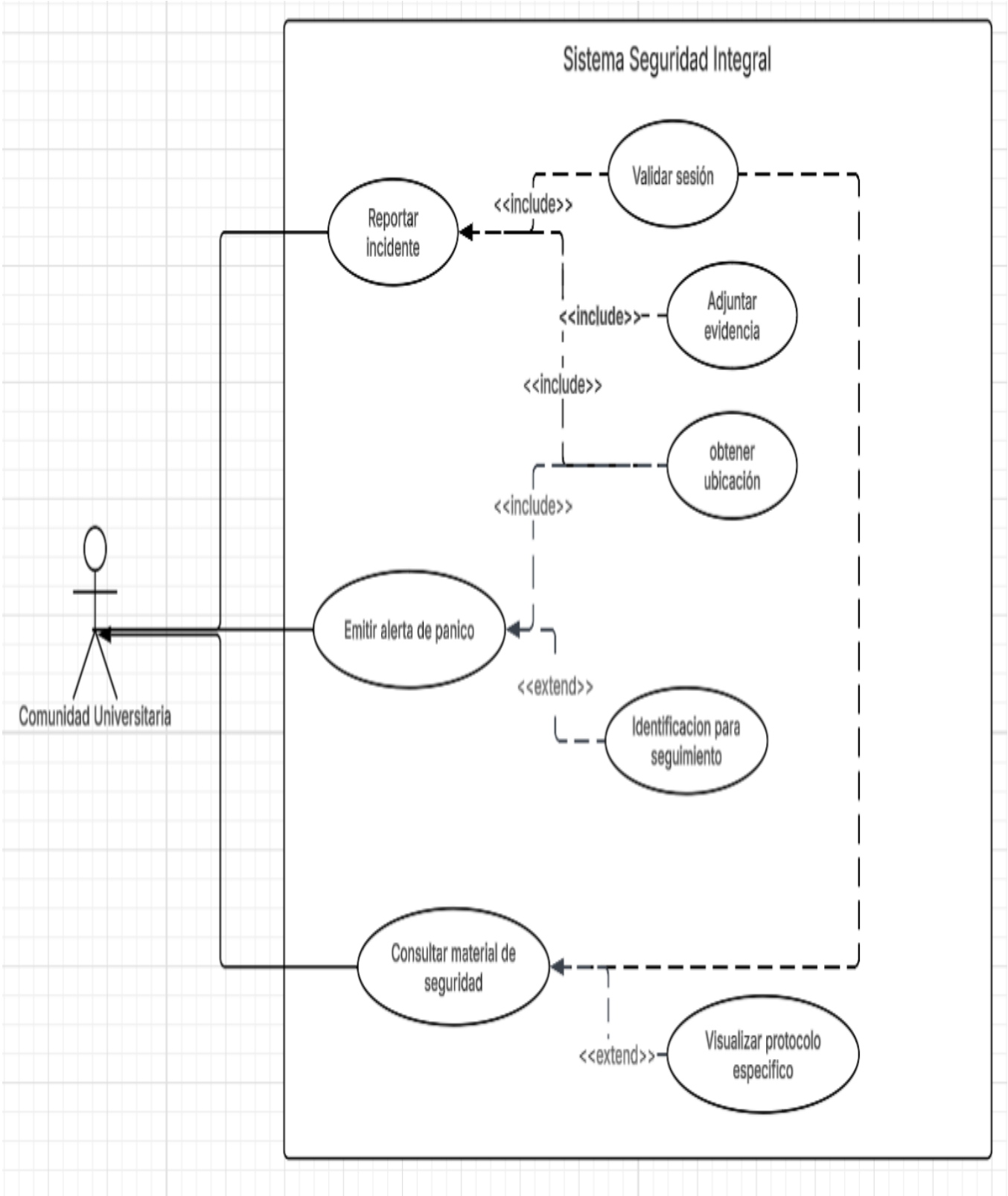
Nota: Elaboración propia

Figura 10 *Diagrama Caso de Uso Administrador*



Nota: Elaboración propia

Figura 11 *Diagrama Caso de Uso Comunidad Universitaria*



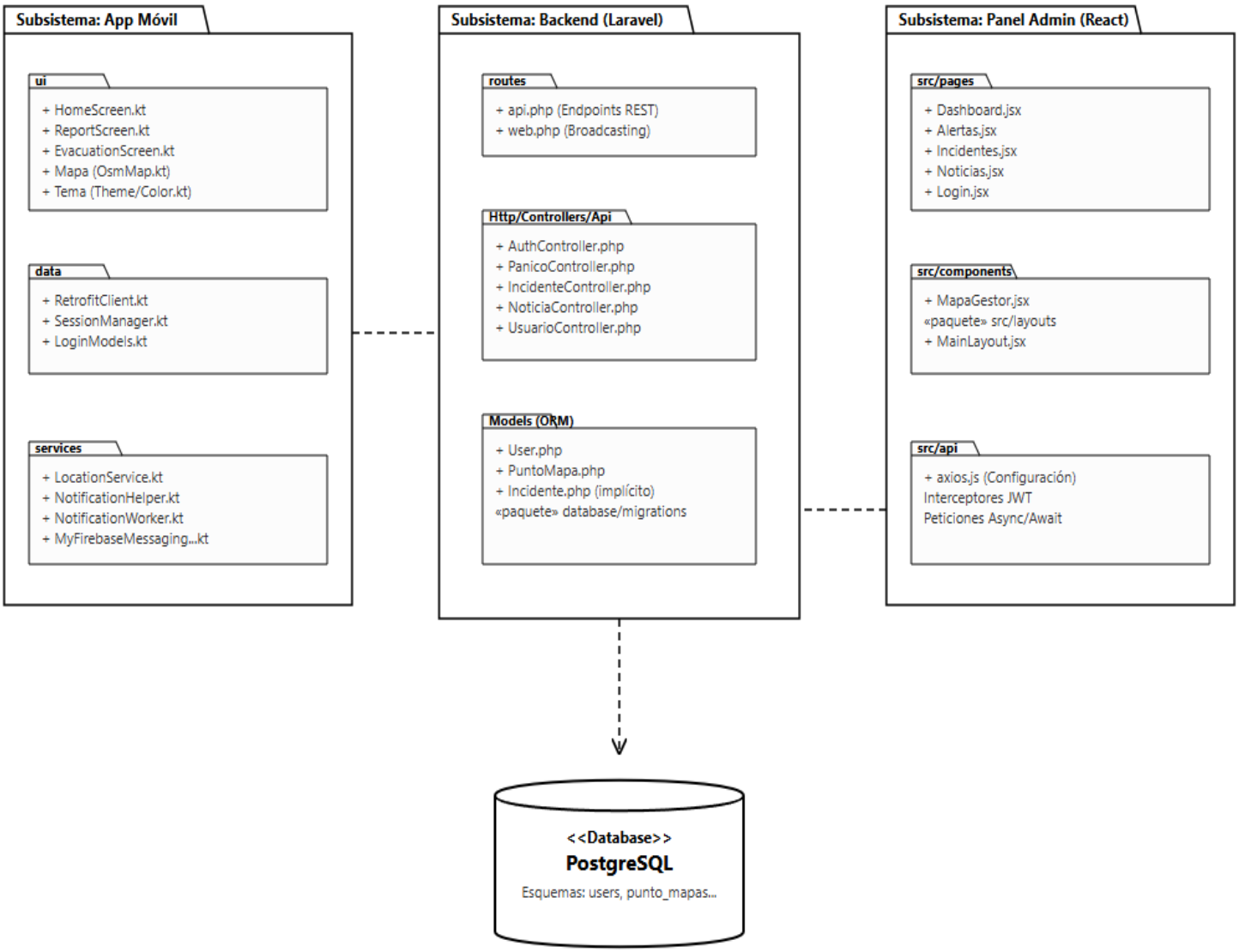
Nota: Elaboración propia

4.3.2. Vista de desarrollo

La vista de desarrollo ilustra la organización estática de los módulos de software. Se evidencia una clara separación en tres subsistemas principales: el aplicativo móvil organizado en paquetes de interfaces, datos y servicios empleando kotlin; el backend estructurado bajo el patrón MVC con los directorios nativos de laravel; el panel administrativo en react con sus páginas, componentes y configuración de peticiones a la API.

Esta distribución modular facilita el mantenimiento de código y asegura la escalabilidad del proyecto.

Figura 12 *Vista de Desarrollo*

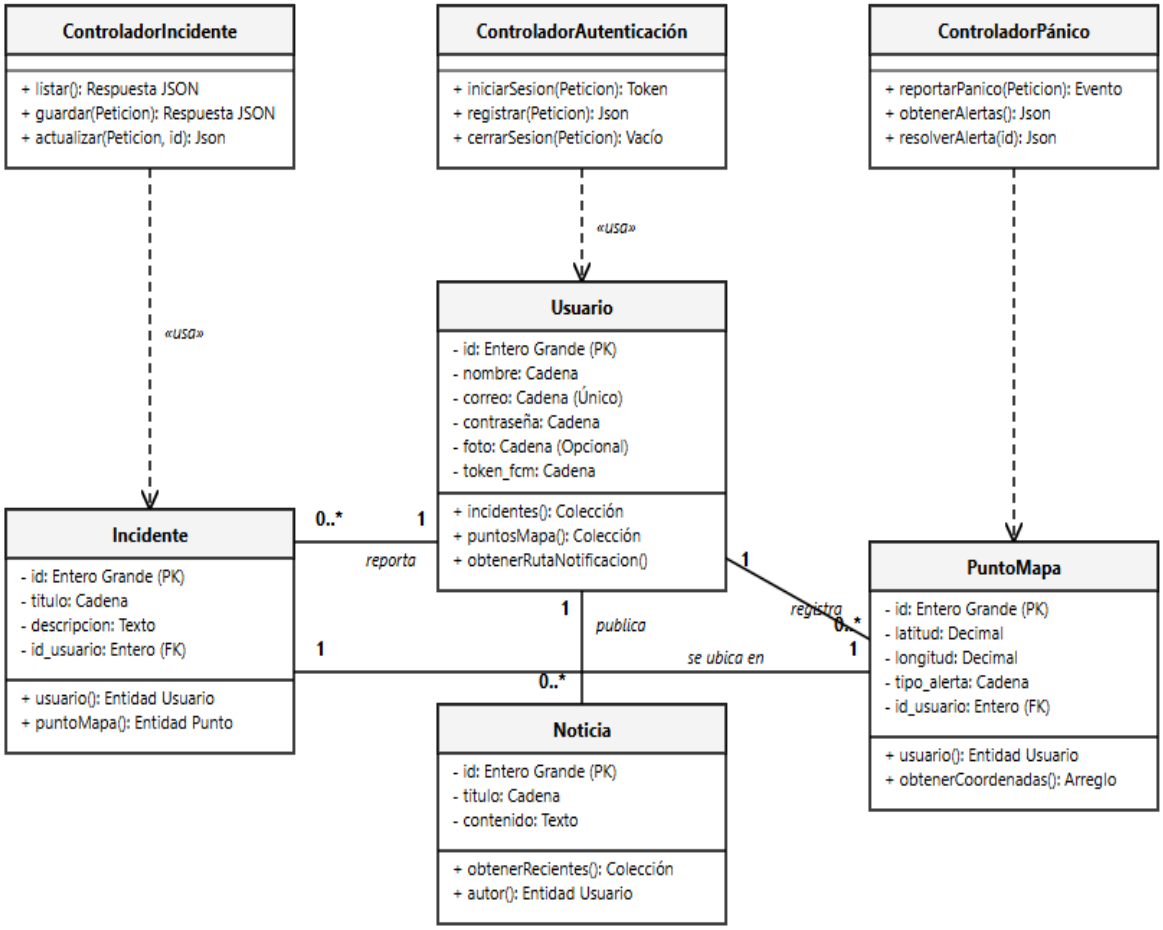


Nota: Elaboración propia

4.3.3. Vista Lógica

La vista lógica representa la estructura interna orientada a objetos y las principales abstracciones del sistema. Mediante un diagrama de clases, se detallan los controladores centrales responsables de la gestión de peticiones. Asimismo, define las entidades del modelo de negocio y las relaciones de multiplicidad entre estas, demostrando como fluyen los datos y como se vinculan obligatoriamente, por ejemplo, un reporte de incidentes con el usuario que lo emite.

Figura 13 *Vista Lógica*



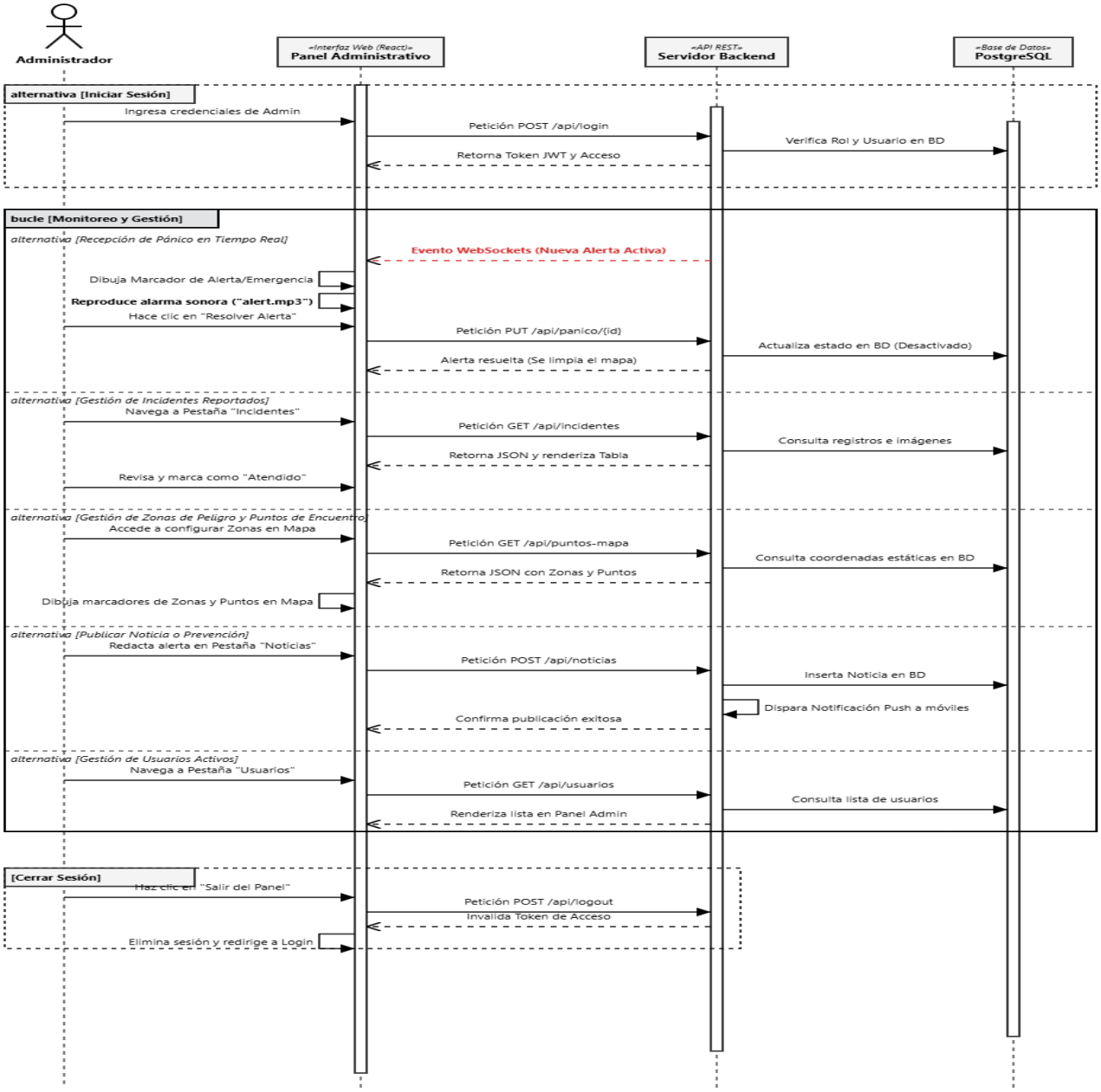
Nota: Elaboración propia

4.3.4. Vista de Procesos

Panel Administrativo

En la vista de proceso corrspondiente al panel administrativo, se modela el comportamiento dinamico del sistema y el flujo de comunicación. El diagrama de secuencia detalla como las interacciones del personal administrativo viajan desde la interfaz web hacia el servidor y concluye en consultas a la base de datos.Destacando visualmente el uso de eventos de red para la recepcion ininterrumpidas de alertas, permitiendo al sistema dibujar marcadores y emitir alarmas sonoras en tiempo real ante una emergencia.

Figura 14 *Vista de Procesos Panel Administrativo*

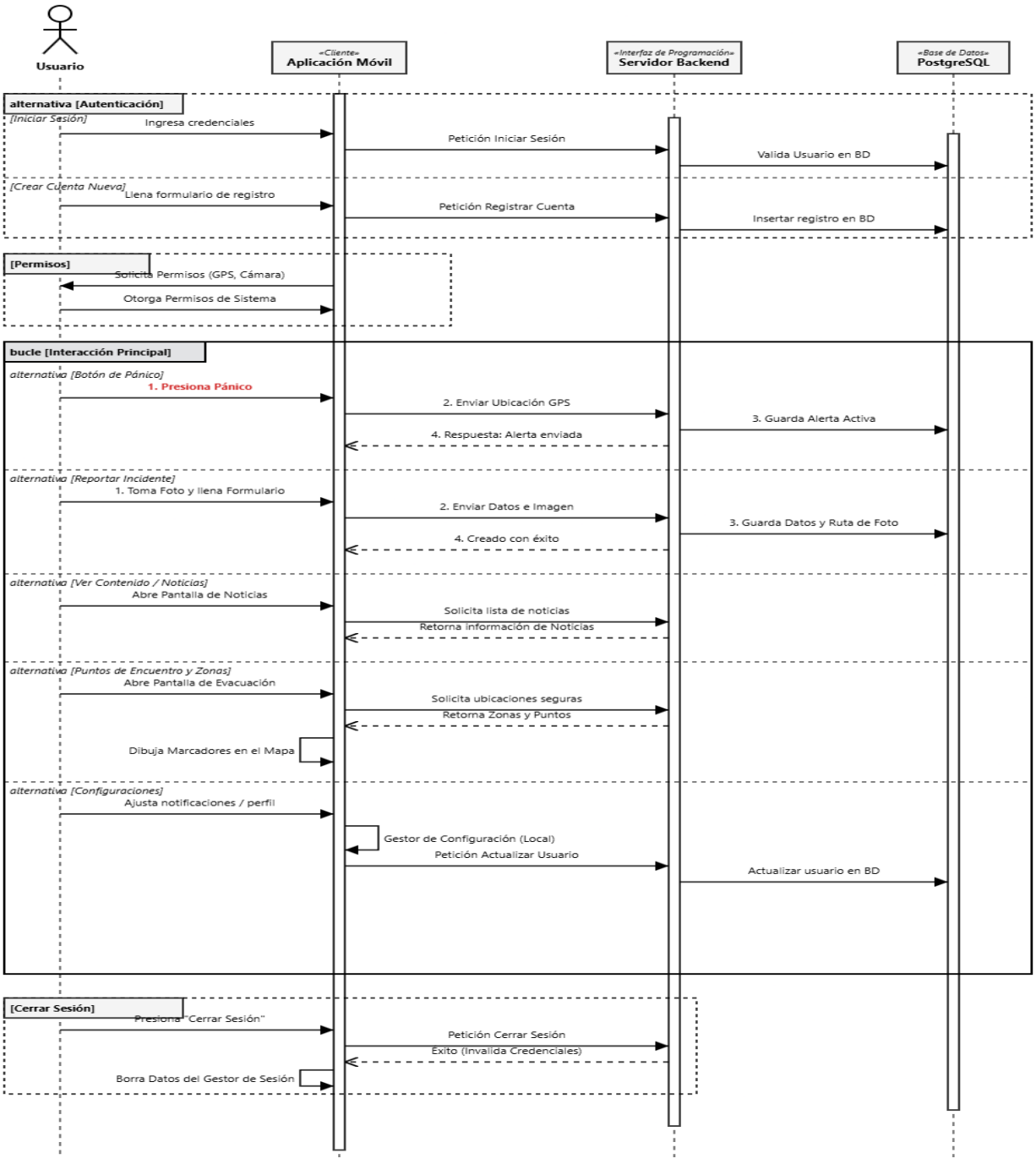


Nota: Elaboraciòn Propia

Aplicación Movil

Esta vista describe la secuencia de operaciones dinamicas desde la perspectiva de la aplicación.Muestra el ciclo de vida de las interacciones criticas, abarcando desde la validacion de credenciales y la solicitud de permisos del dispositivo, hasta el envio de alerta y coordenadas al presionar el botón de pánico. El diagrama ilustra claramente cómo la aplicación móvil se mantiene a la espera de la confirmación del servidor para emitir la respuesta visual del seguridad al estudiante o docente.

Figura 15 *Vista de Procesos App Móvil*

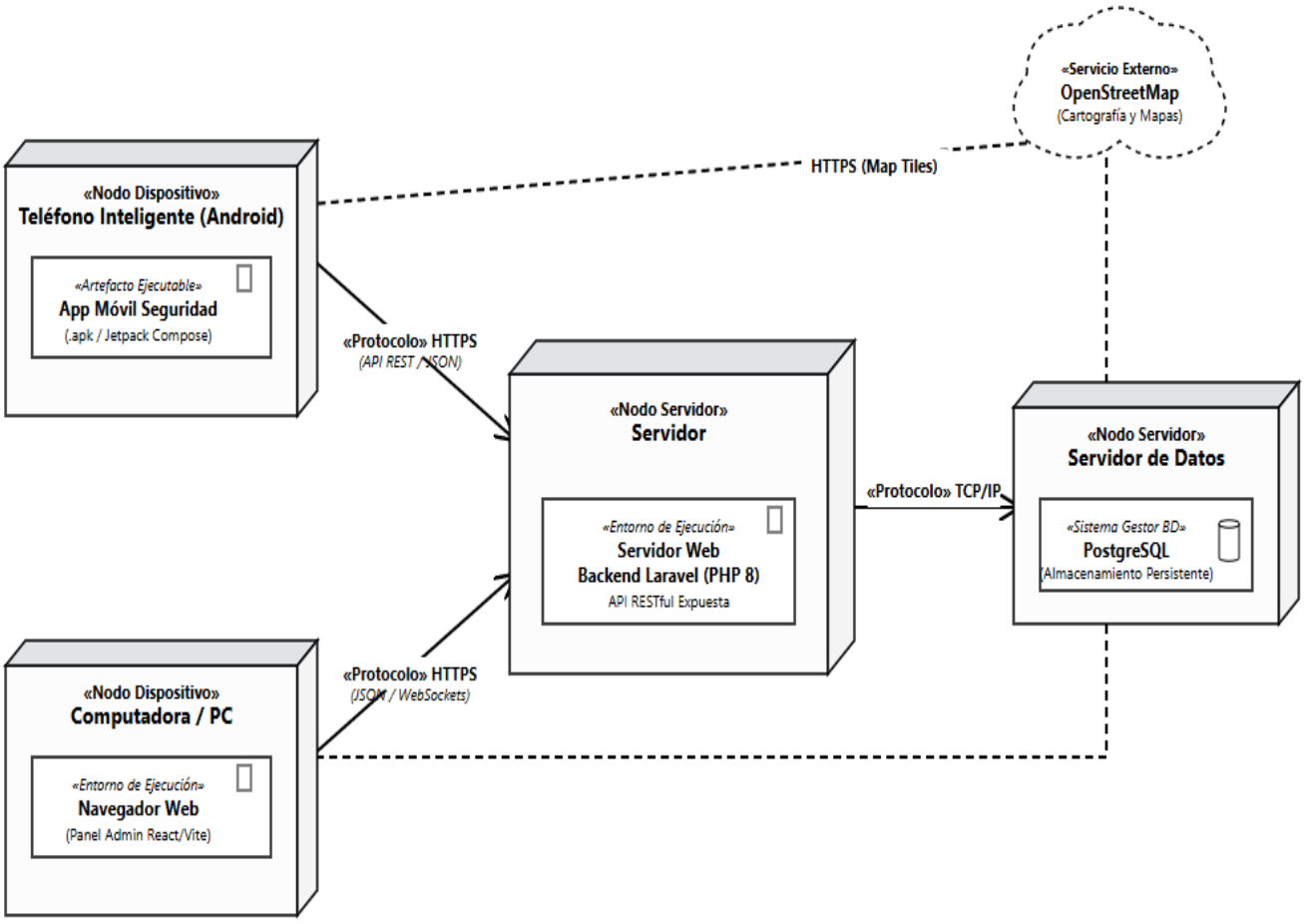


Nota: Elaboración Propia

4.3.5. Vista Física

La Vista Física mapea los componentes de software a la infraestructura de red y hardware en la que se ejecutarán. El diagrama de despliegue muestra la topología del sistema distribuido: los teléfonos inteligentes y computadoras operando como nodos de cliente, comunicándose mediante el protocolo seguro HTTPS con el servidor principal. Expone también el aislamiento del servidor de datos (PostgreSQL) vía TCP/IP y la integración estratégica con servicios externos en la nube, como Firebase (FCM) para la gestión de notificaciones push y OpenStreetMap para la cartografía base.

Figura 16 *Vista Física*



Nota: Elaboración propia

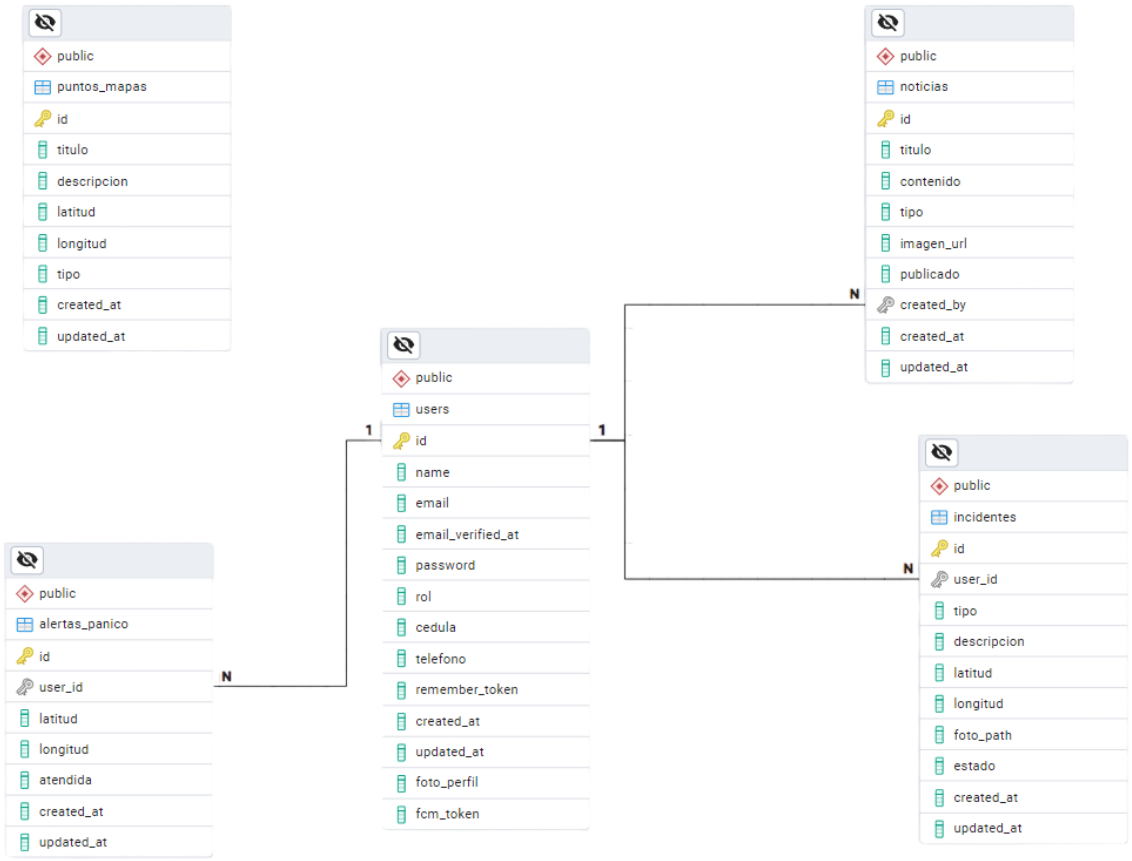
4.4. Diagramas base de datos

4.4.1. Modelo Entidad Relación

El Modelo Entidad-Relación define la arquitectura conceptual de los datos del proyecto. Ilustra de manera gráfica las colecciones fundamentales requeridas para el

funcionamiento del Sistema de Seguridad Integral. Este diagrama permite visualizar gráficamente la estructura de almacenamiento, las llaves primarias y las asociaciones relacionales que garantizan que el flujo de reportes, usuarios y contenidos se gestione sin redundancias.

Figura 17 *Modelo Entidad Relación*



Nota: Elaboración propia

4.4.2. Modelo Relacional

El modelo relacional es la estructura formal de las tablas que se implementarán en el motor de base de datos. Este esquema textual especifica la composición exacta de cada tabla, definiendo sus atributos. Este modelo es el esquema definitivo que asegura la integridad referencial en todo el sistema.

Figura 18 *Modelo Relacional*

users (id, name, email, email_verified_at, password, rol, teléfono, remember_token, created_at, updated_at, foto perfil, fcm_token)
incidentes (id, user_id, tipo, descripción, latitud, longitud, foto_path, estado, created_at, updated_at)
alertas_panico (id, user_id, latitud, longitud, estado, created_at)
punto_mapa (id, titulo, descripción, latitud, longitud, tipo, created_at, updated_at)
noticias (id, titulo, contenido, tipo, imagen_url, publicado, created_by, created_at, updated_at)

Nota: Elaboración propia

SPRINT 2 DESARROLLO DEL SISTEMA ADMINISTRATIVO

4.5. Planificación Sprint

Se efectúo una reunión de planificación con el equipo Scrum para precisar el objetivo y seleccionar las historias de usuario a desarrollar durante este sprint, con el fin de progresar en el desarrollo y diseño de las funcionalidades del sistema Seguridad Integral.

4.5.1. Objetivo del Sprint

Desarrollar y diseñar todas las funcionalidades correspondientes al acceso: visualización y monitoreo de incidentes, monitoreo de alertas de pánico, gestión de contenidos y gestión de usuarios estos eventos se observarán dentro del panel Administrativo en tiempo real.

4.5.2. Actividades Principales

Las actividades planificadas y realizadas durante el sprint fueron:

- Diseño del formulario de entrada que verifica el rol del personal administrativo.
- Implementación de alertas visuales para credenciales incorrectas.
- Menú lateral colapsable para acceder a todas las secciones.
- Sistema con sonido de alarma y notificación flotante si llega una emergencia o incidente.
- Tarjetas con contadores de incidentes pendientes, en revisión y total de alertas.
- Implementación de gráficos visuales para ver la tendencia de incidentes y que tipos de problemas son los más comunes.
- Tabla para ver todos los reportes con filtros por nombre, fecha y tipo de problema.

- Modal de detalle para ver la evidencia, leer la descripción detallada, saber su ubicación y cambiar el estado del incidente.
- Listado priorizado de botones de pánicos activados por la comunidad universitaria.
- Botones directos para abrir la ubicación GPS en el marca y marcar la emergencia como atendida una vez resuelta.
- Formulario para redactar y publicar noticias, protocolos o recomendaciones que se verán en la app móvil.
- Herramienta interactiva para colocar marcadores de puntos de encuentro y zonas de peligro.
- Sección dividida para gestionar el personal administrativo y la comunidad universitaria.
- Formulario para crear o editar usuarios con validación automática de números de cedulas y correos electrónicos reales.
- Modal de configuración para editar perfil, apariencia del sistema y cierre de sesión con advertencia para evitar salidas accidentales del sistema.
- Proceso automático que elimina el token de seguridad y los datos al confirmar su salida, redirigiendo al login.

4.5.3. Sprint Backlog

Se establecieron en las siguientes historia de usuario para ser implementadas dentro de este sprint:

Tabla 16 *Historia de usuario seleccionadas*

Código	Historia de usuario	Descripción	Prioridad
HU001	Autentificación de Usuario.	El usuario inicia sesión en el sistema mediante el uso de credenciales (usuario y contraseña) para acceder a las funcionalidades correspondientes a su rol.	Alta
HU002	Administración de usuarios y asignación de roles.	El director de la unidad crea, edita y elimina las cuentas administrativas además de asignar roles específicos para garantizar el control de acceso.	Alta

HU003	Visualización de estadísticas de reportes.	El director de la unidad visualiza estadísticas del reporte de incidentes y alertas de pánico, para la toma de decisiones y evaluación la seguridad.	Alta
HU004	Atención y respuesta a incidentes y alertas de pánico.	El director y los administradores recibirán todos los reportes de incidentes y alertas de pánico de la app móvil para poder darles seguimiento, cambiar su estado y coordinar la respuesta de emergencia de forma prioritaria.	Alta
HU005	Publicación de contenidos	El administrador crea, publica, edita y elimina noticias, recomendaciones, notificaciones, protocolos de seguridad de manera centralizada para que la información se distribuya inmediatamente a la comunidad universitaria a través de la aplicación móvil.	Alta
HU006	Personalización de cuenta	El director de la unidad y los administradores pueden configurar su perfil, y personalizar apariencia	Baja

Nota: Elaboración propia

4.5.4. Desarrollo Diario (Daily Scrum)

Durante la ejecución del sprint, se llevaron reuniones breves con el equipo de desarrollo. Estas reuniones permitieron dar seguimiento al avance del sprint y detectar posibles obstáculos o cuellos de botella permitiéndonos tomar decisiones rápidas para asegurar el cumplimiento del objetivo.

Bajo este desarrollo el equipo abordo la construcción del sistema administrativo enfocándose en la integración de la interfaz de usuario con la lógica del servidor logrando así la implementación de las actividades principales descritas anteriormente

4.5.5. Revisión del Sprint (Sprint Review)

Al finalizar el sprint, se realizó la demostración del sistema administrativo ante el director del proyecto para validar el cumplimiento de las actividades planteado. Durante esta

sesión, se presentó el incremento funcional del sistema verificando el correcto funcionamiento de los módulos implementados.

A continuación, se detallan los entregables:

Tabla 17 *Ejecución del Sprint 2*

Código	Historia de usuario	Funcionalidad desarrollada	Resultado	Observaciones	Estado
HU001	Autentificación de Usuario.	Inicio de sesión con validación de roles y seguridad.	Acceso seguro implementado	Se verificó el correcto funcionamiento del login, permitiendo el acceso solo a personal autorizado y bloqueando credenciales inválidas mediante alertas visuales.	Completado
HU002	Administración de usuarios y asignación de roles.	Gestión de cuentas administrativas y de la comunidad universitaria.	Módulo de gestión (CRUD) funcional	El módulo permite registrar, editar y listar usuarios sin errores, validando automáticamente que los correos y números de cédula sean únicos y correctos.	Completado
HU003	Visualización de estadísticas de reportes.	Panel de control (Dashboard) con gráficos y contadores.	Dashboard con métricas en tiempo real	El dashboard presenta la información estadística de manera clara, actualizando los gráficos y contadores de incidentes en tiempo real según los reportes recibidos.	Completado
HU004	Atención y respuesta a incidentes y	Monitoreo de alertas de pánico y	módulos de seguimiento y monitoreos	El sistema notifica las emergencias exitosamente mediante	Completado

	alertas de pánico.	reportes de seguridad.	de emergencias	alertas sonoras y visuales, permitiendo ubicar el incidente en el mapa de forma inmediata.	
HU005	Publicación de contenidos	Módulo para la gestión de noticias y comunicados.	Gestor de contenidos y noticias	La herramienta publica las noticias correctamente, almacenando las imágenes y textos para que sean distribuidos a la aplicación móvil sin inconvenientes.	Completado
HU006	Personalización de cuenta	Configuración de perfil y cierre de sesión.	Sistema de Configuración y Cierre de Sesión	Las funciones de edición de perfil y cierre de sesión se ejecutan correctamente, garantizando que la cuenta quede protegida al finalizar el trabajo.	Completado

Nota: Elaboración propia

4.5.6. Retrospectiva del Sprint (Sprint Retrospective)

El equipo de desarrollo realizo una sesión de análisis para evaluar el proceso de trabajo durante el sprint 2. El objetivo fue identificar las prácticas que generaron valor y los problemas técnicos que dificultaron el avance, con el fin de mejorar el rendimiento para los próximos sprints.

Fortalezas:

- El uso de la arquitectura de componentes en react facilito la reutilización de elementos visuales (como los botones, tarjetas y modales) lo que acelero la construcción de las diferentes vistas del panel administrativos.

- La incorporación de herramientas externas como chart.js para los gráficos y el leaflet para los mapas fue exitosa y rápida, permitiendo cumplir con los requisitos visuales sin desarrollar estas funcionalidades desde cero.
- Las reuniones diarias permitieron resolver dudas sobre la estructura, evitando errores de lógica en el desarrollo de los controladores del backend.

Obstáculos:

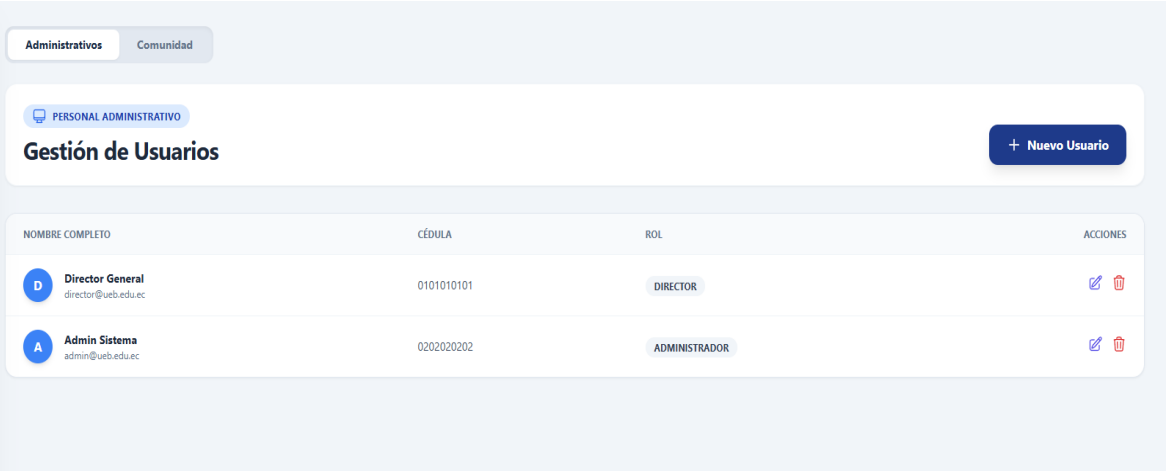
- Se presentaron dificultades iniciales al intentar actualizar los datos del dashboard en tiempo real sin recargar la página. Esto represento un reto en el manejo de los estados de la aplicación, lo cual se solucionó implementando un ciclo de consultas periódicas.
- La previsualización de imágenes antes de subirlas genero errores de formato en el navegador, lo que requirió más tiempo adicional para manejar correctamente los objetos de tipo file.

4.5.7. Resultados del Sprint 2

Administración de usuarios

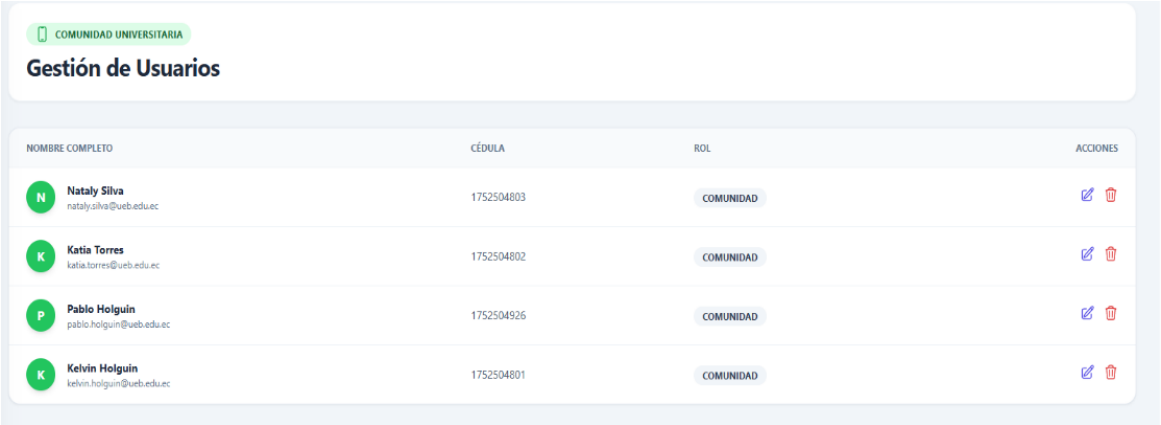
Sección diseñada para el control del personal. La interfaz utiliza pestañas para separar la gestión administrativa y comunidad universitaria.

Figura 19 *Administración de Usuarios*



Nota: Elaboración propia

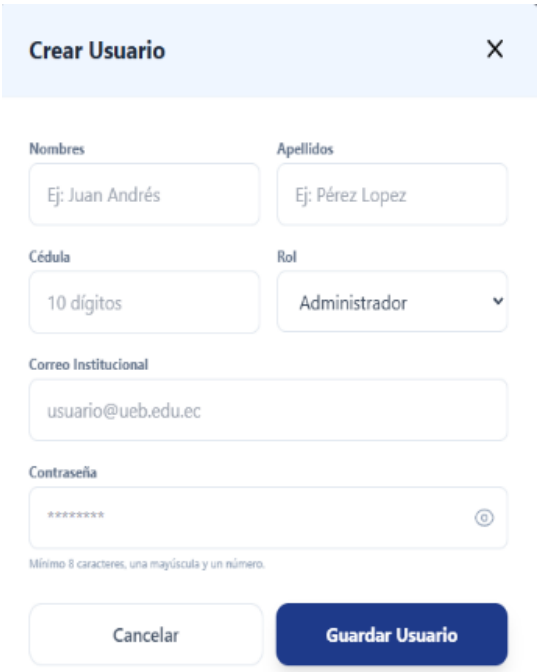
Figura 20 *Gestión de Usuarios*



Nota: Elaboración propia

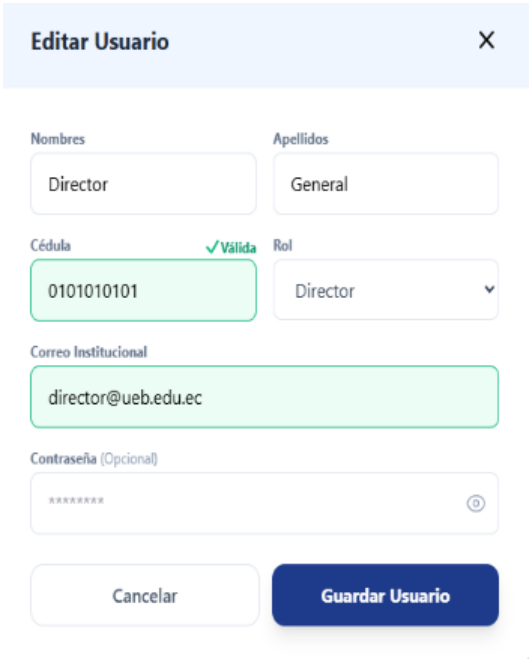
La ventana administrativa cuenta con un formulario valido para crear, editar, y eliminar usuario administrativo asegurando que los correos institucionales y números de cedula sean únicos y correctos antes de registrarlos en el sistema.

Figura 21 *Crear Usuario*



Nota: Elaboración propia

Figura 22 *Editar Usuario*



Nota: Elaboración propia

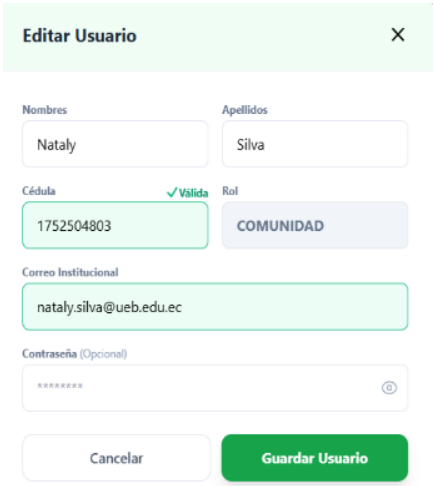
Figura 23 *Acción Eliminar Usuario*



Nota: Elaboración propia

La pestaña de comunidad en cambio solo cuenta con la acción de editar.

Figura 24 *Editar usuario*

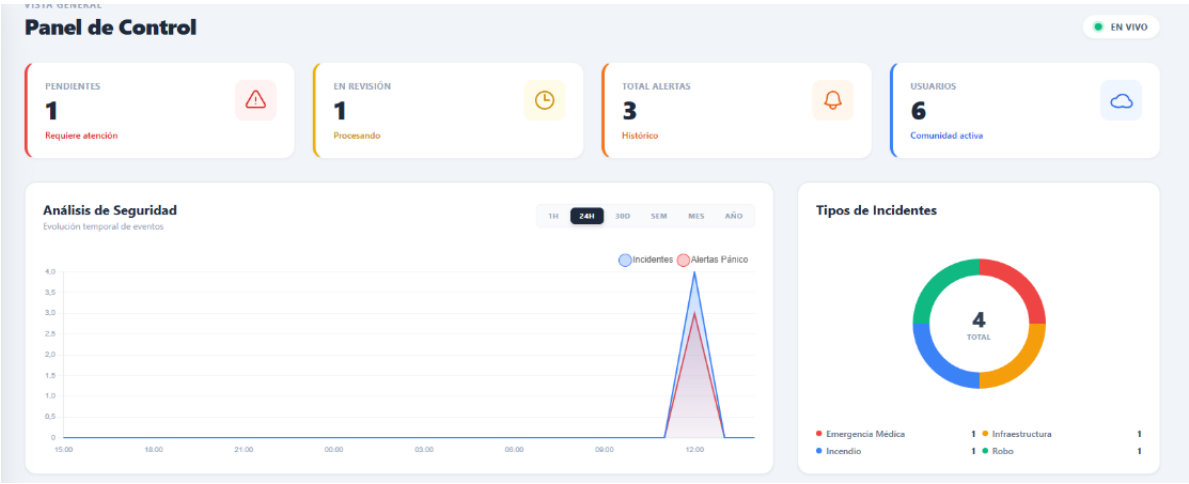


Nota: Elaboración propia

Estadísticas y reportes (Dashboard)

En la página principal del sistema. Se presentan 4 tarjetas de indicadores claves que muestran en tiempo real el conteo de alertas (pendiente, en revisión, Total) y el total de usuarios. Además, integra dos gráficos estadísticos un gráfico de líneas para visualizar la tendencia de incidentes por fechas y un gráfico de donas que clasifica los problemas por tipo (robo, medico, infraestructura, etc.)

Figura 25 *Estadísticas y Reportes*



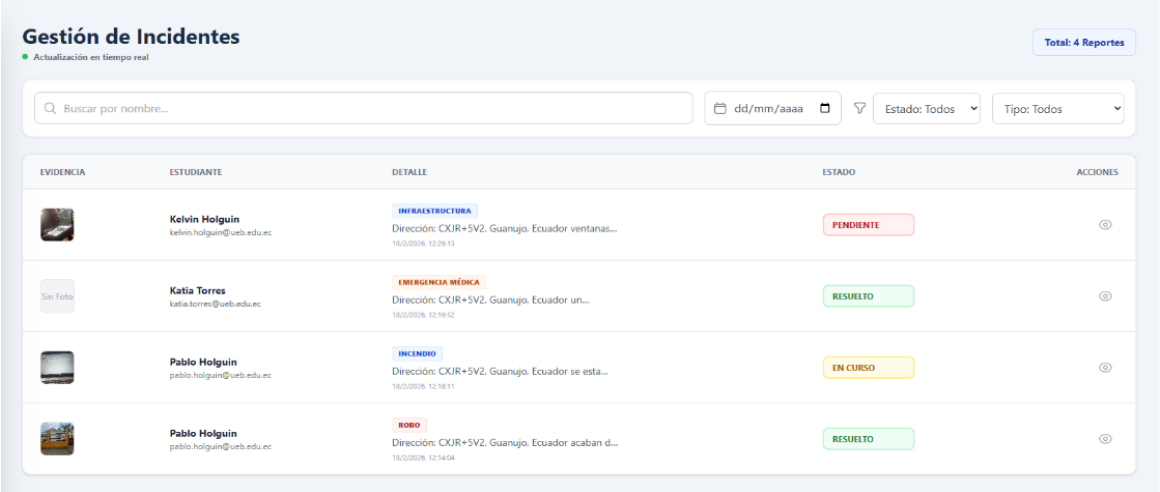
Nota: Elaboración propia

Gestión de Incidentes

El sistema permite gestionar los reportes de incidentes, la interfaz presenta el listado detallado con filtros de búsqueda por nombres y fechas cuenta con un opción para cambiar estado del reporte, además de tener un botón de acción para abrir una ventana del detalle

del reporte el cual permite visualizar la evidencia fotográfica adjunta por el usuario, la descripción detallada, junto con la ubicación.

Figura 26 *Gestión de Incidentes*



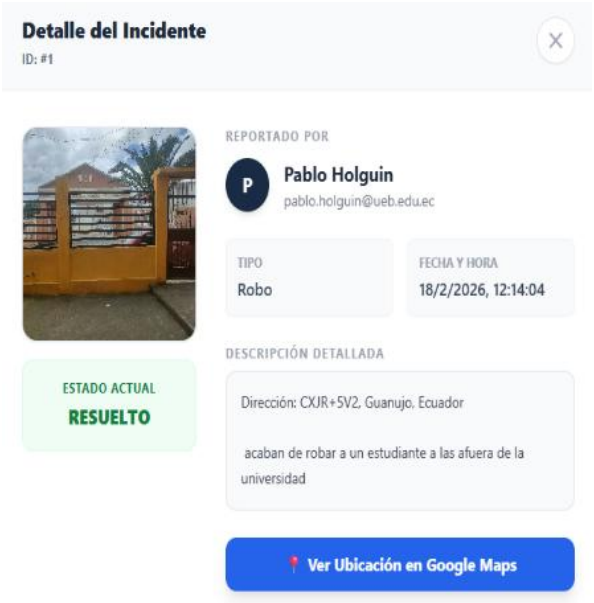
Nota: Elaboración propia

Figura 27 *Cambiar Estado Gestión de Incidentes*



Nota: Elaboración propia

Figura 28 *Detalle de Incidente*



Nota: Elaboración propia

Figura 29 *Ubicación del Incidente*



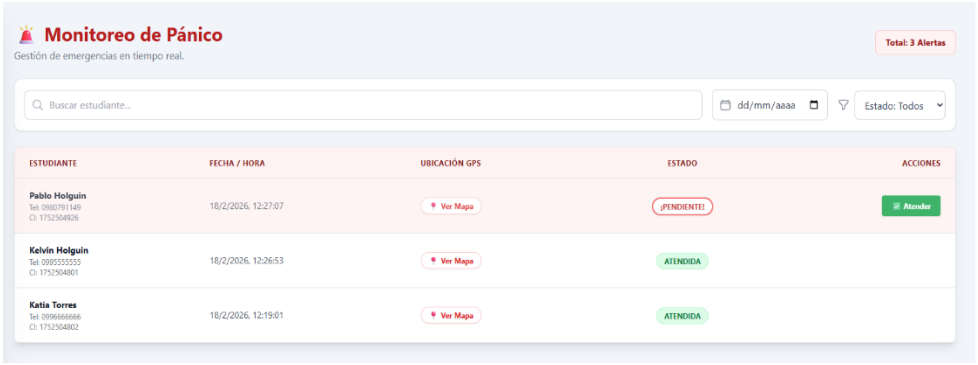
Nota: Elaboración propia

Monitor de Alertas de pánico

Esta interfaz permite la gestión de emergencias críticas. La tabla se actualiza automáticamente cada 10 segundos para mostrar las activaciones del botón de pánico. Al recibir una nueva alerta, el Sistema emite una alarma y resalta la fila correspondiente.

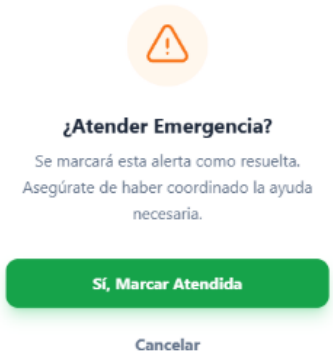
Incluye botón de acción rápida para atender la emergencia y un botón para ver el mapa con la geolocalización exacta del del usuario.

Figura 30 *Monitoreo de Pánico*



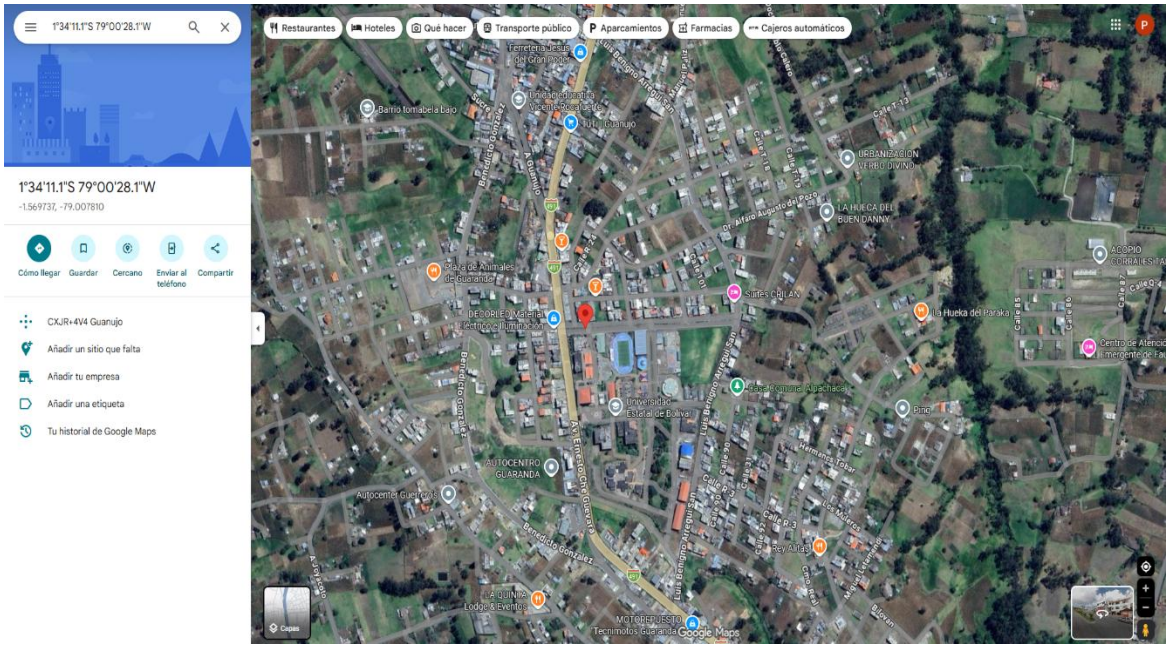
Nota: Elaboración propia

Figura 31 *Atender Emergencia*



Nota: Elaboración propia

Figura 32 *Ubicación de Emergencia de Pánico*

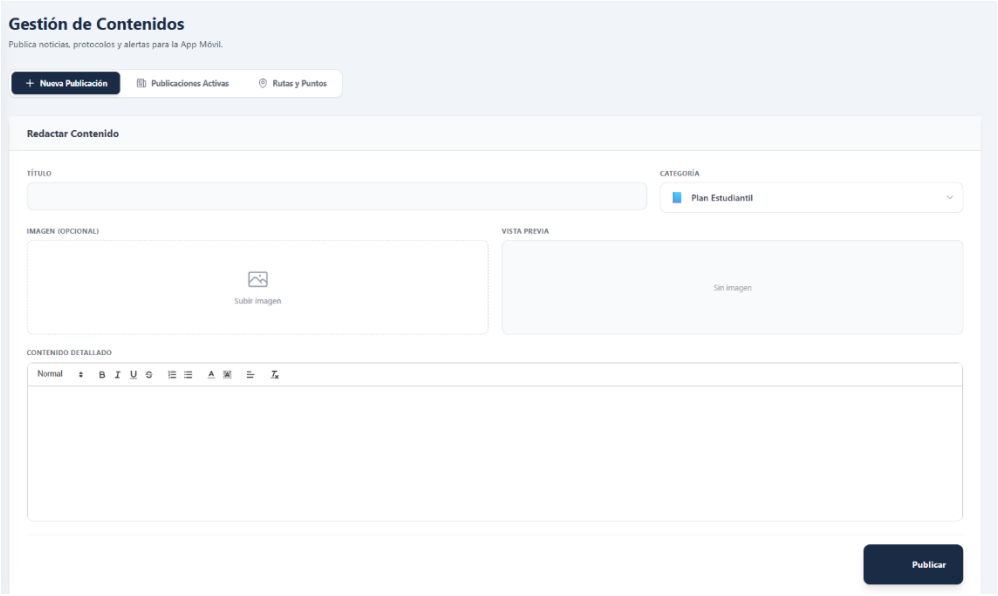


Nota: Elaboración propia

Gestión de Contenidos

Interfaz para publicación de contenidos cuenta con 3 pestañas (nueva publicación, publicaciones Activas, puntos de encuentros y zonas de peligros). En la sección nueva publicación cuenta con un formulario para redactar un nuevos contenido el cual consta del título, categoría, contenido e imágenes al final el botón para publicarlo.

Figura 33 *Gestión de Contenidos*



Nota: Elaboración propia

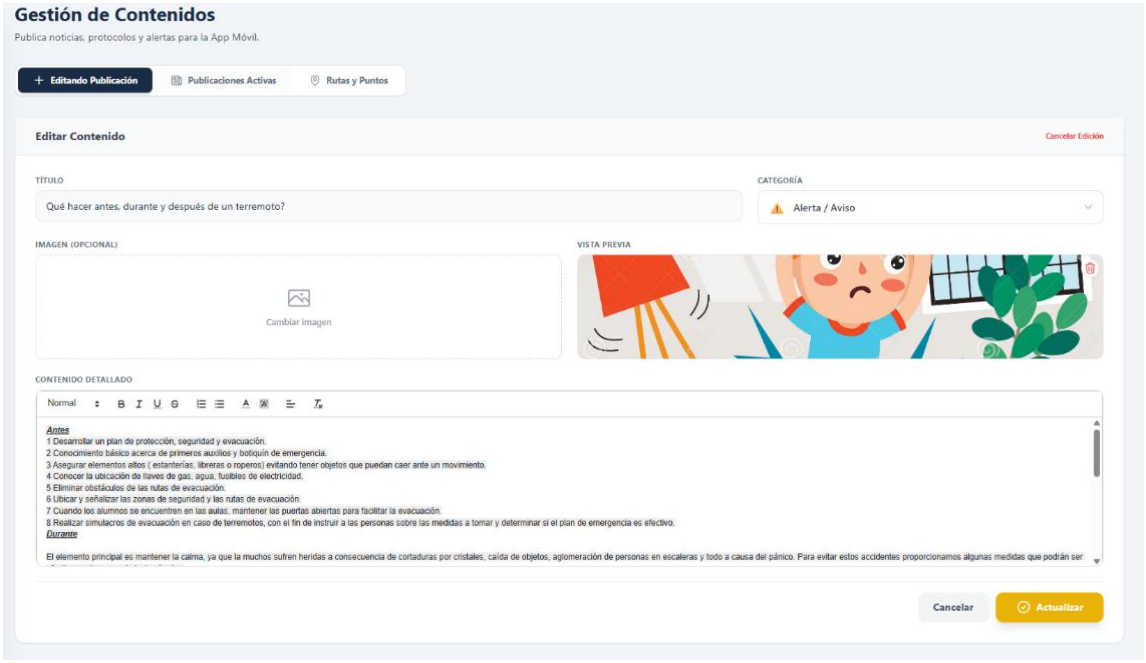
La sección de publicaciones activas cuenta con una interfaz para editar o eliminar las publicaciones activas.

Figura 34 *Sección de Publicaciones*



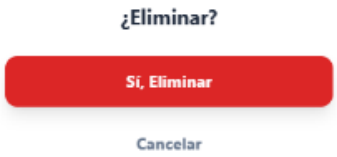
Nota: Elaboración propia

Figura 35 *Sección de Publicaciones Editar*



Nota: Elaboración propia

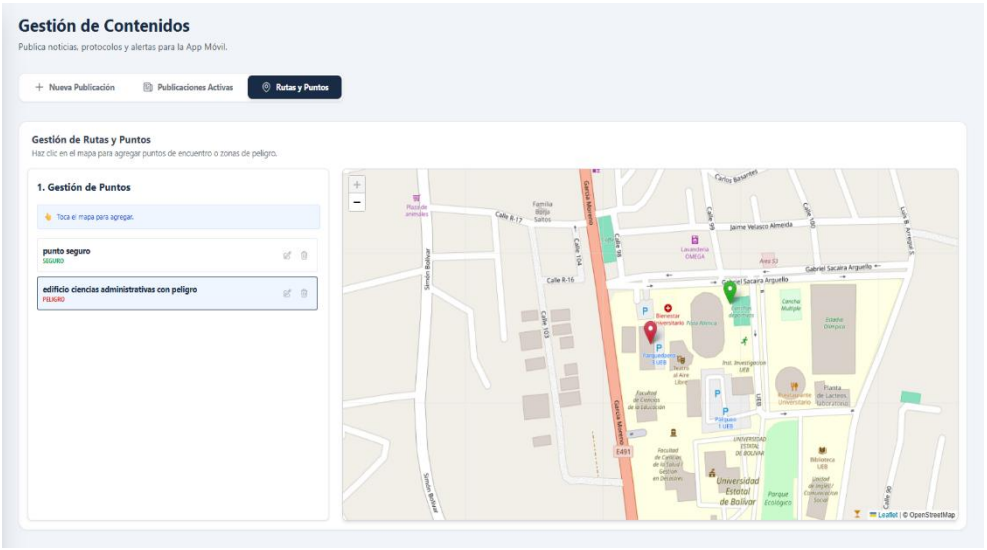
Figura 36 *Sección de Publicaciones Eliminar*



Nota: Elaboración propia

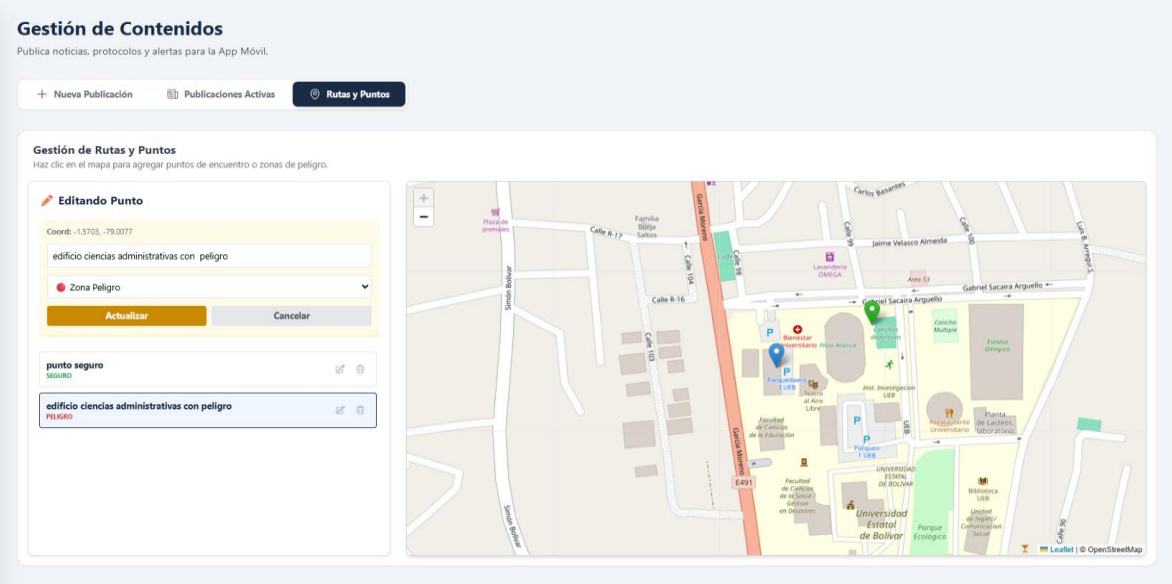
La sección de puntos de encuentro y zonas de peligro contas de una herramienta interactiva desarrollada con leaflet que permite al administrador gestionar la seguridad espacial del campus, la interfaz permite arrastrar y soltar marcadores para definir visualmente la zonas de peligro (marcador rojo) o puntos de encuentro (marcador verde) estas ubicaciones se guardan en la base de datos para ser visualizadas posteriormente en la aplicación así mismo permite editar y eliminar.

Figura 37 *Sección Puntos de Encuentro y Zonas de Peligro*



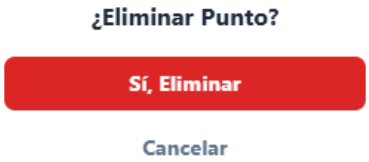
Nota: Elaboración propia

Figura 38 *Editar Puntos de Encuentro y Zonas de Peligro*



Nota: Elaboración propia

Figura 39 *Eliminar Puntos de Encuentro y Zonas de Peligro*

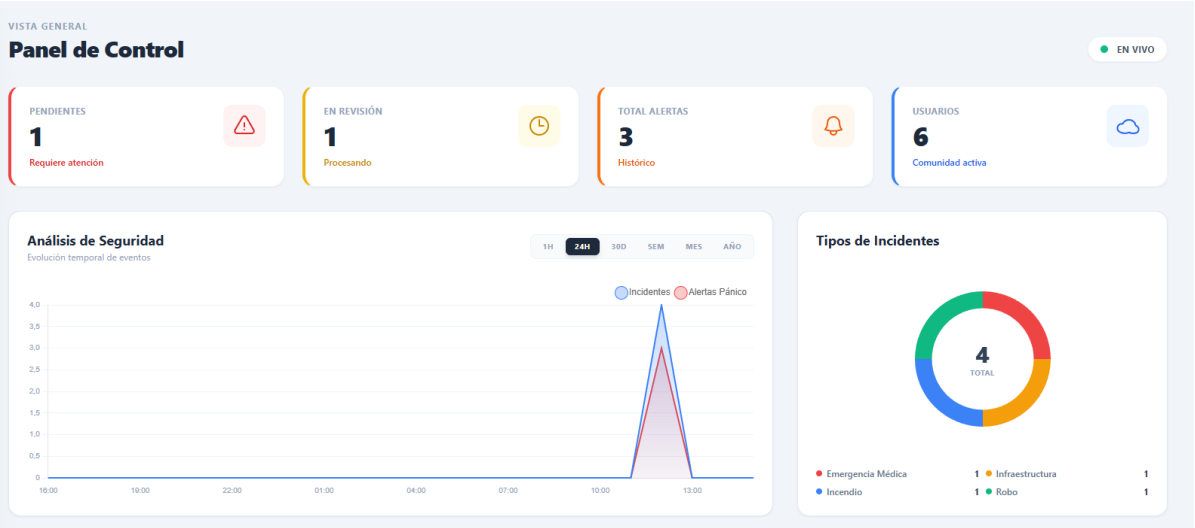


Nota: Elaboración Propia

Configuración y personalización de cuenta

Este módulo permite al usuarios autenticado gestionar su información personal y la seguridad de su sesión. A través de un modal de configuración accesible desde el menú.

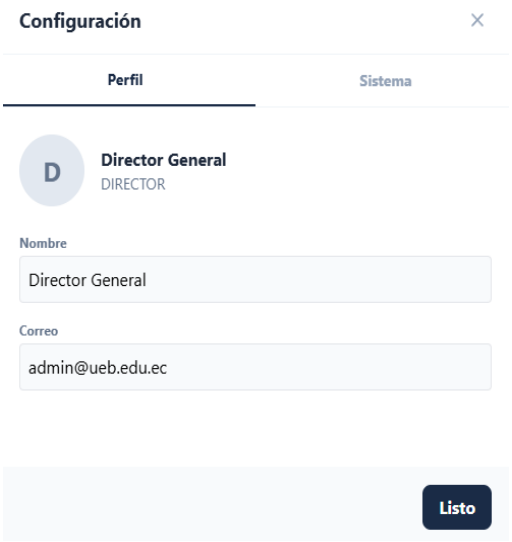
Figura 40 *Configuración y Personalización de Cuenta*



Nota: Elaboración propia

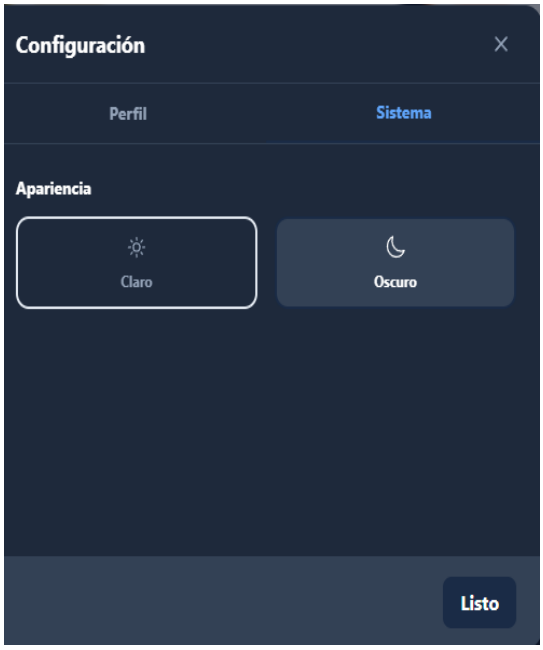
El usuario podrá actualizar sus datos de perfil, así como la apariencia del sistema. Además, se integró la funcionalidad de cierre de sesión la cual se encarga de invalidar el token de acceso al servidor y limpiar el almacenamiento local del navegador, redirigiendo al usuario a la pantalla de inicio para prevenir accesos no autorizados posteriores.

Figura 41 *Actualización de Datos*



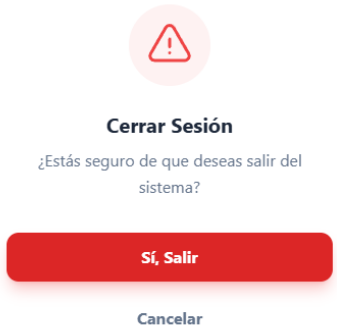
Nota: Elaboración propia

Figura 42 *Configuración Apariencia de Sistema*



Nota: Elaboración Propia

Figura 43 *Ventana para Cerrar Sesión*



Nota: Elaboración propia

SPRINT 3 DESARROLLO DEL APLICATIVO MOVIL

4.6. Planificación del Sprint

En este sprint, el equipo se enfocó en la construcción de la aplicación nativa para Android, priorizando la geolocalización, el reporte de incidentes y la recepción de notificaciones push.

4.6.1. Objetivo del Sprint

Desarrollar una aplicación móvil nativa que permita a la comunidad universitaria autentificarse de forma segura, activar alertas de pánico con geolocalización en tiempo real, genera reportes de incidentes con evidencias multimedia y recibir notificaciones

institucionales, garantizando la sincronización de datos con el servidor central mediante una API REST.

4.6.2. Actividades Principales

Las actividades planificadas y realizadas durante este sprint fueron:

- Diseño del formulario de inicio de sesión y registro que valida las credenciales contra la api institucional
- Implementación de almacenamiento local de sesión para evitar logueos repetitivos.
- Pantalla principal home con diseño intuitivo y accesos directos a las funciones críticas.
- Botón de pánico que activa el envió de ayuda inmediata.
- Servicio de geolocalización en segundo plano para obtener coordenadas exactas del dispositivo.
- Formulario para redactar reportes de seguridad seleccionando la categoría del incidente.
- Integración con la cámara y galería del teléfono para adjuntar evidencia fotográfica en los reportes.
- Listado de noticias donde la comunidad universitaria visualiza los comunicados oficiales.
- Pantalla de detalle de noticia para leer la información completa y ver las imágenes adjuntas.
- Sección de configuración Mis Reportes para consultar el historial y estado de las denuncias realizadas.
- Integración de mapas interactivos (OpenStreetMap) para visualizar la ubicación del usuario en tiempo real.
- Sistema de notificaciones Push (Firebase) para recibir alertas incluso cuando la aplicación está cerrada.
- Pantalla de "Sin Conexión" para gestionar la experiencia de usuario cuando falla el internet.
- Interfaz de perfil de usuario para visualizar los datos personales del estudiante o docente.
- Proceso de cierre de sesión seguro que elimina las credenciales almacenadas y redirige al inicio.

4.6.3. Sprint Backlog

Se establecieron las siguientes historias de usuario del aplicativo móvil para ser implementadas dentro de este sprint:

Tabla 18 *Historia de Usuario Móvil*

Código	Historia de usuario	Descripción	Prioridad
HU007	Autentificación Segura	La comunidad universitaria, inicia sesión de manera segura utilizando las credenciales institucionales para acceder a las funcionalidades de la aplicación.	Alta
HU008	Reportes de incidentes	La comunidad universitaria enviara un reporte detallado de un incidente o condición de riesgo que este observando, incluyendo la ubicación, para que la Unidad de Gestión de Riesgos pueda actuar.	Alta
HU009	Activación de alertas de pánico.	La comunidad universitaria activara una alerta de pánico de manera rápida, preferiblemente en un solo toque, para solicitar ayuda de emergencia de forma inmediata a la Unidad de Gestión de Riesgos.	Alta
HU010	Consulta de material de Seguridad y notificaciones	La Comunidad universitaria accederá fácilmente a las recomendaciones, mochila de emergencia, protocolos de seguridad, alertas, plan estudiantil, puntos de encuentros y zonas de peligro emitidos por la administración en una sección dedicada para mantener informado y saber cómo actuar en diferentes situaciones.	Alta
HU011	Configuración general	La Comunidad universitaria accederá fácilmente a las configuraciones generales y cierre de sesión.	Media

Nota: Elaboración propia

4.6.4. Desarrollo diario (Daily Scrum)

Durante la ejecución del sprint, se llevaron a cabo reuniones breves con el equipo de desarrollo. Estas reuniones permitieron dar avances del sprint y detectar impedimentos o posibles obstáculos permitiendo tomar decisiones para asegurar el cumplimiento del objetivo.

Bajo este desarrollo, el equipo abordó la construcción del aplicativo móvil enfocándose en la integración de componentes nativos de Android.

4.6.5. Revisión del Sprint (Sprint Review)

Al finalizar el sprint, se realizó la demostración del aplicativo móvil instalado en un dispositivo físico ante el director del proyecto para validar el cumplimiento de las actividades planteadas, Durante esta sesión, se presentó el incremento funcional del sistema verificando el correcto funcionamiento de los módulos implementados.

A continuación, se detallan los entregables:

Tabla 19 *Ejecución del Sprint 3*

Código	Historia de usuario	Funcionalidad desarrollada	Resultado	Observaciones	Estado
HU007	Autenticación segura	Inicio de sesión y gestión de tokens.	Módulo de acceso móvil	Se verificó que la app valida el correo institucional y mantiene la sesión activa al reiniciar el dispositivo, bloqueando el acceso si las credenciales son erróneas.	Completado
HU008	Reportes de incidentes	Formularios de denuncias o reportes con evidencias fotográficas y ubicación GPS.	Gestor de reportes	El módulo permite seleccionar el tipo de incidente, adjuntar la ubicación, adjuntar una foto de cámara galería, incluyendo la ubicación	Completado

				GPS y una breve detalle del reporte.	
HU009	Activación de alertas de pánico	Botón de emergencia con envío de ubicación GPS	Sistema de alertas tempranas	Al pulsar el botón de evacuación, la alerta llega al panel administrativo con la ubicación exacta en menos de 3 segundos cumpliendo los requisitos de inmediatez.	Completado
HU010	Consulta de material de Seguridad y notificaciones	Feed de noticias y sistema de alertas push.	Canal de Comunicación oficial	Las noticias se visualizan correctamente con sus imágenes. Las notificaciones de prueba llegaron al dispositivo instantáneamente incluso con la app cerrada.	Completado
HU011	Configuración general	Visualización de perfil, y configuraciones generales	Gestión de perfil, ajustes, y sesión	La interfaz muestra el perfil del usuario, el historial de reportes, historial de alertas, ajustes y notificaciones, soporte, y el botón de cerrar sesión que limpia los datos locales, redirigiendo al login.	Completado

Nota: Elaboración propia

4.6.6. Retrospectiva del Sprint (Sprint Retrospective)

El equipo de desarrollo realizó una sesión de análisis para evaluar el proceso de trabajo durante el sprint 3.

Fortalezas:

- El uso de estas tecnologías modernas permitió desarrollar interfaces complejas con menos código y mayor estabilidad.
- La implantación de OpenStreetMap resulto exitosa, permitiendo visualizar la ubicación del usuario y los puntos de seguridad sin depender de licencias de pago de Google maps.

Obstáculos:

- Se presentaron dificultades para gestionar los permiso de ubicación en tiempo de ejecución en versiones recientes de Android, lo que requirió ajustes específicos en el AndroidManifest.
- La conversión de las fotografías capturadas por la cámara a un formato adecuado para la subida al servidor genero errores iniciales de memoria, que se solucionaron optimizando el proceso de compresión.

4.6.7. Resultados del Sprint 3

A continuación, se presentan las interfaces graficas desarrolladas durante este sprint, describiendo la funcionalidad implementada en el Aplicativo Móvil.

Módulo de Alerta de Pánico

Interfaz diseñada para situaciones de emergencia. Al presionar el botón nos pedirá acceso a la ubicación y nos dirige al encender el GPS, una vez encendido regresamos y nos mostrará una notificación que confirma el envío de la alerta. Internamente el sistema captura las coordenadas GPS y las envía al centro de monitoreo, además de mostrarnos una notificación en el inicio de confirmación de ayuda.

Figura 44 Modulo de Alerta de Pánico



Nota: Elaboración propia

Figura 45 *Permisos de Ubicación*



Nota: Elaboración propia

Figura 46 Notificación de Confirmación de Ayuda



Nota: Elaboración propia

Figura 47 *Confirmación de Ayuda en Camino*



Nota: Elaboración propia

Modulo reporte de incidente

Formulario que permite al usuario documentar un suceso. Incluye selectores para el tipo de incidente, ubicación del incidente, botón para activar cámara o galería, permitiendo adjuntar evidencia fotográfica un campo de texto para la descripción o detalle del incidente y el botón para el envío del reporte.

Figura 48 *Modulo de Reporte de Incidente*



Nota: Elaboración propia

Figura 49 Botón Activar Cámara



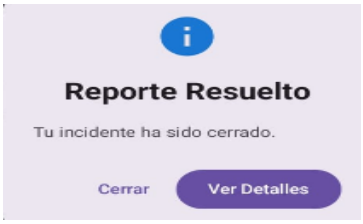
Nota: Elaboración propia

Figura 50 *Estado del Reporte*



Nota: Elaboración propia

Figura 51 *Detalle del Reporte*



Nota: Elaboración propia

Módulo de noticias y comunicados

Módulo informativo integral diseñado para centralizar la difusión de material preventivo y de actuación. Esta interfaz generaliza la gestión de contenidos, permitiendo a la comunidad universitaria acceder a protocolos de seguridad, el plan estudiantil, guías de la mochila de emergencia, recomendaciones y alertas vigentes. La información se presenta mediante tarjetas visuales que, al ser seleccionadas, despliegan el detalle completo con imágenes ilustrativas, garantizando que el estudiante tenga a la mano toda la información necesaria para actuar ante una emergencia.

Figura 52 *Protocolos de Seguridad*



Nota: Elaboración propia

Figura 53 *Detalle del Protocolo de Seguridad*



Nota: Elaboración propia

Modulo mapa de puntos de encuentro y zonas de peligro

Interfaz gráfica con una herramienta visual basada en OpenStreetMap que muestra las zonas de peligro y puntos de encuentro del campus universitario, definidos por la Unidad de Gestión de Riesgos.

Figura 54 *OpenStreetMap Zonas de Peligro y Puntos de Encuentro*



Nota: Elaboración propia

Módulo de Configuración general y perfil

Interfaz con visualización de perfil y configuraciones generales como el historial de reportes, historial de alertas, ajustes y notificaciones, ayuda y soporte, y por último el botón de cerrar sesión, el cual elimina el token de acceso del dispositivo y redirige a la pantalla de inicio, protegiendo la plataforma de usos no autorizados.

Figura 55 *Configuración General*



Nota: Elaboración propia

Figura 56 *Visualización de Perfil*



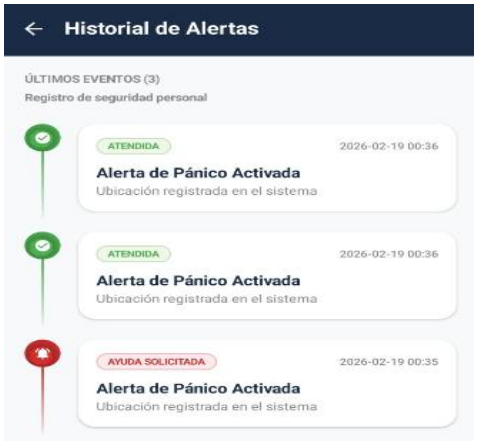
Nota: Elaboración propia

Figura 57 *Historial de mis Reportes*



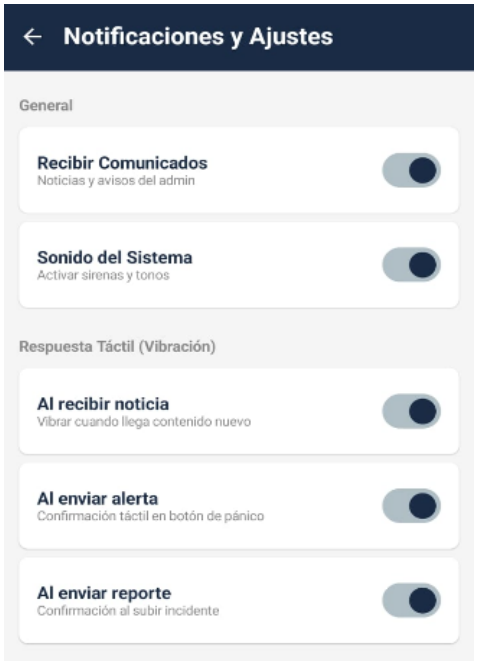
Nota: Elaboración propia

Figura 58 *Historial de Alertas*



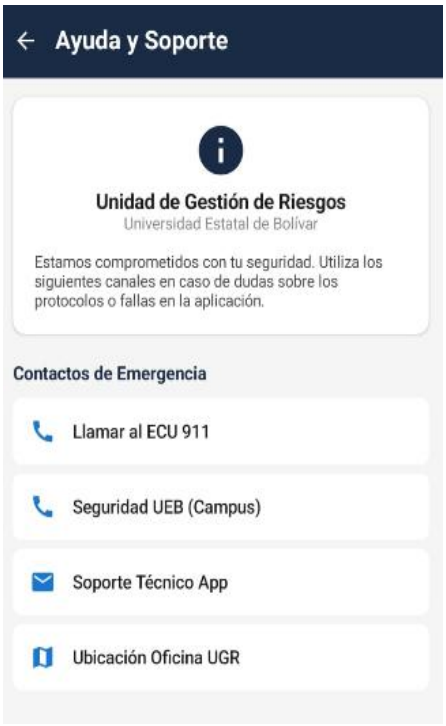
Nota: Elaboración propia

Figura 59 *Notificaciones y Ajustes*



Nota: Elaboración propia

Figura 60 *Ayuda y Soporte*



Nota: Elaboración propia

SPRINT 4 IMPLEMENTACIÒN, PRUEBAS Y ENTREGAS

4.7. Planificación Sprint

Para este último sprint, el equipo scrum se reunió con los stakeholders para definir la estrategia de transición del entorno de desarrollo al entorno de producción. Se priorizó la configuración de servidores, la compilación de la aplicación móvil y la ejecución de un plan de prueba exhaustivo para garantizar la calidad de producto final antes de su entrega a la Unidad de Gestión de Riesgos.

4.7.1. Objetivo del Sprint

Desplegar el panel administrativo en un servidor web de producción, generar el instalador APK del aplicativo móvil, ejecutar pruebas integrales y capacitar al personal administrativo para la entrega oficial del sistema de Seguridad Integral.

4.7.2. Actividades Principales

Las actividades planificadas y realizadas durante este sprint fueron:

- Configuración del servidor web de producción para el despliegue del backend y la base de datos.
- Compilación y despliegue del panel administrativo en el servidor.

- Generación del archivo APK.
- Ejecución de pruebas de funcionalidad.
- Realización de pruebas de rendimiento.
- Validación de seguridad, cifrado de datos y protección de endpoints.
- Capacitación al director de la Unidad y a los administradores sobre el uso del sistema.
- Elaboración y entrega del manual de usuario.

4.7.3. Sprint Backlog

Se establecieron loas siguientes tareas técnicas para ser implementadas dentro de este sprint de cierre.

Tabla 20 *Backlog Sprint Cierre*

Código	Tarea	Descripción	Prioridad	Estado
T01	Despliegue Backend	Configuración del servidor, migración de base de datos y despliegue de la API REST en producción.	Alta	Pendiente
T02	Despliegue Fronted	Compilación del código de React y publicación del panel web administrativo.	Alta	Pendiente
T03	Generación de APK	Compilación de la aplicación móvil Android para la comunidad universitaria.	Alta	Pendiente
T04	Ejecución de pruebas	Realización de pruebas de rendimiento, usabilidad, funcionalidad y seguridad	Alta	Pendiente
T05	Capacitación	Entrenamiento al personal de la UGR y entrega de manuales	Media	Pendiente

Nota: Elaboración propia

4.7.4. Desarrollo diario (Daily Scrum)

Durante la ejecución del sprint 4, las reuniones se centraron en monitorear la estabilidad de servidor tras el despliegue. Estas breves sesiones permitieron al equipo identificar y corregir rápidamente problemas menores de configuración de red y asegurar que la conexión entre el aplicativo móvil y el servidor fueran constante y sin latencia.

4.7.5. Revisión del Sprint (Sprint Review)

Al finalizar el sprint, se realizó la presentación final del sistema operando al 100% ante el Director de la Unidad de Gestión de Riesgos. Se simuló un escenario real de emergencia donde un estudiante emitió una alerta de pánico desde su teléfono móvil y el panel web administrativo recibió la notificación y la ubicación GPS en menos de 3 segundos.

El sistema fue validado y aceptado en su totalidad.

4.7.6. Retrospectiva del Sprint (Sprint Retrospective)

El equipo de desarrollo realizo la última sesión de análisis para evaluar el proceso de implementación.

Fortalezas:

- La separación de la arquitectura (Cliente-Servidor) facilito enormemente el despliegue, ya que el panel web y la API se pudieron configurar de manera independiente sin afectar el rendimiento. La capacitación al personal fue rápida gracias a la interfaz intuitiva lograda en el Sprint 2.

4.7.7. Resultados del Sprint 4

El sistema de Seguridad Integral se encuentra operando en un entorno de producción establece. El personal de la Unidad de Gestión de Riesgos posee las credenciales maestras y cuenta con las competencias necesarias para administrar usuarios, contenidos y coordinar emergencias.

4.8. Pruebas

4.8.1. Pruebas de Rendimiento

Las pruebas de rendimiento permiten realizar la evaluación de la eficiencia, velocidad de respuesta y estabilidad del sistema. Para el desarrollo de esta prueba y analizar el rendimiento, se empleó la herramienta Apache JMeter, la cual permitió evaluar métricas claves como la velocidad de respuesta de la API REST bajo concurrencia. El objetivo primordial del desarrollar estas pruebas es validar que el sistema mantenga tiempos óptimos en situaciones de emergencia.

Tabla 21 *Pruebas de Rendimiento*

Nombre de prueba	Método	Resultado	Tiempo de respuesta (ms)	Descripción
Interfaz del panel administrativo.	GET	Aprobado	25	Valida que el dashboard del personal de la UGR esté disponible y cargue los recursos correctamente.
Endpoint de Reporte de incidentes	POST	Aprobado	180	Verifica que la subida de datos e imágenes multimedia desde el aplicativo móvil funcione sin cuellos de botella.
Activación de Alerta de Pánico	POST	Aprobado	45	Comprueba la latencia del servicio de red para notificar una emergencia al administrador en tiempo real.
Listado de Usuarios e Historial	GET	Aprobado	35	Valida que la consulta a la base de datos PostgreSQL retorne los registros de manera eficiente.

Nota: Elaboración propia

4.8.2. Pruebas de Usabilidad

Las pruebas de usabilidad se enfocaron en medir la facilidad de uso, la claridad de la interfaz y la experiencia de usuario al momento de interactuar con las plataformas. Se consideraron criterios como efectividad, consistencia y accesibilidad.

Tabla 22 *Pruebas de Usabilidad*

Prueba de Usabilidad				
Responsable: Pablo Holguin, Nataly Silva		**Nivel de satisfacción**		
No	**Criterio evaluado**	**Bajo**	**Medio**	**Alto**
01	Eficiencia de Uso			X
02	Efectividad			X
03	Consistencia			X
04	Aprendizaje			X
05	Memorabilidad			X
06	Accesibilidad			X
07	Errores y recuperación		X	
08	Navegación		X	

Nota: Elaboración propia

4.8.3. Prueba de Funcionalidad

Este tipo de pruebas se orienta a comprobar que las funciones del sistema respondan correctamente a los requerimientos establecidos y operen según lo previsto. Las validaciones se ejecutaron en base a las historias de usuarios desarrolladas en los Sprints 2 y 3

Tabla 23 *Pruebas de Funcionalidad*

Prueba de Funcionalidad			
Responsable: Pablo Holguin, Nataly Silva	**Sprint 2**		
	Evaluación de satisfacción		
Historias de Usuarios	**Bajo**	**Medio**	**Alto**
Autentificación de Usuario			X

Administración de usuarios y asignación de roles	X
Visualización de estadísticas de reportes	X
Atención y respuesta a incidentes y alertas de pánico	X
Publicación de contenidos	X
Personalización de cuenta	X

Nota: Elaboración propia

Tabla 24 *Prueba de Funcionalidad Sprint 3*

Prueba de Funcionalidad			
Responsable: Pablo Holguin, Nataly Silva	**Sprint 3**		
	Evaluación de Satisfacción		
Historias de Usuarios	**Bajo**	**Medio**	**Alto**
Autentificación Segura			X
Reporte de incidente			X
Activación de alertas de pánico			X
Consulta de material de Seguridad y notificaciones			X
Configuración general			X

Nota: Elaboración propia

4.8.4. Pruebas de Seguridad

Este tipo de prueba está diseñada para garantizar la confidencialidad, integridad y disponibilidad del sistema, especialmente considerando la sensibilidad de los datos manejados (geolocalización en tiempo real y reportes de emergencias). En lugar de un escaneo genérico, las pruebas se enfocaron en la validación de la arquitectura API REST y la robustez del Backend (Laravel) y los clientes (React/Android) ante intentos de vulneración intencionados utilizando herramientas como Postman.

Tabla 25 *Pruebas de Seguridad*

Vulnerabilidad	Nivel de severidad	Descripción de la prueba	Resultados
Ataque de fuerza bruta y spam	Alto	Se intento enviar múltiples peticiones de alertas de pánico para intentar saturar el servidor y la base de datos	**Aprobado** Se implemento limitaciones de peticiones nativo de Laravel. Este servidor bloquea automáticamente la IP del usuario si detecta peticiones anómalas repetitivas, protegiendo el panel de falsas alarmas masivas
Inyección SQL	Alto	Se ingresaron comandos maliciosos de SQL en los formularios de inicio de sesión y en la descripción de los reportes de incidentes de la app móvil	Aprobado El sistema utiliza Eloquent ORM de laravel, el cual emplea enlaces de parámetros PDO. El código malicioso es tratado como texto simple, neutralizando por completo el ataque
Escalada de Privilegios y acceso a Endpoints+	Alto	Se simuló ser un usuario con rol de "Comunidad Universitaria" e intentar acceder (vía Postman con un token válido) a las rutas de la API de creación de administradores y edición de noticias.	Aprobado El sistema de middlewares de laravel valido correctamente los roles. El servidor denegó el acceso devolviendo un código 403, confirmando que la barrera entre roles es sólida.

Nota: Elaboración propia

CONCLUSIONES

- Se cumplió con el desarrollo integral de la aplicación móvil de seguridad para la Unidad de Gestión de Riesgos de la Universidad Estatal de Bolívar. Esta plataforma soluciona la carencia de un sistema oficial digitalizado, centralizando el monitoreo, empoderando a la comunidad estudiantil y fortaleciendo una cultura institucional enfocada en la prevención.

- Se logro analizar a profundidad las necesidades de la Unidad de Gestión de Riesgos mediante técnicas de observación y entrevista dirigidas. Esto permitió estructurar requerimientos precisos en forma de historias de usuario, garantizando que el diseño del sistema atienda directamente las deficiencias comunicacionales y la falta de registros estadísticos formales frente a emergencias institucionales.

- Se diseñaron con éxito las funcionalidades principales bajo una arquitectura Cliente-Servidor desacoplada, seleccionando tecnologías altamente eficientes para el sistema móvil. El uso de Kotlin y el patrón MVVM para la aplicación nativa en Android aseguró una interfaz fluida e intuitiva, mientras que el soporte backend permitió establecer módulos funcionales robustos como la alerta de pánico inmediata y la centralización de contenidos.

- La aplicación móvil se implementó satisfactoriamente siguiendo las iteraciones del marco de trabajo Scrum. El producto final dota a la comunidad universitaria de una herramienta fiable, capaz de emitir alertas geolocalizadas en tiempo real y reportar incidentes con evidencia multimedia, cumpliendo asi con los estándares de calidad esperados y optimizando sustancialmente los tiempos de respuesta ante eventualidades.

RECOMENDACIONES

- Se recomienda a la Unidad de Gestión de Riesgos establecer un proceso de evaluación periódica del sistema para continuar analizando nuevos requerimientos operativos. Es fundamental recopilar retroalimentación constante de la comunidad estudiantil y docente para adaptar la plataforma frente a futuros cambios en los protocolos de actuación o en la infraestructura del campus.

- Dado que el alcance inicial se centró en diseñar funcionalidades para plataformas Android, se sugiere planificar a mediados plazos el desarrollo de una versión nativa para el sistema operativo iOS. Esto garantizará que la totalidad de la comunidad universitaria tenga acceso a la herramienta de seguridad, manteniendo la misma arquitectura de conexión con el servidor principal.

- Para asegurar la aceptación y el uso correcto de la aplicación móvil recién implementada, se aconseja ejecutar campañas masivas de socialización y realizar simulacros de evacuación o en los que se utilice el botón de pánico de la herramienta. Adicionalmente, el personal administrativo debe mantener rutinas de actualización de software y de copias de seguridad de las bases de datos para salvaguardar la disponibilidad del servicio.

- Se sugiere para futuras fases del proyecto evaluar la viabilidad técnica de integrar la plataforma con servicios de emergencia externos a la universidad, como el Ecu 911. Dado que el alcance inicial determinó que las alertas y reportes se manejan de forma exclusivamente interna por la Unidad de Gestión de Riesgos, crear un canal de interoperabilidad o comunicación automatizada con entidades de seguridad pública permitirá una capacidad de respuesta mucho más robusta ante incidentes críticos o delictivos que superen las competencias de la institución.

BIBLIOGRAFÍA

Asamblea Constituyente del Ecuador. (2008). *Constitución de la República del Ecuador*. Quito: Resgistro Oficial 449.

Asamblea Nacional del Ecuador . (2009). *Ley de Seguridad Públiva y del Estado.* Registro Oficial 35 al 28.

Astera. (05 de marzo de 2025). *¿Qué es una base de datos? Una guía completa*. Obtenido de https://www.astera.com/es/type/blog/what-is-a-database/

Ayala, M., & Sigcha, M. (08 de 2022). *Aplicación móvil con administración web para la seguridad comunitaria de la ciudad de Latacunga.* Obtenido de Universidad Técnica de Cotopaxi: https://repoadmin.utc.edu.ec/items/88550c44-5fb3-4ef0-9f18-35020fb68f74

Cachipuendo, E. (2019). *Diseño e implementación de una aplicación móvil para alertar una emergencia de personas vulnerables sobre un teléfono inteligente.* Obtenido de Universidad de las Américas: https://dspace.udla.edu.ec/handle/33000/11586

Chancas, G. (2021). *Introduccion A La Seguridad Integral e Importancia.* Obtenido de https://es.scribd.com/document/673031283/Introduccion-a-La-Seguridad-Integral-e-Importancia

Cristancho, F. (26 de julio de 2022). *¿Qué es Flutter?* Obtenido de talently: https://talently.tech/blog/que-es-flutter/

GoDaddy. (28 de septiembre de 2023). *Laravel: Qué es y para qué sirve*. Obtenido de https://www.godaddy.com/resources/latam/stories/laravel-que-es-para-que-sirve

Gomila, J. (13 de 10 de 2023). *¿Qué es el PHP? Definición, Historia, Características y Recursos Destacados*. Obtenido de frogames: https://cursos.frogamesformacion.com/pages/blog/que-es-el-php

Goodwin, M. (09 de 04 de 2024). *¿Qué es una API?* Obtenido de IBM: https://www.ibm.com/es-es/think/topics/api

Hashemi, C. (06 de junio de 2024). *DFINICIÓN Software*. Obtenido de TechTarget: https://www.techtarget.com/searchapparchitecture/definition/software

Hernández, U. (22 de febrero de 2015). *MVC (Model, View, Controller) explicado.* Obtenido de codigofacilito: https://codigofacilito.com/articulos/mvc-model-view-controller-explicado

Loarte, B. (2024). Desarrollo de una guía metodológica para el desarrollo de aplicaciones del lado del servidor. *Ciencia Digital, Vol.8*(No.4), 54-74. Obtenido de

https://cienciadigital.org/revistacienciadigital2/index.php/CienciaDigital/article/view/3206

Loayan, S. (08 de enero de 2025). *¿Qué es la Metodología Agile y cómo revoluciona la gestión de proyectos?* Obtenido de asana: https://asana.com/es/resources/agile-methodology

Loja, C., & Márquez, J. (2023). *Desarrollo de una aplicación móvil multiplataforma para el proyecto ESPOL Alert.* Obtenido de Escuela Superiro Politécnica del Litoral: https://dspace.espol.edu.ec/handle/123456789/58454

López, S. (17 de 05 de 2020). *Firebase: qué es, para qué sirve, funcionalidades y ventajas*. Obtenido de DIGITAL55: https://digital55.com/blog/que-es-firebase-funcionalidades-ventajas-conclusiones/

Lucena, P. (05 de mayo de 2023). *Tecnologías de Información: ¿Qué es el framework?* Obtenido de Universidad Cesuma: https://www.cesuma.mx/blog/que-es-el-framework.html

Maida, E., & Pacienzia, J. (2015). *Metodologías de desarrollo de software.* Obtenido de Universidad Católica de Argentina: https://repositorio.uca.edu.ar/bitstream/123456789/522/1/metodologias-desarrollo-software.pdf

Montenegro, A. (12 de febrero de 2024). *Como funciona la Metodología Sprint*. Obtenido de Función Pública: https://www.funcionpublica.gov.co/eva/red/publicaciones/funciona-metodologia-sprint

Mora.Melo, M. (2022). *Desarrollo de una aplicación para celular que permita dar aviso de emergencias de salud y seguridad.* Obtenido de Universidad Católica de Colombia: https://redcol.minciencias.gov.co/Record/UCATOLICA2_1b6cef89c313da247e78e165e9c6630d

Nava, J. (18 de NOVIEMBRE de 2024). *¿Qué son Stakeholders? tipos existen y sus características*. Obtenido de EMPOWERTALENT: https://empowertalent.com/que-son-stakeholders/#:~:text=%C2%BFQu%C3%A9%20son%20los%20stakeholders?,de%20negocio%20y%20tomar%20decisiones.

Pérez, J. (20 de agosto de 2021). *Android - Qué es, características, definición y concepto*. Obtenido de Definición.DE: https://definicion.de/android/

Rodríguez, K. (2023). *Scrum: Metodología ágil*. Obtenido de UNICISO: https://www.uniciso3.com/info/SCRUM.pdf

Schwaber, K., & Sutherland, J. (noviembre de 2020). *La Guía Definitiva a Scrum: Las reglas del juego*. Obtenido de jeronimopalacios: https://repositorio.uvm.edu.ve/handle/123456789/59

Secretaria de Gestión de Riesgos . (2014). *Reglamento del Sistema Nacial Descentralizado de Gestión de Riesgos.* Registro Ofical 334.

Universidad Estatal de Bolivar. (s.f.). *Normativa institucional sobre seguridad y bienestar universitario* .

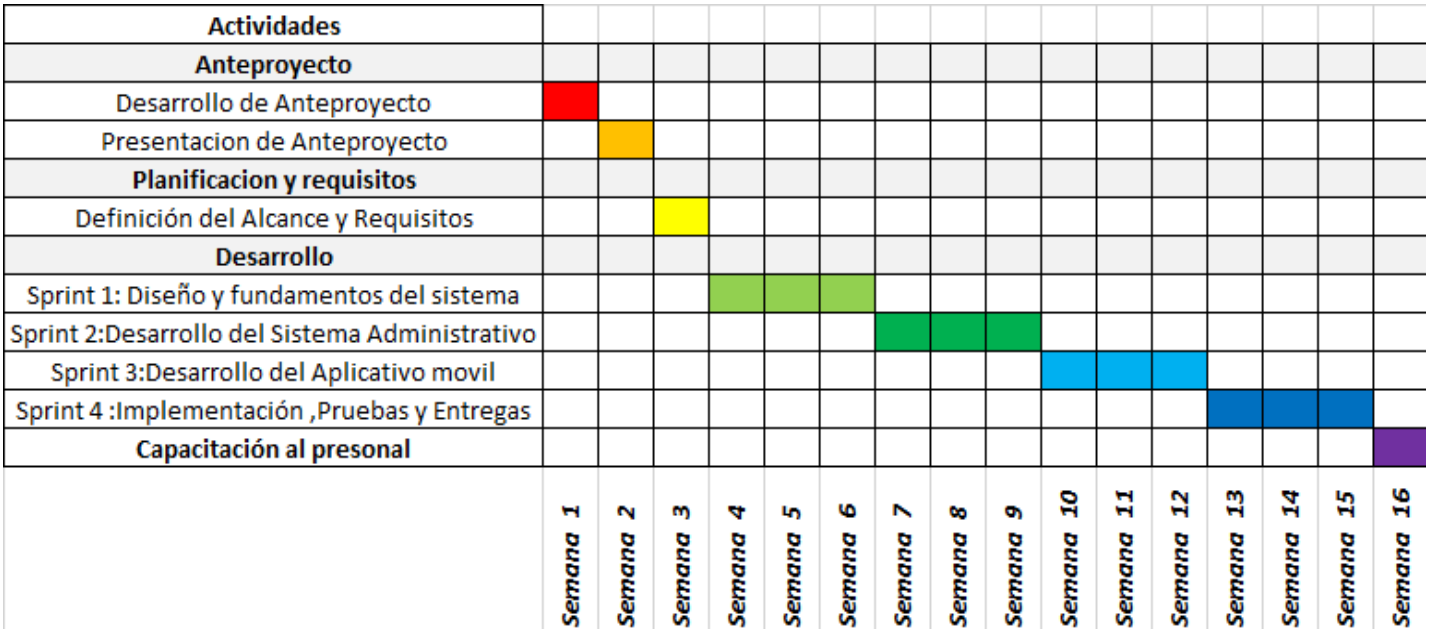
Zemsania. (2018). *Metodologías de Gestión de Proyectos*. Obtenido de Digital Talent: https://www.dtagency.tech/cursos/metodologias_gestion_proyectos/tema_2-ModeloAgile.pdf

Zendesk. (28 de 02 de 2024). *¿Qué significa feedback?* Obtenido de https://www.zendesk.com.mx/blog/feedback-que-es/

ANEXOS

Anexo 1

- **Cronograma (Gantt)**



Anexo 2

- **Presupuesto Ejecutado**

Descripción	Cantidad	Valor Unitario	Total
1 Equipos informáticos			
Computadoras para el desarrollo	2	1000	2000
Impresora multifuncional	1	500	500
2 Infraestructura y servicios			
Conexión Segura e Internet	6 meses	25	150
3 Material de Oficina			
Resma de Papel	2	3,80	7,60
Tintas para Impresora	2	35	70
Otros insumos (Esferos, Cuadernos.etc)	2		5
Imprevistos		100	100
Total Estimado			2.832,62

Anexo 3

- **Instrumentos de recopilación de datos**

ENTREVISTA

Entrevistado: Ing. Paúl Sánchez Franco.

Entrevistadores: Nataly Silva

Pablo Holguin

Objetivo de la entrevista:

Obtener información clave sobre los procesos de gestión de riesgos y seguridad institucional de la Unidad de Gestión de Riesgos.

1. ¿Cómo se gestionan actualmente los incidentes y emergencias dentro de la universidad?

Respuesta:

Actualmente, los incidentes se gestionan de forma manual y verbal. Los reportes suelen recibirse directamente por llamadas telefónicas o de manera presencial, y posteriormente se registran en documentos físicos o reportes informales, lo que dificulta el seguimiento adecuado y la consolidación de información.

2. ¿Cuáles considera que son las principales debilidades del sistema actual de gestión de riesgos?

Respuesta:

Las principales debilidades son la falta de un sistema digital oficial, la ausencia de registros centralizados, la demora en la atención de emergencias y la inexistencia de estadísticas confiables que permitan evaluar los riesgos y mejorar la planificación preventiva.

3. ¿Qué tipo de incidentes se presentan con mayor frecuencia en la universidad?

Respuesta:

Entre los incidentes más frecuentes se encuentran emergencias médicas, fallas en la infraestructura, situaciones de inseguridad, robos menores y eventos que requieren evacuación preventiva. Estos casos requieren una respuesta rápida y coordinada.

4. ¿Considera necesario contar con una aplicación móvil para la gestión de la seguridad universitaria?

Respuesta:

Sí, es totalmente necesario. Una aplicación móvil permitiría mejorar la comunicación con

la comunidad universitaria, agilizar los reportes de incidentes, emitir alertas inmediatas y fortalecer la capacidad de respuesta de la Unidad de Gestión de Riesgos.

5. ¿Qué funcionalidades considera indispensables en una aplicación móvil de seguridad integral?

Respuesta:

Las funcionalidades indispensables serían: reporte de incidentes con geolocalización, botón de alerta de pánico, emisión de notificaciones y alertas, acceso a protocolos de seguridad, puntos seguros, zonas de peligro y recomendaciones preventivas para la comunidad universitaria.

6. ¿Qué beneficios considera que aportaría esta aplicación a la comunidad universitaria?

Respuesta:

La aplicación contribuiría a reducir los tiempos de respuesta ante emergencias, mejorar la comunicación institucional, fortalecer la cultura de prevención y brindar mayor seguridad física y emocional a estudiantes, docentes y personal administrativo.

7. ¿Cree usted que la comunidad universitaria utilizaría una aplicación de este tipo?

Respuesta:

Sí, siempre que la aplicación sea intuitiva, fácil de usar y esté vinculada a las credenciales institucionales. La comunidad universitaria suele responder positivamente a herramientas tecnológicas que mejoren su seguridad y bienestar.

Anexo 5

- **Manual de usuario**

MANUAL DE USUARIO

SISTEMA DE SEGURIDAD INTEGRAL UNIVERSITARIO
PANEL DEL CONTROL ADMINISTRATIVO

AUTORES:
PABLO DAVID HOLGUIN RIOS
NATALY DOMENICA SILVA VILLAGRAN

CONTENIDO

1 **El Panel de Control Web (Para Administradores)**…..……100

1.1 ¿Cómo ingresar al Panel?.....100

1.2 Conociendo el Menú Principal……..100

1.3 ¿Cómo gestionar loa Incidentes reportados?.....103

1.4 ¿Cómo atender una Alerta de Pànico?.....105

1.5 ¿Cómo publicar Noticias o Avisos?..... 106

1.6 ¿Cómo administrar a los Usuarios?.....108

1.7 Configuración de cuenta…….110

INDICE DE FIGURAS

Figura 1 Panel de Ingreso……..……..100
Figura 2 Página Principal…….……..101
Figura 3 Tarjetas de colores……....101
Figura 4 Graficas de Análisis……..……..…..102
Figura 5 Menú Lateral…….…..……..….102
Figura 6 Sección de Gestión de Incidentes……..……..103
Figura 7 Filtro de Búsqueda……...…103
Figura 8 Reportes……..….103
Figura 9 Estado del Incidente…….…104
Figura 10 Detalle del Incidente…….…….104
Figura 11 Sección de Alertas……..105
Figura 12 Filtro de Búsqueda…….105
Figura 13 Lista de Información…….….105
Figura 14 Estado de Alerta…….106
Figura 15 Sección de Nueva Publicación……..106
Figura 16 Editar y Eliminar Publicaciones …….……..107
Figura 17 Sección de Puntos Seguros y Zonas de Peligro……...107
Figura 18 Sección Listado……...…108
Figura 19 Editar Usuarios……..…….108
Figura 20 Eliminar Usuarios……..……109
Figura 21 Sección de Comunidad……..……109
Figura 22 Acción Editar…….110
Figura 23 Configuración de Cuenta……110
Figura 24 Configurar Perfil……111
Figura 25 Apariencia del Sistema…….…….111

1. Panel de control web administrativo

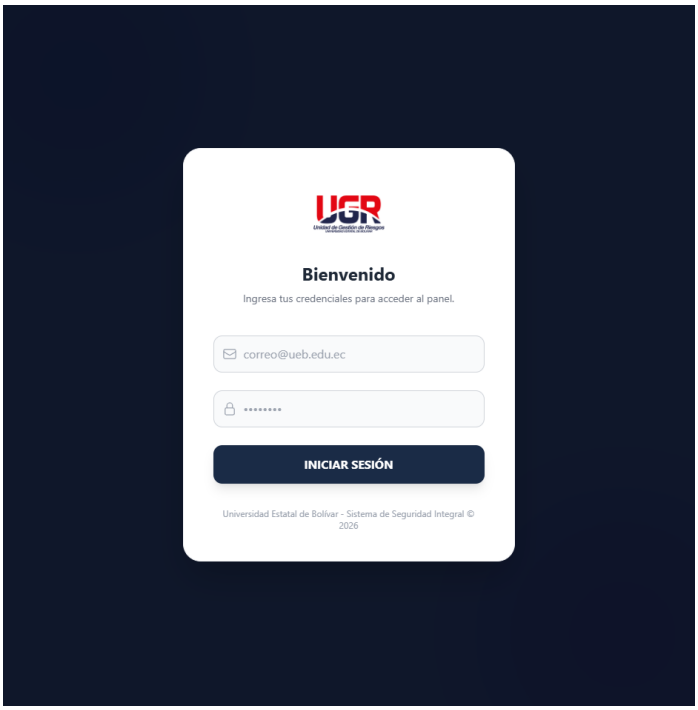
El panel de control administrativo es la central de operaciones. Desde aquí, el personal autorizado puede ver en tiempo real las emergencias, revisar reportes y enviar comunicados.

1.1. ¿Cómo ingresar al Panel?

1. Abra su navegador web preferido (Chrome, Brave, Edge, etc.)
2. Ingrese a la dirección web (URL) del sistema.
3. Vera una pantalla de inicio de sesión.
4. Escriba sus credenciales (Correo electrónico institucional y contraseña).
5. Haga clic en el botón Iniciar Sesión.

Nota: *Las credenciales administrativas serán entregada al personal de la unidad de gestión de riesgos por parte de los desarrolladores*.

Figura 1 *Panel de Ingreso*

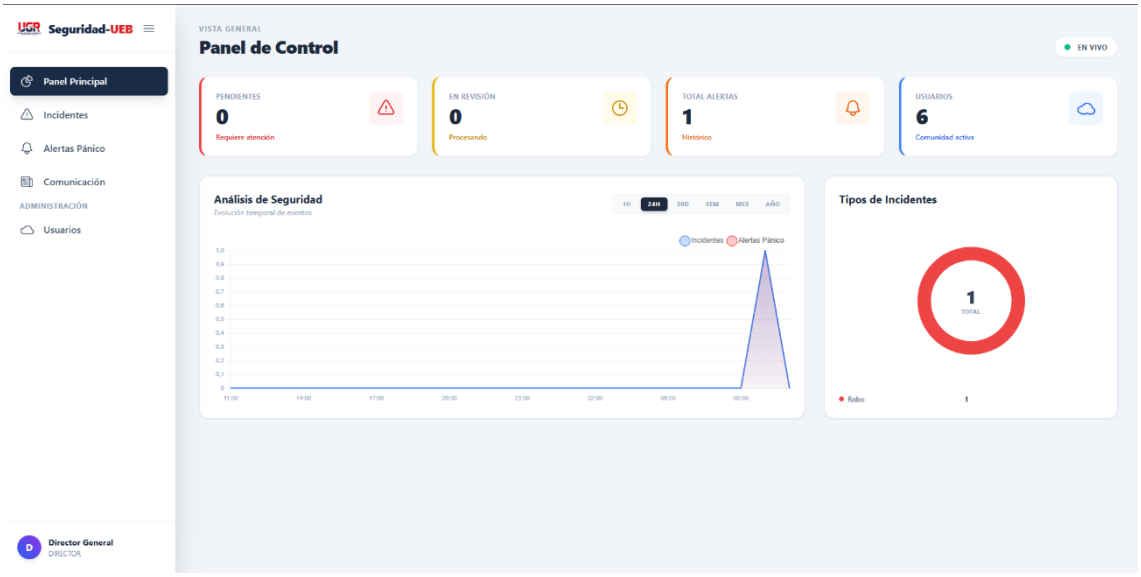


Nota: Elaboración Propia

1.2. Conociendo la página principal.

Al ingresar, la primera pantalla que vera es el dashboard o tablero principal. Que es la vista donde el director y los administradores pueden ver el resumen de lo que está pasando.

Figura 2 *Página Principal*

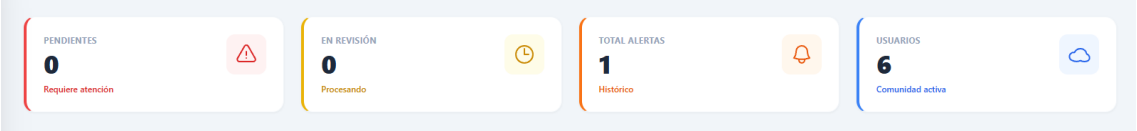


Nota: Elaboración Propia

La vista principal del director contiene:

1. Tarjetas de colores con cada resumen: cuantos incidentes pendientes hay en este momento, cuantos incidentes pendientes esperan revisión, el historial de alertas de pánico y el total de usuarios registrados en la app y el panel administrativo.

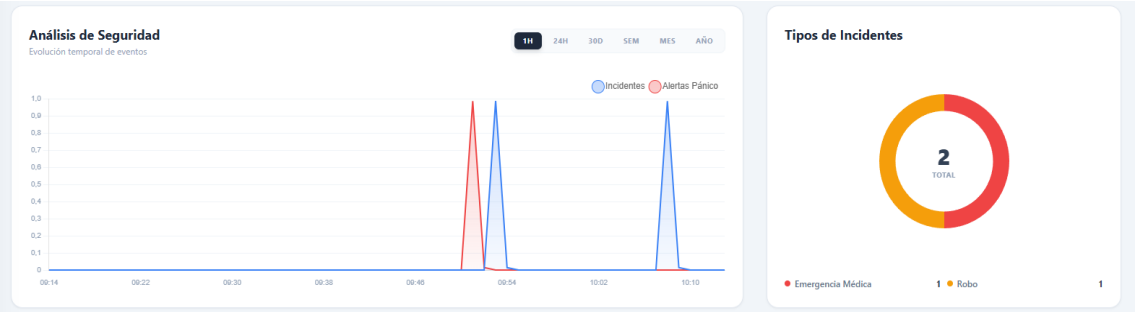
Figura 3 *Tarjetas de Colores*



Nota: Elaboración Propia

2. **Graficas de análisis:** Debajo de las tarjetas, encontrará representaciones visuales que ayudaran a entender la situación de seguridad en el campus: Análisis de seguridad y tipos de incidentes.

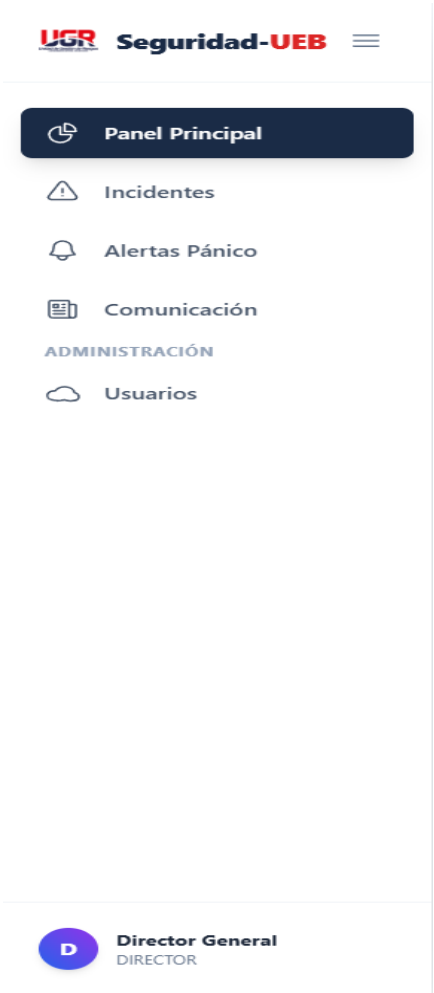
Figura 4 *Graficas de Análisis*



Nota: Elaboración Propia

3. **Menú lateral:** A la izquierda encontraremos un menú colapsable con las opciones y enlaces a cada sección: Panel Principal, Incidentes, Alertas de pánico, comunicación y usuarios; además de cuenta con la opción de configuración de cuenta al final del menú.

Figura 5 *Menú Lateral*

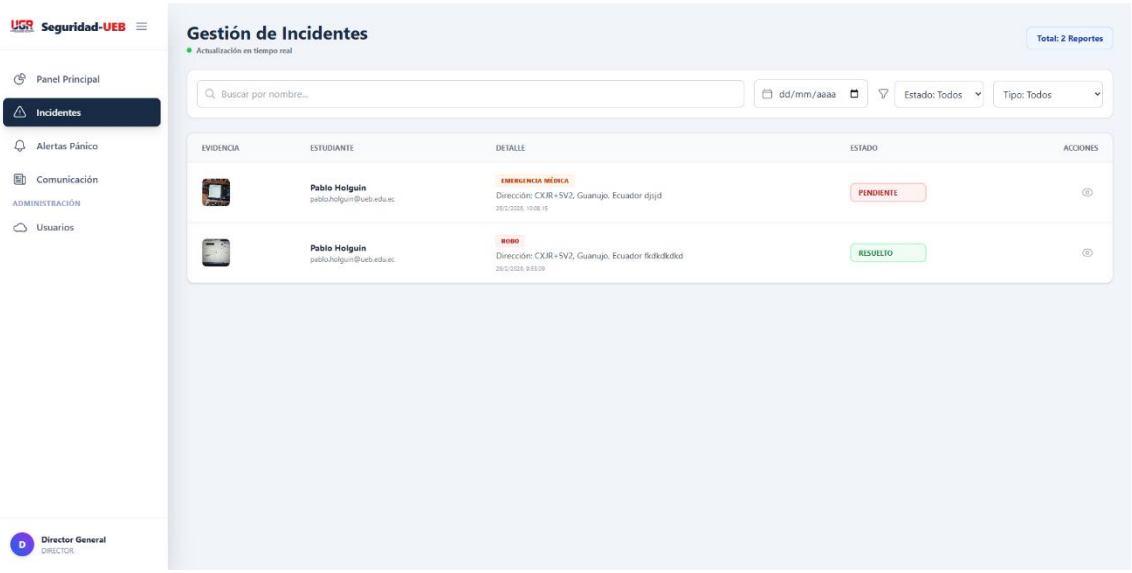


Nota: Elaboración Propia

1.3. ¿Cómo gestionar los incidentes reportados?

1. Valla a la sección **Incidentes** del menú.
2. Visualizara la sección de gestión de incidentes.

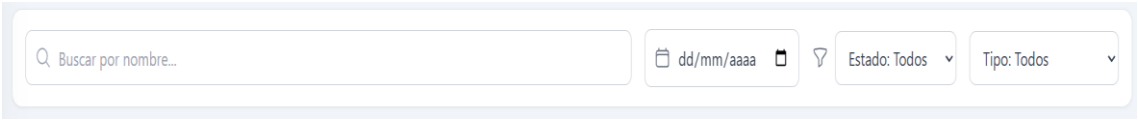
Figura 6 *Sección de Gestión de Incidentes*



Nota: Elaboración Propia

3. Contiene un apartado de filtro para búsquedas especificas ya sea por nombre del usuario que reporto, fechas, estado y tipo.

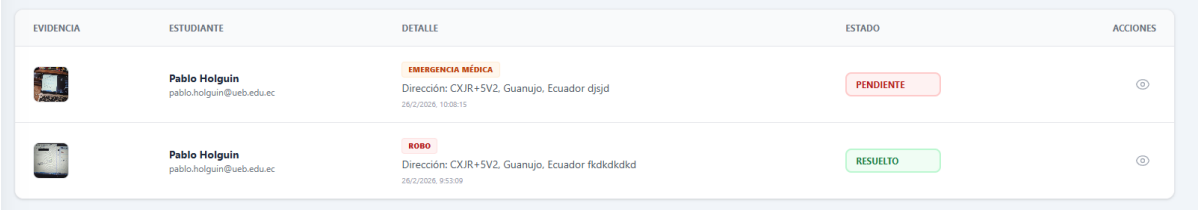
Figura 7 *Filtro de Búsquedas*



Nota: Elaboración Propia

4. Vera un listado con todos los reportes. Los más recientes aparecerán primero

Figura 8 *Reportes*

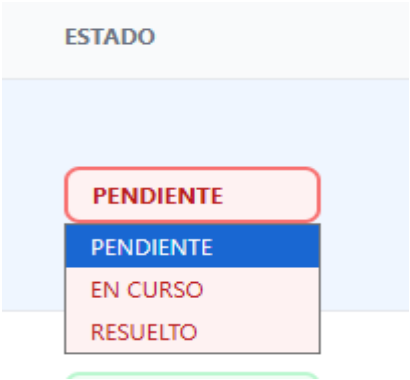


Nota: Elaboración Propia

Todo incidente nuevo entra como pendiente conforme su equipo lo investigue, podrán actualizar el estado haciendo clic sobre el:

- Cambie a proceso cuando el personal este trabajando en ello.
- Cambie a resuelto cuando el problema se haya solucionado.

Figura 9 *Estado del Incidente*



Nota: Elaboración Propia

5. Para cada reporte en acción tiene un botón donde podrá ver el reporte completo: Quien lo reportó, evidencia fotográfica, que tipo de incídete es, la fecha y hora que genero el reporte, la descripción detalla además de contener un botón para ver la ubicación del reporte.

Figura 10 *Detalle del Incidente*



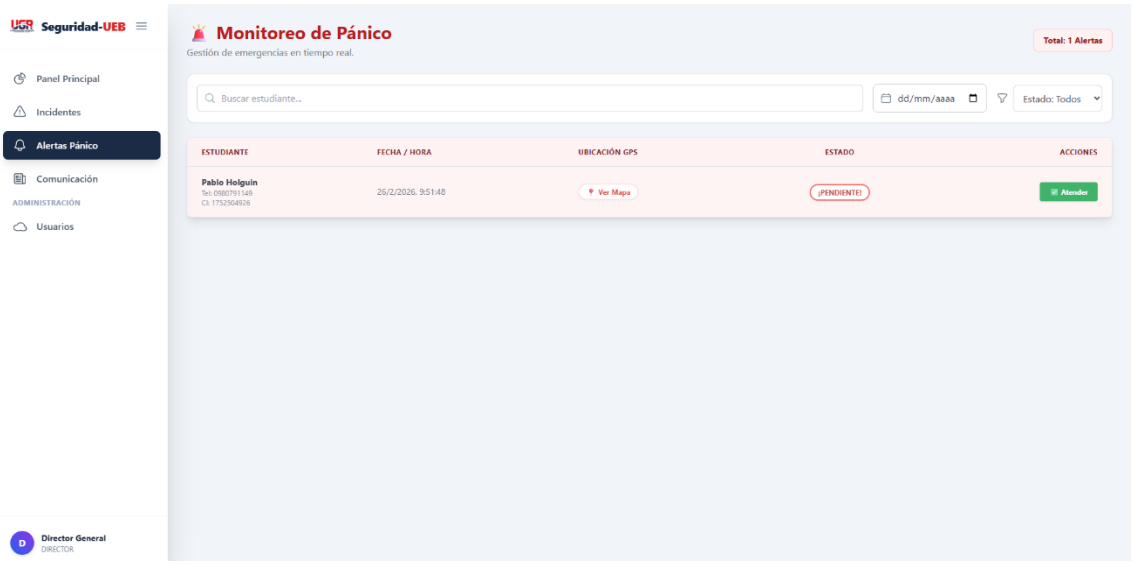
Nota: Elaboración Propia

1.4. ¿Cómo atender una alerta de Pánico?

1. Valla a la sección **Alertas** en el menú.

2. Visualizara la gestión de emergencia para el monitoreo de alertas de pánico.

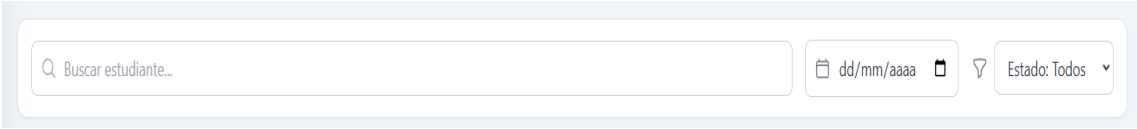
Figura 11 *Sección de Alertas*



Nota: Elaboración Propia

3. Contiene un apartado de filtro para búsquedas especificas ya sea por nombre del usuario que reporto, fechas y estado.

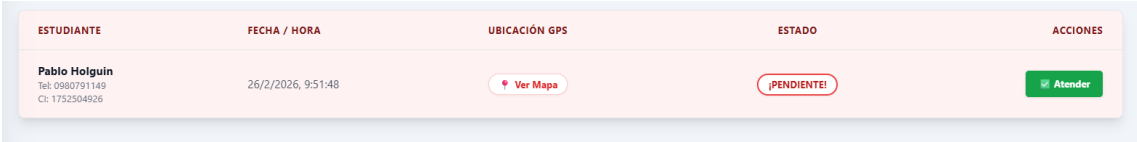
Figura 12 *Filtro de Búsqueda*



Nota: Elaboración Propia

4. Si hay una emergencia, la verá en una lista con la información del usuario, la fecha y hora, botón para ver la ubicación gps, el estado y el botón para atender la alerta.

Figura 13 *Lista de Información*



Nota: Elaboración Propia

5. Una vez que el quipo atiende y resuelve la emergencia actualiza el estado de la alerta.

Figura 14 *Estado de Alerta*

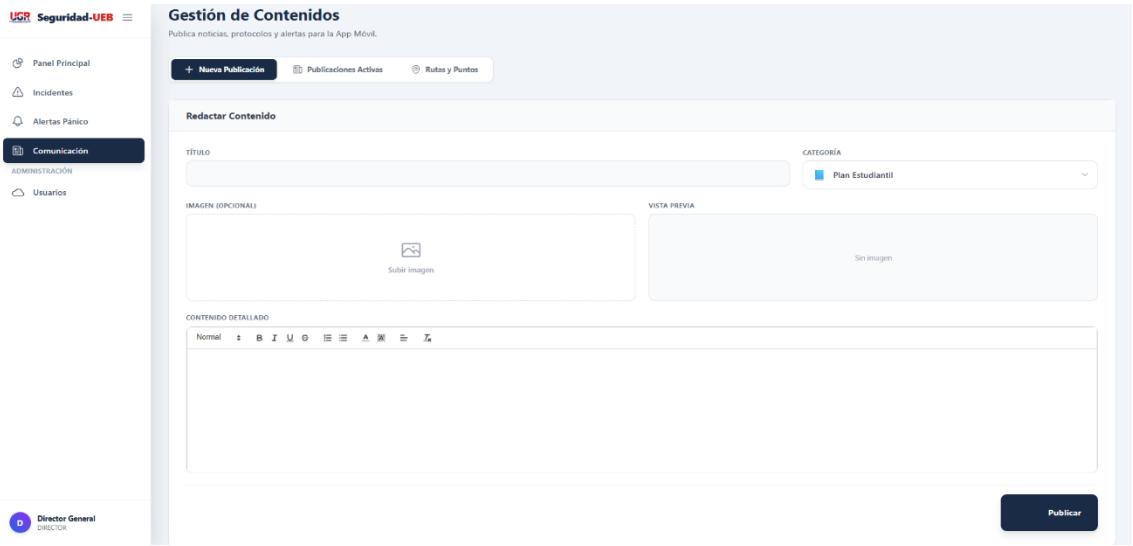


Nota: Elaboración Propia

1.5.¿Cómo publicar noticias o aviso?

1. Valla a la sección de comunicación en el menú.
2. Visualizara el apartado de gestión de contenidos donde cuenta con un submenú para ir a diferentes secciones de contenidos.
3. Tenemos la sección de nueva publicación donde redactara el contenido.

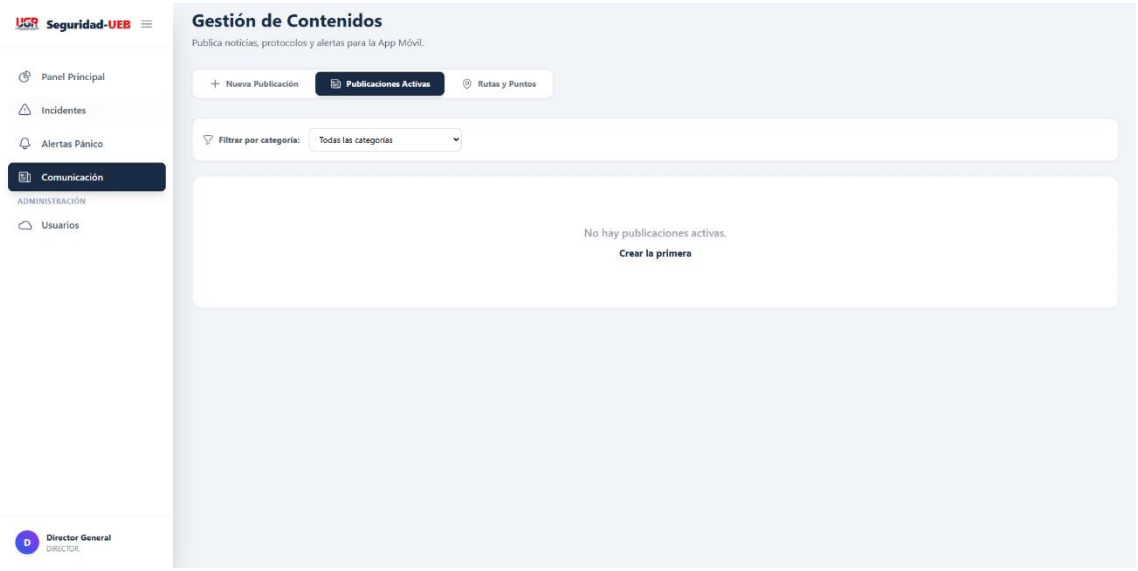
Figura 15 *Sección de Nueva Publicación*



Nota: Elaboración Propia

4. La sección de publicaciones activas mostrara los contenidos ya publicados donde se podrán editar y eliminar.

Figura 16 *Editar y Eliminar Publicaciones*



Nota: Elaboración propia

5. La sección de puntos seguros permite gestionar en un mapa los puntos seguros y zonas de peligro.

Figura 17 *Sección de Puntos Seguros y Zonas de Peligro*



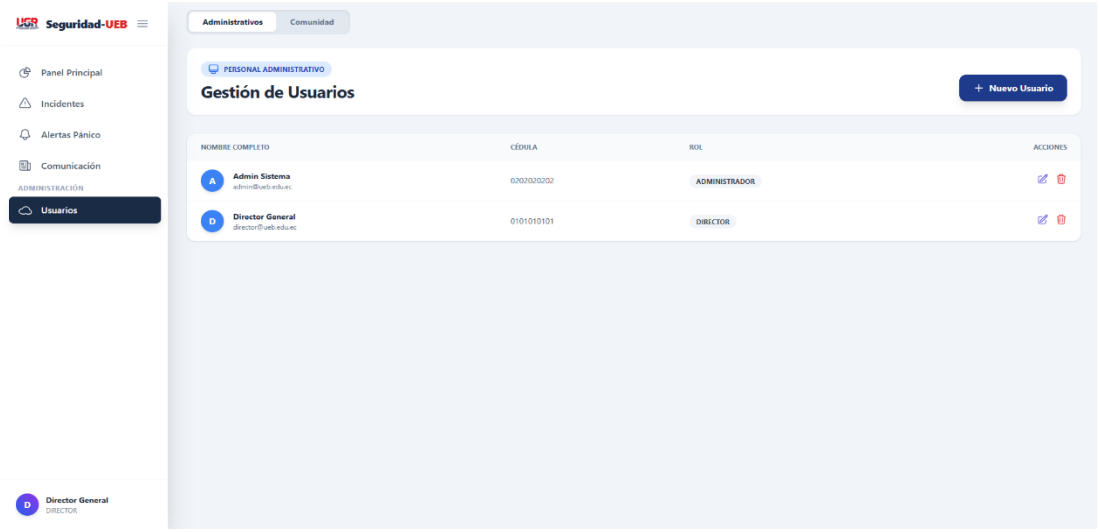
Nota: Elaboración Propia

1.6. ¿Cómo administrar los usuarios?

1. Valla a Usuarios en el menú
2. Visualizara la gestión de usuarios.
3. Esta sección cuenta con un submenú para tipos de usuarios

4. En la sección administrativo pueden ver el listado del personal administrativo.

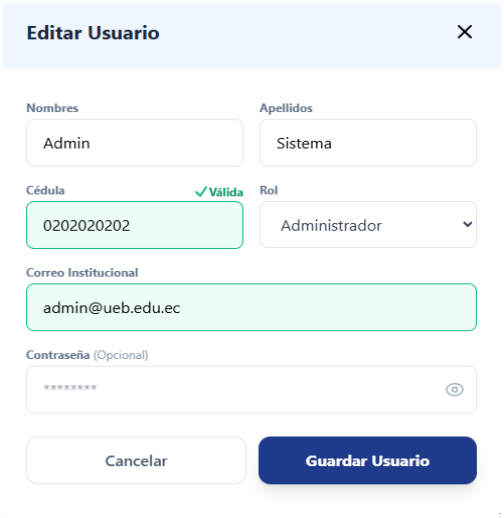
Figura 18 *Sección Listado*



Nota: Elaboración Propia

5. Cuenta con acciones para editar y eliminar usuarios administrativos

Figura 19 *Editar Usuarios*



Nota: Elaboración Propia

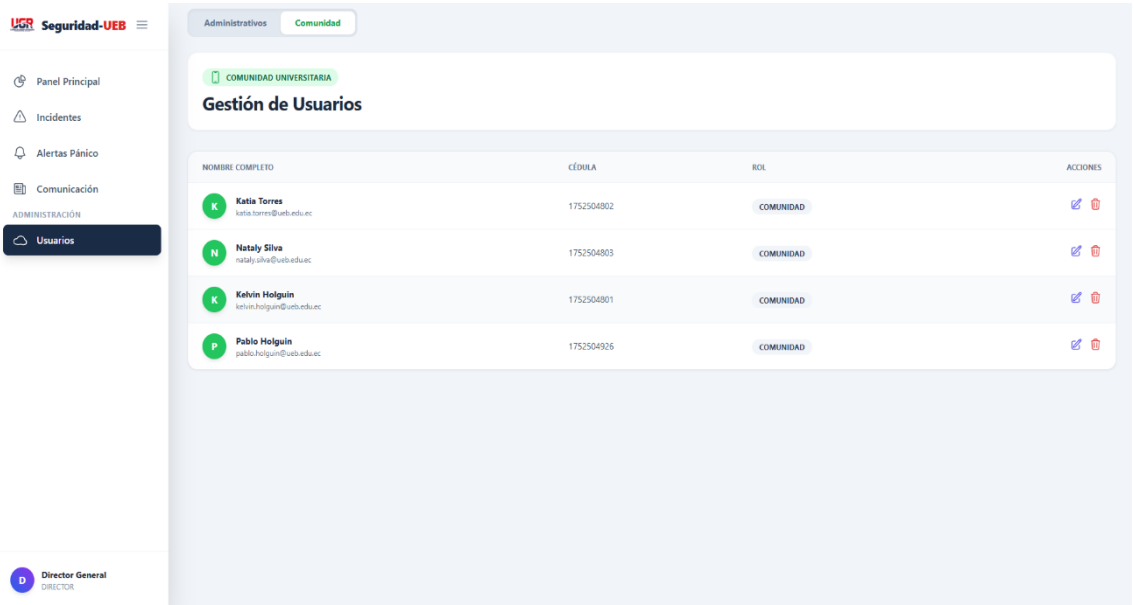
Figura 20 *Eliminar Usuarios*



Nota: Elaboración Propia

6. La sección de comunidad puede ver el listado de usuarios que ocupan el aplicativo móvil

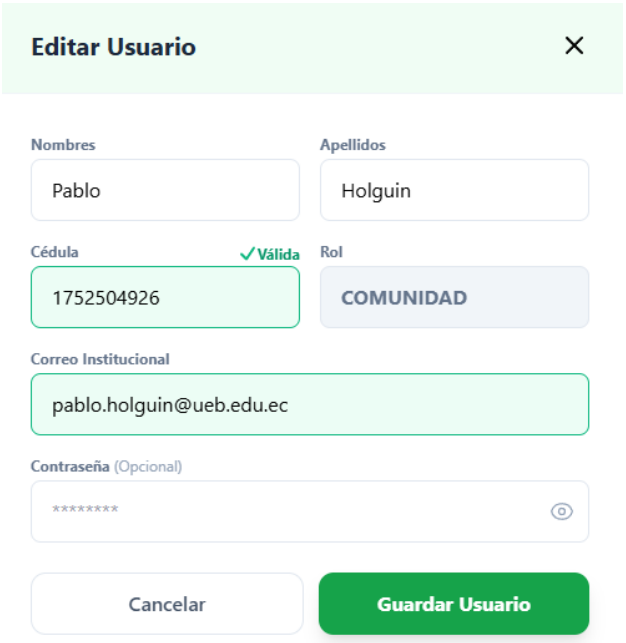
Figura 21 *Sección de Comunidad*



Nota: Elaboración Propia

7. Cuenta con la acción para editar

Figura 22 *Acción Editar*

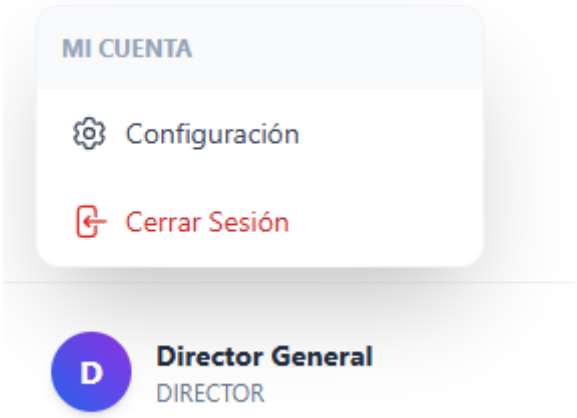


Nota: Elaboración Propia

1.7.Configuración de cuenta

1. Ir al perfil que se encuentra en la parte inferir del menú
2. Nos desplegara un menú para configuración y cierre de sesión.

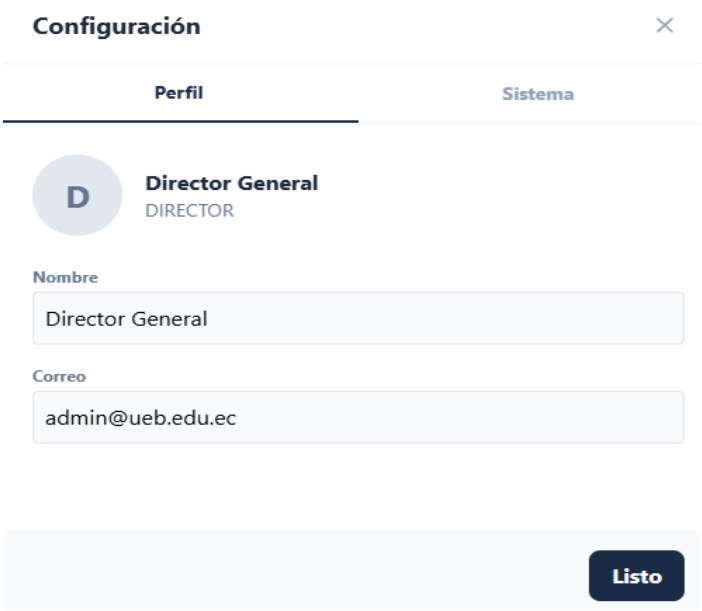
Figura 23 *Configuración de Cuenta*



Nota: Elaboración Propia

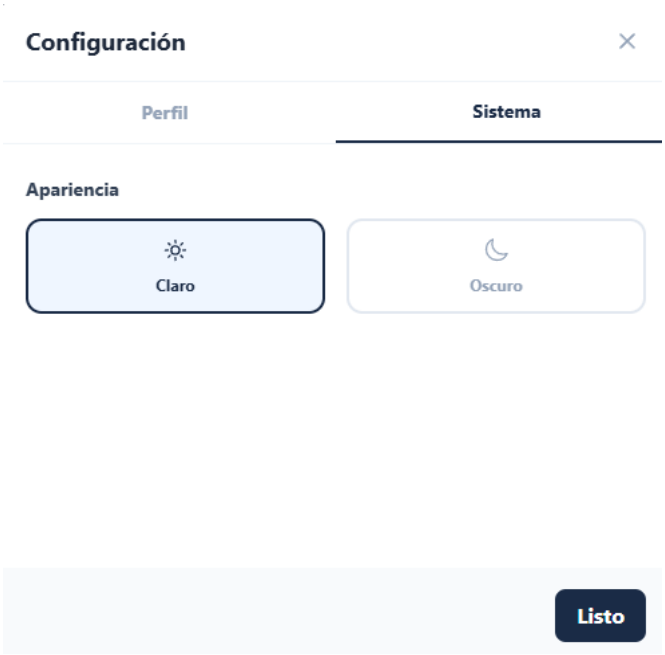
3. En la sección de configuración podemos configurar el perfil y la apariencia del sistema.

Figura 24 *Configurar Perfil*



Nota: Elaboración Propia

Figura 25 *Apariencia del Sistema*



Nota: Elaboración Propia

MANUAL DE USUARIO

APLICACIÓN MÓVIL DE SEGURIDAD INTEGRAL PARA LA UNIDAD DE GESTIÓN DE RIESGOS EN LA UNIVERSIDAD ESTATAL DE BOLÍVAR

AUTORES:
PABLO DAVID HOLGUIN RIOS
NATALY DOMENICA SILVA VILLAGRAN

CONTENIDO

1 **Aplicación Móvil**……115

1.1 Primeros Pasos: Registro e Inicio de Sesión……..115
1.2 Uso del Botón de Pánico……116
1.3 ¿Cómo Reportar un Incidente?.....117
1.4 ¿Dónde leer Noticias, Recomendaciones, Protocolos, Alertas, etc.?.....120
1.5 Menú Inferior……121
1.6 Configuración de la Cuenta……122

ÍNDICE DE FIGURAS

Figura 1 Inicio de Sesión……115
Figura 2 Registrarse……...115
Figura 3 Página Principal……...116
Figura 4 Botón de Pánico……...116
Figura 5 Notificación de Confirmación de Ayuda…….117
Figura 6 Información de la Ayuda en Camino……117
Figura 7 Reportar……...118
Figura 8 Nuevo Reporte…….118
Figura 9 Tipo de Incidente…….118
Figura 10 Ubicación Pendiente……..119
Figura 11 Evidencia Fotográfica y Detalle…….119
Figura 12 Estado del Reporte…….120
Figura 13 Información de Reporte……..120
Figura 14 Sección de Contenidos Protocolos…….121
Figura 15 Contenido Completo……..121
Figura 16 Menú Inferior ……122
Figura 17 Configuración Generales…….……..122
Figura 18 Datos Personales……123
Figura 19 Mis Reportes……..123
Figura 20 Historial de Alertas……124
Figura 21 Notificaciones y Ajustes……124
Figura 22 Contactos de Emergencia……...125
Figura 23 Cerrar Sesión …….125

1. Aplicación móvil de seguridad integral

1.1 Primeros pasos: Registro e Inicio de Sesión.

1. Al abrir la app por primera vez, verá la pantalla de inicio de sesión.

Figura 1 *Inicio de Sesión*



Nota: Elaboración Propia

2. Toque en registrase aquí.

Figura 2 *Registrarse*



Nota: Elaboración Propia

3. Llene el formulario con su nombre, correo institucional, cedula, teléfono, contraseña presione en registrarse, se enviará un correo para verificación de cuenta, una vez verificado podrá iniciar sesión con su correo institucional y la contraseña que acaba de crear.

3. Ingresara a la página principal del aplicativo.

Figura 3 *Página Principal*



Nota: Elaboración Propia

1.2 Uso del botón de Pánico

1. En la pantalla principal en la primera tarjeta se encuentra el botón de pánico llamado emergencia.

Figura 4 *Botón de Pánico*



Nota: Elaboración Propia

2. En caso de emergencia, presiónelo la aplicación leerá su ubicación GPS actual y enviará una alerta inmediata a la Unidad de Gestión de Riesgos.

Nota: Asegúrese de tener el GPS encendido

3. Se mostrará una notificación de confirmación de ayuda, así como una ventana con información de la ayuda en camino.

Figura 5 *Notificación de Confirmación de Ayuda*



Nota: Elaboración Propia

Figura 6 *Información de la Ayuda en Camino*

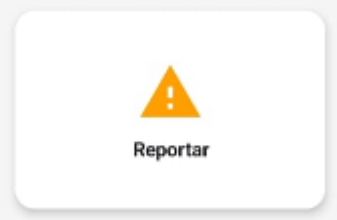


Nota: Elaboración Propia

1.3 ¿Cómo reportar un incidente?

1. En la pantalla principal presione la tarjeta de reporte la cual lo llevara a la sección para crear un nuevo reporte

Figura 7 *Reportar*



Nota: Elaboración Propia

Figura 8 *Nuevo Reporte*



Nota: Elaboración Propia

2. Seleccione el tipo de incidente (ej. robo, emergencia médica, acoso, incendio, infraestructura otros)

Figura 9 *Tipo de Incidente*



Nota: Elaboración Propia

3. Seleccione el botón ubicación pendiente para saber la ubicación exacta donde se encuentra(obligatorio).

Figura 10 *Ubicación Pendiente*



Nota: Elaboración Propia

4. Opcionalmente puede añadir evidencia fotográfica y el detalle del suceso.

Figura 11 *Evidencia Fotográfica y Detalle*



Nota: Elaboración Propia

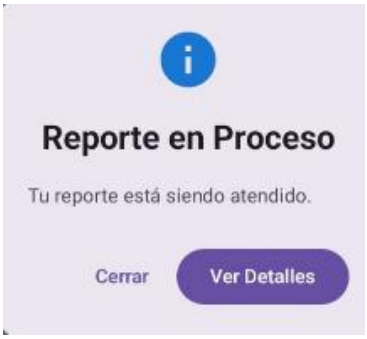
5. Presione el botón **Enviar reporte** para que el equipo de gestión de riesgo reciba su incidente o reporte.
6. Una vez enviado el reporte en la página principal se mostrará el estado del reporte (Pendiente, En Curso, Resuelto), así como también mostrará una ventana con información del reporte.

Figura 12 *Estado del Reporte*



Nota: Elaboración Propia

Figura 13 *Información de Reporte*

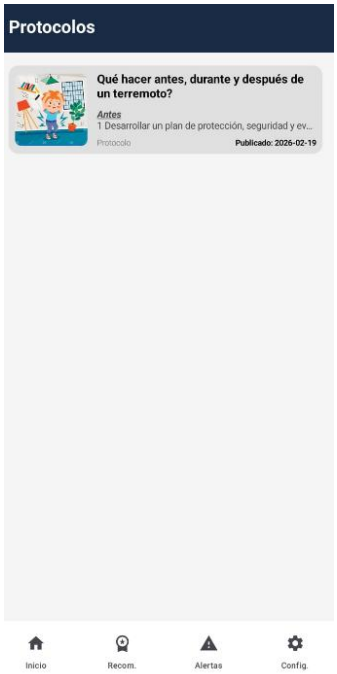


Nota: Elaboración Propia

1.4 ¿Dónde leer Noticias, Recomendaciones, protocolos, alertas, etc.?

1. En la página principal contamos con tarjetas protocolos, recomendaciones, alertas, plan estudiantil, etc. las cuales no lleva a las secciones con los contenidos de información.

Figura 14 *Sección de Contenidos Protocolos*



Nota: Elaboración Propia

2. Seleccione cualquier noticia para leer el contenido completo.

Figura 15 *Contenido Completo*



Nota: Elaboración Propia

1.5 Menú Inferior

1. En la parte inferior tendremos un menú con 4 opciones las cuales no llevaran a los apartados: inicio para mantenernos en la página principal,

recomendaciones y alertas para ver las noticias o contenidos publicados, y configuraciones.

Figura 16 *Menú Inferior*

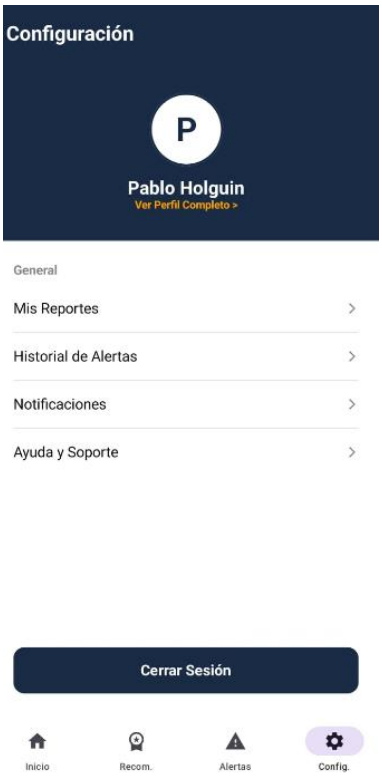


Nota: Elaboración Propia

1.6 Configuración de Cuenta

1. Al seleccionar la opción de configuración en el menú nos llevara al apartado de configuraciones generales

Figura 17 *Configuraciones Generales*



Nota: Elaboración Propia

2. donde tendremos varios opciones como:
    - Ver perfil completo: muestra nuestros datos personales

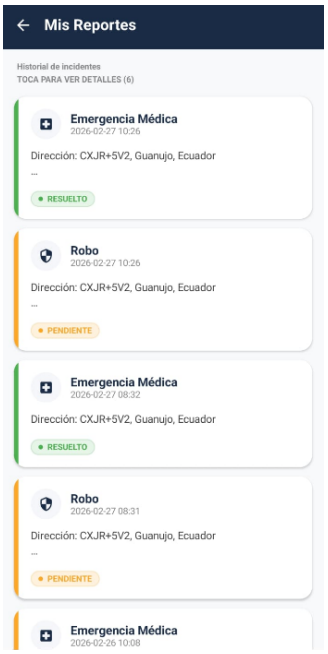
Figura 18 *Datos Personales*



Nota: Elaboración Propia

- Historial del reportes: muestra el historial de incidentes, con opción para ver el detalle

Figura 19 *Mis Reportes*



Nota: Elaboración Propia

- Historial de alerta.

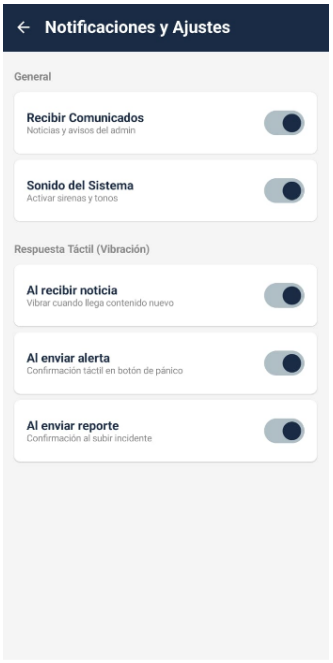
Figura 20 *Historial de Alertas*



Nota: Elaboración Propia

- Ajustes y notificaciones: podemos activar o desactivar el sonidos de notificaciones, así como respuestas de vibración

Figura 21 *Notificaciones y Ajustes*



Nota: Elaboración Propia

- Ayuda y soporte: sección con contactos de emergencia

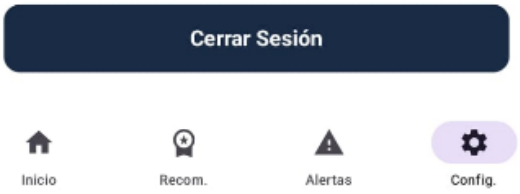
Figura 22 *Contactos de Emergencia*



Nota: Elaboración Propia

- Cerrar sesión

Figura 23 *Cerrar Sesión*



Nota: Elaboración Propia

MANUAL TECNICO

SISTEMA DE SEGURIDAD INTEGRAL PARA LA UNIDAD DE GESTION DE RIESGOS EN LA UNIVERSIDAD ESTATAL DE BOLIVAR.

AUTORES:

PABLO DAVID HOLGUIN RIOS

NATALY DOMENICA SILVA VILLAGRAN

VERSION: 1.0

FECHA: Febrero 2026

CONTENIDO

1 Introducción……130

2 Objetivo……..130

3 Creadores……130

4 Tecnología Utilizada…….……..130

5 Requerimientos Técnicos…….……..131

6 Estructura de Archivos y Carpetas…….……....131

7 Estructura del Sistema……..……..……132

7.1 Backend (API Rest)……..132

7.2 FrontEnd (Panel Administrador)……..……135

7.3 Aplicación Móvil……..……137

INDICE DE TABLAS

Tabla 1 Desarrolladores……130

Tabla 2 Tecnologías Utilizadas……130

INDICE DE FIGURAS

Figura 1 Backend y Frontend……132
Figura 2 Repositorio Aplicativo Móvil…….....133
Figura 3 Modelos…….....133
Figura 4 Controladores…….....134
Figura 5 Migraciones……134
Figura 6 Rutas……135
Figura 7 Frontend Panel Administrativo…….……...136
Figura 8 Aplicativo Movil…….……137

1. Introducción

El presente manual técnico tiene como objetivo documentar todo el funcionamiento interno del Sistema de Seguridad Integral de la Unidad de Gestión de Riesgos (UGR), especificando todos los recursos necesarios para su ejecución y futura evolución del sistema.

Por lo cual, este documento va orientado al personal técnico y desarrolladores de la Universidad Estatal de Bolívar, para que puedan entender la estructura interna que está detrás del funcionamiento del sistema (API, Panel Web y App Móvil). Este manual facilitará que cualquier técnico en la materia del desarrollo de software pueda incorporar nuevas funcionalidades, mejoras y crear un sistema más completo.

2. Objetivos

Facilitar una guía técnica y completa sobre el sistema de seguridad y alertas de la UGR, que ayude a desarrolladores y personal técnico a comprender, manipular y mantener el sistema de forma eficiente, prolongando su vida útil y haciéndolo sostenible a largo plazo.

3. Creadores

Tabla 1 *Desarrolladores*

Participantes:	Pablo David Holguin Rios Nataly Domenica Silva Villagran
Área:	Carrera de Software – UEB
Teléfonos:	0980791149 – 0968381035
Rol:	Desarrolladores

Nota: Elaboración propia

4. Tecnologías utilizadas

Tabla 26 *Tecnologías Utilizadas*

Servidor web:	Nginx/Apache
Framework:	Laravel versión 12.39.0
Gestor de base de datos:	PostgreSQL
Framework Frontend:	React
Desarrollo Móvil:	Android Kotlin
Gestor de dependencias:	Composer, NPM (Node.js), Gradle (Android)
Control de versiones:	GitHub

Nota: Elaboración propia

5. Requerimientos Técnicos

Los requerimientos mínimos para el funcionamiento de este sistema se dividen en categorías Hardware y Software:

Hardware:

PC (Entorno Local): Computadora de escritorio o laptop con procesador Intel i5/i7 o equivalente, mínimo 8GB de RAM.

Móvil: Dispositivo Android con versión 8.0 (Oreo) o superior, con GPS y conexión a red (Wi-Fi o datos).

Software:

Entorno Backend: PHP versión 8.1 o superior, Composer.

Entorno Frontend: Node.js v18+.

Entorno Móvil: Android Studio con Kotlin DSL.

Navegador web: Firefox, Chrome, o Edge para la ejecución y uso del panel.

6. Estructura de Archivos y Carpetas

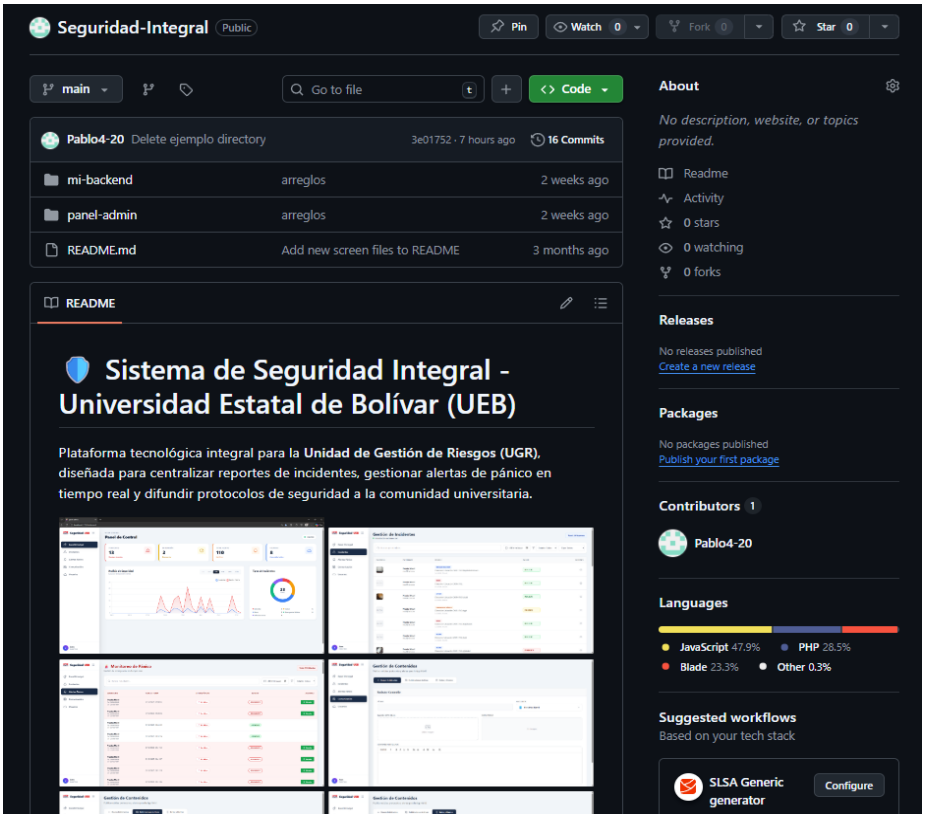
El sistema está dividido en 2 repositorios para garantizar la separación de responsabilidades:

1. https://github.com/Pablo4-20/Seguridad-Integral/tree/main

Mi-backend: Contiene toda la lógica local del servidor, API y base de datos (Laravel).

Panel-admin: Contiene el código fuente de la interfaz administrativa web (React).

Figura 1 *Repositorio Backend y Frontend*



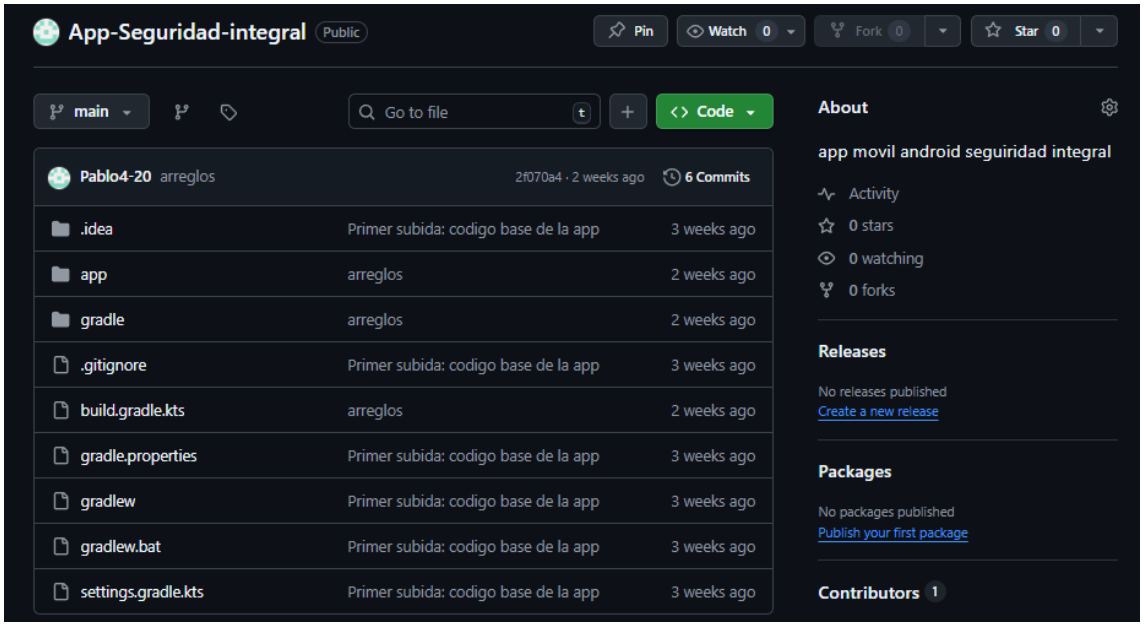
Nota: Elaboración Propia

Para conectar el backend con la base de datos, se debe acceder al archivo. env en la carpeta mi-backend y colocar los datos como el nombre de la base de datos, el usuario y la contraseña .

2. https://github.com/Pablo4-20/App-Seguridad-integral

App-seguridad-integral: Contiene el código fuente de la aplicación móvil para usuarios (Android/Kotlin).

Figura 2 *Repositorio Aplicativo Móvil*



Nota: Elaboración Propia

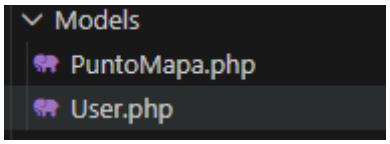
7. Estructura del Sistema

7.1. Backend

Dado que el framework Laravel utiliza el patrón Modelo Vista Controlador (MVC), es fundamental para tener una organización limpia.

Modelos: Responsables de la interacción con la base de datos. Aquí se encuentran modelos como User.php (usuarios) y PuntoMapa.php (ubicaciones de incidentes).

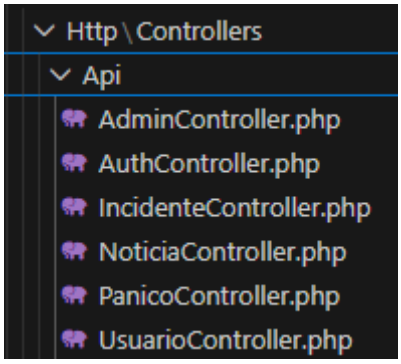
Figura 3 *Modelos*



Nota: Elaboración Propia

Controladores: Actúan como intermediarios entre las rutas API y los modelos.

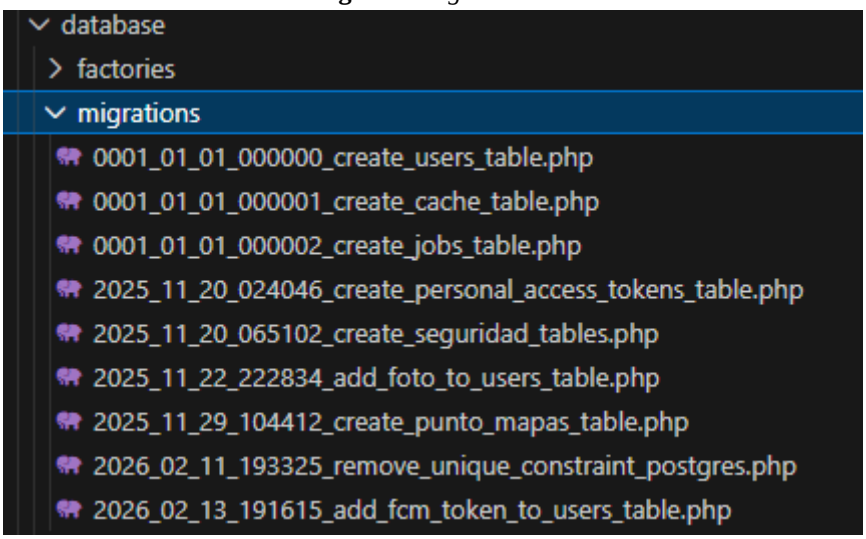
Figura 4 *Controladores*



Nota: Elaboración Propia

Migraciones: Sistema de control de versiones para la base de datos, donde se crean las tablas de usuarios, alertas y reportes.

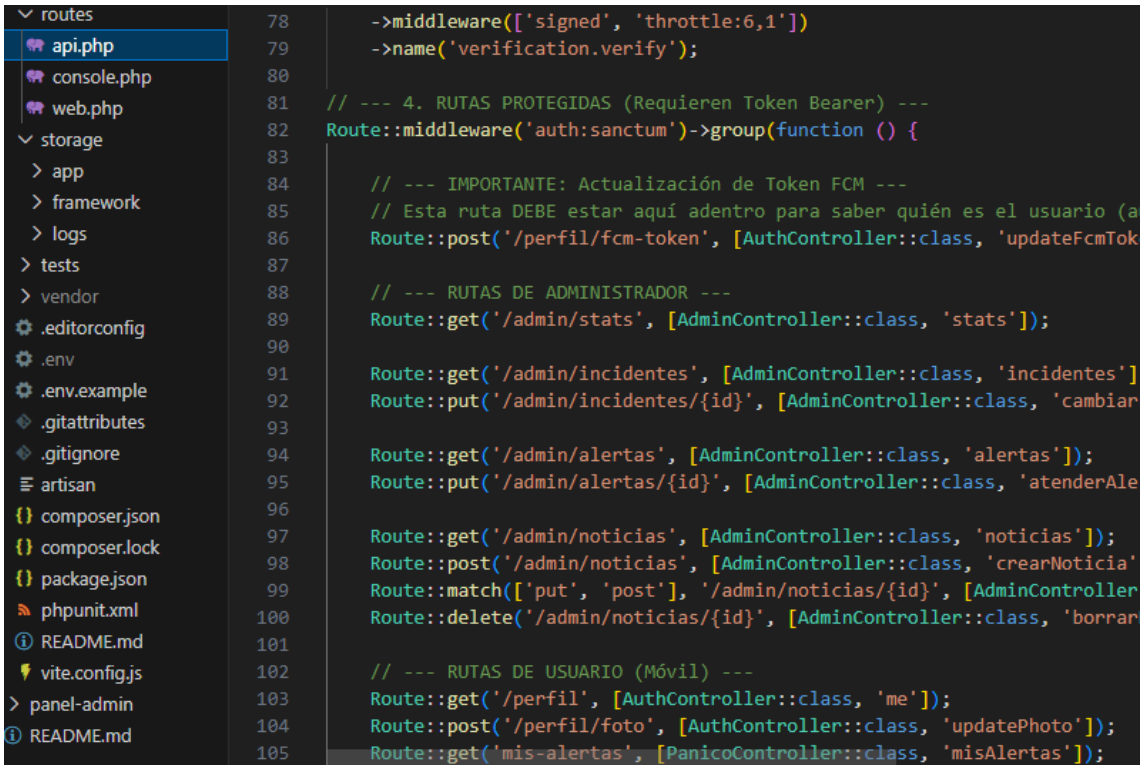
Figura 5 *Migraciones*



Nota: Elaboración Propia

Rutas: Dirigen las solicitudes (GET, POST, etc.) desde la aplicación móvil o el panel web hacia los controladores.

Figura 6 *Rutas*



Nota: Elaboración Propia

7.2. Frontend Panel administrador

El panel web utiliza una arquitectura basada en componentes React.

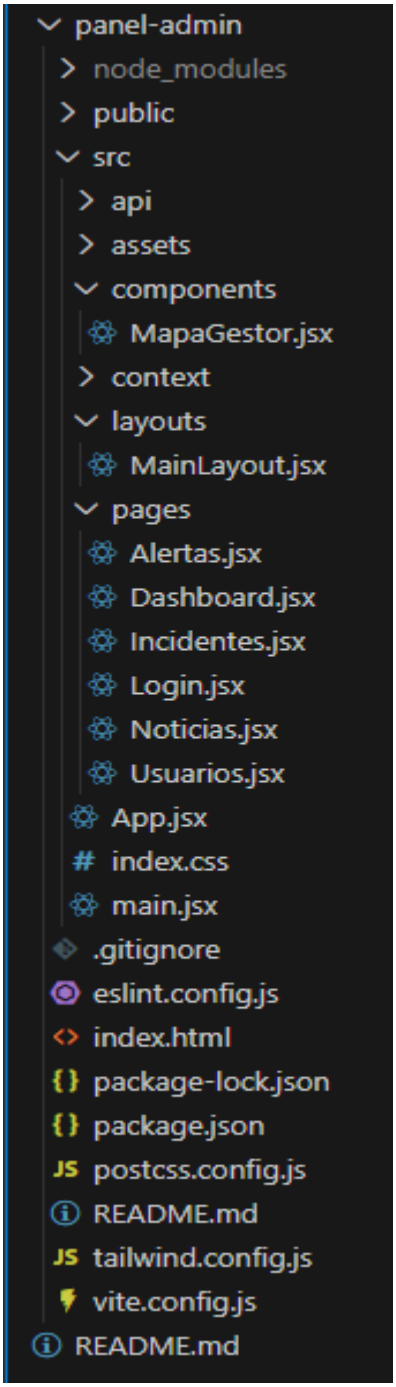
Páginas / Vistas: Son las responsables de la interfaz de usuario. Archivos. jsx como Dashboard.jsx, Incidentes.jsx, Noticias.jsx y MapaGestor.jsx.

Servicios HTTP: Encargado de hacer las peticiones seguras al Backend enviando el token de autorización.

Estilos: Se utiliza Tailwind CSS en lugar de archivos CSS puros, permitiendo diseñar interfaces modernas directamente en las clases de los componentes React.

Recursos Multimedia: Archivos de audio utilizados para notificar sonoramente a los administradores cuando ingresa una alerta de pánico.

Figura 7 *Frontend Panel Administrativo*



Nota: Elaboración Propia

7.3. Aplicativo móvil

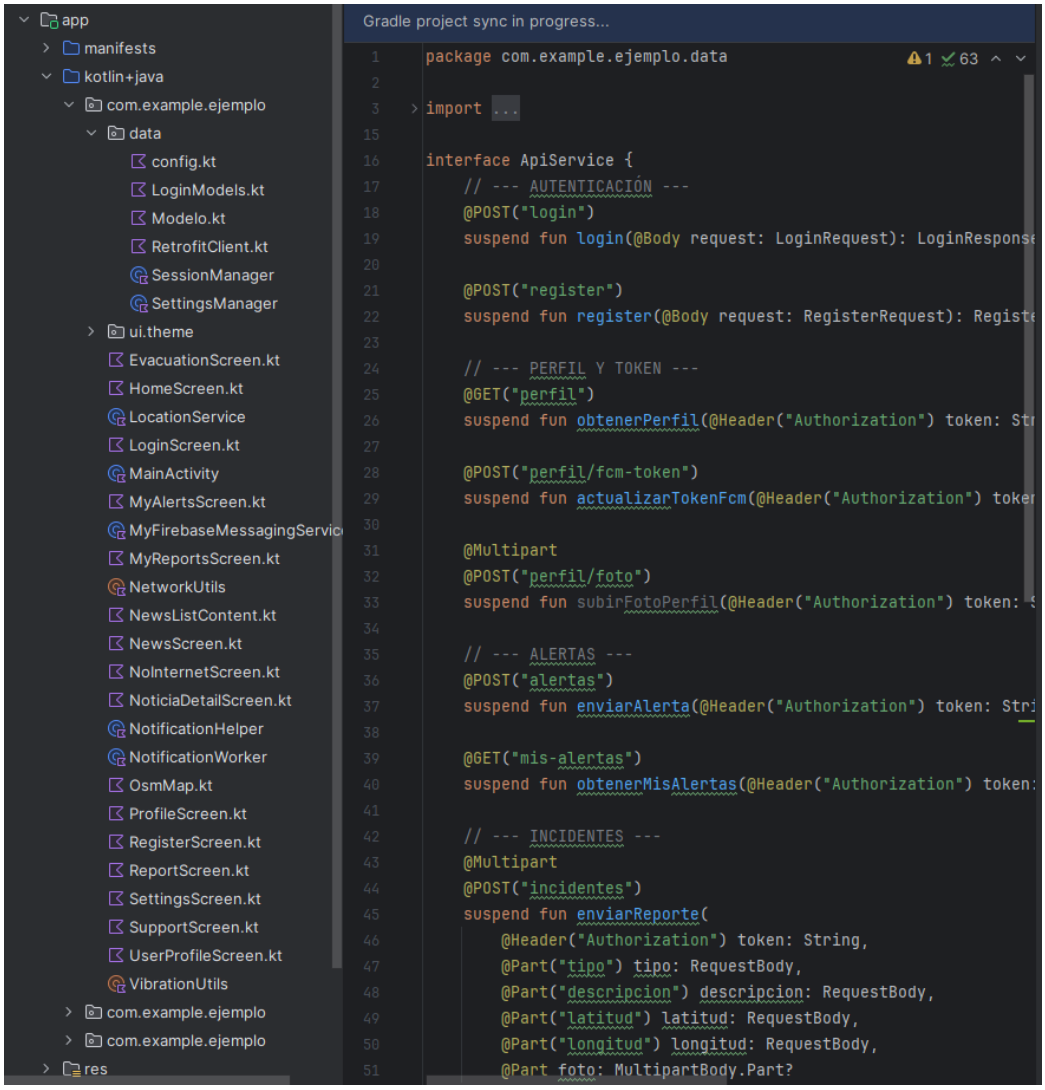
Vistas / UI: Utiliza Jetpack Compose para interfaces declarativas puras (HomeScreen.kt, EvacuationScreen.kt, ReportScreen.kt).

Consumo de API (RetrofitClient.kt): Se comunica con mi-backend para validar usuarios, descargar noticias y enviar reportes. Gestiona los tokens localmente con SessionManager.kt.

Servicios en Segundo Plano: (LocationService.kt y NotificationWorker.kt): Se encargan de gestionar la obtención de coordenadas para el botón de pánico y la generación de notificaciones locales en el sistema operativo del teléfono, funcionando de manera nativa sin dependencias externas.

Manejo de Mapas (OsmMap.kt): Integración con OpenStreetMap para mostrar rutas seguras y puntos de encuentro de forma independiente.

Figura 8 *Aplicativo Móvil*



Nota: Elaboración Propia

ING. MARICELA ARACELI ESPÍN MOREJÓN EN CALIDAD DE DIRECTORA DEL TRABAJO DE INTEGRACIÓN CURRICULAR,

CERTIFICA

Que el trabajo de integración curricular denominado APLICACIÓN MÓVIL DE SEGURIDAD INTEGRAL PARA LA UNIDAD DE GESTIÓN DE RIESGOS EN LA UNIVERSIDAD ESTATAL DE BOLÍVAR, presentado por: PABLO DAVID HOLGUIN RIOS y NATALY DOMENICA SILVA VILLAGRAN estudiantes de la **carrera de Software** pasó el análisis de coincidencia no accidental en la herramienta COMPILATIO, reflejando un **porcentaje de similitud del 10%**, como se puede evidenciar en el documento adjunto.

Guaranda, 27 de febrero del 2026

Atentamente,



ING. MARICELA ESPÍN
Directora

CERTIFICADO DE ANÁLISIS
magister

Holguin_Titulacion Proyecto Tecnologico Tesis

10%
Textos sospechosos

- **< 1%** Similitudes
  - < 1 % similitudes entre comillas
  - 0 % entre las fuentes mencionadas
- **3%** Idiomas no reconocidos (ignorado)
- **10%** Textos potencialmente generados por la IA

Nombre del documento: Holguin_Titulacion Proyecto Tecnologico Tesis.docx
ID del documento: cf89b4b6ef7d888cf4310e07931900ec9251ce8f
Tamaño del documento original: 13,5 MB

Depositante: MARICELA ARACELI ESPIN MOREJON
Fecha de depósito: 27/2/2026
Tipo de carga: interface
fecha de fin de análisis: 27/2/2026

Número de palabras: 19.038
Número de caracteres: 133.117

Ubicación de las similitudes en el documento:



Fuentes principales detectadas

N°	Descripciones	Similitudes	Ubicaciones	Datos adicionales
1	**www.dtagency.tech** https://www.dtagency.tech/cursos/metodologias_gestion_proyectos/tema_2-ModeloAgile.pdf **1 fuente similar**	3%		Palabras idénticas: **3%** (610 palabras)
2	**www.godaddy.com** | Laravel: Qué es y para qué sirve https://www.godaddy.com/resources/latam/stories/laravel-que-es-para-que-sirve	< 1%		Palabras idénticas: **< 1%** (68 palabras)
3	**ribuni.uni.edu.ni** | Desarrollo de sistema web para determinar y visualizar la geol... http://ribuni.uni.edu.ni/4083/1/95678.pdf **1 fuente similar**	< 1%		Palabras idénticas: **< 1%** (53 palabras)

Fuentes con similitudes fortuitas

N°	Descripciones	Similitudes	Ubicaciones	Datos adicionales
1	**hdl.handle.net** | Diseno e implantacion de un sistema de capacitacion en el secto... https://hdl.handle.net/20.500.14330/TES01000021061	< 1%		Palabras idénticas: **< 1%** (22 palabras)
2	**doi.org** | Frameworks del lado del Servidor: Caso de Estudio Node JS, Django y Lar... https://doi.org/10.33386/593dp.2025.1.2729	< 1%		Palabras idénticas: **< 1%** (23 palabras)
3	**dspace.udla.edu.ec** | Repositorio Digital Universidad De Las Américas: Diseño e i... https://dspace.udla.edu.ec/handle/33000/11586	< 1%		Palabras idénticas: **< 1%** (18 palabras)
4	**Documento de otro usuario** #7d640e Viene de de otro grupo	< 1%		Palabras idénticas: **< 1%** (14 palabras)
5	**repositorio.untels.edu.pe** | Desarrollo de un aplicativo móvil para agilizar la gest... http://repositorio.untels.edu.pe/jspui/handle/123456789/264	< 1%		Palabras idénticas: **< 1%** (15 palabras)

Fuentes mencionadas (sin similitudes detectadas)

Estas fuentes han sido citadas en el documento sin encontrar similitudes.

1 https://www.astera.com/es/type/blog/what-is-a-database/
2 https://es.scribd.com/document/673031283/Introduccion-a-La-Seguridad-Integral-e-Importancia
3 https://talently.tech/blog/que-es-flutter/
4 https://www.ibm.com/es-es/think/topics/api
5 https://www.techtarget.com/searchapparchitecture/definition/software