UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS

PROYECTO DE GRADO

SISTEMA WEB PARA LA SEGURIDAD DE APLICACIONES MÓVILES Y SU IMPACTO EN ESTUDIANTES VULNERABLES A POSIBLES RIESGOS TECNOLÓGICOS

CASO: UNIDAD EDUCATIVA SAN MARTÍN DE PORRES

Para Optar al Título de Licenciatura en Ingeniería de Sistemas

MENCIÓN: GESTIÓN Y PRODUCCIÓN

Postulante : Leo Charly Quispe Marquez

Tutor Metodológico : M. Sc. Lic. Ing. Marisol Arguedas Balladares

Tutor Especialista : Lic. Ing. Humberto Aguilar Lobo, PhD

Tutor Revisor : Lic. Santos Zenon Quispe Apaza

EL ALTO – BOLIVIA

2024

DECLARACIÓN JURADA DE AUTENTICIDAD Y RESPONSABILIDAD

Yo **Leo Charly Quispe Marquez estudiante con C.I. 9885757 LP** mediante la presente **declaro** de manera pública que la propuesta del **TRABAJO DE GRADO** titulada **"SISTEMA WEB PARA LA SEGURIDAD DE APLICACIONES MÓVILES Y SU IMPACTO EN ESTUDIANTES VULNERABLES A POSIBLES RIESGOS TECNOLÓGICOS CASO: UNIDAD EDUCATIVA SAN MARTÍN DE PORRES",** es original, siendo resultado de mi trabajo personal y no constituye una copia o replica de trabajos similares elaborados,

Autorizo la publicación del resumen de mi propuesta en internet y me comprometo a responder a todos los cuestionamientos que se desprenden de su lectura.

Asimismo, me hago responsable ante la universidad o terceros, de cualquiera irregularidad o daño que pudiera ocasionar, por el incumplimiento de lo declarado.

De identificarse falsificación, plagio, fraude, o que el **TRABAJO DE GRADO** haya sido publicado anteriormente; asumo las consecuencias y sanciones que de mi acción se deriven, responsabilizándome por todas las cargas legales que se deriven de ello sometiéndome a las normas establecidas y vigentes de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

El Alto, noviembre de 2024

……

Nombre: Leo Charly Quispe Marquez
C.I. 9885757 L.P.
e-mail: leonfico330@gmail.com

Dedicatoria

Dedico este proyecto de grado a Dios, por su infinita bondad, por estar siempre a mi lado y por darme la sabiduría y la perseverancia necesarias para llegar hasta aquí.

A mis padres, cuyo amor, sacrificio y enseñanzas me han dado el valor para enfrentar los retos y superar los obstáculos. Gracias por ser mi refugio y mi mayor apoyo. A mis hermanos, por su compañía constante, por ser mi ejemplo de lucha y por su apoyo incondicional. Este logro es, en gran parte, gracias a ustedes.

Agradecimientos:

Quiero agradecer de todo corazón a Dios, por guiarme con su luz, por darme fuerza cuando más la necesité y por cada bendición recibida en este proceso. A mis padres, por ser mi mayor inspiración, por su sacrificio, amor y por estar siempre presentes. Gracias por creer en mí y por enseñarme el verdadero valor del esfuerzo. Cada uno de sus gestos y palabras ha sido fundamental para llegar hasta aquí.

A mis hermanos, por su amor y apoyo inquebrantables, por estar a mi lado en cada paso de este viaje, por su comprensión y por ser mi mayor fuente de aliento.

A mis tutores, M. Sc. Lic. Ing. Marisol Arguedas Balladares, Lic. Ing. Humberto Aguilar Lobo PhD, y Lic. Santos Zenon Quispe Apaza, por su dedicación, paciencia y orientación. Gracias por brindarme su sabiduría, por su apoyo constante y por ayudarme a encontrar el camino cuando las dudas surgían. Su guía ha sido esencial para el éxito de este proyecto.

A mis amigos, quienes, con su compañía, risas y apoyo constante, hicieron que este camino fuera mucho más llevadero. Gracias por ser mi refugio en los momentos difíciles y por celebrar conmigo cada pequeña victoria. Su amistad ha sido un regalo invaluable en este proceso.

Resumen

Este proyecto desarrolla un sistema web para evaluar la seguridad de aplicaciones móviles, enfocado en proteger a los estudiantes de riesgos tecnológicos. La pandemia de COVID-19 aumentó el uso de dispositivos móviles en entornos educativos, exponiendo a los estudiantes a amenazas informáticas, especialmente al descargar aplicaciones de fuentes no confiables. En respuesta, el sistema, implementado en Laravel, permite a los usuarios analizar aplicaciones para identificar permisos sospechosos y proteger sus datos personales. El proyecto tiene como objetivos diagnosticar la seguridad de las aplicaciones móviles, educar en seguridad digital y fomentar una comunidad informada sobre riesgos tecnológicos. Para ello, se utilizó APKTool para analizar las aplicaciones y detectar permisos invasivos. Además, la plataforma incluye medidas de seguridad como autenticación y encriptación de contraseñas, y permite pruebas de funcionalidad y usabilidad del sistema. El sistema fue implementado con éxito, proporcionando a la comunidad educativa una herramienta eficaz para analizar aplicaciones móviles y promover prácticas seguras en el entorno digital. Las recomendaciones incluyen integrar alertas en tiempo real y módulos adicionales de capacitación para mejorar la seguridad digital.

Palabras Claves: seguridad de aplicaciones móviles, sistema web, protección de datos personales, seguridad digital.

Índice de Contenidos

Pág.

Índice de Figuras

Pág.

Índice de Tablas

Pág.

Índice de Ecuaciones

Pág.

CAPITULO I

MARCO PRELIMINAR

INGENIERÍA
DE SISTEMAS
UNIVERSIDAD PUBLICA DE EL ALTO

CAPITULO I
MARCO PRELIMINAR

1.1. INTRODUCCIÓN

En la era digital actual, el uso de aplicaciones móviles se ha convertido en una parte integral de nuestra vida diaria, abarcando desde la comunicación hasta las transacciones financieras. Sin embargo, la seguridad en los dispositivos móviles depende mucho de las aplicaciones que se les instala, llegando a ser una preocupación cada vez mayor dada a la creciente cantidad de datos personales y sensibles que manejan.

El presente proyecto de grado se enfoca en la creación de un sistema web destinado a estudiantes para en comprender y fortalecer la seguridad en aplicaciones móviles. Esta iniciativa nace de la necesidad crítica de formar individuos capacitados en identificar vulnerabilidades e implementar prácticas seguras.

El proyecto propuesto se fundamenta en una combinación de teoría y práctica, ofreciendo módulos interactivos, recursos multimedia y ejercicios prácticos que abarcan desde los conceptos básicos hasta técnicas avanzadas de seguridad en aplicaciones móviles. Este enfoque integral busca empoderar a los estudiantes con las habilidades necesarias para enfrentar los desafíos de seguridad inherentes al uso de aplicaciones móviles en la actualidad.

El sistema web se desarrollará utilizando el Método de Diseño Hipermedia Orientado a Objetos, con el cual nos permite realizar un desarrollo del sistema mediante los datos recolectados y así lograr la prevención y orientación sobre la seguridad en aplicaciones móviles, cuidando la exposición de los datos personales y asegurar un buen manejo de las aplicaciones móviles en sus dispositivos de los estudiantes.

1.2. ANTECEDENTES

1.2.1. Antecedentes Institucionales

De acuerdo al manual de funciones de la institución, exprese que:

La Unidad Educativa “San Martín de Porres” que es parte de las Escuelas Populares Don Bosco, queda ubicado en la Ciudad de El Alto del Departamento de La Paz, carretera a Viacha zona San Martín, en la Avenida Raúl Salmón y calle Rafael Bustamante, fue fundada el 26 de marzo del año 2.000 por el Excelentísimo Monseñor Jesús Juárez Obispo de la Diócesis de El Alto, juntamente con los miembros de la Junta Vecinal de la Zona San Martín de Porres.

La zona San Martín que alberga 23 años a la Unidad Educativa del mismo nombre, el cual tiene una visión de formar buenos cristianos y honrados ciudadanos para responder a las necesidades socioculturales, equitativa, religiosos y de calidad, así también contempla una misión a base de la práctica de valores como amor, la razón y la religión, en base al sistema preventivo de Don Bosco.

La unidad educativa está compuesta por varios elementos como:

- Estructura administrativa: Director, Personal Directivo, Docentes, personal administrativo
- Estudiantes: Nivel inicial, Nivel primario, Nivel secundario.
- Instalaciones: Aulas, laboratorios, biblioteca, áreas recreativas.

1.2.1.1. Misión y Visión

Visión

Buscamos una Unidad Educativa que dé respuestas a las necesidades socio culturales, equitativa, religioso y de alta calidad, para que formemos una sociedad más justa, logrando la formación del hombre integral; así mismo se considera a la persona como centro de acción educadora y centro de todo dinamismo existencial, llegando a formar: “Buenos cristianos y Honrados Ciudadanos” (U.E. San Martin de Porres, 2023).

Misión

Como parte de una sociedad educativa Salesiana, inspirados por el Sistema Educativo de Don Bosco, fundamentada en los pilares de RAZÓN, RELIGIÓN y AMABILIDAD, cultivamos los valores humanos y cristianos, ofreciendo apoyo en el proceso enseñanza, aprendizaje de cada uno de nuestros destinatarios, formando integralmente, hombres y mujeres, responsables, competentes y comprometidos que respondan a sus actitudes al tiempo histórico en que viven (U.E. San Martin de Porres, 2023).

1.2.1.2. Objetivo

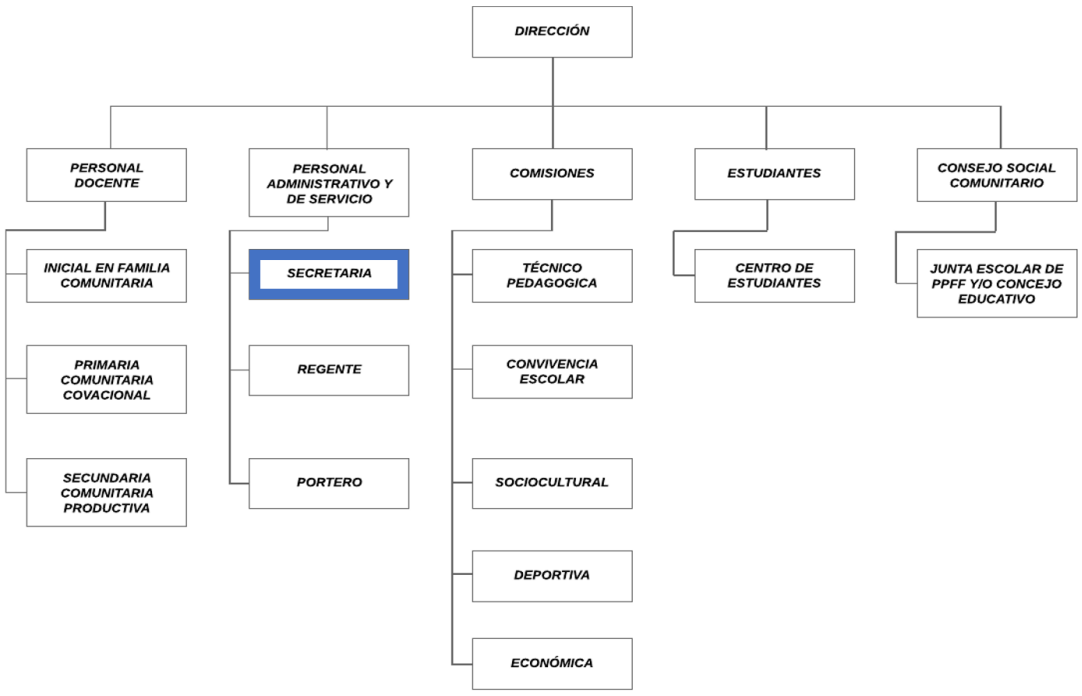
El objetivo de la Unidad Educativa, es brindar una educación integral que abarque aspectos académicos, sociales, emocionales y físicos para el desarrollo completo de los estudiantes, también proporcionar una sólida formación académica en diferentes áreas del conocimiento, promoviendo el aprendizaje significativo y la adquisición de habilidades que les permitan enfrentar desafíos académicos y profesionales

Promover valores éticos y morales, como el respeto, la responsabilidad, la tolerancia, la solidaridad y la honestidad, que contribuyan a la formación integral de individuos íntegros y conscientes de su entorno, también fomentar el desarrollo de habilidades específicas como el pensamiento crítico, la resolución de problemas, la creatividad y el trabajo en equipo. Apoyando el crecimiento personal y emocional de los estudiantes proporcionándoles un ambiente seguro que fomente la autoestima, preparando a los estudiantes para enfrentar desafíos de la vida cotidiana, brindándoles herramientas para tomar decisiones informadas.

También tiene como objetivo, fomentar la participación activa de la comunidad educativa, incluyendo a padres, profesores y estudiantes, para crear un entorno colaborativo que potencie el aprendizaje y el desarrollo integral de los estudiantes (U.E. San Martin de Porres, 2023).

1.2.1.3. Organigrama

Figura 1.
Organigrama de la U.E. San Martin de Porres



Nota. Es la distribución de las unidades de trabajo del colegio, donde en el departamento de "secretaria" estará establecido el Sistema Web. (Extraído del Estatuto del Colegio)

1.2.2. Antecedentes afines al proyecto de grado

A continuación, se describe los antecedentes internacionales consultados para el desarrollo del presente proyecto.

1.2.2.1. Antecedentes Internacionales

A continuación, se describe los antecedentes internacionales consultados:

(Rosel Miguel Castillo Romero, 2022) Desarrollo de una aplicación web y móvil para la gestión de riesgos de seguridad de la información aplicado a una empresa de consultoría de sistemas: Dentro de la arquitectura del software se hará referencia a servicios web consumidos por una ampliación móvil, por lo cual se define de forma teórica. Según la World Wide Web Consortium (W3C), un servicio web lo define como aquel sistema de software asignado con el objetivo de permitir y facilitar la interacción de máquina a máquinaInter operativa mediante una red. Adicionalmente se determina como un conjunto de protocolos con el propósito de intercambiar diversos datos por medio de aplicaciones de software, la idea general es que diversas aplicaciones de software desarrollados en distintos lenguajes de programación y ejecutadas sobre cualquier plataforma, pueda consumir los servicios web con la finalidad de intercambiar cualquier dato de manera íntegra en redes de computadores (pág. 26).

(D. Adrián Rojo Becerril, s. f., 2020)**|:** En este proyecto se aborda cubrir la necesidad detectada en la mitigación de riesgos de seguridad en los usuarios como es el análisis desde el punto de vista de seguridad de las aplicaciones que instalan en los teléfonos inteligentes los usuarios. Este análisis requiere ser realizado desde dos aproximaciones diferentes:

a) De manera estática: revisión del código fuente de la aplicación móvil y extracción de los componentes que la forman, identificación de estructuras sospechosas en el código, obtención de direcciones IP y web.

b) De manera dinámica: monitorizando actividad de la aplicación durante su uso, siendo posible obtener los recursos a los que accede la aplicación, tráfico que esta genera.

Para ello se ha recopilado información del sistema operativo Android, sus aplicaciones y se han probado diversas herramientas de análisis de aplicaciones disponibles en el mercado, eligiendo para el desarrollo de la plataforma la herramienta Marco de seguridad móvil (MobSF) pues nos permite realizar el análisis desde las dos aproximaciones antes mencionadas. Para la realización del análisis dinámico es usado un dispositivo virtual con Android. Se documenta el manejo de la herramienta, así como el significado de los reportes generados (pág. 6).

1.2.2.2. Antecedentes Nacionales

A continuación, se describe los antecedentes nacionales:

(De Redes et al., s. f., 2020)Uso de Redes Sociales por parte De Jóvenes, En Tiempos De Pandemia: En esta primera unidad, se trabajó el tema del uso de celulares y redes sociales en el período de la cuarentena generada por la pandemia del COVID-19, en la provincia de Tiraque1. La metodología es cualitativa, basada en entrevistas y grupos de discusión virtual, con 30 jóvenes de último año de colegio. Se utilizó infografías para generar la discusión y también para presentar los resultados. La actividad duró una semana. Se trata de una combinación entre investigación e intervención, porque al tiempo de recabar información, se genera interacción que convierte el dato obtenido en contenido para la producción de materiales didácticos (pág. 1).

(Elizabeth S., 2022) Uso del celular en el aula y ciberbullying en Bolivia: Las redes sociales, se han introducido en la vida de las personas que antes eran ajenas al fenómeno de Internet. No es extraño oír hablar por la calle de Facebook y no necesariamente entre los más jóvenes. La extraordinaria capacidad de comunicación y de poner en contacto a las personas que tienen las redes ha provocado que un gran número de personas las esté utilizando con fines muy distintos. Se utilizan para encontrar y entablar diálogo con amistades perdidas tiempo atrás, para debatir sobre los temas más variados, apoyar causas de todo tipo, organizar encuentros de amigos, ex-compañeros de estudios o para dar a conocer congresos y conferencias, a través de los cuales no sólo se dan detalles sobre el encuentro, sino que las personas pueden confirmar su asistencia o ausencia al evento (pág. 16).

1.3. PLANTEAMIENTO DEL PROBLEMA

1.3.1. Problema principal

En la actualidad, las aplicaciones móviles se han convertido en una parte fundamental de la vida cotidiana. Durante la pandemia en la ciudad de El Alto, los estudiantes de los colegios fueron pasando sus clases de manera virtual y la mayoría optó por conseguir un dispositivo móvil para no perder sus clases virtuales, en la actualidad ante el retorno a la modalidad de clases presenciales, el uso de las aplicaciones móviles en entornos educativos ha experimentado un crecimiento exponencial.

Estas herramientas tecnológicas ofrecen numerosas ventajas en términos de accesibilidad y recursos pedagógicos, pero al mismo tiempo plantean desafíos cruciales en cuanto a la seguridad y protección de los datos de los estudiantes y la integridad del proceso educativo, el desconocimiento sobre los riesgos hace que los estudiantes sean vulnerables a ataques cibernéticos con fines delictivos.

Se observo que los estudiantes utilizan una amplia variedad de aplicaciones móviles para diferentes propósitos, tanto en comunicación y entretenimiento, llegando a presenciar que no todos saben los riesgos que traen algunas aplicaciones móviles dejando vulnerable a los estudiantes, por lo cual algunas aplicaciones móviles pueden recopilar información personal de los usuarios como nombres, direcciones, números de teléfono, e incluso datos financieros, Esto puede poner en riesgo la privacidad y seguridad los estudiantes, así como exponerlos a posibles engaños o estafas.

Por un lado, las aplicaciones sospechosas pueden contener contenido inapropiado, como violencia, drogas, pornografía o promover conductas peligrosas como el bullying.

Además, algunas aplicaciones móviles sospechosas pueden ser adictivas y generar una dependencia en los estudiantes. La constante exposición y uso de estas aplicaciones pueden ocasionar distracciones en clase, dificultad en concentración y disminución del rendimiento académico, ya que están en constante manejo del dispositivo haciendo uso de aplicaciones de forma inadecuada.

1.3.2. Problemas secundarios

Se ha determinado los siguientes problemas secundarios:

- Vulnerabilidad a ataques cibernéticos lo que resulta en la filtración de información confidencial, robo de datos personales y compromiso de la privacidad de los estudiantes.

- La ausencia de actualización en las aplicaciones, son expuestas a posibles ataques y amenazas cibernéticas.

- Deficiencia en capacitación a estudiantes lo cual hace que realicen acciones que pongan en peligro la seguridad de sus aplicaciones y datos personales

- Descarga de aplicaciones de fuentes no confiables lo cual aumenta el riesgo de instalar aplicaciones maliciosas o comprometidas

1.3.3. Formulación del problema

¿De qué manera generar un Sistema Web para la seguridad de aplicaciones móviles y su impacto en estudiantes vulnerables a posibles riesgos tecnológicos en la Unidad Educativa San Martin de Porres?

1.4. OBJETIVOS

1.4.1. Objetivo General

Desarrollar un Sistema Web para la Seguridad de Aplicaciones Móviles y su impacto en Estudiantes Vulnerables a Posibles Riesgos Tecnológicos en la Unidad Educativa San Martin de Porres.

1.4.2. Objetivos Específicos

- Diagnosticar las aplicaciones móviles para evaluar su nivel de seguridad informática

- Realizar pruebas de seguridad a las aplicaciones móviles mediante las herramientas de ingeniería inversa.

- Identificar las aplicaciones que usan permisos sospechosos para evitar el acceso a la información del dispositivo móvil.

- Verificar fuentes confiables para descargar aplicaciones móviles garantizando la seguridad de los dispositivos e información personal.

- Diseñar un sistema web didáctico para proporcionar información y recursos sobre las mejores prácticas de seguridad en aplicaciones móviles

1.5. JUSTIFICACIÓN

De acuerdo con Rut (2019), actualmente este dispositivo es considerado una herramienta indispensable, es difícil una persona que no posea un teléfono móvil, sea por trabajo, por ocio o por la simple necesidad de hacer y recibir llamadas.

Entre los mayores usuarios de este medio tecnológico, se observa que son, especialmente los jóvenes quienes hacen un uso excesivo del celular y hasta de manera dependiente, ya sea por el uso de servicio de telefonía, el WhatsApp, mensajes de texto, Internet, etc.

Una investigación realizada por dos estudiantes de la Facultad de Ciencia Económicas y Financieras de la “UAJMS”, sobre el problema del mal uso del teléfono móvil indica que los estudiantes sí hace uso del celular en aula sin la autorización respectiva, llegando a la conclusión: “El mal uso del celular en los jóvenes influye mucho en el bajo nivel académico, la distracción en clase hace que él pierda la concentración y desvíe la explicación del docente. Además, pierde la noción de las tareas y ejercicios propuestos. El estudiante en lugar de tener contacto personal, este realiza por medio de él, causando problemas adictivos y dependencia al uso del móvil llamado ‘nomofobia’.

Dentro de esta investigación también se indica que los alumnos pasan el mayor tiempo del día navegando por internet, esto crea una problemática significativa pues solamente pierden el tiempo en estar en redes sociales, pudiendo aprovecharlo en acciones positivas, como la convivencia familiar y con amigos, el poder realizar algún tipo de ejercicio, leer libros, entre otras cosas, que puedan mejorar su salud y su desempeño académico. Martínez, A. R. (2020).

1.5.1. Justificación Técnica

Se desarrollará un sistema web educativo dirigido a los estudiantes, con el objetivo de proporcionarles información valiosa sobre la seguridad de las aplicaciones móviles. Esto responde a la realidad de que la mayoría de los estudiantes poseen dispositivos móviles con sistema operativo Android, y a menudo los utilizan para actividades recreativas, comunicarse y mantenerse conectados a través de redes sociales. Sin embargo, muchos desconocen los riesgos asociados con el uso de aplicaciones móviles, lo que los lleva a descargar aplicaciones desde plataformas de distribución digital sin tener conocimiento sobre las posibles vulnerabilidades que puedan comprometer la seguridad de sus dispositivos. Este desconocimiento puede resultar en la instalación de aplicaciones que contienen configuraciones maliciosas, las cuales pueden alterar el dispositivo y permitir el acceso no autorizado a los datos personales de los usuarios. La necesidad de sensibilizar a los estudiantes sobre estas amenazas es fundamental para reducir los riesgos de seguridad, garantizando que las aplicaciones que utilizan sean confiables y seguras.

1.5.2. Justificación Económica

El desarrollo de este sistema web contribuirá a la orientación de los estudiantes en el uso responsable y seguro de las herramientas de ingeniería inversa. A través de un acceso libre e interactivo al sistema, se busca prevenir los riesgos emocionales y sociales causados por el uso indebido de aplicaciones móviles, como la manipulación por parte de atacantes, la trata y tráfico de personas, o la adicción a contenidos sospechosos y potencialmente peligrosos. Al educar a los estudiantes sobre la seguridad de las aplicaciones y los métodos para identificar vulnerabilidades, se facilita la prevención de estos problemas. De esta manera, el sistema no solo proporciona educación en ciberseguridad, sino que también representa un ahorro significativo en términos de recursos económicos, ya que reduce la necesidad de intervenciones costosas y tratamientos especializados para los afectados por los problemas mencionados, como el tratamiento de la adicción o la asistencia a víctimas de delitos digitales.

1.5.3. Justificación Social

El sistema web centrado en la seguridad de aplicaciones móviles proporcionará a los estudiantes información crucial sobre herramientas de seguridad informática, lo que fomentará una toma de decisiones más informada y responsable al descargar e instalar aplicaciones en sus dispositivos móviles. Esta educación les permitirá comprender los riesgos asociados con las aplicaciones y cómo prevenir vulnerabilidades, promoviendo un entorno más seguro en el uso de tecnologías móviles. Además, esta iniciativa será igualmente beneficiosa para los padres y tutores, quienes al adquirir conocimientos sobre la seguridad digital de sus hijos, podrán ofrecerles un apoyo más efectivo en la protección de su información personal y emocional, brindándoles la tranquilidad necesaria para que utilicen sus dispositivos de forma segura y confiable. De este modo, no solo se incrementa la seguridad individual de los estudiantes, sino que también se fortalece la protección y el bienestar familiar en el entorno digital.

1.6. METODOLOGÍA DE INVESTIGACIÓN

1.6.1. Enfoque de Investigación

Se utilizará un enfoque cuantitativo debido a que el objetivo principal es obtener datos numéricos que permitan medir el nivel de conocimiento de los estudiantes respecto a la seguridad en aplicaciones móviles. Este enfoque es adecuado para evaluar la relación entre el conocimiento y el uso de aplicaciones móviles en los estudiantes. Además, permite realizar un análisis estadístico sobre las prácticas relacionadas con la seguridad móvil. Este enfoque proporciona la base para obtener información objetiva y clara sobre las percepciones y comportamientos de los estudiantes (Sampieri, Fernández, & Baptista, 2014).

1.6.2. Diseño de la Investigación

El diseño será ex post facto correlacional, lo cual significa que se analizarán los datos ya existentes sobre el uso de aplicaciones móviles y la seguridad en ellas, sin manipular ni controlar las variables independientes. El diseño ex post facto es adecuado para estudios que buscan identificar y analizar relaciones entre variables en situaciones que no pueden ser controladas directamente. Se busca establecer la correlación entre el nivel de conocimiento de los estudiantes y sus prácticas de seguridad al usar aplicaciones móviles (Hernández, Fernández & Baptista, 2014; Sampieri, 2014).

1.6.3. Técnicas y Herramientas de Recolección de Datos

Se empleará Google Forms como herramienta para administrar las encuestas de manera eficiente. Esta herramienta permitirá recolectar datos numéricos exactos sobre el conocimiento de los estudiantes sobre la seguridad en aplicaciones móviles y su comportamiento en cuanto a la instalación y uso de aplicaciones. La encuesta será estructurada con preguntas cerradas, incluyendo opciones múltiples, para obtener una medición precisa del nivel de conocimiento y las actitudes de los estudiantes respecto a la seguridad (Google, n.d.).

1.6.4. Población y Muestra

La población estará compuesta por más de 600 estudiantes de la institución educativa seleccionada. Sin embargo, debido a la accesibilidad y a limitaciones logísticas, se seleccionará una muestra de un solo curso, que contará con 33 estudiantes. Esta muestra es representativa del grupo objetivo, lo que permite obtener datos suficientes para el análisis, considerando el rango de edades y el nivel educativo de los estudiantes.

1.6.5. Procedimiento de Investigación

Fase 1: Diseño y validación de la encuesta. En esta fase se creará una encuesta estructurada para evaluar el nivel de conocimiento de los estudiantes sobre la seguridad en aplicaciones móviles (ver Anexo 3).

Fase 2: Aplicación de la encuesta. La encuesta será aplicada a los estudiantes seleccionados del curso elegido. Se hará de manera digital a través de Google Forms, lo que facilitará la recolección de datos.

Fase 3: Recolección de respuestas. Las respuestas se almacenarán automáticamente en una base de datos proporcionada por Google Forms, lo que permitirá acceder a los datos en tiempo real.

Fase 4: Análisis de los datos. Los datos recolectados se analizarán utilizando herramientas estadísticas como Excel para calcular la correlación entre el nivel de conocimiento de los estudiantes y las prácticas relacionadas con la seguridad en las aplicaciones móviles.

1.6.6. Técnicas de Análisis de Datos

El análisis de los datos se realizará utilizando herramientas estadísticas descriptivas. Se emplearán gráficos de barras y otras representaciones visuales para mostrar la distribución de respuestas sobre la seguridad de las aplicaciones móviles. Las frecuencias y porcentajes de cada respuesta se calcularán para obtener una representación clara del nivel de conocimiento y las prácticas de seguridad de los estudiantes. Los resultados se analizarán cualitativamente para identificar tendencias y patrones en las percepciones de los estudiantes (ver anexo 4).

1.7. METODOLOGÍA PARA EL DESARROLLO DEL SOFTWARE

1.7.1. Metodología OOHDM

El método de Diseño Hipermedia Orientado a Objetos (OOHDM), es un enfoque sistemático para el diseño de aplicaciones hipermedia orientadas a objetos. Fue desarrollada para guiar el proceso de diseño de sistemas hipermedia complejos, como sitios web interactivos, sistemas de información y aplicaciones multimedia lo cual será de gran utilidad para el desarrollo del sistema web.

La metodología OOHDM se basa en los siguientes principios y características:

- Orientación a objetos: Utiliza el paradigma de programación orientada a objetos para modelar los elementos del sistema. Los objetos y sus interacciones son la base del diseño.

- Hipermedia: Se enfoca en la creación de sistemas que incluyen diferentes tipos de contenido multimedia (texto, imágenes, video, audio) interconectados mediante enlaces.

- Proceso de diseño estructurado: Proporciona un conjunto de fases y actividades bien definidas que guían desde la conceptualización hasta la implementación y mantenimiento del sistema.

1.6.2. Fases de la metodología OOHDM: El método de Diseño Hipermedia Orientado a Objetos, se compone de cinco fases que guían el diseño de sistemas hipermedia orientados a objetos. Estas fases son: Obtención de requerimientos, Modelo Conceptual, Diseño Navegacional, Diseño de Interfaz Abstracta, e implementación.

Las fases de la metodología son:

Para aplicar la metodología de desarrollo se seguirá las 5 fases del método de Diseño Hipermedia Orientado a Objetos:

- Obtención de Requerimientos: Para el desarrollo del sistema web, tiene como objetivo la orientación sobre cómo prevenir ataques cibernéticos en aplicaciones móviles a los estudiantes de la U.E San Martin de Porres, y así fomentar los pasos para identificar aplicaciones con funciones sospechosas que vulneran la privacidad de los estudiantes.

Mediante las encuestas realizadas al curso 5to “B” del Colegio, se pudo observar que la mayoría posee un dispositivo móvil y accede a aplicaciones móviles ya sea en redes sociales, mensajería y navegación web.

- Modelo Conceptual: Después del análisis y obtención de datos, el sistema web tendrá una vista de las herramientas de seguridad informática en la que estarán los datos más esenciales para hacer un buen análisis y reconocimiento de aplicaciones móviles y sus funcionamientos dentro del dispositivo móvil.

  También tendrá acceso a toda la información sobre informática de ingeniería inversa en aplicaciones móviles con diversas herramientas de acceso libre o instalación de software en ordenadores para que así el estudiante pueda manejar los recursos de la mejor manera y así evitar ataques cibernéticos.

- Diseño Navegacional: El sistema web será acceso libre donde el estudiante podrá acceder sin necesidad de registrar una cuenta, dentro del sistema accederán al contenido que ayudara a los estudiantes a ver cómo funciona las herramientas mediante laboratorios y ejercicios prácticos ya que contara con acceso a las páginas y secciones de para realizar el estudio sobre las aplicaciones móviles.

- Diseño de Interfaz Abstracta: Las representaciones visuales que tendrá el sistema web, será una sección de bienvenida, botones de acción, categoría de herramientas y sus descripciones, enlaces o descargas de herramientas, valoración de usuarios, etc.

- Implementación: Al implementar el sistema web a la Unidad Educativa San Martin de Porres, los estudiantes podrán ingresar a la página de inicio con una breve introducción de sobre la importancia de la seguridad en aplicaciones móviles y una introducción sobre el uso para navegar en el sistema.

También tendrá información sobre el por qué es vital proteger la información, dando descripción sobre las amenazas y riesgos en las aplicaciones móviles y dando consejos sobre contraseñas seguras, actualizaciones de software, entre otros.

El sistema web contara con videos informativos que explicarán los conceptos sobre como interactuar con las herramientas de seguridad informática en aplicaciones móviles, donde los estudiantes podrán hacer consultas y e interactuar mediante comentarios y prácticas de seguridad informática.

El sistema está diseñado para ser intuitivo y atractivo para los estudiantes del colegio, ofreciendo contenido educativo relevante, interacciones interactivas, y herramientas para evaluación y mejora continua en seguridad de aplicaciones móviles.

1.7.2. Metodología de desarrollo de software

1.7.2.1. UML (Lenguaje de Modelado Unificado)

El Lenguaje de Modelado Unificado (UML por sus siglas en inglés) es un conjunto de notaciones y herramientas de un sistema de software, utiliza varios tipos de diagramas para representar distintos aspectos y perspectivas de un sistema de software.

Los diagramas se utilizan para visualizar un sistema desde diferentes perspectivas, de forma que un diagrama es una proyección de un sistema. Representan una vista resumida de los elementos que contribuyen un sistema. UML tiene nueve diagramas fundamentales, agrupados en dos grandes grupos, uno para modelar la estructura estática del sistema y otro para modelar el comportamiento dinámico (Universidad de Alcalá, 2002).

Para el desarrollo se utilizarán los diagramas que proporciona la metodología UML:

Modelado de la Estructura Estática del Sistema

- Diagrama de Clases: Representa la estructura estática del sistema, mostrando las clases, sus atributos, métodos y relaciones entre ellas.
- Diagrama de Objetos: Muestra una instancia específica del sistema, enfocándose en los objetos y sus relaciones en un momento particular.
- Diagrama de Componentes: Describe los componentes del sistema y sus relaciones, enfocándose en los aspectos físicos y de implementación.
- Diagrama de Despliegue: Representa la disposición física de los artefactos de software en el hardware, mostrando nodos y sus conexiones.

Modelado del Comportamiento Dinámico del Sistema

- Diagrama de Casos de Uso: Representa las interacciones entre usuarios y el sistema, identificando casos de uso y actores.
- Diagrama de Secuencia: Describe la secuencia de interacciones entre objetos a lo largo del tiempo, mostrando cómo se comunican.

- Diagrama de Colaboración (Comunicación): Muestra cómo los objetos colaboran entre sí para cumplir un objetivo, resaltando las relaciones entre objetos.

- Diagrama de Estados: Modela el ciclo de vida de un objeto, sus estados y las transiciones entre ellos.

- Diagrama de Actividades: Representa el flujo de control entre actividades, mostrando procesos o acciones secuenciales.

Estos diagramas se utilizarán en diferentes etapas del desarrollo del Sistema Web, para capturar y comunicar aspectos estructurales y comportamentales del sistema.

1.7.3. Pruebas de Software

Las pruebas que podrían ser relevantes para un sistema desarrollado con OOHDM se pueden aplicar diferentes tipos de pruebas para verificar su funcionalidad y calidad.

- Pruebas de Unidad: Verifican el funcionamiento individual de cada unidad o componente del sistema hipermedia para asegurar que funcionen según lo esperado.

- Pruebas de Integración: Evalúan la interacción entre los diversos componentes del sistema para asegurar que se integren correctamente y funcionen juntos como un todo coherente.

- Pruebas de Aceptación del Usuario: Se realizan para confirmar que el sistema cumple con los requisitos del usuario final y que es fácil de usar.

- Pruebas de Seguridad: Evalúan la seguridad del sistema, buscando vulnerabilidades y comprobando la resistencia a posibles ataques.

- Pruebas de Rendimiento: Verifican el rendimiento del sistema, incluyendo su velocidad, capacidad de respuesta y capacidad para manejar un determinado número de usuarios o carga de trabajo.

- Pruebas de Usabilidad: Evalúan la facilidad de uso y la experiencia del usuario para garantizar que el sistema sea intuitivo y fácil de navegar.

1.8. HERRAMIENTAS

Herramientas necesarias para el desarrollo del sistema web:

- XAMPP (v 3.3): Es una herramienta de software desarrollada por Apache Friends, donde proporciona un entorno de desarrollo local que incluye Apache, MySQL, PHP y Perl. *Apache Friends. (2023). XAMPP Apache + MariaDB + PHP + Perl. https://www.apachefriends.org/index.html*

- Composer (v 2.3): Es una herramienta de administración de dependencias para PHP creada por Nils Adermann y Jordi Boggiano. *Adermann, N. & Boggiano, J. (2022). A Dependency Manager for PHP. https://getcomposer.org/*

- Visual Studio Code (v 1.84): Es un editor de código fuente desarrollado por Microsoft. Se ha vuelto muy popular entre los desarrolladores debido a su interfaz fácil de usar, su amplia gama de funcionalidades y su gran cantidad de extensiones disponibles. *Redmond, WA: Microsoft Corporation*

- Laravel (v 9): Es un marco de aplicación web de código abierto para php, creado principalmente por Taylor Otwell y contribuido por una comunidad de desarrolladores, sigue el patrón de arquitectura MVC (Modelo-Vista-Controlador). *Otwell, T. (2023). The PHP Framework for Web Artisans. Laravel. https://laravel.com/*

- Bootstrap (v 4.6): Es un framework de código abierto creado por Mark Otto y Jacob Thornton, desarrolladores de Twitter. *Otto, M. (2010). Build fast, responsive sites with Bootstrap. getbootstrap. https://getbootstrap.com/*

- Node.js: Es un entorno de ejecución de JavaScript creado por Ryan Dahl, utiliza el motor V8 de Google Chrome para ejecutar código JavaScript fuera del navegador, lo que significa que se puede usar para desarrollar aplicaciones de servidor, herramientas de línea de comandos e incluso aplicaciones de escritorio. *Dahl, R. (2009). Node.js® is an open-source, cross-platform JavaScript runtime environment. nodejs. https://nodejs.org/*

- CSS (Cascading Style Sheets): Es un estándar web desarrollado por un grupo de trabajo de World Wide Web Consortium (W3C). *Bos, B. (2023). Cascading Style Sheets. w3. https://www.w3.org/Style/CSS/*

- HTML (HyperText Markup Language): Es un lenguaje estándar para crear páginas web y su desarrollo se atribuye principalmente a un grupo de trabajo de World Wide Web Consortium (W3C), liderado por Tim Berners-Lee, quien es reconocido como uno de los creadores de la web. *Berners, T. (2023). HTML Living Standard. HTML. https://www.w3.org/TR/html/*

- PHP (Hypertext Preprocessor): Es un lenguaje de programación de código abierto que fue creado inicialmente por Ramus Lerdorf en 1994. Sin embargo, su desarrollo ha sido llevado a cabo por un grupo de contribuyentes en la comunidad de código abierto. *Lerdorf, R. (2023). Php A popular general-purpose scripting language that is especially suited to web development. Fast, flexible and pragmatic, PHP powers everything from your blog to the most popular websites in the world. php. https://www.php.net/*

1.9. LÍMITES Y ALCANCES

1.9.1. Limites

EL sistema web se limitará a dispositivos u ordenadores sin acceso a internet ya que importante contar con un punto de acceso a internet y así navegar el Sistema Web de manera eficiente.

El sistema web solo tendrá compatibilidad con las aplicaciones con extensión “APK”, ya que la mayoría de los estudiantes según las encuestas tienen dispositivos móviles con el sistema operativo Android.

1.9.2. Alcances

Módulos educativos:

- Introducción a la seguridad en aplicaciones móviles: Información básica sobre amenazas comunes, prácticas de seguridad y consejos para proteger dispositivos móviles.

- Protección de datos personales y privacidad: Guía sobre la importancia de la privacidad y cómo configurar adecuadamente las opciones de privacidad en aplicaciones y dispositivos móviles.

- Seguridad en contraseñas y autenticación: Enseñanza sobre la creación de contraseñas sólidas y la implementación de autenticación de dos factores en aplicaciones y dispositivos.

- Identificación y prevención de malware: Información sobre cómo reconocer y evitar la instalación de aplicaciones maliciosas, virus y malware en dispositivos móviles.

- Concientización sobre ataques de phishing: Explicación de cómo identificar y evitar ser víctima de ataques de phishing en dispositivos móviles.

Interactividad y recursos:

- Simulaciones y ejercicios interactivos: Creación de ejercicios prácticos y simulaciones para que los estudiantes practiquen habilidades de seguridad.

- Vídeos y tutoriales interactivos: Desarrollo de contenido multimedia para explicar conceptos de seguridad de manera más dinámica y visual.

Funcionalidades del sistema:

- Foros y comunidades: Espacios interactivos donde los estudiantes puedan hacer preguntas, intercambiar información y discutir temas relacionados con la seguridad móvil.

- Evaluaciones y pruebas de conocimiento: Cuestionarios o evaluaciones interactivas para que los estudiantes puedan poner a prueba su comprensión sobre seguridad móvil.

Características de implementación:

- Diseño responsivo y accesibilidad: Garantizar que el sistema sea accesible desde dispositivos móviles y adaptado a diferentes tamaños de pantalla.

- Análisis y retroalimentación: Herramientas de análisis para evaluar el uso del sistema y recopilar retroalimentación de los estudiantes para mejorar el contenido y la experiencia de usuario.

Este enfoque puede ayudar a los estudiantes del colegio a desarrollar habilidades esenciales en seguridad móvil a través de un sistema educativo interactivo, proporcionándoles conocimientos fundamentales para proteger sus dispositivos y datos personales.

1.10. APORTES

Un sistema web en Laravel para estudiantes de colegio complementa la seguridad en aplicaciones móviles al ofrecer un acceso seguro y centralizado a información educativa. Al proveer una plataforma web, se minimizan riesgos inherentes a las vulnerabilidades de las aplicaciones móviles, asegurando un entorno más controlado y protegido para la gestión de datos sensibles de los estudiantes.

El sistema web tendrá características evolutivas aplicando nuevos módulos educativos y actualizando los contenidos conforme evolucionen las amenazas de seguridad en aplicaciones móviles, incluirá módulos de aprendizaje, consejos de seguridad y materiales educativos destinados a aumentar la conciencia sobre las amenazas potenciales y como proteger su información personal.

CAPITULO II

MARCO TEÓRICO

INGENIERÍA
DE SISTEMAS
UNIVERSIDAD PUBLICA DE EL ALTO

CAPITULO II

MARCO TEÓRICO

2.1. INTRODUCCIÓN

La seguridad en aplicaciones, tanto móviles como web, es crucial en un mundo donde cada vez más datos sensibles y personales están en juego. Proteger la información de los usuarios no solo implica implementar tecnologías avanzadas; también requiere establecer políticas de seguridad sólidas que fomenten la confianza y resguarden la privacidad. En el ámbito educativo, donde el uso de aplicaciones móviles y sistemas web se ha vuelto cotidiano, garantizar la seguridad de estos entornos se convierte en una necesidad urgente para evitar ciberataques y salvaguardar la integridad de los estudiantes. Esto pone de relieve la importancia de incorporar medidas de seguridad efectivas, como la validación de datos y el control de accesos, para crear un espacio seguro que favorezca el aprendizaje y el desarrollo personal de todos los usuarios.

- **Sistema:** Un "sistema" se define como un conjunto de componentes interrelacionados que trabajan juntos para alcanzar un objetivo común, permitiendo la interacción de los elementos de forma ordenada y eficiente (Stair & Reynolds, 2021). En este contexto, el sistema será una plataforma diseñada para monitorear y gestionar la seguridad en el uso de aplicaciones móviles, protegiendo a los usuarios ante posibles vulnerabilidades tecnológicas. El término también implica que el sistema tiene un enfoque holístico, integrando distintos módulos y funcionalidades para cumplir su propósito de protección y educación en seguridad digital (O’Brien & Marakas, 2019).

- **Web:** El término "web" hace referencia al entorno digital que facilita la conectividad y el acceso a información mediante Internet, específicamente utilizando la World Wide Web (Berners-Lee et al., 2006). Una plataforma web es accesible desde cualquier dispositivo con conexión a Internet, lo que permite que los estudiantes y otros usuarios accedan al sistema sin limitaciones físicas o geográficas. En el caso del proyecto, la web será el medio por el cual se desplegará el sistema de seguridad, asegurando que los usuarios puedan ingresar y utilizar las funciones de monitoreo y protección en tiempo real (He & Freeman, 2020).

- **Seguridad en Aplicaciones Móviles:** La seguridad en aplicaciones móviles implica una serie de prácticas y tecnologías destinadas a proteger las aplicaciones de accesos no autorizados, malware y otras amenazas de seguridad. Según la OWASP Foundation (2023), estas medidas son especialmente importantes en aplicaciones que manejan datos personales o educativos, ya que la falta de controles adecuados puede poner en riesgo la privacidad de los usuarios. Las aplicaciones móviles son altamente susceptibles a ciberataques debido a su distribución masiva y al acceso a información sensible, lo que resalta la necesidad de implementar autenticación segura, cifrado de datos y auditorías de seguridad continuas. Además, estudios de Symantec (2021) y Android Developers (2021) enfatizan que estas prácticas no solo protegen los datos de los usuarios, sino que también ayudan a prevenir la exposición a contenido inapropiado y el riesgo de ciberacoso, especialmente en entornos educativos.

- **Impacto en Estudiantes Vulnerables:** El "impacto en estudiantes vulnerables" se refiere a los efectos adversos que puede tener el uso de tecnologías digitales en aquellos alumnos que carecen de recursos, educación o conocimientos previos en ciberseguridad. Según un informe de Symantec (2021), los estudiantes en situaciones de vulnerabilidad enfrentan mayores riesgos de sufrir ciberacoso, pérdida de privacidad y exposición a contenido inapropiado, ya que suelen tener menos acceso a educación en seguridad digital. Además, estudios de la UNICEF (2020) destacan que estos alumnos son más susceptibles a ser víctimas de ataques digitales debido a la falta de supervisión adecuada y conocimientos sobre prácticas seguras en internet. Este proyecto busca mitigar estos riesgos proporcionando un entorno digital seguro y educativo que permita a los estudiantes y sus familias aprender sobre la importancia de la ciberseguridad y así proteger su privacidad en el entorno digital.

Posibles Riesgos Tecnológicos: Los "riesgos tecnológicos" abarcan una variedad de amenazas que enfrentan los usuarios de dispositivos y servicios digitales, incluyendo la pérdida de privacidad, la exposición a malware, y la vulnerabilidad a ataques de phishing o ciberacoso. Según un informe de la Agencia Europea de Seguridad de Redes y de la Información (ENISA, 2021), los riesgos tecnológicos son particularmente preocupantes en contextos educativos, donde los estudiantes, especialmente los más jóvenes o aquellos con acceso limitado a educación en ciberseguridad, son más vulnerables a ataques y manipulaciones digitales. Estos riesgos se agravan debido a la falta de formación en seguridad digital en muchas instituciones educativas. El proyecto, por lo tanto, también examina y aborda estos riesgos, proporcionando un sistema de monitoreo y defensa que contribuye a crear un entorno más seguro para los estudiantes y otros usuarios.

2.2. DEFINICIÓN Y FUNCIONALIDADES DE LOS SISTEMAS WEB

Un sistema web se define como un conjunto de aplicaciones o servicios que se ejecutan a través de servidores web y son accesibles desde cualquier dispositivo con conexión a internet (Berners-Lee, 1996). Estos sistemas permiten la interacción de múltiples usuarios de forma simultánea, independientemente de su ubicación geográfica. Según Berners-Lee (1996), los sistemas web democratizan el acceso a la información y proporcionan una plataforma unificada para el intercambio de datos a nivel global.

Entre las principales funcionalidades de un sistema web se encuentran:

- Accesibilidad Universal: Los usuarios pueden acceder al sistema desde cualquier lugar y dispositivo con conexión a internet (W3C, 2008).

- Gestión de Información: Los sistemas web permiten almacenar, procesar y recuperar datos de manera eficiente a través de bases de datos centralizadas (Beal, 2020).

- Seguridad: En sistemas que manejan datos sensibles, como los que se desarrollan para aplicaciones móviles, la seguridad es un componente crítico, asegurando que solo los usuarios autorizados puedan acceder a información confidencial (ISO/IEC 27000, 2018).

2.2.1. Importancia de los Sistemas Web en el Entorno Educativo

Los sistemas web han revolucionado el ámbito educativo, facilitando el acceso remoto a materiales de enseñanza y promoviendo la interacción entre estudiantes y docentes. Según un estudio de la OECD (2022), el uso de plataformas web permite actividades colaborativas, evaluaciones en línea y el acceso a recursos multimedia, lo que enriquece significativamente la experiencia de aprendizaje. Además, herramientas interactivas como foros de discusión y módulos de evaluación son elementos clave que fomentan el aprendizaje activo y la participación de los estudiantes. Un aspecto crucial de este proyecto es la integración de tutoriales interactivos sobre ciberseguridad, que ayudan a los estudiantes a aprender de manera autodidacta sobre la protección digital (UNESCO, 2021).

Para reforzar la seguridad en el entorno educativo, el proyecto también utiliza herramientas como APKTool, una aplicación que permite la ingeniería inversa de archivos APK de Android. Esta herramienta descompila las aplicaciones, exponiendo su estructura interna y los permisos solicitados, lo cual es fundamental para identificar comportamientos sospechosos y evaluar riesgos de seguridad. Según la OWASP Mobile Security Testing Guide (2023), el uso de APKTool facilita la detección de vulnerabilidades y ayuda a garantizar que el entorno educativo sea seguro y controlado para los estudiantes.

2.2.1.1. Conceptos de Términos Clave en el Desarrollo del Trabajo de Grado

A continuación, se definen términos clave para el desarrollo del proyecto:

- **Ciberataques en Dispositivos Móviles:** Son ataques informáticos dirigidos a dispositivos móviles con el objetivo de comprometer la seguridad de los datos personales o alterar el comportamiento de las aplicaciones. Según Floyd (2006), la naturaleza ad hoc de las redes móviles facilita la penetración de estos ataques, especialmente cuando los dispositivos carecen de protocolos de seguridad robustos.

- **Educación en Seguridad Informática:** Se refiere a la capacitación de los usuarios en prácticas de protección digital, lo cual es esencial en entornos educativos donde los estudiantes están expuestos a diversos riesgos digitales. Según el informe de la UNESCO (2021), la educación en ciberseguridad permite a los estudiantes identificar vulnerabilidades, comprender los riesgos asociados con el uso de dispositivos digitales y tomar medidas preventivas para proteger sus datos. La capacitación en seguridad informática es fundamental para equipar a los estudiantes con las herramientas necesarias para navegar de manera segura en el entorno digital, especialmente a medida que los ataques cibernéticos se vuelven más sofisticados y prevalentes.

- **Aplicaciones Sospechosas y Permisos Móviles:** Se refiere a aquellas aplicaciones que solicitan permisos innecesarios o presentan conductas anómalas que podrían comprometer la privacidad del usuario. Balapour et al. (2020) señalan que la percepción de privacidad influye en la seguridad percibida por los usuarios, destacando la importancia de evaluar cuidadosamente los permisos solicitados antes de instalar una aplicación.

2.3. REVISIÓN DE TRABAJOS Y REPOSITORIOS RELACIONADOS

La revisión de trabajos previos y repositorios de código abierto permite contextualizar el desarrollo del presente proyecto, proporcionando una base sólida para la implementación de sistemas web centrados en la seguridad de aplicaciones móviles. A continuación, se detallan los hallazgos más relevantes.

2.3.1. Revisión de Trabajos de Grado

- **Análisis de seguridad en aplicaciones móviles (OWASP, 2023)**: Se ha estudiado ampliamente el uso de herramientas como el **Mobile Security Framework (MobSF)** para realizar auditorías de seguridad en aplicaciones móviles, identificando vulnerabilidades y riesgos potenciales. Este enfoque es relevante para el proyecto actual, ya que permite integrar un análisis de seguridad profundo en el desarrollo de aplicaciones móviles (OWASP, 2023).

- **Iniciativas globales de ciberseguridad en educación (UNESCO, 2021)**: La UNESCO ha impulsado programas educativos que abordan la importancia de educar a los estudiantes sobre los riesgos digitales y promover la seguridad en el uso de tecnologías. Estos programas son fundamentales para el contexto educativo en el que se enmarca el presente proyecto, subrayando la necesidad de integrar prácticas de seguridad en el entorno escolar (UNESCO, 2021).

2.3.2. Revisión de repositorios

Además de los trabajos académicos, se analizaron repositorios de código abierto en plataformas como GitHub que ofrecen ejemplos prácticos de gestión de usuarios, roles y seguridad en sistemas web:

- **Crater Invoice (Laravel & Vue.js)**: Una aplicación de facturación que incluye un sistema avanzado de gestión de usuarios y seguridad. Su estructura de roles y permisos es una referencia para la implementación de controles de acceso en el proyecto actual (Crater, 2020).

- **Bagisto (Laravel eCommerce)**: Una plataforma de comercio electrónico con un robusto sistema de gestión de usuarios, roles y permisos. Su enfoque en la autenticación y control de acceso es útil para la creación de sistemas que manejen múltiples tipos de usuarios, como superadministradores, administradores y estudiantes (Bagisto, 2021).

- **Spatie Laravel-Permission**: Un paquete ampliamente utilizado en Laravel para la gestión de permisos y roles. Su flexibilidad para definir permisos de forma granular se adapta bien al sistema propuesto, permitiendo un control preciso sobre las acciones permitidas a cada usuario según su rol (Spatie, 2022).

2.3.3. Conclusión de la Revisión

La revisión de trabajos de grado y repositorios de código abierto proporciona una base sólida para el desarrollo del sistema web propuesto. Estos estudios y recursos destacan la importancia de implementar sistemas seguros que protejan los datos de los usuarios y controlen el acceso mediante roles y permisos. La combinación de enfoques académicos y herramientas prácticas permite una comprensión integral de los desafíos y soluciones asociados con la seguridad en sistemas web.

2.4. METODOLOGÍAS

2.4.1. Metodología OOHDM

La metodología Object-Oriented Hypermedia Design Method (OOHDM) se implementa en el diseño de este sistema web para crear una plataforma educativa centrada en la seguridad en aplicaciones móviles. Este sistema tiene como objetivo capacitar a estudiantes y miembros de la comunidad educativa (incluidos padres y personal administrativo) en prácticas seguras para el uso y desarrollo de aplicaciones. OOHDM permite estructurar el contenido de manera intuitiva, facilitando el acceso a recursos sobre seguridad en aplicaciones móviles.

De este modo, los usuarios comprenden conceptos clave, como la protección de datos y la gestión de permisos, fundamentales para la seguridad y privacidad en el entorno digital (Schwabe & Rossi, 1995).

Justificación de la Selección de OOHDM

La elección de OOHDM es particularmente relevante debido a su enfoque orientado a objetos, que facilita la representación y gestión de relaciones complejas entre usuarios (estudiantes, docentes, padres) y sus niveles de acceso y permisos. Este enfoque permite estructurar la información de forma lógica y coherente. Además, OOHDM posibilita la creación de módulos que categorizan los permisos de las aplicaciones en niveles de riesgo (bajo, medio, alto), ayudando a los estudiantes a evaluar los riesgos asociados a las aplicaciones antes de instalarlas, lo cual es crucial en la educación digital (Schwabe & Rossi, 1995). Este sistema proporciona un aprendizaje seguro y efectivo, promoviendo prácticas de ciberseguridad en la comunidad educativa.

2.4.2. Diagrama UML y Aplicación en OOHDM

El Lenguaje Unificado de Modelado (UML) complementa la metodología OOHDM en el desarrollo de una plataforma educativa de seguridad en aplicaciones móviles. A través de UML, se puede visualizar de manera estructurada la arquitectura y los procesos de interacción del sistema, facilitando la implementación y comprensión de cada fase de OOHDM.

2.4.3. Fases de la Metodología OOHDM

La metodología Object-Oriented Hypermedia Design Method (OOHDM) se organiza en cinco fases fundamentales para el diseño y desarrollo de sistemas web educativos y de hipertexto. Cada fase aborda un aspecto crítico del sistema, desde la estructura conceptual hasta la implementación operativa. A continuación, se detallan cada una de estas fases aplicadas al desarrollo de un sistema educativo de seguridad en aplicaciones móviles (Schwabe & Rossi, 1995).

2.4.3.1. Fase 1: Obtención de requerimientos

Constituye la fase inicial y se enfoca en definir la estructura de datos y la lógica de negocio del sistema. Aquí, se identifican los actores y casos de uso principales, estableciendo una base que refleja las necesidades educativas y los objetivos de aprendizaje. En este sistema web, los actores clave incluyen:

- **Estudiante:** Es el usuario principal que busca aprender sobre la seguridad en aplicaciones móviles. Puede acceder a contenidos educativos y registrar su progreso a través de formularios.

- **Administrador:** Se encarga de gestionar el contenido educativo y garantizar el correcto funcionamiento de la plataforma. Además, supervisa el progreso de los estudiantes y la calidad del contenido.

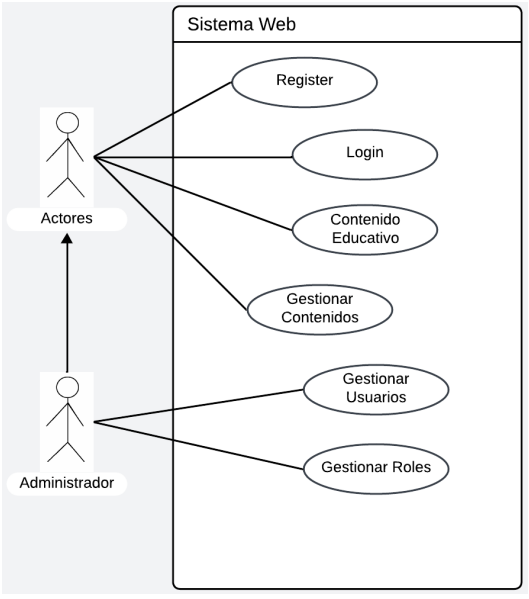
- **Director:** Proporciona información sobre avisos, noticias y eventos del colegio, apoyando la comunicación institucional y promoviendo la importancia de la seguridad en aplicaciones móviles.

- **Docentes:** Promueven el aprendizaje sobre cómo protegerse en aplicaciones móviles y facilitan la comprensión de los contenidos. También son responsables de comunicar eventos y noticias relevantes.

- **Secretarios y Asistentes:** Proporcionan información y ayudan en la logística del sistema educativo, encargándose de la difusión de avisos y eventos escolares.

- **Padres de Familia:** Acceden a información sobre seguridad en aplicaciones móviles, así como a avisos, noticias y eventos del colegio. Se mantienen informados sobre actividades en el entorno educativo.

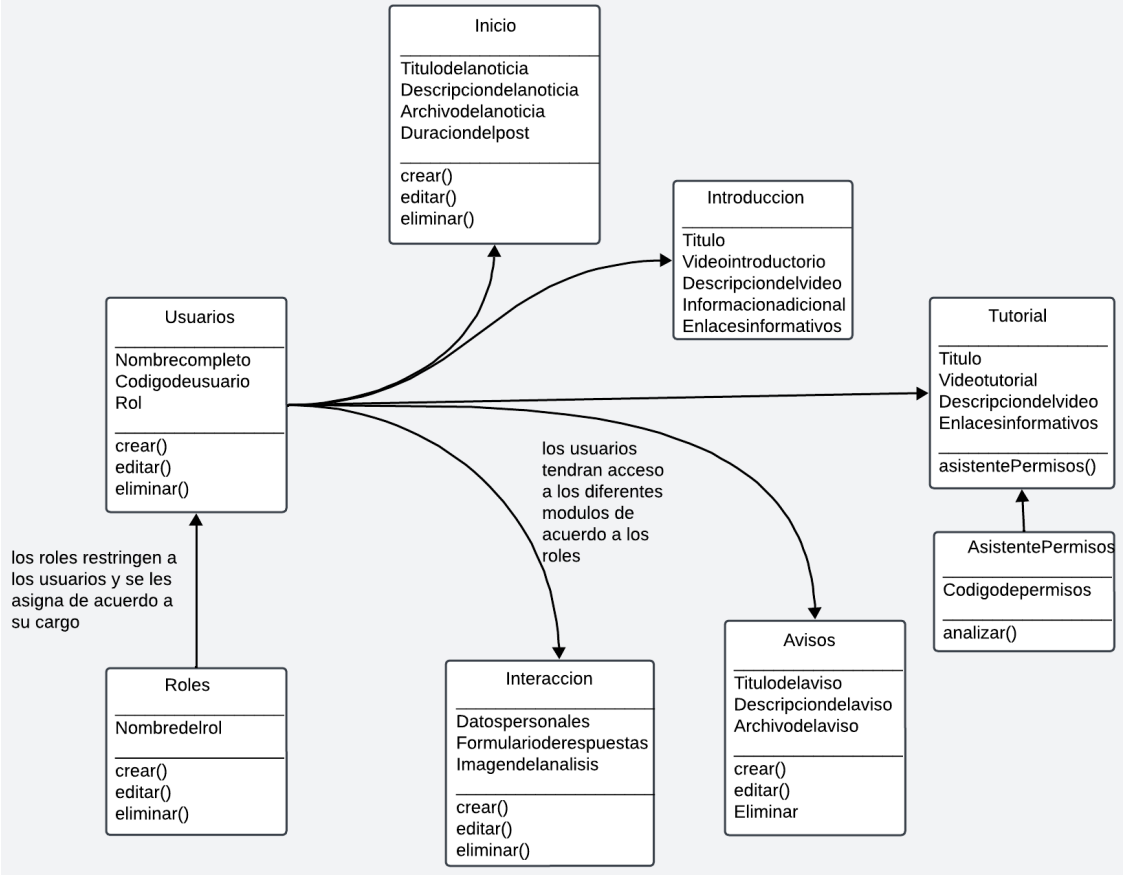
Figura 2.
Diseño de Casos de Uso



Nota. Diagrama de casos de uso que muestra los actores interactuando con el sistema.

Estos actores interactúan con el sistema a través de casos de uso definidos, tales como Acceso a Módulos de Seguridad, Visualización de Tutoriales, y Análisis de Permisos en AndroidManifest.xml, lo cual proporciona una estructura clara y funcional para el sistema (Schwabe & Rossi, 1995).

Figura 3:
Ejemplo del Modelo Conceptual



Nota. Este diagrama muestra las principales entidades y relaciones del sistema, incluyendo las funciones de los usuarios y las vistas de la plataforma.

2.4.3.2. Fase 2: Modelo conceptual

En el modelo conceptual, se define la arquitectura de navegación, permitiendo que los usuarios se desplacen de manera intuitiva entre los distintos módulos del sistema. La navegación adecuada es crucial para que los usuarios encuentren y accedan fácilmente a los recursos educativos y herramientas de análisis. La estructura modular incluye:

Módulo de Autenticación: Para registro e inicio de sesión.

Módulo de contenido Educativo: Acceso a contenidos educativos sobre seguridad en aplicaciones móviles.

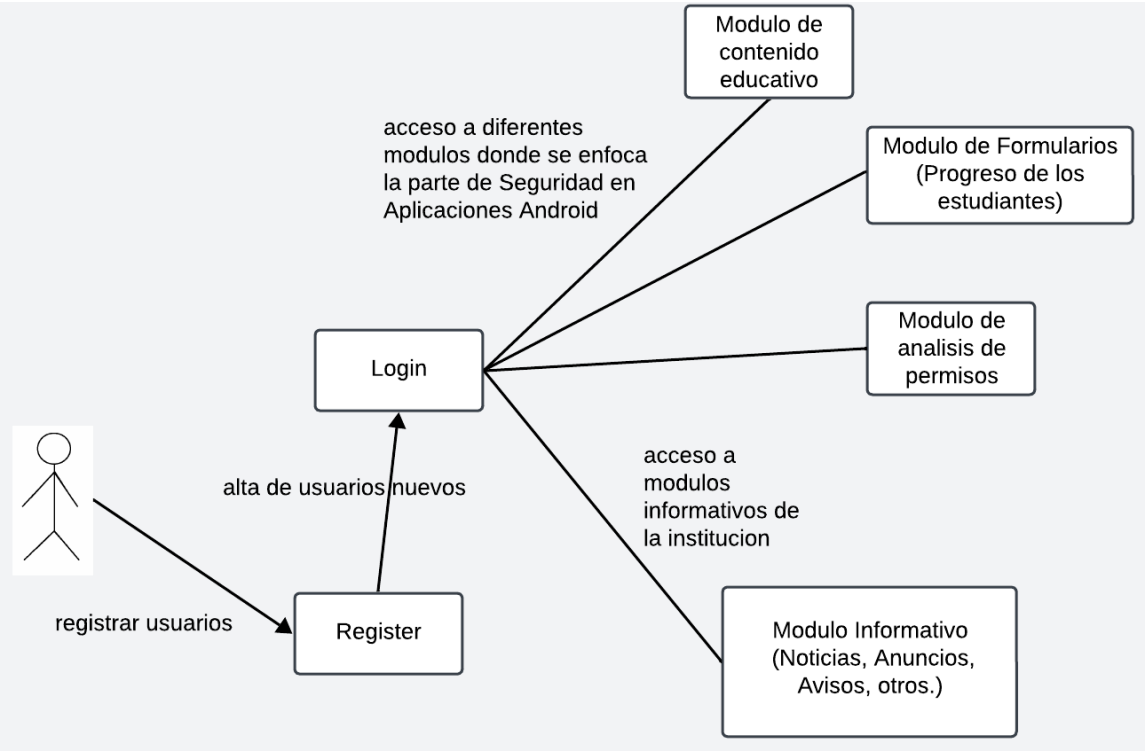
Módulo de Análisis de Permisos: Proporciona herramientas para analizar permisos de aplicaciones Android encontradas en el archivo AndroidManifest.xml.

Módulo de Formularios de Aprendizaje: Donde los estudiantes documentan su aprendizaje y progresión.

Modulo Informativo Donde los usuarios registrados podrán enterarse de las novedades que pasa en la institución.

Este diseño de navegación organiza los menús y enlaces, asegurando rutas claras que faciliten la experiencia de aprendizaje y alineando la estructura del sistema con los objetivos educativos (Garzotto et al., 1993).

Figura 4.
Ejemplo de Modelo Navegacional



Nota. Este diagrama muestra como el usuario navegara dentro del sistema.

2.4.3.3. Fase 3: Diseño navegacional

El diseño navegacional especifica la estructura visual y funcional de los componentes de la interfaz, sin considerar aún detalles gráficos. En esta fase, se definen elementos como menús, botones y formularios, organizándolos para facilitar la interacción del usuario. Para este proyecto, se emplea Blade como motor de plantillas en Laravel, permitiendo vistas dinámicas y coherentes con una disposición clara de los módulos educativos. Los componentes de la interfaz incluyen:

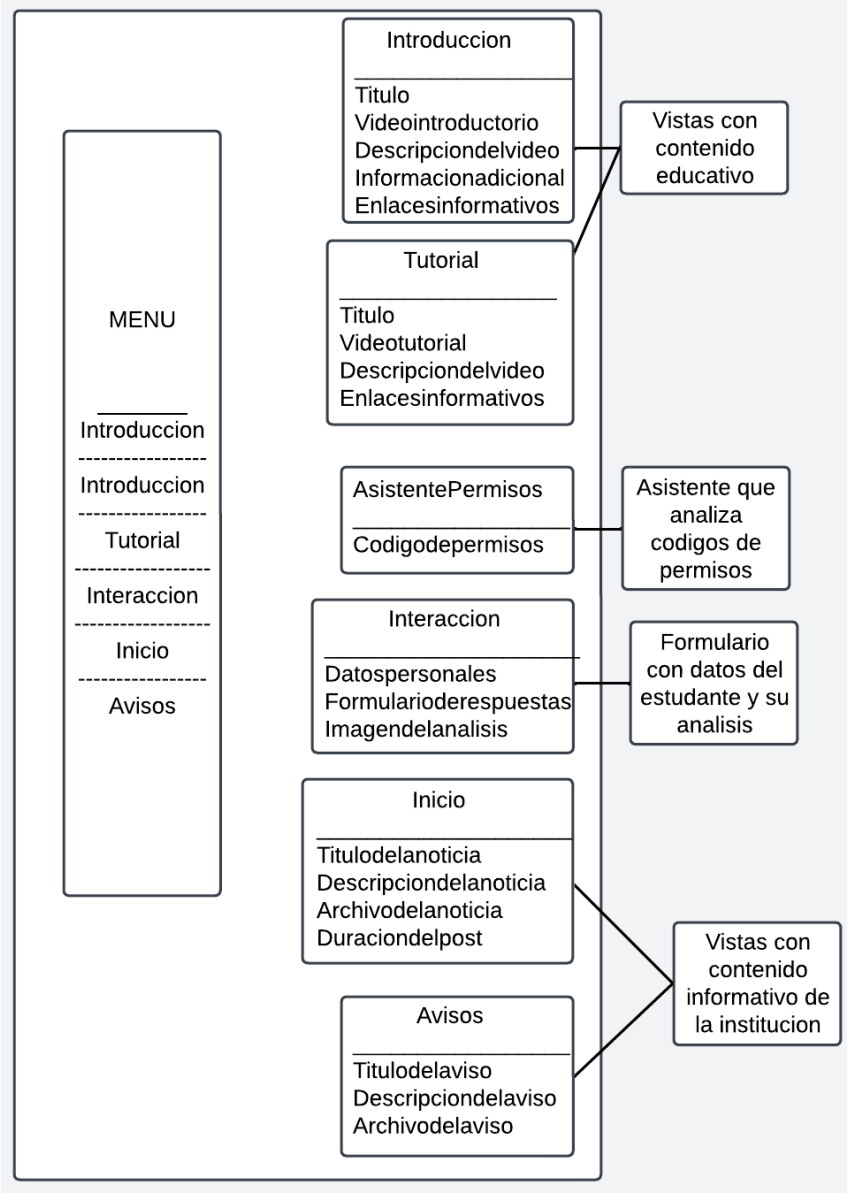
Explorando la Seguridad en los Celulares: Video educativo introductorio.

Aprende a Decompilar con "APKTool" y Asistente de Análisis de Permisos: Tutoriales y herramientas de análisis de permisos en aplicaciones móviles.

Formulario de Evaluación: Documento de progreso que los estudiantes completan, registrando su aprendizaje y opiniones.

Esta fase se enfoca en hacer que la experiencia de usuario sea intuitiva y que los usuarios puedan acceder a cada módulo educativo sin dificultad (Rossi et al., 1996).

Figura 5.
Ejemplo de Modelo de Presentación del Sistema



Nota. Este modelo del sistema muestra la interfaz principal del sistema web

2.4.3.4. Fase 4: Diseño de la Interfaz Abstracta

En el diseño de la interfaz abstracta, se concreta el diseño abstracto en elementos visuales específicos. Se eligen colores, tipografías y estilos gráficos que hacen que la interfaz sea atractiva y funcional, asegurando una experiencia de usuario positiva. En un sistema educativo, los detalles visuales ayudan a mantener la atención y facilitan la navegación en el contenido. En este proyecto, se utilizan:

Estilos de Colores y Fuentes: Que mejoran la legibilidad y organización.

Interacción Visual: Elementos visuales destacados para secciones importantes, como el acceso al asistente de análisis y formularios.

Esta fase asegura que el sistema sea visualmente coherente y que los elementos estéticos refuercen la funcionalidad y accesibilidad del sistema (Schwabe & Rossi, 1995).

2.4.3.5. Fase 5: Implementación

La fase de implementación convierte los modelos de diseño en un sistema operativo. Utilizando Laravel 8 y herramientas como HTML, CSS y JavaScript, los desarrolladores crean la estructura del sistema, integrando módulos de navegación, análisis y tutoriales. Durante la implementación, se realizan pruebas unitarias y de integración para asegurar la correcta interacción entre módulos. Además, se recolecta retroalimentación de usuarios finales, como estudiantes y administradores, permitiendo ajustes que optimicen la funcionalidad y usabilidad del sistema.

Esta fase final asegura que el sistema esté listo para su uso real, cumpliendo los objetivos propuestos y brindando un entorno seguro y educativo para los usuarios (Schwabe & Rossi, 1995).

2.5. HERRAMIENTAS DE DESARROLLO

Para el desarrollo del sistema web educativo centrado en la seguridad de aplicaciones móviles, se han utilizado una serie de herramientas tecnológicas que facilitan tanto la construcción del sistema como su ejecución. Las herramientas seleccionadas aseguran que el sistema sea eficiente, escalable y fácil de mantener, proporcionando a los desarrolladores las capacidades necesarias para gestionar roles, permisos, bases de datos, y la interacción de los usuarios. Las herramientas para la realización del desarrollo del proyecto son:

- **Framework Laravel**: Se utilizará Laravel en su versión 8.83.27, un framework PHP conocido por su elegancia y facilidad para desarrollar aplicaciones web robustas (Laravel Documentation, 2023, https://laravel.com).

- **Servidor Local**: Para el desarrollo, se empleó XAMPP versión 3.2.2, un entorno que incluye Apache, MySQL y PHP, lo que permite ejecutar aplicaciones de manera local (Apache Friends, n.d., https://www.apachefriends.org).

- **Base de Datos**: La base de datos utilizada fue MySQL versión 10.1.37-MariaDB, que ofrece un sistema de gestión de bases de datos relacional (MariaDB Foundation, n.d., https://mariadb.org).

- **Gestión de Dependencias**: Se utilizó Composer en su versión 2.7.7, una herramienta para la gestión de dependencias en PHP que permite instalar y actualizar bibliotecas (Composer, n.d., https://getcomposer.org).

- **Frontend**: La plataforma Node.js fue utilizada en su versión 16.20.2, permitiendo la gestión de dependencias frontend y la optimización de recursos (Node.js, n.d., https://nodejs.org).

- **Frameworks y Bibliotecas CSS**: Se integró Bootstrap en su versión 4.6.2, un framework CSS que facilita el diseño responsivo y atractivo de interfaces web (Bootstrap, n.d., https://getbootstrap.com).

- **Editor de Código**: Se utilizó Visual Studio Code en su versión 1.95.1, un editor de código fuente que proporciona soporte para múltiples lenguajes y herramientas de depuración (Microsoft, n.d., https://code.visualstudio.com).

- **HTML5**: El lenguaje de marcado HTML5, que proporciona las bases para la creación de estructuras web interactivas y dinámicas, es esencial en el desarrollo frontend de aplicaciones web modernas (W3C, 2014, https://www.w3.org/TR/html5).

- **Autenticación y Autorización**: Para la gestión de roles y permisos se usaron:

    - Spatie Laravel Permission en su versión 5.11.1, que facilita la implementación de roles y permisos en aplicaciones Laravel (Spatie, n.d., https://spatie.be/docs/laravel-permission).

    - Laravel UI en su versión 3.4.6, que proporciona una forma sencilla de implementar interfaces de autenticación (Laravel Documentation, 2023, https://laravel.com/docs/8.x/authentication).

El diseño del sistema web educativo se basa en los principios de la Metodología de Diseño Hipermedia Orientada a Objetos (OOHDM) y en el uso de herramientas como Laravel 8 para gestionar las vistas, controladores, y la base de datos. El objetivo del diseño es crear una experiencia de usuario fluida y fácil de usar, que permita a los estudiantes acceder a tutoriales, realizar análisis de aplicaciones móviles, y a los administradores gestionar los roles y permisos.

2.5.1. Arquitectura del Sistema

El sistema sigue una arquitectura de tres capas basada en el patrón MVC (Modelo-Vista-Controlador):

- **Modelo**: Define la estructura de los datos y la interacción con la base de datos. Laravel utiliza Eloquent ORM, que facilita la gestión de los modelos y la interacción con las tablas de MySQL.

- **Vista**: Se encarga de mostrar la interfaz de usuario. Las plantillas Blade de Laravel permiten la integración de contenido dinámico como videos, textos, y formularios interactivos.

- **Controlador**: Gestiona la lógica de negocio y las solicitudes del usuario, y actúa como intermediario entre el modelo y la vista.

2.5.2. Diseño de las Vistas

El sistema cuenta con varias vistas principales, que están organizadas de acuerdo con los módulos educativos y las funcionalidades específicas de administración:

- **Inicio SMP**: Esta vista es la página principal del sistema, donde se presenta información sobre la unidad educativa. En esta sección, se pueden publicar contenidos informativos que incluyen:

    - Título

    - Detalles

    - Imágenes

    - Videos

    - Duración de la publicación

- **Explorando la Seguridad en los Celulares**: Esta vista proporciona información introductoria sobre la seguridad en aplicaciones móviles. Incluye un video que explica los riesgos asociados y recomendaciones de seguridad. No contiene campos configurables.

- **Aprende a Decompilar con "APKTool"**: Vista dedicada a un video tutorial que enseña el uso de la herramienta APKTool. También incluye:

    - Un enlace que redirige a un video en YouTube sobre cómo descargar la herramienta.

    - Un botón para abrir el asistente de análisis de permisos, fundamental para el funcionamiento del sistema.

- **Resultado de Análisis por los Estudiantes**: Esta vista es esencial para que los estudiantes registren sus análisis de seguridad de aplicaciones móviles. Contiene un formulario con los siguientes campos:

    - Nombre

    - Curso

    - Fecha

    - Nombre y fuente de la aplicación

    - Categoría de la aplicación

    - Imágenes con las observaciones encontradas

- Preguntas de selección (únicas y múltiples)

Esta vista incluye validaciones para asegurar la correcta entrada de datos antes de ser almacenados en la base de datos.

- **Avisos Importantes PPFF**: Esta vista permite a los administradores o superadministradores publicar anuncios relacionados con la unidad educativa. Los avisos incluyen los siguientes campos:
    - Título del aviso
    - Imagen del aviso
    - Detalles del aviso
- **Panel Informativo**: Esta vista está diseñada para la gestión de usuarios y roles. Solo es accesible para los usuarios registrados que están a la espera de la aprobación de un administrador o superadministrador. Una vez aprobados, los usuarios obtienen acceso a las vistas relacionadas con su rol. Además, incluye contenido sobre la seguridad en aplicaciones móviles.
- **Usuarios Registrados**: En esta vista se listan los usuarios registrados, permitiendo al administrador asignarles roles para que puedan navegar por el sistema. Además, tiene la capacidad de crear nuevos usuarios y editar o eliminar usuarios existentes. Los campos incluyen:
    - Nombre completo
    - Código de usuario
    - Contraseña

- Confirmar contraseña
- Rol (donde se selecciona el rol que tendrá el usuario)

- **Designación de Roles**: Esta vista permite al administrador crear roles personalizados para los usuarios y definir los permisos necesarios para navegar por el sistema.
- **Registro de Actividades**: En esta sección se registran las actividades que cada usuario realiza en el sistema, facilitando un seguimiento de las interacciones y comportamientos de los usuarios.

2.5.3. Diseño de la Interfaz de Usuario

El diseño de la interfaz se centra en la simplicidad y accesibilidad. Se utilizan plantillas Blade para generar contenido dinámico, permitiendo que las vistas se actualicen fácilmente con base en los datos del servidor.

- **Interfaz de Usuario para Estudiantes**: Los estudiantes tienen acceso a las vistas de Inicio SMP, Explorando la Seguridad en los Celulares, Aprende a Decompilar con “APKTool”, Resultado de Análisis por los Estudiantes, donde pueden registrar sus conclusiones y ver los avisos de la unidad educativa. La navegación es intuitiva y las secciones están claramente etiquetadas para facilitar el aprendizaje autónomo.

- **Interfaz de Usuario para Administradores y Superadministrador**: El dashboard ofrece opciones para gestionar usuarios, roles y permisos mediante una interfaz clara. Los administradores pueden crear, eliminar y dar de alta a nuevos usuarios, asignando roles a cada uno.

2.5.4. Interacción con la Base de Datos

La estructura de la base de datos se organiza de manera que la tabla Inicio funcione como la tabla principal, con relaciones a las tablas Introducción, Tutorial, Interacción, y Avisos. Las interacciones de los usuarios se gestionan a través de Eloquent ORM de Laravel, lo que facilita la creación de relaciones entre tablas y garantiza la integridad de los datos.

- **Dashboard**: Esta tabla principal proporciona un punto de entrada para los usuarios, ofreciendo una visión general de las funcionalidades del sistema. Contiene enlaces a diferentes secciones, como Usuarios, Roles, Interacciones y Avisos, y presenta información relevante sobre la actividad del sistema.

- **Inicio**: Tabla secundaria que almacena contenido informativo relevante sobre la unidad educativa, incluyendo publicaciones que pueden tener campos como título, detalles, imágenes, videos y duración de la publicación. Esta tabla complementa el Dashboard al proporcionar información adicional a los usuarios.

- **Introducción**: Tabla que almacena información estática, como videos y textos explicativos sobre la seguridad de aplicaciones móviles. Su contenido es fijo y esencial para la educación inicial de los usuarios, ayudando a establecer un contexto de seguridad.

- **Tutorial**: Similar a la tabla Introducción, esta tabla contiene datos estáticos relacionados con tutoriales específicos, incluidos enlaces a videos y descripciones. La funcionalidad de esta tabla es clave para la capacitación de los usuarios en el uso de herramientas como APKTool.

- **Interacción**: Esta tabla es crucial para el registro de las interacciones de los estudiantes con el sistema. Almacena datos sobre los análisis de aplicaciones móviles realizados por los usuarios, así como las conclusiones que han generado. Los campos incluyen el nombre del estudiante, la aplicación analizada, categorías y respuestas a preguntas específicas, lo que permite hacer seguimiento al progreso de los estudiantes.

- **Avisos**: Tabla destinada a la publicación de avisos importantes para los usuarios registrados. Almacena detalles como el título del aviso, imágenes asociadas y el contenido descriptivo. Esta funcionalidad es esencial para mantener a los usuarios informados sobre eventos y actualizaciones.

- **Usuarios**: Tabla que almacena la información de los usuarios registrados en el sistema. Contiene campos como nombre completo, código de usuario, contraseña (almacenada de manera segura) y el rol asignado a cada usuario. Esta tabla es fundamental para la gestión de acceso y la autenticación dentro del sistema.

- **Roles**: Esta tabla define los diferentes roles que pueden asignarse a los usuarios (por ejemplo, superadministrador, administrador, estudiante). Permite la gestión de permisos y accesos a diferentes partes del sistema, asegurando que cada usuario tenga el nivel adecuado de autorización.

- **Registro de Actividades**: Esta tabla registra las actividades realizadas por los usuarios dentro del sistema, permitiendo un seguimiento detallado de las acciones llevadas a cabo. Incluye campos como la fecha de la actividad, la descripción de la acción realizada y el usuario que la ejecutó. Este registro es útil para auditorías y para garantizar la transparencia en el uso del sistema.

La gestión de estas tablas se realiza a través de Eloquent ORM de Laravel, que simplifica las operaciones de base de datos y permite establecer relaciones entre tablas de manera intuitiva. Eloquent no solo facilita la creación, lectura, actualización y eliminación de registros (CRUD), sino que también garantiza la integridad de los datos al aplicar validaciones y restricciones a nivel de base de datos (Laravel, 2023).

2.5.5. Seguridad en el Diseño

La seguridad es un aspecto fundamental en el diseño de sistemas web, especialmente cuando se gestionan datos sensibles de los usuarios. En el caso de este proyecto, la seguridad se aborda mediante una serie de implementaciones y configuraciones dentro de Laravel, utilizando sus herramientas predeterminadas y paquetes adicionales para garantizar un entorno seguro tanto para los usuarios como para la integridad de la información del sistema (Laravel, 2023).

2.5.5.1. Autenticación y Autorización

La autenticación de los usuarios en el sistema se gestiona mediante el sistema predeterminado de Laravel, que utiliza sesiones para la autenticación y protección del acceso. El control de acceso es manejado a través del paquete Spatie Laravel-Permission, que permite definir roles y permisos específicos para cada usuario (Spatie, s.f.). Esto garantiza que los usuarios solo tengan acceso a las funcionalidades y vistas que les son asignadas según su rol. Los usuarios deben iniciar sesión para acceder al sistema y sus funciones. El acceso a ciertas vistas está restringido según el rol del usuario, lo que significa que, por ejemplo, un usuario recién registrado no podrá acceder a módulos o realizar ciertas acciones hasta que su cuenta sea aprobada por un administrador o superadministrador.

2.5.5.2. Cifrado de Contraseñas

Las contraseñas de los usuarios se cifran utilizando el algoritmo bcrypt, garantizando que las contraseñas no se almacenen en texto claro en la base de datos. Laravel maneja este proceso automáticamente en el momento del registro o la actualización de las contraseñas, proporcionando una capa adicional de seguridad en caso de que la base de datos sea comprometida (Laravel, s.f.).

2.5.5.3. Protección contra Ataques CSRF (Cross-Site Request Forgery)

El sistema de Laravel incluye protección contra ataques CSRF de manera predeterminada. Cada formulario en el sistema incluye un token CSRF que debe ser validado antes de procesar la solicitud, lo que asegura que las peticiones solo provengan de fuentes legítimas y no de sitios maliciosos que intenten manipular el sistema a través de solicitudes falsas (Otwell, 2023).

2.5.5.4. Protección contra Inyección SQL

Laravel utiliza su Eloquent ORM y el Query Builder para interactuar con la base de datos, lo que previene eficazmente la inyección SQL. Estas herramientas gestionan la entrada del usuario de forma segura, evitando que los datos maliciosos alteren las consultas SQL (Laravel, s.f.).

2.5.5.5. Validación y Sanitización de Datos

El sistema de validación de Laravel se utiliza para garantizar que todos los datos proporcionados por los usuarios sean correctos y adecuados antes de ser procesados o almacenados en la base de datos. Las reglas de validación se aplican en todos los formularios clave del sistema, como los registros de usuario y la interacción con las aplicaciones móviles (Otwell, 2023).

2.5.5.6. Auditoría y Registro de Actividades

El sistema emplea el paquete Spatie Laravel Activitylog para registrar todas las actividades clave dentro de la aplicación. Estas actividades incluyen acciones como el inicio de sesión de los usuarios, cambios en los roles y permisos, y cualquier otra operación administrativa relevante (Spatie, s.f.).

2.6. PRUEBAS DE SOFTWARE

Las pruebas de software son un proceso fundamental para garantizar la calidad y el correcto funcionamiento del sistema antes de su despliegue en producción. Estas pruebas permiten identificar errores, asegurar que las funcionalidades cumplan con los requisitos establecidos y verificar que el sistema sea robusto y seguro. En el contexto de este proyecto, donde el sistema web maneja roles, permisos y datos sensibles, las pruebas de software desempeñan un papel esencial para garantizar que todas las interacciones funcionen sin problemas.

2.6.1. Importancia de las Pruebas de Software

El objetivo principal de las pruebas es identificar y corregir posibles defectos antes de que el sistema sea utilizado por los usuarios finales. Las pruebas aseguran que:

- El sistema cumpla con los requerimientos funcionales y no funcionales.
- Las interacciones con los módulos educativos.
- Los roles y permisos definidos mediante Spatie Laravel-Permission funcionen de acuerdo con las especificaciones, restringiendo el acceso a los módulos según el rol del usuario.
- El sistema sea seguro, evitando accesos no autorizados o fugas de información.

2.6.2. Tipos de Pruebas Aplicadas

En este proyecto, se aplicarán dos tipos de pruebas principales: pruebas de caja negra y pruebas de caja blanca.

2.6.2.1. Pruebas de Caja Negra

Las pruebas de caja negra, también conocidas como pruebas funcionales, se centran en verificar la funcionalidad del sistema sin necesidad de acceder a su código interno. Estas pruebas se realizan desde la perspectiva del usuario, asegurándose de que las entradas generen las salidas correctas, tal como se espera en el comportamiento del sistema. En este sistema, las pruebas de caja negra incluyen:

- **Pruebas de las vistas**: Verificar que las vistas de Inicio SMP, Explorando la Seguridad en los Celulares, Aprende a Decompilar con “APKTool”, Resultado de Análisis por los Estudiantes, Avisos Importantes PPFF, Usuarios Registrados, Designación de Roles y Registro de Actividades funcionen correctamente, mostrando los contenidos y videos de manera adecuada.

- **Pruebas de validación**: Comprobar que el formulario de interacción para los estudiantes valide correctamente los datos introducidos (nombre, curso, nombre de la aplicación, imagen, etc.) y que los errores sean manejados de forma adecuada.

- **Pruebas de permisos**: Verificar que los roles asignados (superadministrador, administrador, estudiante, general) restringen o permiten el acceso a las vistas del sistema según las especificaciones.

2.6.2.3. Pruebas de Caja Blanca

Las pruebas de caja blanca, también conocidas como pruebas estructurales, se enfocan en la lógica interna del sistema, analizando el código fuente para garantizar que los flujos de trabajo y las rutas de ejecución funcionen correctamente. Este tipo de pruebas permiten identificar posibles errores en la implementación del código que no son evidentes desde una perspectiva externa. En este proyecto, las pruebas de caja blanca incluyen:

- **Revisión del flujo de trabajo en Laravel**: Verificar que las rutas y controladores del sistema gestionen correctamente las solicitudes de los usuarios y que las operaciones CRUD (Crear, Actualizar y Eliminar) funcionen de acuerdo con lo esperado.

- **Pruebas de lógica de roles y permisos**: Evaluar que los permisos definidos mediante Spatie Laravel-Permission controlen el acceso de forma segura y correcta, asegurando que los usuarios no puedan acceder a vistas o realizar acciones fuera de su rol.

- **Revisión de la seguridad del sistema**: Evaluar la robustez del cifrado de contraseñas y la autenticación de usuarios en Laravel.

2.6.3. Pruebas Unitarias y de Integración

Además de las pruebas de caja negra y caja blanca, se realizarán pruebas unitarias y pruebas de integración utilizando las herramientas de testing que proporciona Laravel.

- **Pruebas unitarias**: Se utilizan para probar componentes individuales del sistema, como los modelos y controladores, asegurando que cada unidad de código funcione de manera independiente. Laravel incluye PHPUnit para la ejecución de estas pruebas. Por ejemplo, se probarán los modelos que gestionan las tablas de inicio, interacción, avisos, usuarios y roles verificando que los datos se guarden y recuperen correctamente.

- **Pruebas de integración**: Evalúan cómo interactúan entre sí las diferentes partes del sistema. En este caso, las pruebas de integración verificarán la interacción entre las vistas, los controladores y la base de datos. Por ejemplo, se probará que cuando un estudiante sube un análisis en la vista de interacción, los datos se almacenen correctamente en la tabla correspondiente y sean accesibles para el administrador.

2.6.4. Pruebas de Seguridad

Las pruebas de seguridad son esenciales en un sistema que maneja datos sensibles, como el acceso a roles y permisos. En este proyecto, las pruebas de seguridad evaluarán:

- **Cifrado de contraseñas**: Verificar que las contraseñas de los usuarios estén correctamente cifradas utilizando el algoritmo bcrypt y que no puedan ser descifradas fácilmente.

- **Control de acceso**: Asegurar que los usuarios recién registrados no tengan acceso a vistas o funcionalidades sin la aprobación del administrador.

- **Protección contra vulnerabilidades**: Realizar pruebas para evitar vulnerabilidades comunes como la inyección de SQL y los ataques XSS (cross-site scripting).

2.7. SEGURIDAD DEL SISTEMA

La seguridad del sistema es uno de los componentes más críticos del desarrollo de aplicaciones web, especialmente cuando se manejan datos sensibles como los roles y permisos de los usuarios. En este proyecto, se aplican diversas medidas de seguridad, tanto físicas como lógicas, para garantizar que el sistema sea seguro y protegido frente a amenazas internas y externas.

2.7.1. Seguridad Física

La seguridad física se refiere a las medidas implementadas para proteger los servidores y la infraestructura que soporta el sistema web. En el contexto de este proyecto, aunque el sistema se ejecuta en un entorno de desarrollo local con XAMPP, es fundamental considerar la seguridad física en un futuro despliegue en servidores web.

Las **normativas ISO/IEC 27000** son un estándar internacional que define las mejores prácticas para la gestión de la seguridad de la información, incluida la seguridad física. Estas normativas establecen medidas clave para garantizar la protección de los recursos tecnológicos y la infraestructura crítica en las organizaciones. Entre las recomendaciones más relevantes se incluyen:

- **Control de acceso a los servidores**: Limitar el acceso físico a los servidores donde se alojan los datos y la aplicación. Solo el personal autorizado debe tener acceso al hardware (ISO/IEC 27001, 2013).

- **Sistemas de respaldo**: Implementar sistemas de copias de seguridad regulares para prevenir la pérdida de datos en caso de fallos del hardware o ataques físicos (ISO/IEC 27002, 2013).

- **Protección contra desastres**: Sistemas de contingencia, como fuentes de alimentación ininterrumpida (UPS) y estrategias de recuperación ante desastres, deben estar implementados para asegurar la continuidad del servicio (ISO/IEC 27005, 2018).

2.7.2. Seguridad Lógica

La seguridad lógica es el conjunto de medidas implementadas a nivel de software para proteger el sistema de ataques cibernéticos. En este proyecto, se abordan diferentes capas de seguridad lógica:

2.7.2.1. Seguridad en la Base de Datos

El sistema utiliza MySQL como base de datos, y para asegurar su integridad y seguridad, se aplican diversas medidas clave:

- **Cifrado de contraseñas**: Las contraseñas de los usuarios se almacenan de forma segura utilizando el algoritmo bcrypt**,** un algoritmo de hash que emplea una función irreversible para garantizar la protección de las contraseñas, incluso si los atacantes logran acceder a la base de datos. El uso de bcrypt se recomienda ampliamente en la documentación oficial de Laravel (Laravel, n.d.).

- **Acceso restringido a la base de datos**: El sistema limita el acceso a la base de datos mediante una gestión de permisos detallada, utilizando el paquete Spatie Laravel-Permission. Este paquete permite definir roles y niveles de acceso específicos para los usuarios, ayudando a reducir el riesgo de accesos no autorizados. Se puede consultar la documentación oficial de Spatie para más detalles sobre su implementación (Spatie, n.d.).

- **Backups regulares**: Se implementan políticas de copias de seguridad periódicas para proteger los datos ante fallos del sistema o ataques. El ISO/IEC 27001 recomienda la implementación de estrategias de backup como una medida clave para garantizar la seguridad de la información en las organizaciones (ISO/IEC 27001, 2013).

- **Protección mediante reCAPTCHA**: El sistema utiliza Google reCAPTCHA para evitar ataques automatizados, como intentos de acceso no autorizado y ataques de fuerza bruta. ReCAPTCHA ayuda a verificar que las interacciones provienen de usuarios humanos y no de bots, lo cual reduce el riesgo de spam y otros ataques automáticos (Google, 2023).

2.7.2.2. Seguridad en el Código

La seguridad en el código es fundamental para evitar vulnerabilidades comunes en aplicaciones web. Algunas de las prácticas aplicadas en el desarrollo de este sistema incluyen:

- **Protección contra inyección SQL**: Laravel previene automáticamente los ataques de inyección SQL mediante su Eloquent ORM, que utiliza consultas preparadas y sanitiza los datos antes de interactuar con la base de datos. Esta práctica está documentada en la guía oficial de Laravel (Laravel, n.d.).

- **Prevención de XSS (Cross-Site Scripting)**: Las plantillas Blade de Laravel escapan automáticamente las entradas de los usuarios, evitando que código malicioso sea ejecutado en el navegador del usuario. Laravel incluye protecciones contra XSS de forma predeterminada, como se menciona en su documentación (Laravel, n.d.).

- **Protección CSRF (Cross-Site Request Forgery)**: Laravel incluye automáticamente tokens CSRF para todas las solicitudes POST, lo que asegura que cada petición provenga de una fuente confiable y no de un atacante. Esta protección es una capa esencial de seguridad en aplicaciones web, tal como se describe en la documentación de Laravel (Laravel, n.d.).

2.7.2.3. Seguridad en el Sistema

La seguridad es un aspecto clave en el diseño de este sistema, y se asegura mediante un control de acceso que define claramente los roles y permisos de los usuarios. Utilizando el paquete Spatie Laravel-Permission, se configuran los siguientes roles y permisos:

- **Administrador**: Tiene acceso completo al sistema, lo que incluye la capacidad de crear, modificar y eliminar roles y usuarios, así como asignar roles y permisos a otros usuarios.

- **Director**: Puede gestionar roles y usuarios, asignando permisos a los nuevos usuarios y controlando su acceso a diferentes partes del sistema.

- **Secretario/Asistente**: Gestionar los módulos informativos del sistema, pero su acceso a módulos administrativos está restringido, como la gestión de usuarios y roles.

- **Docente**: Al igual que el secretario, puede gestionar los módulos informativos, pero su acceso a áreas administrativas está limitado, sin poder modificar roles o gestionar usuarios.

- **Estudiante**: Puede gestionar los módulos educativos donde puede cargar sus conclusiones sobre aplicaciones móviles. Su acceso a áreas administrativas está restringido.

- **Padres de Familia**: Pueden acceder a los módulos informativos y educativos, pero con un acceso limitado, sin la posibilidad de modificar o gestionar ningún módulo.

El **dashboard** es visible para todos los usuarios que se registran en el sistema, sin embargo, estos usuarios no podrán interactuar con otras vistas ni acceder a funciones completas del sistema hasta que se les haya otorgado la autorización correspondiente. Este control de acceso asegura que solo los usuarios con permisos apropiados puedan modificar datos o acceder a información sensible, garantizando la seguridad de los recursos del sistema.

2.8. MÉTRICAS DE CALIDAD

2.8.1. Métrica McCall

El modelo McCall se utiliza para evaluar la calidad del software basándose en una serie de características clave que impactan la experiencia del usuario, el rendimiento y la capacidad de mantenimiento del sistema. Las métricas más relevantes para este proyecto son las siguientes:

- **Fiabilidad**: Evalúa la capacidad del sistema para realizar las funciones esperadas de manera consistente sin fallos. Las pruebas de fiabilidad se basan en la cantidad de errores detectados durante las pruebas de caja negra y caja blanca, lo cual es fundamental para asegurar que el sistema funcione de manera estable y confiable en todo momento (Sommerville, 2011).

- **Mantenibilidad**: Mide la facilidad con la que el sistema puede ser modificado, corregido y actualizado. El uso del framework Laravel, con su estructura modular y el enfoque en controladores y modelos bien definidos, facilita la mantenibilidad y simplifica las tareas de modificación y expansión del sistema (Laravel Documentation, n.d.).

- **Eficiencia**: Verifica que el sistema utiliza los recursos de manera adecuada, manteniendo tiempos de carga rápidos y una respuesta eficiente, incluso con múltiples usuarios conectados simultáneamente. La optimización de las consultas a la base de datos y la correcta gestión de las sesiones contribuyen a la eficiencia global del sistema (Rademacher & Trautner, 2019).

- **Usabilidad**: Evalúa la facilidad de uso de las interfaces para los usuarios finales, como estudiantes, administradores y padres. Se mide a través de la claridad de las vistas y la facilidad para completar las tareas requeridas, como subir un análisis o publicar un aviso. La usabilidad es crucial para garantizar que los usuarios no se enfrenten a barreras tecnológicas al interactuar con el sistema (Krug, 2014).

2.8.2. Métricas de Seguridad

Aunque el modelo McCall se centra principalmente en atributos como la fiabilidad y la usabilidad, también es importante integrar métricas relacionadas con la seguridad para garantizar la protección de los datos y la integridad del sistema. Las métricas de seguridad implementadas en este proyecto incluyen:

- **Captcha**: El sistema utiliza la tecnología Captcha para proteger las rutas de inicio de sesión, registro y otros formularios críticos. Esto previene ataques automatizados, como los intentos de adivinación de contraseñas (fuerza bruta) y otros métodos de spam. La implementación de Captcha asegura que los usuarios sean humanos y no bots, mejorando la seguridad y evitando el uso malicioso del sistema (OWASP Foundation, n.d.).

- **Criptografía**: Las contraseñas y datos sensibles se gestionan mediante criptografía robusta utilizando algoritmos como bcrypt en Laravel. Este proceso asegura que las contraseñas no se almacenen en texto claro en la base de datos, lo que previene el acceso no autorizado incluso si la base de datos se ve comprometida (OWASP Foundation, 2017).

- **Fiabilidad**: Mantener copias de seguridad periódicas garantiza que, incluso en situaciones de fallo, el sistema pueda restaurar datos importantes sin comprometer su estabilidad. Esto se conecta directamente con la fiabilidad del sistema, asegurando que la pérdida de datos no afecte su funcionamiento general (NIST, 2013).

- **Seguridad**: Los backups cifrados y protegidos aseguran que no solo los datos activos, sino también las copias, estén protegidas contra accesos no autorizados. Además, permiten mantener la integridad de la información, un aspecto crucial en la seguridad del sistema (NIST, 2013).

2.9. ESTIMACIÓN DE COSTOS

La planificación de costos es esencial en proyectos de software para asignar adecuadamente los recursos y asegurar que el desarrollo se mantenga dentro del presupuesto. En este proyecto, el objetivo es crear un sistema web educativo en Laravel orientado a la seguridad en aplicaciones móviles. Para estimar los costos, se empleará el modelo COCOMO II (Constructive Cost Model), que se basa en factores como el esfuerzo de desarrollo, la complejidad del sistema y el tiempo requerido para completar el proyecto (Boehm & Lane, 2003).

2.9.1. COCOMO II (Modelo Constructivo de Costos)

COCOMO II, una actualización del modelo original desarrollado por Barry Boehm, permite calcular el esfuerzo, la duración y los costos asociados al desarrollo de software según tres categorías de proyectos: Orgánico, Semi-desarrollado y Embebido. Este sistema educativo, debido a sus características de autenticación, permisos y gestión de usuarios, se clasifica como un proyecto semi-desarrollado. Herramientas como Laravel, MySQL y Spatie Laravel-Permission son opciones de bajo costo, pero su integración y configuración aumentan la complejidad del desarrollo, afectando la estimación final del costo (Boehm, 2000).

2.9.2. Factores Clave en COCOMO II

- **Esfuerzo de Desarrollo**: Medido en persona-meses (PM), representa el trabajo continuo de una persona en un mes. Este proyecto incluirá tareas de configuración de bases de datos, desarrollo de vistas en Laravel y gestión de roles.

- **Tamaño del Proyecto**: Calculado en Líneas de Código (LOC) o Puntos de Función (PF). En este sistema, el tamaño del proyecto se determina con base en las funcionalidades de autenticación, análisis de APKs y los módulos educativos interactivos.

- **Complejidad del Sistema**: La implementación de roles, autenticación y permisos, además de los módulos de tutoriales y análisis, implica una complejidad moderada.

La fórmula de COCOMO II para calcular el esfuerzo es:

(1)Ecuación para calcular el Esfuerzo (PM)

$$Esfuerzo\ (PM) = A * \ (Tamaño)^b * \Pi\ Factores\ de\ costo \quad (1)$$

Donde:

- **A** es una constante que varía según el tipo de proyecto.

- **Tamaño** representa las líneas de código o puntos de función.

- **B** es un exponente que refleja el impacto del tamaño en el esfuerzo.

- **Factores de Costo**: Factores adicionales que modifican el esfuerzo estimado, como la experiencia del equipo, los requisitos de seguridad y las restricciones de tiempo.

Factores de Costo Específicos en COCOMO II

Para este proyecto, los factores de costo adicionales incluyen:

- **Experiencia del Equipo:** La experiencia en Laravel y PHP influye en la eficiencia y precisión del desarrollo.

- **Requerimientos de Seguridad:** Implementar medidas de seguridad para la gestión de roles y permisos añade complejidad.

- **Restricciones de Plazos:** Si el proyecto requiere entregas en plazos cortos, el esfuerzo estimado aumenta, ya que se debe agilizar el desarrollo.

CAPITULO III

MARCO APLICATIVO

INGENIERÍA
DE SISTEMAS
UNIVERSIDAD PUBLICA DE EL ALTO

CAPITULO III

MARCO APLICATIVO

3.1. INTRODUCCIÓN

En este capítulo se aborda la aplicación práctica de las cinco fases de la metodología OOHDM (Método de Diseño Hipermedia Orientado a Objetos) para el desarrollo de un sistema web. Esta metodología, que integra principios del diseño orientado a objetos con la creación de hipermedias, proporciona un marco estructurado y flexible para la construcción de aplicaciones web interactivas y dinámicas. A lo largo de este capítulo, se describirá el proceso de aplicación de cada fase, desde la planificación inicial hasta la implementación del sistema.

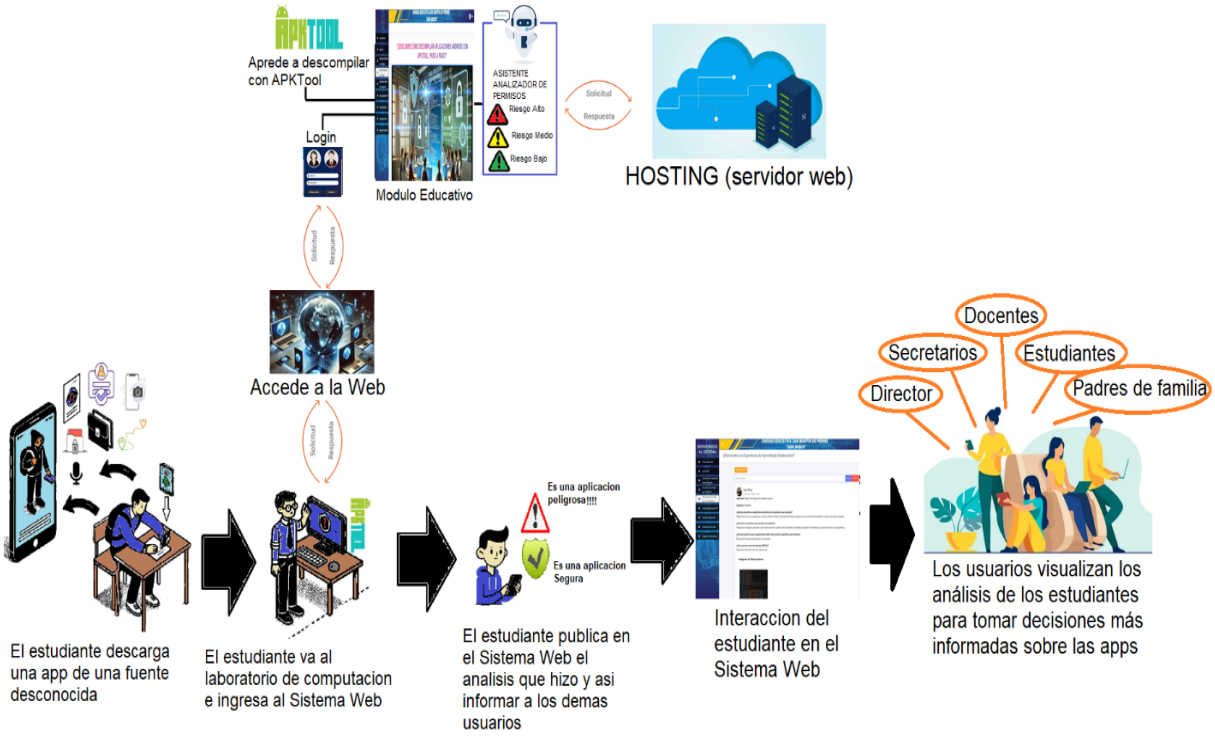
3.2. ANÁLISIS DE LA SITUACIÓN ACTUAL

En la Unidad Educativa San Martín de Porres, una gran mayoría de los estudiantes dispone de un teléfono móvil, situación que se aceleró debido a la pandemia de COVID-19, pues los colegios adoptaron las clases virtuales para continuar con la educación en un entorno seguro y accesible. Al contar con un dispositivo móvil, los estudiantes sienten una atracción natural por descargar aplicaciones móviles Android, principalmente desde la Google Play Store, que se caracteriza por un nivel alto de seguridad en sus aplicaciones (Ruiz & Martínez, 2021). No obstante, en los casos en que la aplicación deseada no está disponible en la Play Store o requiere de un pago, algunos estudiantes recurren a sitios web externos y no oficiales, donde las aplicaciones pueden descargarse gratuitamente. Estos sitios, sin embargo, a menudo carecen de garantías de seguridad, lo cual expone a los estudiantes a riesgos como la descarga de malware o software no deseado sin que ellos estén al tanto de los peligros asociados (González & Hernández, 2020).

Para abordar esta problemática y concientizar a los estudiantes sobre los riesgos de descargar aplicaciones de fuentes no seguras, se ha propuesto un sistema web educativo que les permita analizar dichas aplicaciones antes de su instalación. Este sistema incluye módulos educativos sobre seguridad en aplicaciones móviles y emplea la herramienta externa APKTool para decompilada las aplicaciones, facilitando la revisión del archivo AndroidManifest.xml. Mediante un asistente en el sistema, los estudiantes pueden revisar y comprender los permisos solicitados por cada aplicación, clasificándolos según su nivel de riesgo (alto, medio, bajo). Esto permite a los estudiantes tomar las precauciones necesarias y, además, publicar en el sistema web los resultados de sus análisis para que otros usuarios registrados puedan consultarlos antes de descargar aplicaciones desde fuentes externas no verificadas, promoviendo una comunidad informada y protegida (López, 2022).

3.2.1. Esquema del sistema web

Figura 6
Esquema del Sistema Web



Nota. Este es un esquema donde se tiene como objetivo desarrollar el sistema web.

3.2.2. Métodos de Recolección de Datos

En esta fase, se realizaron varias técnicas de recolección de datos para obtener información clave sobre las necesidades del sistema, específicamente de los estudiantes. A continuación, se detallan los métodos y herramientas utilizadas.

Tabla 1

Recolección de datos: Entrevista

Método de Recolección	Descripción	Fuente de Datos
Entrevistas	Se llevó a cabo una entrevista con los estudiantes para conocer sus conocimientos en las seguridades en las aplicaciones móviles	Estudiantes

Nota. En esta tabla se muestran los detalles de la recolección de datos.

3.2.3. Herramientas de Recolección de Datos

Para la recopilación de datos, se emplearon herramientas tecnológicas que facilitaron el análisis de la información obtenida.

Tabla 2

Recolección de datos: Google Forms

Herramienta	Descripción	Ventajas
Google Forms	Plataforma en línea utilizada para crear encuestas y formularios personalizados. Los datos fueron recolectados de forma automática y organizada.	Sencillez en el diseño de formularios, recolección automática de datos, integración con Google Sheets para análisis posterior.

Nota. En esta tabla se muestra el detalle de la recolección de datos.

3.2.4. Actores y Sus Roles

En el sistema web, los actores son principalmente los estudiantes, quienes son los principales usuarios finales de la plataforma.

Tabla 3

Identificación de actores

Actor	Atributos	Métodos
Estudiantes	Nombre, Código de Usuario, Rol → Estudiante	Ver Módulos del Sistema Web, Análisis de Permisos, Completar Formulario

Nota. En esta tabla se ve al Estudiantes como actor principal.

3.2.5. Requerimientos Funcionales y No Funcionales

3.2.5.1. Requerimientos Funcionales

Los requerimientos funcionales definen las funcionalidades específicas que el sistema debe cumplir según las necesidades de los estudiantes. A continuación, se detallan:

Tabla 4

Requerimientos funcionales

ID de Requerimiento	Descripción del Requerimiento	Prioridad	Actor(es) Involucrado(s)	Fuente
RF-01	Los estudiantes deben poder registrarse en la plataforma utilizando su código de usuario y una contraseña.	Alta	Estudiantes	Entrevistas con estudiantes
RF-02	Los estudiantes deben poder acceder a materiales de estudio de manera sencilla y organizada.	Alta	Estudiantes	Encuesta y entrevistas

ID de Requerimiento	Descripción del Requerimiento	Prioridad	Actor(es) Involucrado(s)	Fuente
RF-03	Los estudiantes deben poder analizar los permisos en el sistema y evaluar las aplicaciones mediante una herramienta externa.	Media	Estudiantes	Encuestas y entrevistas
RF-04	Los estudiantes deben poder completar el formulario de inscripción y registrar sus datos correctamente.	Alta	Estudiantes	Encuestas y entrevistas

Nota. En esta tabla se muestra los requerimientos funcionales para el estudiante.

3.2.5.2. Requerimientos No Funcionales

Los requerimientos no funcionales abordan aspectos como el rendimiento, la seguridad, la accesibilidad y otros criterios del sistema. A continuación, se detallan los principales requerimientos:

Tabla 5

Requerimiento no Funcionales

ID de Requerimiento	Descripción del Requerimiento No Funcional	Prioridad	Fuente
RNF-01	El sistema debe ser accesible desde dispositivos móviles y de escritorio, optimizado para una experiencia fluida.	Alta	Reunión con estudiantes
RNF-02	El sistema debe ser capaz de manejar al menos 100 estudiantes concurrentes sin experimentar lentitud en la carga.	Alta	Análisis técnico
RNF-03	El sistema debe tener un nivel de seguridad alto, cifrando todas las contraseñas y datos sensibles.	Alta	Requerimiento de seguridad
RNF-04	El sistema debe ser accesible en varios idiomas (por ejemplo, inglés y español).	Media	Requerimiento de internacionalización

Nota. En esta tabla se muestra los requerimientos no funcionales en el sistema.

3.2.6. Identificación de las Entidades (Clases)

Tabla 6

Identificación de Entidades

Usuario	Atributos	Métodos
Administrador	Nombre, Código de Usuario, Rol->Administrador	Gestionar el Sistema Supervisar y Administrar Roles y Usuarios. Gestionar el Registro de Actividades
Director	Nombre, Código de Usuario, Rol->Director	Ver Módulos del Sistema Web. Supervisar y Administrar Usuarios. Informar Novedades de la Institución.

Usuario	Atributos	Métodos
Secretario/Asistente	Nombre, Código de Usuario, Rol->Secretario/Asistente	Ver Módulos del Sistema Web. Supervisar Usuarios. Informar Novedades de la Institución.
Docentes	Nombre, Código de Usuario, Rol->Docentes	Ver Módulos del Sistema Web. Informar Novedades de la Institución.
Estudiantes	Nombre, Código de Usuario, Rol->Estudiantes Nombre, Código de Usuario, Rol	Ver Módulos del Sistema Web Análisis de Permisos. Completar Formulario
Padres de Familia	Nombre, Código de Usuario, Rol->Padre de Familia	Ver Módulos del Sistema Web.

Nota. En esta tabla se describe a las identidades que estarán en el sistema web.

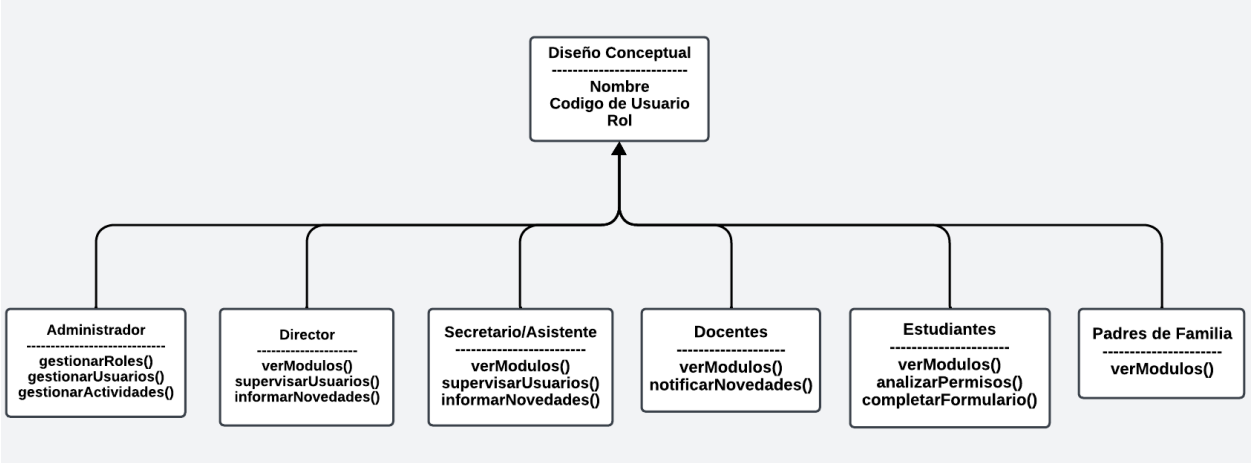
METODOLOGÍA OOHDM Y SU INTEGRACIÓN CON UML EN EL DISEÑO DE NAVEGACIÓN

Figura 7
Las cinco fases de OOHDM

Nota. Las cinco fases del modelo OOHDM. Basado en Schwabe y Rossi (1995), Garzotto et al. (1993) y Rossi et al. (1996).

3.2.7. Fase 1: Obtención de requerimientos

Figura 8
Diseño Conceptual



Nota. Este diseño conceptual ilustra como el sistema web propuesto para la seguridad en aplicaciones móviles está estructurado.

3.2.7.1. Módulos de Aprendizaje y Formularios

Tabla 7

Descripción de módulos

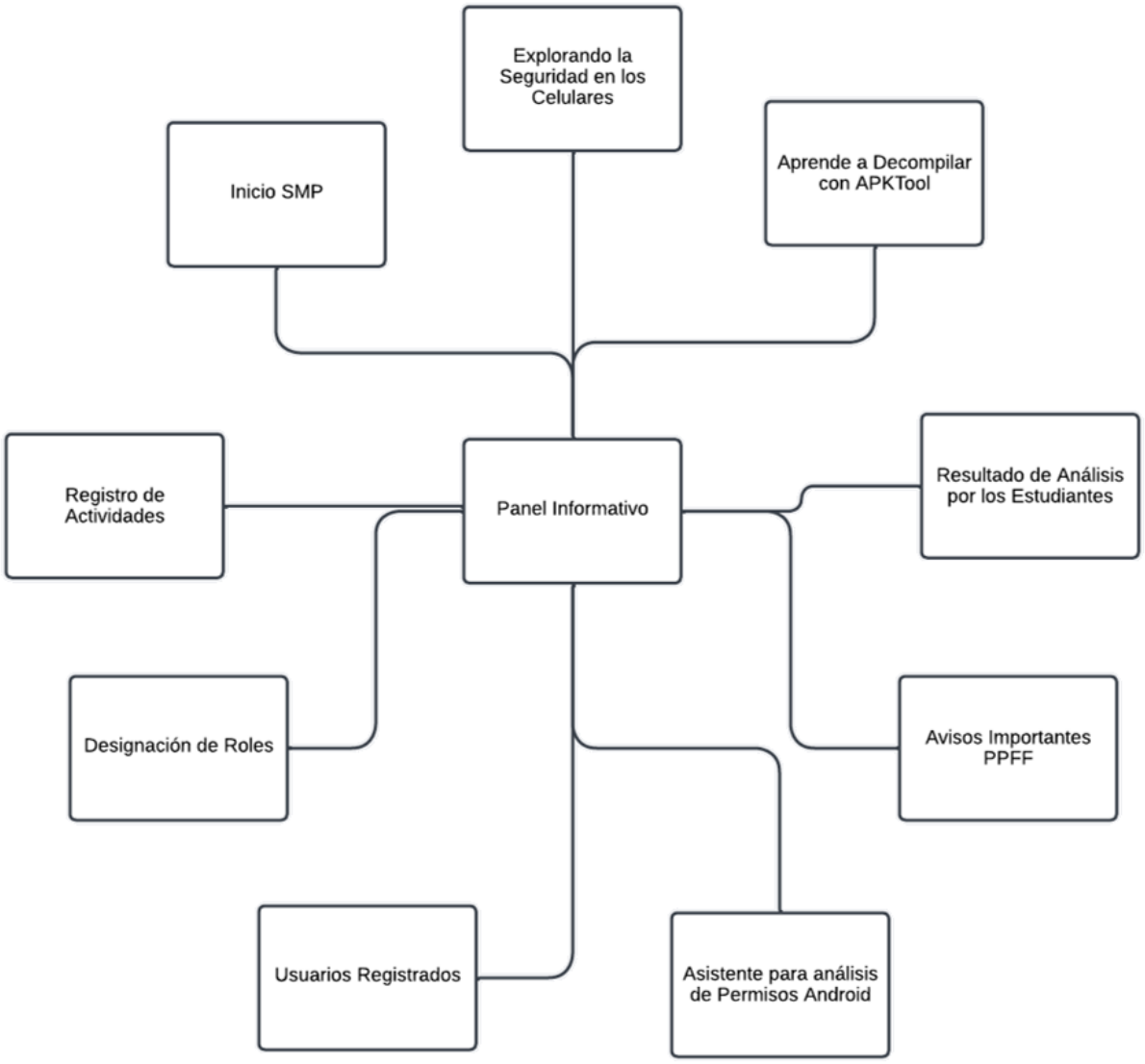
Modulo	Contenido	Descripción	Tipo de modulo
Panel Informativo (Dashboard)	Videos breves sobre la seguridad en aplicaciones móviles. Tarjetas de navegación	Contiene videos cortos explicando sobre las herramientas para la seguridad en aplicaciones móviles. Contiene tarjetas de navegación en el sistema web, en el cual no tiene los permisos para modificar registros	Módulo de Navegación
Inicio SMP	Información de la institución. Noticias y eventos	Contiene información de la institución como el nombre, misión y visión, valores. Contiene publicaciones por parte del personal administrativo	Módulo de Información Institucional

Modulo	Contenido	Descripción	Tipo de modulo
Explorando la Seguridad en los Celulares	Video introductorio sobre la seguridad en aplicaciones móviles Detalles e información sobre los riesgos tecnológicos	Contiene un video de introducción donde se explica sobre la seguridad en aplicaciones móviles. Contiene información sobre los riesgos tecnológicos según “Kaspersky”	Modulo Educativo
Aprende a Decompilada con APKTool	Video tutorial sobre como decompilada aplicaciones Android Detalles	Contiene un video tutorial completo de como decompilada aplicaciones móviles Android (APK) En los detalles esta la información de la aplicación decompilada en el tutorial	Modulo educativo
Resultado de Análisis por los Estudiantes	Formulario	Contiene la información del estudiante más las respuestas sobre lo aprendido en el análisis de una aplicación Android	Modulo Educativo
Avisos Importantes PPFF	Información Administrativa	Contiene publicaciones como avisos, comunicados, convocatorias y otros.	Módulo de Información Institucional
Asistente para análisis de Permisos Android	Análisis de permisos	Contiene información sobre los permisos en una aplicación Android	Modulo Educativo
Usuarios Registrados	Información de Usuarios	Contiene la información de los usuarios registrados en el sistema	Modulo Administrativo
Designación de Roles	Información de los roles	Contiene la información de los roles creados	Modulo administrativo

Modulo	Contenido	Descripción	Tipo de modulo
Registro de Actividades	Información de las actividades en el sistema	Contiene el registro de los movimientos de cada usuario	Modulo Administrativo

Nota. En esta tabla se detallan los módulos principales en el sistema web.

Figura 9
Diagrama de Clases: Módulos del Sistema Educativo

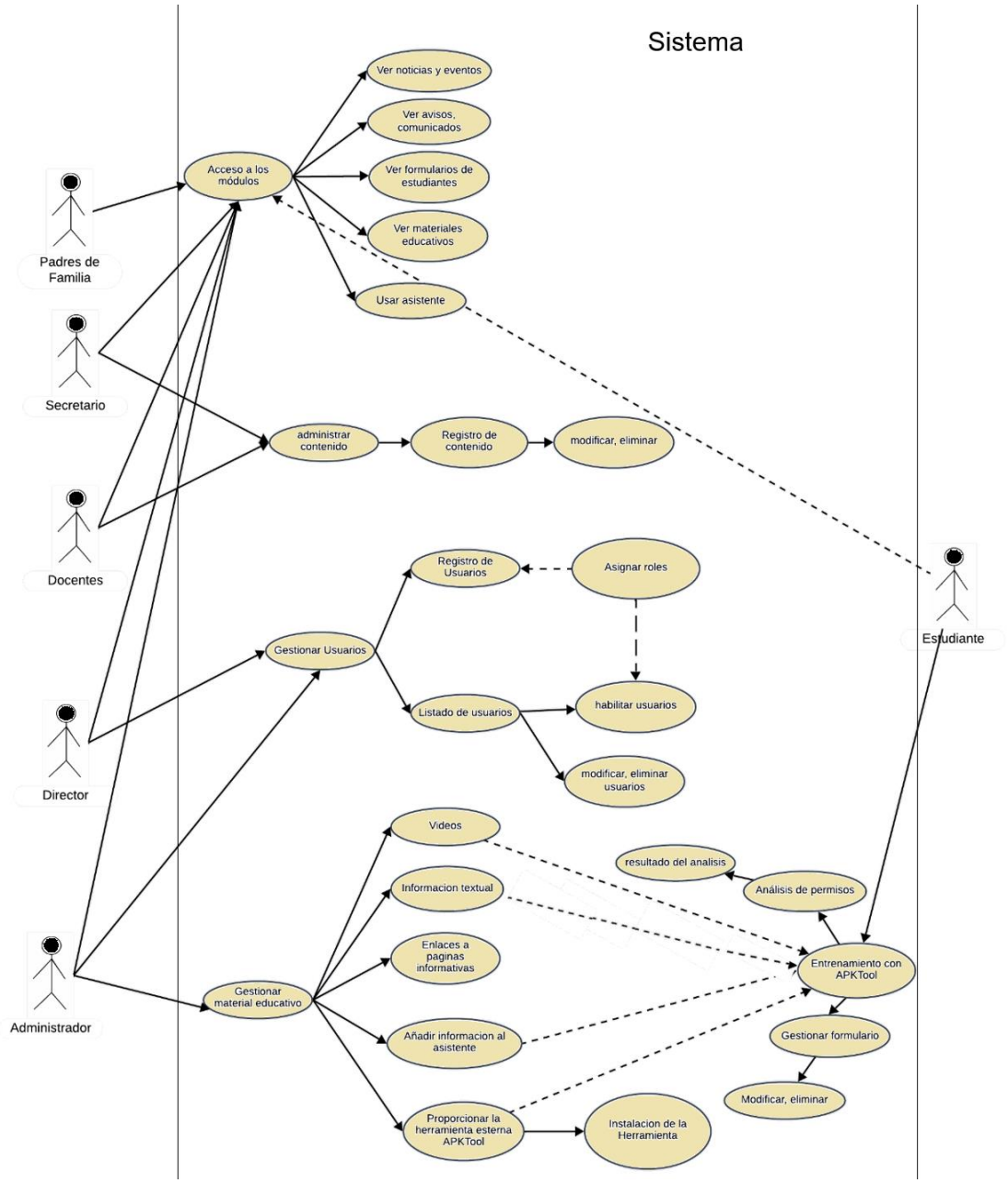


Nota. El diagrama ilustra las relaciones entre estos módulos dentro del sistema.

3.2.8. Fase2: Diseño Conceptual

3.2.8.1. Diagrama de Caso de Uso

Figura 10
Caso de Uso del Sistema

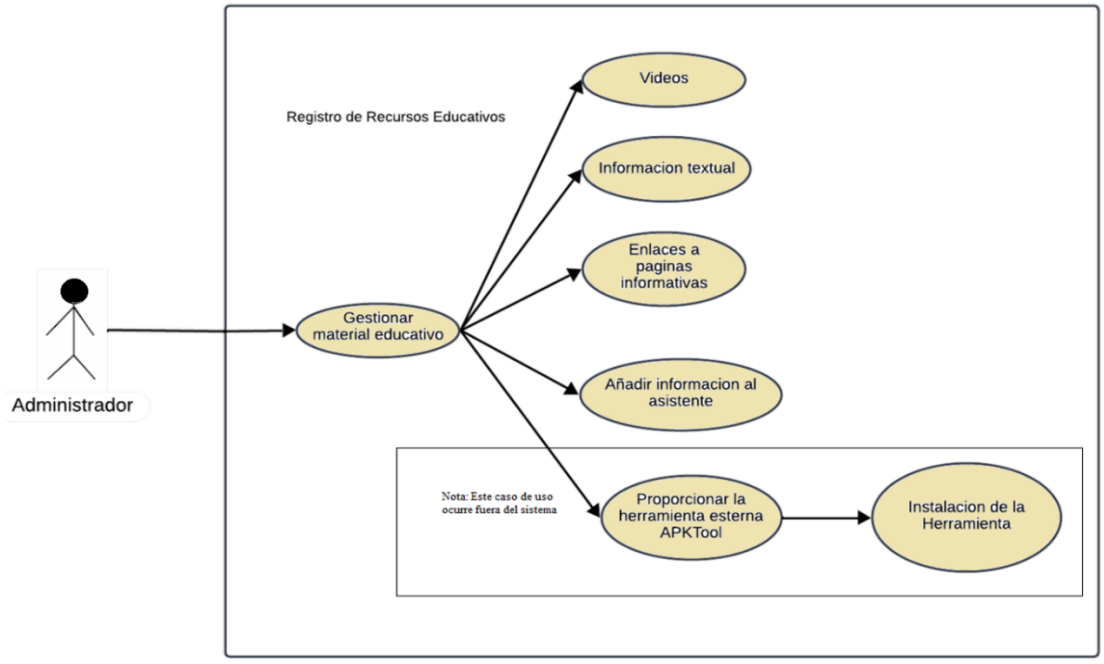


Nota. El diagrama ilustra como los Usuarios interactúan en el sistema web.

A continuación, se describirán los diferentes casos de uso del sistema web, con el fin de ilustrar las actividades que realizará cada usuario dentro del sistema.

Figura 11

Caso de Uso: Registro de Recursos Educativos



Nota. El diagrama ilustra como el Administrador interactuara en el sistema para gestionar material educativo.

Tabla 8

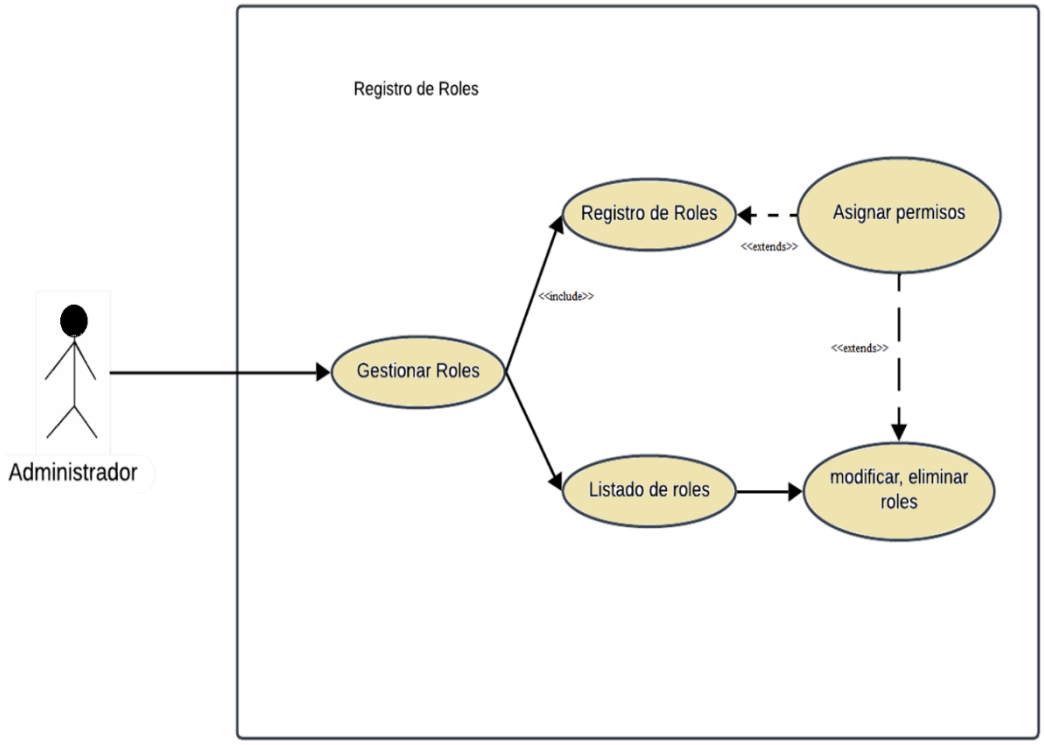
Descripción del caso de uso: Registro de Recursos Educativos

Caso de Uso	**Registro de Recursos Educativos**
Actores	Administrador
Tipo	Primario (Administrador)
Descripción	El administrador sube manualmente videos educativos sobre la seguridad en aplicaciones móviles al igual que información textual y enlaces a paginas informativas. También carga datos sobre los permisos Android encontrados hasta la actualidad, también de proporcionar la herramienta externa APKTool en el laboratorio de la institución educativa

Nota. En esta tabla se describe el caso de uso del Administrador para administrar material educativo al sistema

Figura 12

Caso de Uso: Registro de Roles



Nota. El diagrama ilustra como el Administrador interactuara en el sistema para la gestión de roles

Tabla 9

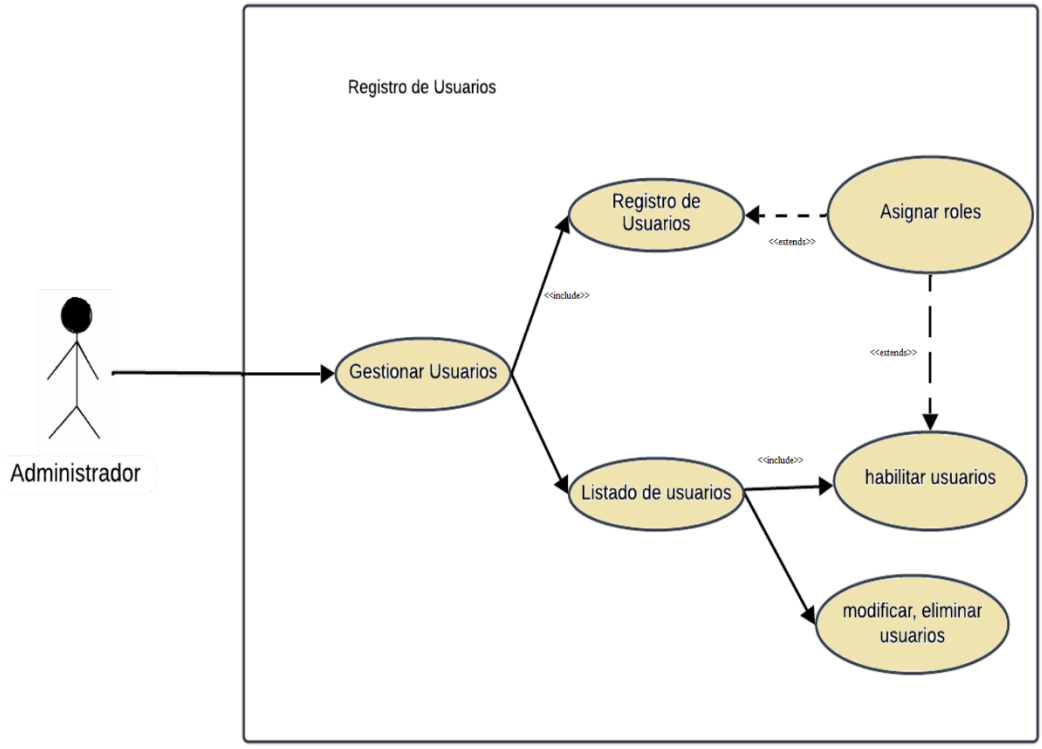
Descripción del caso de uso: Registro de Roles

Caso de Uso	**Registro de Roles**
Actores	Administrador
Tipo	Primario (Administrador)
Descripción	El administrador crea nuevos roles de acuerdo a los usuarios, también puede modificar o eliminar los datos

Nota. En esta tabla se describe las funciones del Administrador en base a los roles.

Figura 13

Caso de Uso: Registro de Usuarios



Nota. El diagrama ilustra como el Administrador y el Director interactuara en el sistema

Tabla 10

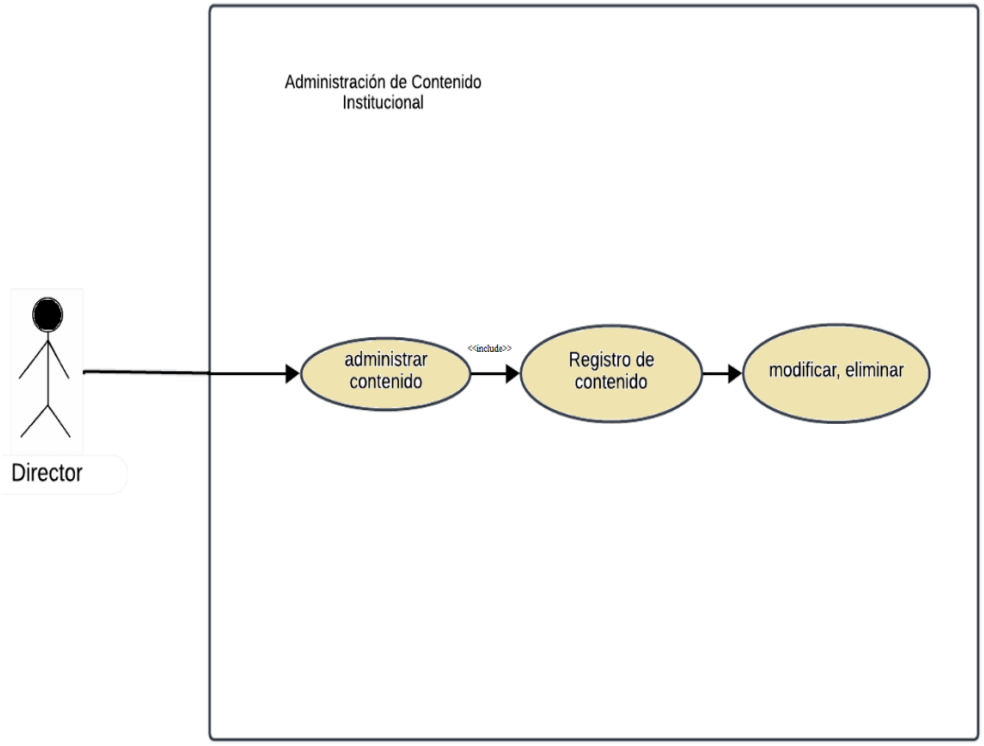
Descripción del caso de uso: Registro de Usuarios

Caso de Uso	**Registro de Usuarios**
Actores	Administrador, Director
Tipo	Primario (Administrador)
Descripción	El administrador o el Director esta encargada de gestionar los usuarios tanto para crear, modificar y habilitar asignándoles un rol dependiendo del usuario al que se habilitar

Nota. En esta tabla se describe las funciones del Administrador y Director.

Figura 14

Caso de Uso: Administración de Contenido Institucional



Nota. El diagrama ilustra como el Director, Secretario, Asistente y Docentes interactuara en el sistema.

Tabla 11

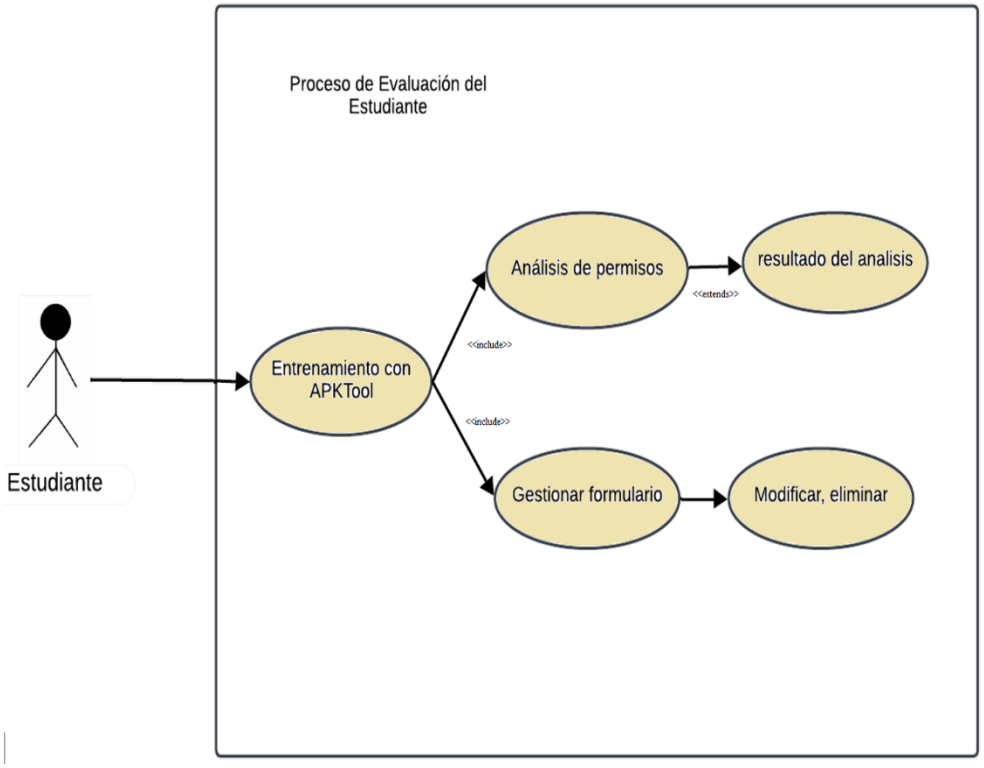
Descripción del caso de uso: Registro Administración de Contenido Institucional

Caso de Uso	**Administración de Contenido Institucional**
Actores	Director, Secretario, Asistente, Docentes
Tipo	Primario
Descripción	Los usuarios con los roles asignados podrán administrar las publicaciones institucionales, lo que incluye la creación, modificación y eliminación de contenido

Nota. En esta tabla se describe las funciones de Director, Secretario, Asistente, Docentes.

Figura 15

Caso de Uso: Proceso de Evaluación del Estudiante



Nota. El diagrama ilustra como el Estudiante interactuara en el sistema.

Tabla 12

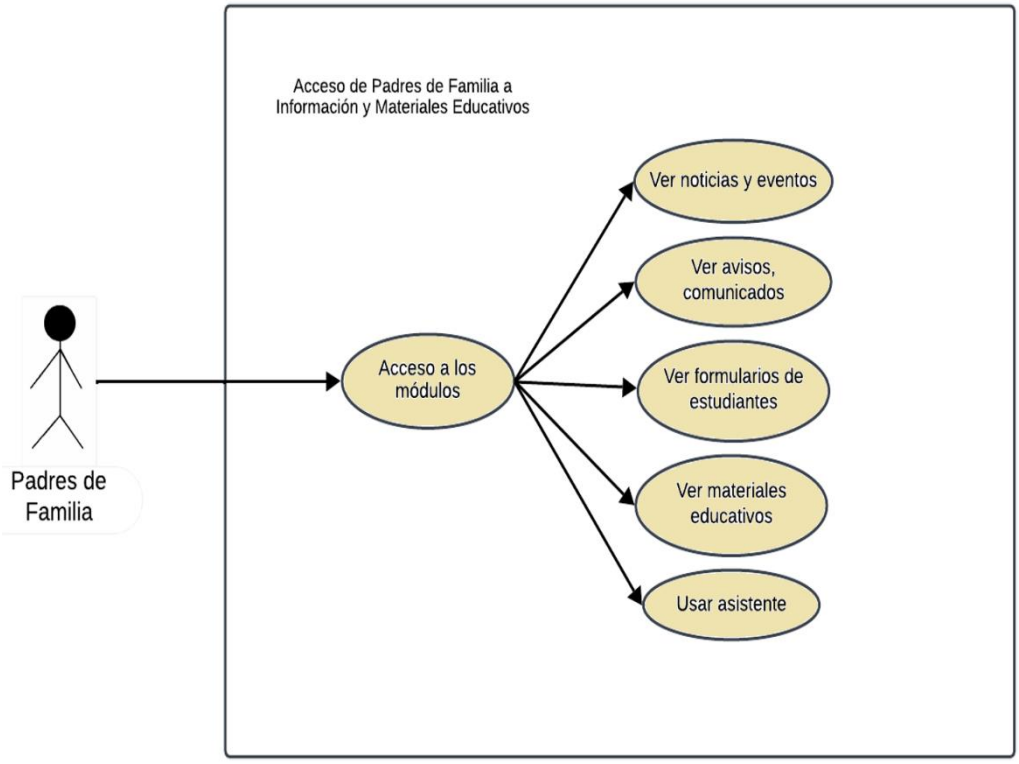
Descripción del caso de uso: Proceso de Evaluación del Estudiante

Caso de Uso	**Proceso de Evaluación del Estudiante**
Actores	Estudiante
Tipo	Primario
Descripción	El estudiante recibirá un entrenamiento sobre la decompilación aplicaciones móviles en los módulos con contenido educativo con la herramienta externa APKTool y también hará uso del asistente, que detallará los permisos del archivo AndroidManifest.xml. Además, podrá gestionar uno o más formularios en los que ingresará sus datos personales y proporcionará respuestas relacionadas con el análisis de aplicaciones Android

Nota. En esta tabla se muestra las funciones del estudiante.

Figura 16

Caso de Uso: Acceso de Padres de Familia a los Módulos del Sistema



Nota. El diagrama ilustra como los Padres de Familia interactuara en el sistema.

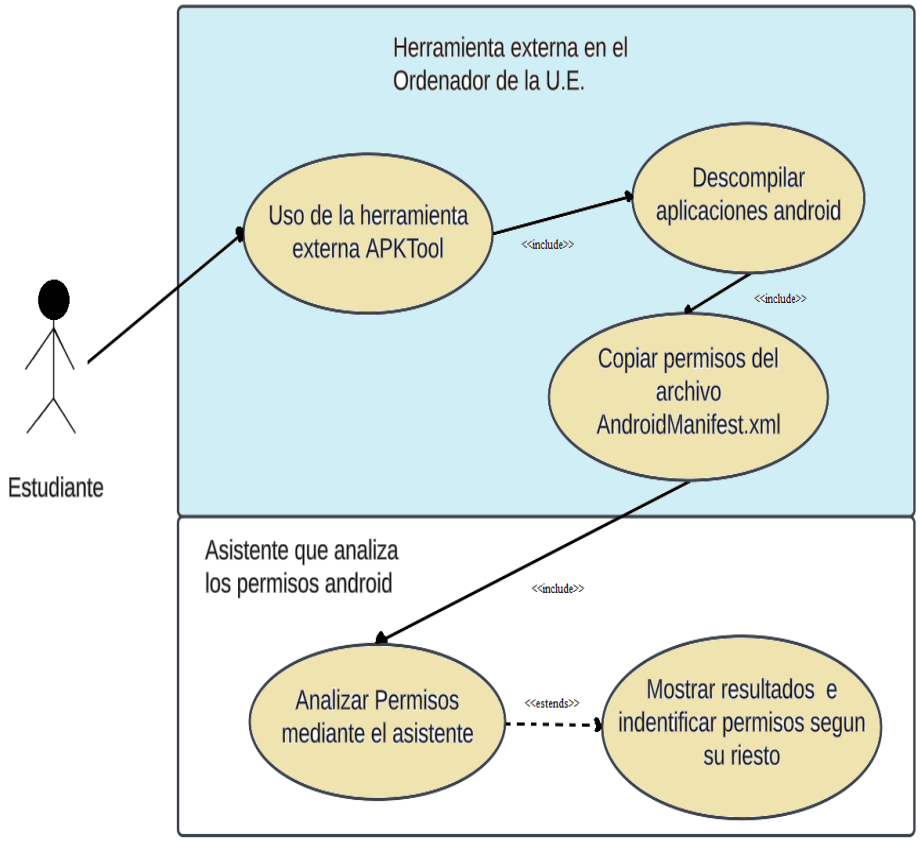
Tabla 13

Descripción del caso de uso: Accesos de Padres de Familia a los Módulos del Sistema

Caso de Uso	**Acceso de Padres de Familia a los Módulos del Sistema**
Actores	Padres de Familia
Tipo	Primario
Descripción	Los usuarios como padres de familia solo podrán ver contenido del sistema, no tendrán acceso a módulos para gestionar modificar o eliminar y tampoco vera los módulos del administrador

Nota. En esta tabla se muestra las funciones del padre de familia.

Figura 17
Caso de Uso: Análisis con la Herramienta Externa



Nota. El diagrama ilustra como el Estudiante interactuara en el sistema ya también usando la herramienta externa.

Tabla 14

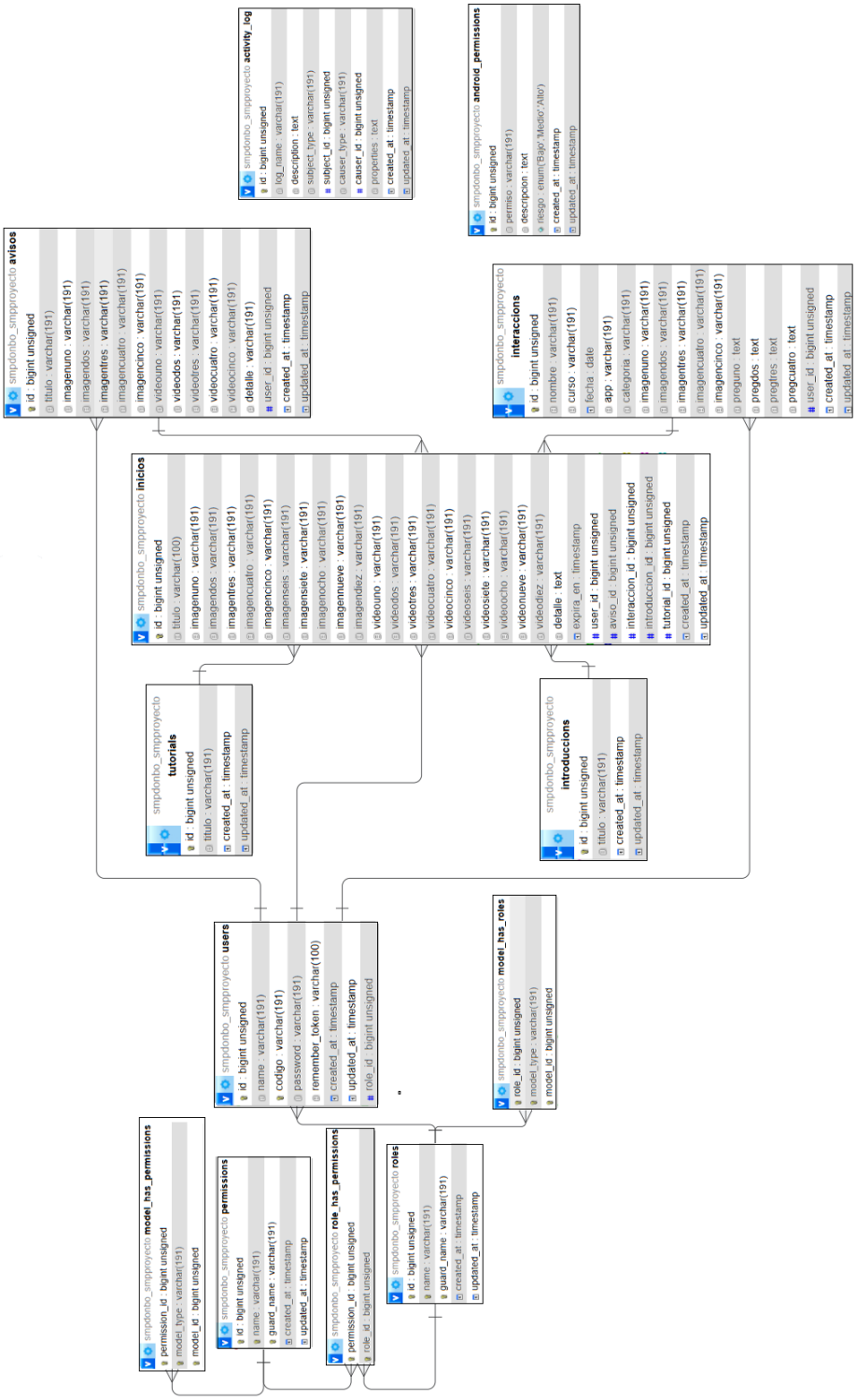
Descripción del caso de uso: Análisis con la Herramienta Externa "APKTool"

Caso de Uso	**Análisis con la Herramienta Externa “APKTool”**
Actores	Estudiantes
Tipo	Primario
Descripción	Los estudiantes harán uso de la herramienta externa “APKTool” para decompilada aplicaciones Android para luego ser analizados por el asistente de permisos para así identificar los permisos sospechosos

Nota. En esta tabla se muestra las funciones del estudiante usando la herramienta externa.

3.2.8.2. Modelo conceptual

Figura 18
Modelo Conceptual de Sistema

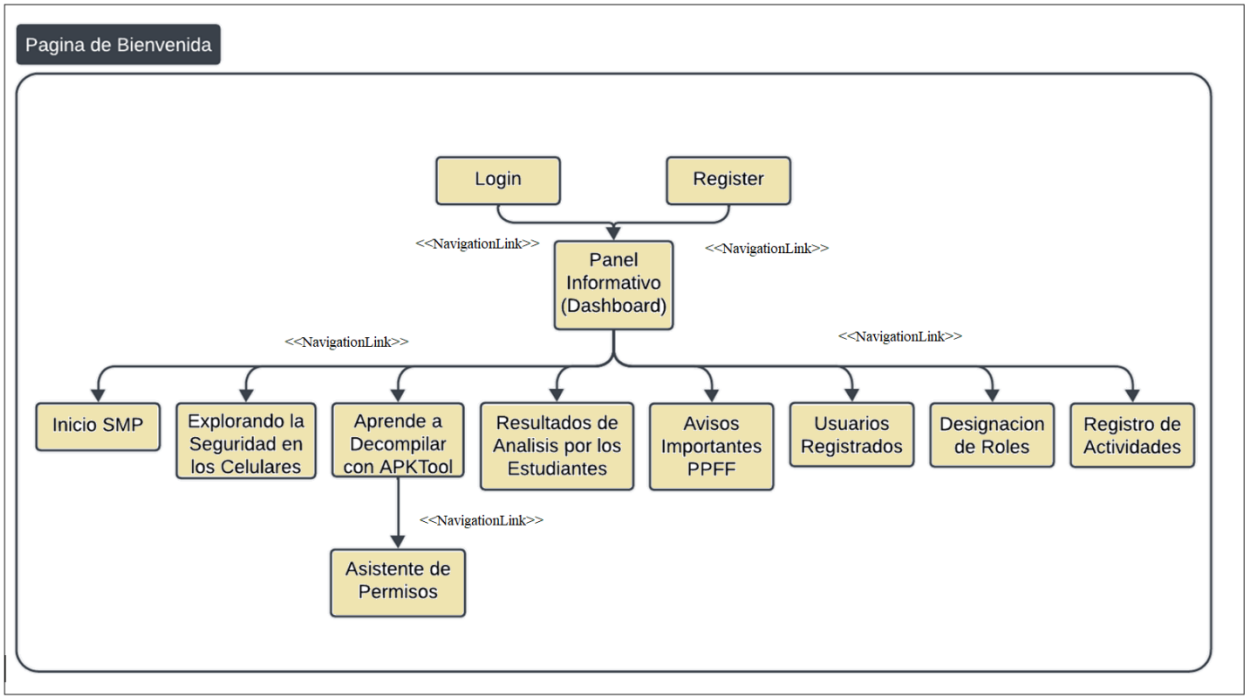


Nota. En este modelo conceptual ilustra las relaciones entre los principales componentes del sistema.

3.2.9. Fase 3: Diseño Navegacional

3.2.9.1. Diagrama de navegación del sistema

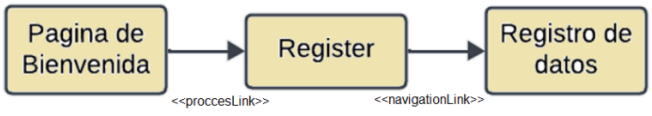
Figura 19
Organización de las Zonas de Interacción dentro del Sistema



Nota. En este diagrama se ve la navegación que tendrán los usuarios dentro del sistema.

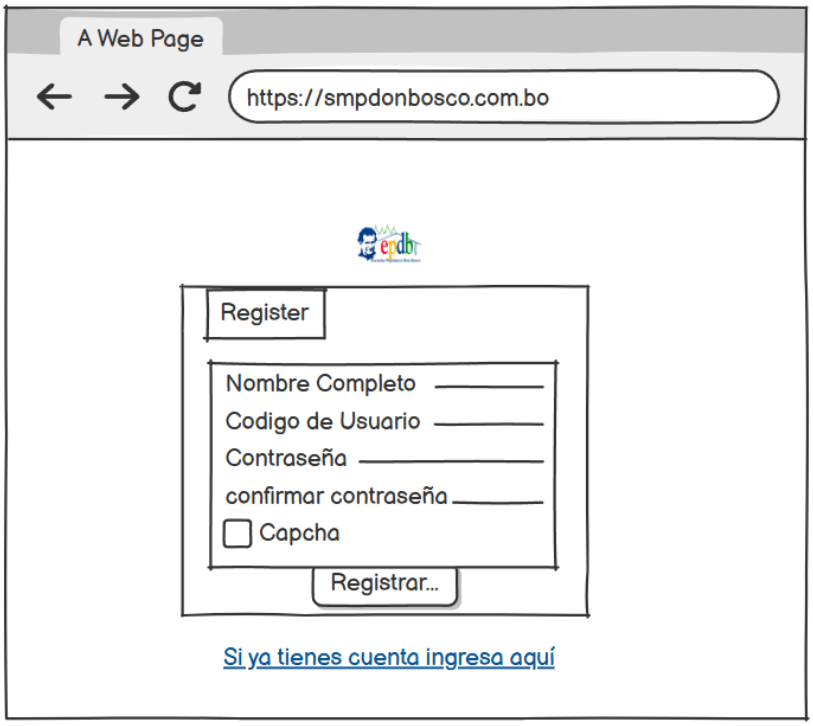
A continuación, se procederá a describir en detalle los diagramas de navegación del sistema web, los cuales ilustran cómo los usuarios interactúan con las diferentes secciones y funcionalidades de la plataforma. Estos diagramas representan visualmente el flujo de navegación dentro del sistema, facilitando la comprensión de las rutas disponibles para los usuarios y las interacciones entre las distintas páginas o módulos del sistema.

Figura 20
Diagrama de navegación Register



Nota. El diagrama muestra el proceso de navegación para el registro de nuevos usuarios.

Figura 21
Wireframe de la pantalla Register

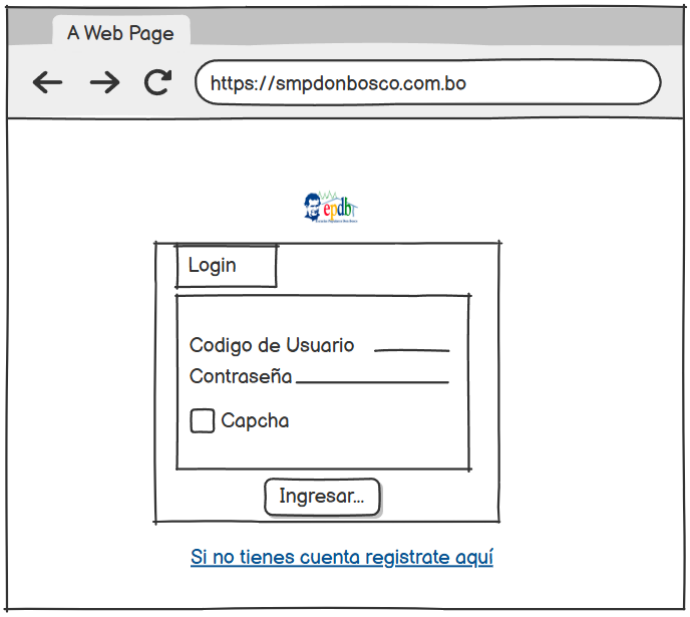


Nota. Este wireframe muestra la disposición de los campos para nombre, código de usuario, contraseña, confirmación de contraseña y CAPTCHA, junto con el botón de registro.

Figura 22
Diagrama de navegación Login

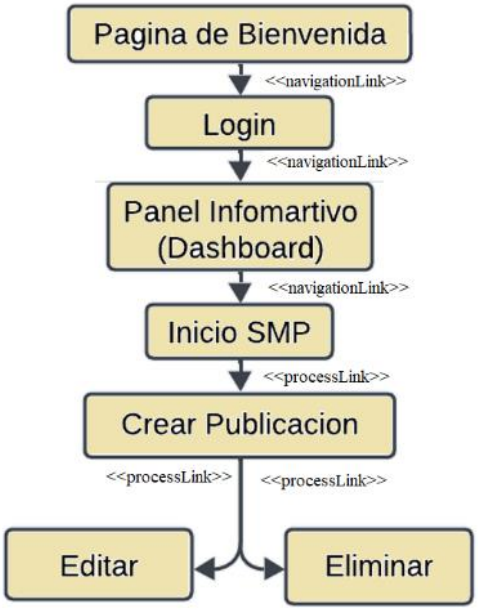
Nota. El diagrama muestra el proceso de navegación para el Login de los usuarios registrados en el sistema

Figura 23
Wireframe de la pantalla Login



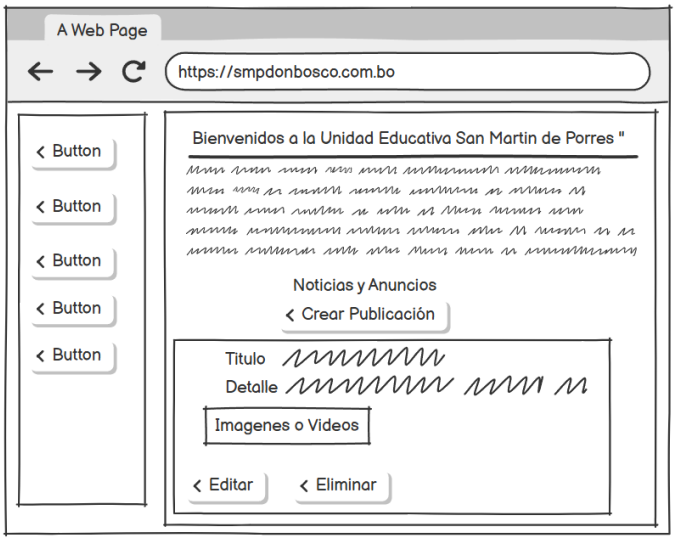
Nota. Este wireframe muestra la disposición de los campos para ingresar el código de usuario, contraseña y CAPTCHA, junto con el botón "Ingresar" para completar el inicio de sesión.

Figura 24
Diagrama de navegación Inicio SMP



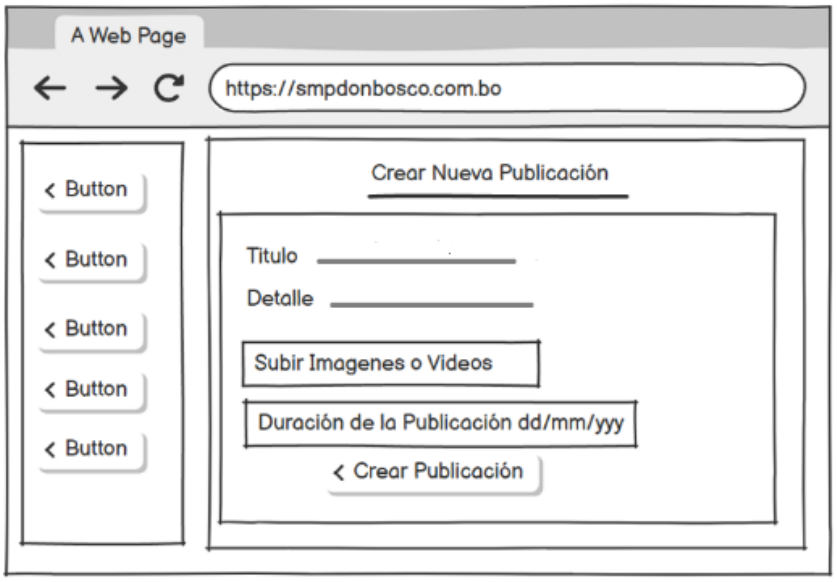
Nota. El diagrama muestra el proceso de navegación en el módulo Inicio SMP

Figura 25
Wireframe del módulo Inicio SMP



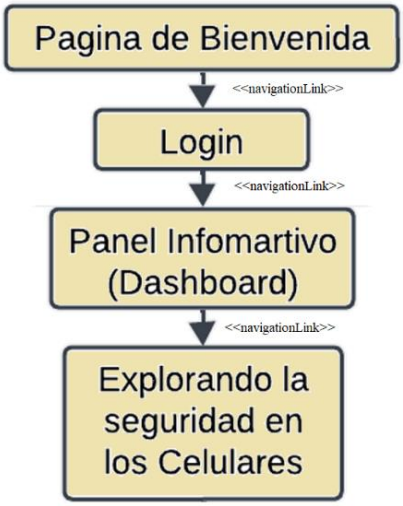
Nota. En este wireframe muestra el contenido del módulo Inicio SMP juntamente con las funciones de crear publicación, editar y eliminar.

Figura 26
Wireframe de la disposición de los elementos de la pantalla de Inicio SMP->crear publicación



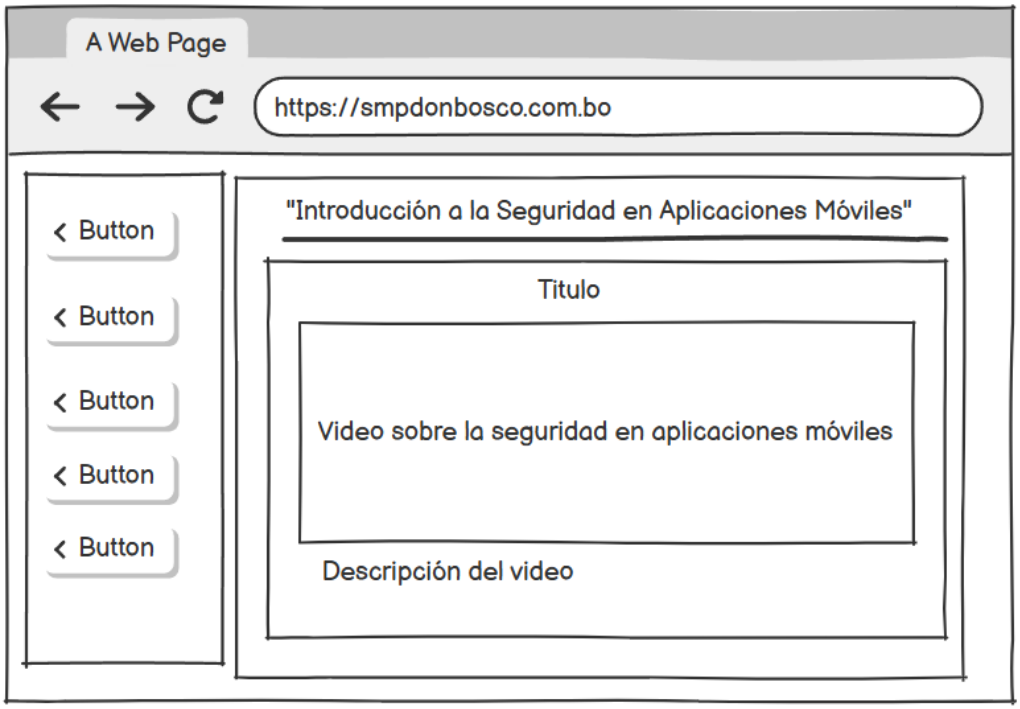
Nota. Este wireframe muestra el formulario para crear una publicación, que incluye campos para título, detalle, subir imagen y subir video (opcionales), así como la duración de la publicación, con un botón Publicar para guardar la entrada.

Figura 27
Diagrama de navegación del módulo Explorando la Seguridad en Aplicaciones Móviles



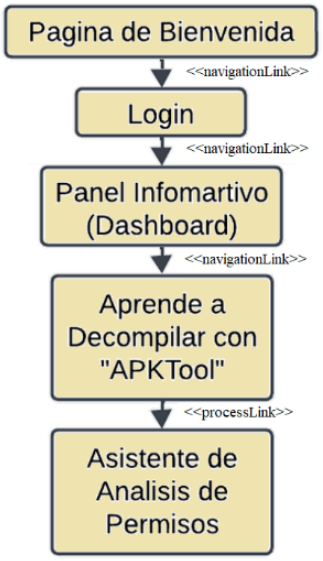
Nota. El diagrama muestra el proceso de navegación en el módulo de Explorando la Seguridad en los Celulares.

Figura 28
Wireframe del módulo Explorando la Seguridad en Aplicaciones Móviles



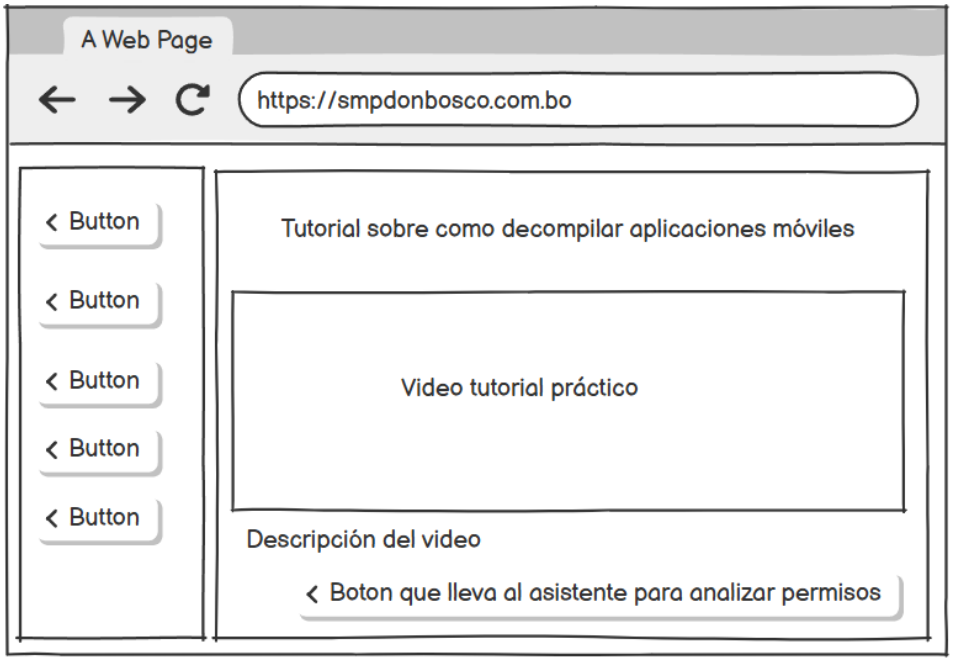
Nota. El wireframe muestra el título, un video reproductor destacado para contenido educativo y una descripción debajo que ofrece detalles adicionales sobre los riesgos en aplicaciones Android y cómo mitigarlos.

Figura 29
Diagrama de navegación para el módulo Aprende a Decompilar con "APKTool"



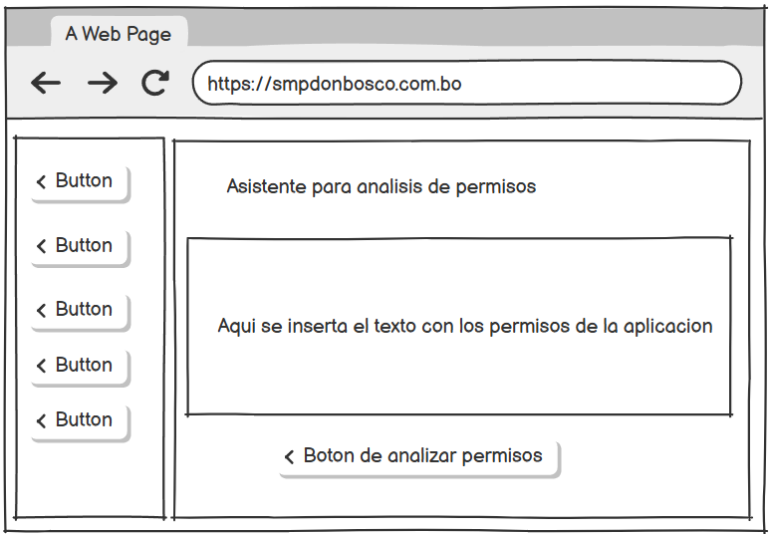
Nota. El diagrama muestra el proceso de navegación en el módulo Aprende a Decompilar con APKTool y juntamente con la función de Análisis de Permisos.

Figura 30
Wireframe del módulo Aprende a Decompilar con "APKTool"



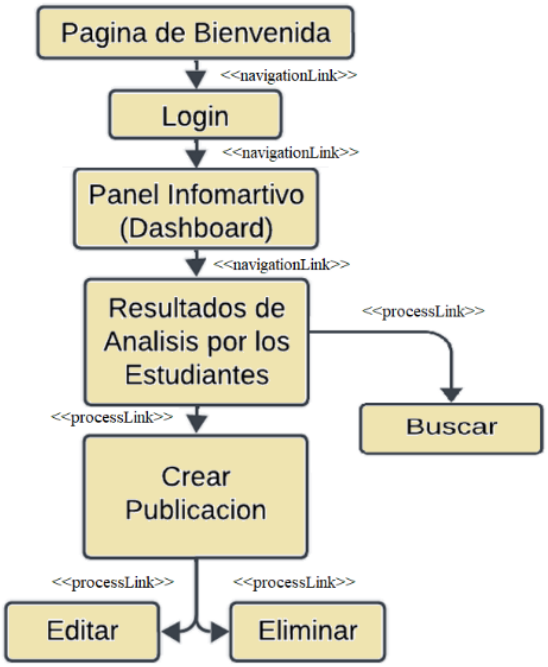
Nota. El wireframe muestra el título "¡Descubre cómo decompilar aplicaciones Android con APKTool!", un video tutorial destacado y una descripción que explica el contenido. Al final, se incluye un botón para acceder al "Asistente para Análisis de Permisos".

Figura 31
Wireframe del módulo Asistente para el Análisis de Permisos



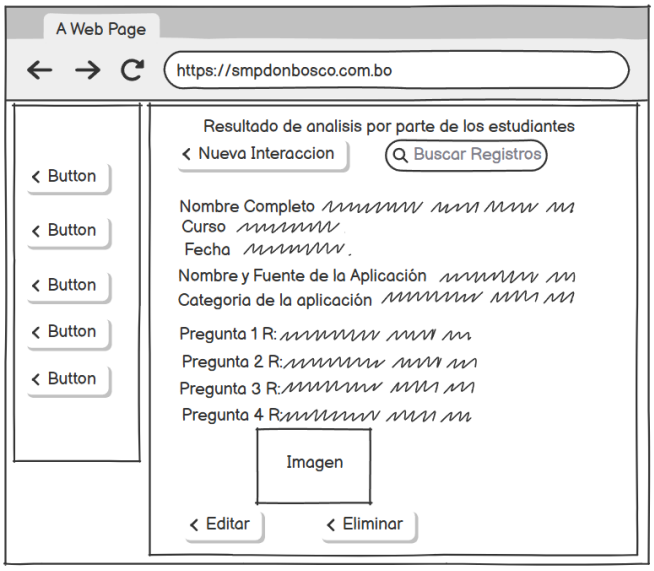
Nota. El wireframe muestra el título "Asistente para análisis de permisos APK", un campo de texto donde el usuario puede pegar los permisos del archivo AndroidManifest.xml, y un botón "Analizar Permisos" para procesar y analizar los permisos proporcionados.

Figura 32
Diagrama de navegación: Resultado de Análisis por los Estudiantes



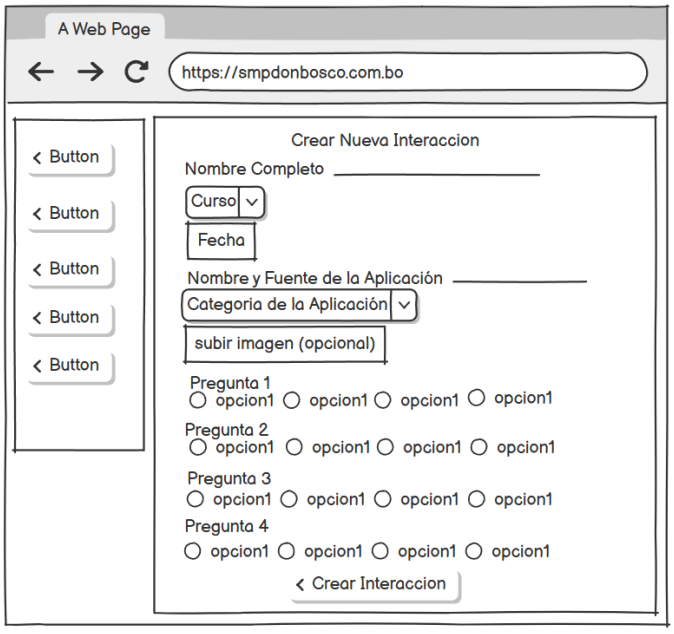
Nota. El diagrama muestra el proceso de navegación en el módulo Resultado de Análisis por los Estudiantes.

Figura 33
Wireframe del módulo Resultado de Análisis por parte de los estudiantes



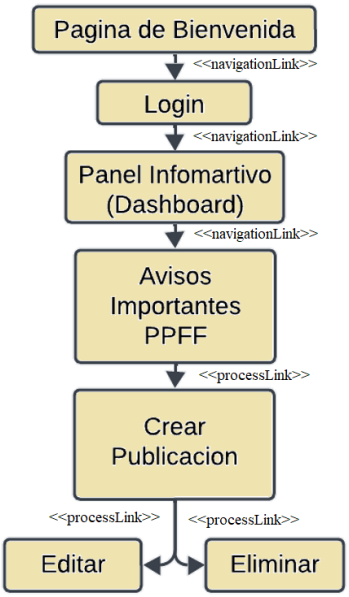
Nota. Este wireframe muestra los resultados del Análisis por parte de los Estudiantes juntamente con las funciones de buscar, nueva interacción, editar y eliminar el análisis.

Figura 34
Wireframe del formulario de registro de los estudiantes



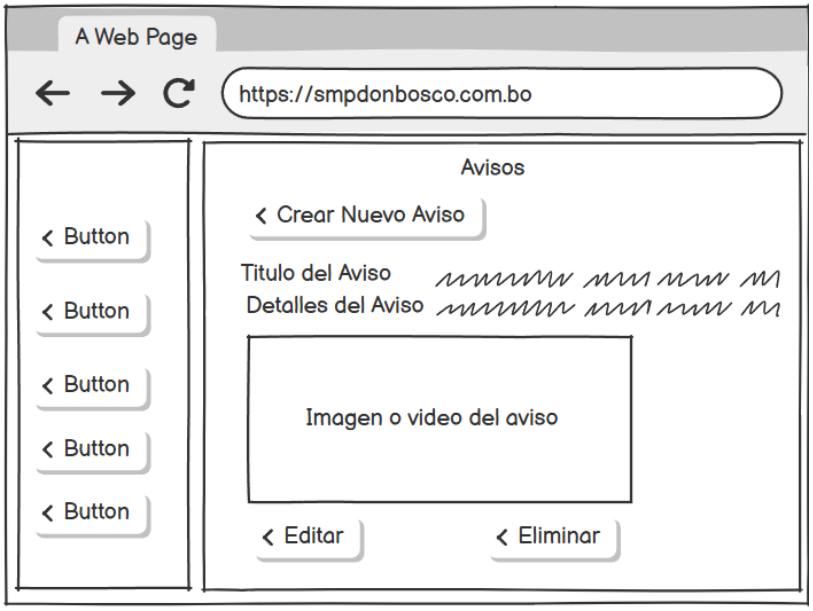
Nota. El wireframe muestra un formulario con campos para ingresar el nombre completo, seleccionar curso, fecha del análisis, nombre y fuente de la aplicación, seleccionar categoría, y subir imagen. Además, incluye preguntas de selección única y múltiple sobre la seguridad y funcionalidad de la aplicación. Finalmente, un botón "Guardar" permite enviar el formulario para su procesamiento.

Figura 35
Diagrama de navegación al módulo Avisos Importantes PPFF



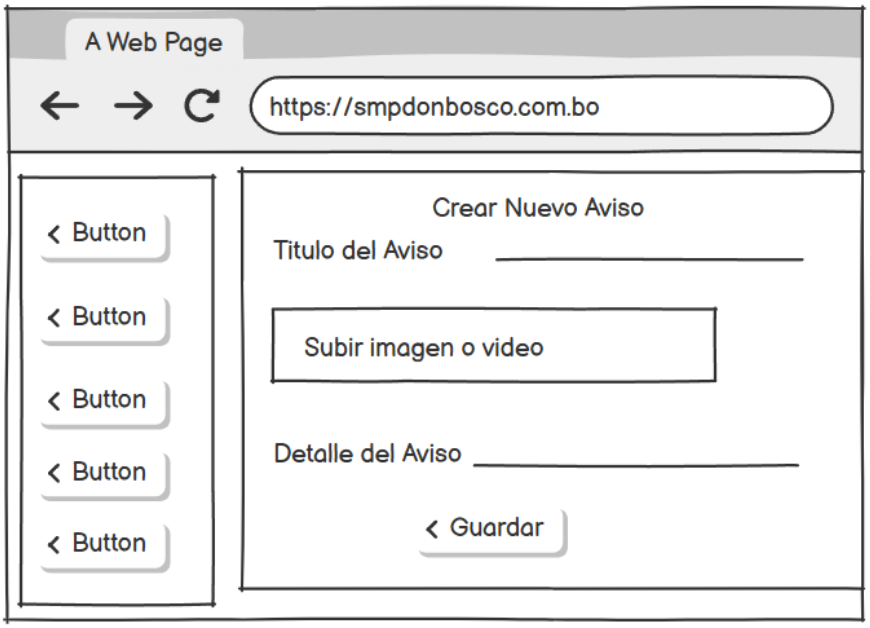
Nota. El diagrama muestra el proceso de navegación en el Módulo Avisos Importantes PPFF.

Figura 36
Wireframe del Módulo Avisos Importantes PPFF



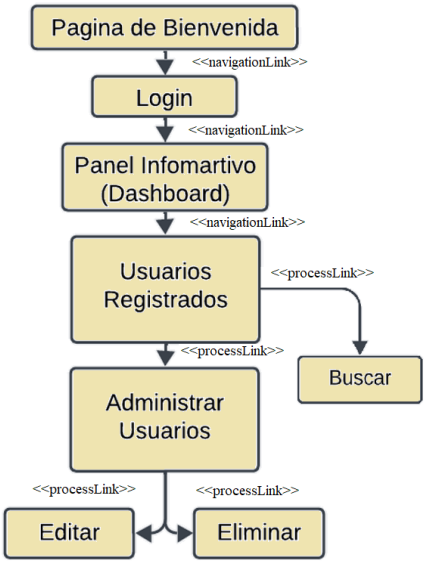
Nota. Este wireframe muestra las publicaciones de avisos de la institución juntamente con las funciones de crear, editar y eliminar Avisos.

Figura 37
Wireframe del formulario de Registro de Avisos Importantes



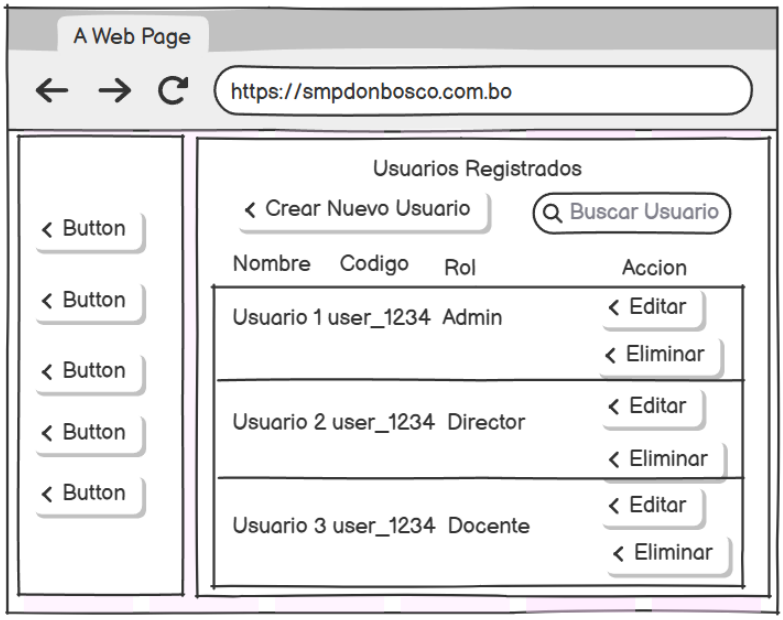
Nota. Este wireframe muestra un formulario con campos para ingresar el título del aviso, subir imagen y video (opcionales), y escribir los detalles del aviso. Un botón "Guardar Aviso" permite almacenar la información y los archivos (si se han subido).

Figura 38
Diagrama de navegación al módulo Usuarios Registrados



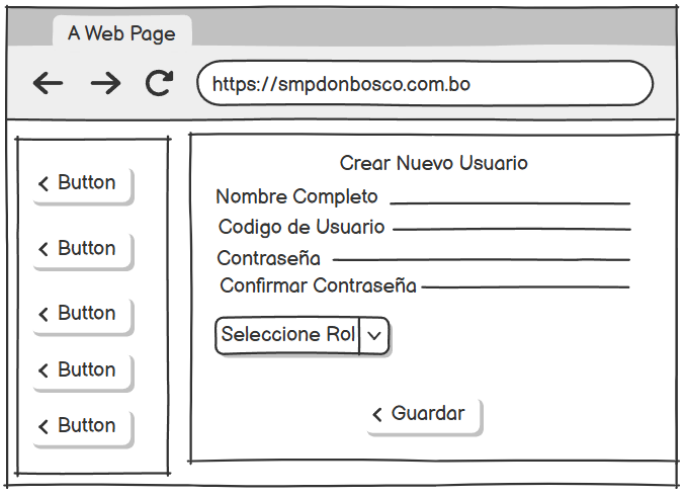
Nota. El diagrama muestra el proceso de navegación en el módulo de Usuarios Registrados.

Figura 39
:Wireframe del módulo Usuarios Registrados



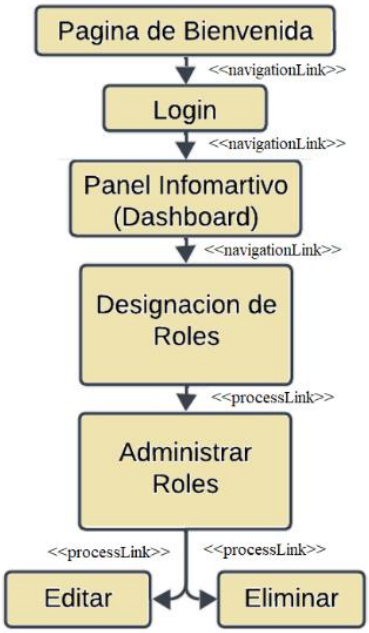
Nota. Este wireframe muestra los registros de los usuarios conjuntamente con la función de buscar, crear, editar y eliminar usuarios.

Figura 40
Este wireframe muestra el formulario de Registro de Nuevo Usuario



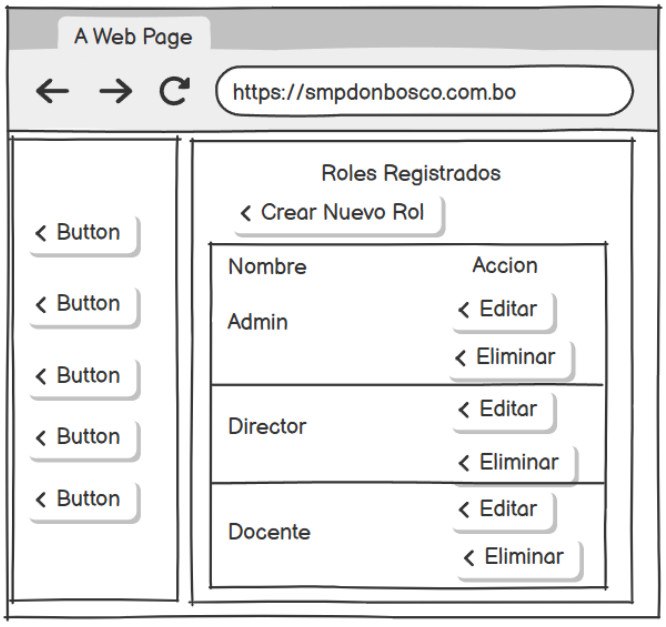
Nota. Este wireframe presenta un formulario con campos para ingresar el nombre completo, código de usuario, contraseña (y su confirmación), y seleccionar el rol del usuario (Administrador, Estudiante o Profesor). Un botón "Guardar" permite registrar al nuevo usuario, activándose solo cuando todos los campos están completos y las validaciones son correctas.

Figura 41
Diagrama de navegación al módulo Designación de Roles



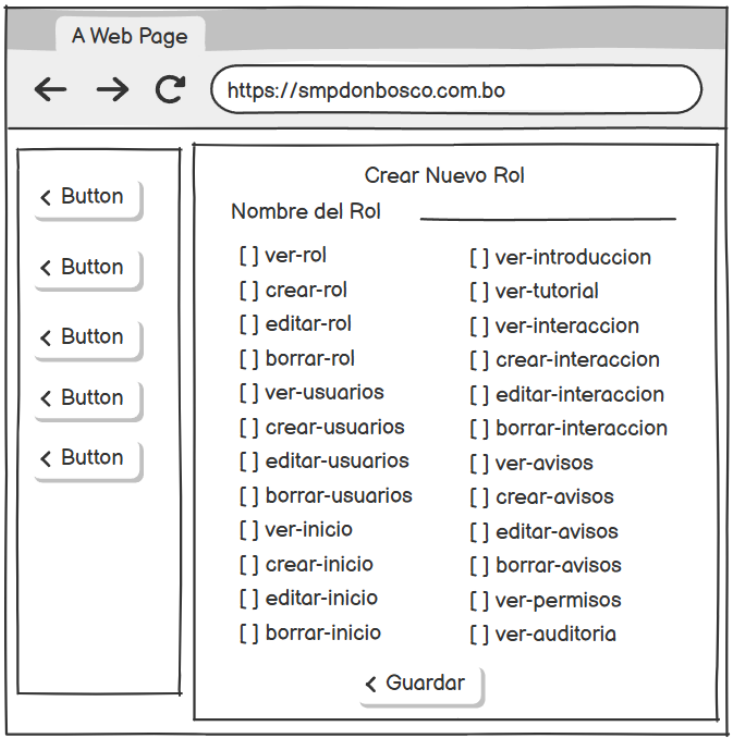
Nota. El diagrama muestra el proceso de navegación en el módulo Designación de Roles.

Figura 42
Wireframe del módulo Roles Registrados



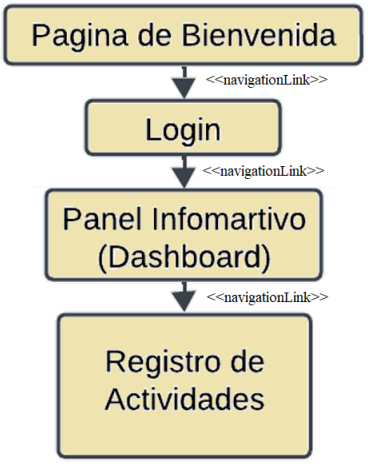
Nota. En este wireframe se ven los roles registrados con las funciones de crear, editar y eliminar.

Figura 43
Este wireframe muestra el formulario de Registro de Nuevo Rol



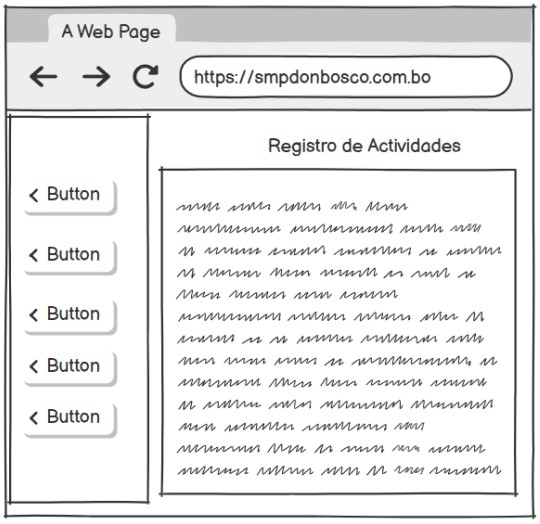
Nota. El formulario incluye un campo de texto para ingresar el nombre del rol y una lista de permisos representados por casillas de verificación. Un botón "Guardar" permite almacenar el nuevo rol y sus permisos.

Figura 44
Diagrama de navegación al módulo Registro de Actividades



Nota. El diagrama muestra el proceso de navegación en el módulo de Registro de Actividades.

Figura 45
Wireframe del módulo Registro de Actividades



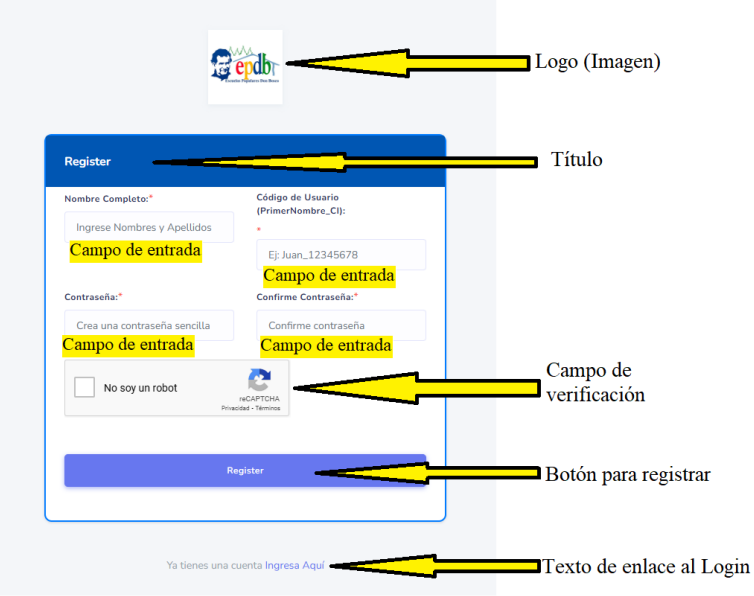
Nota. El wireframe muestra una tabla con las columnas: Usuario (nombre del usuario), Descripción (acción realizada), y Fecha (fecha de la acción).

3.2.10. Fase 4: Diseño de la Interfaz Abstracta

Figura 46
Pantalla de Bienvenida con Imagen de Fondo y Navegación Inicial

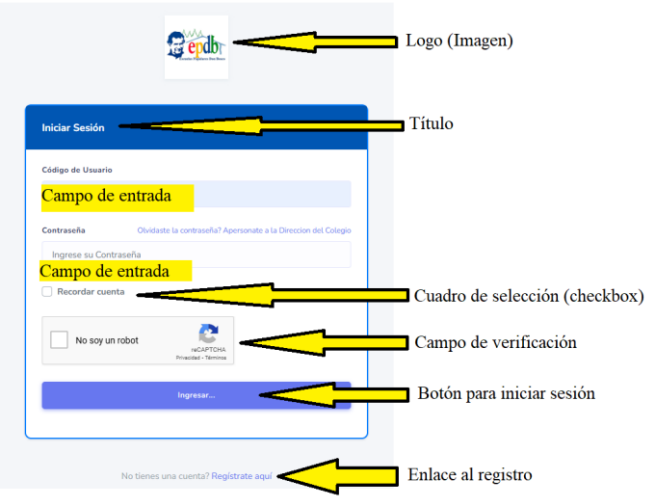
Nota. La vista de bienvenida muestra un fondo institucional con el título de la institución. En el centro, se encuentra el botón Acceder al Sistema, y a los costados los botones de ingresar y regístrate, facilitando el acceso al sistema para los usuarios.

Figura 47
Pantalla de Registro de Usuario con Verificación de CAPTCHA



Nota. La vista Register permite que un nuevo usuario se registre en el sistema. Incluye campos para nombre completo, código de usuario, contraseña y confirmar contraseña, y también una verificación de CAPTCHA. Al completar el formulario, el usuario puede hacer clic en el botón de Registrar para crear una cuenta.

Figura 48
Pantalla de Login (Ingreso al Sistema)



Nota. La vista Login permite a los usuarios registrados acceder al sistema. Presenta campos para ingresar el código de usuario y la contraseña, así como un botón de ingreso para iniciar sesión. También incluye un CAPTCHA para validar la autenticidad de la solicitud.

Figura 49

Pantalla de Dashboard con Menú Lateral y Header

Nota. En el Panel Informativo (Dashboard) muestra el menú y tarjetas donde están videos y funciones a los demás módulos, donde el usuario nuevo accederá directamente a este módulo.

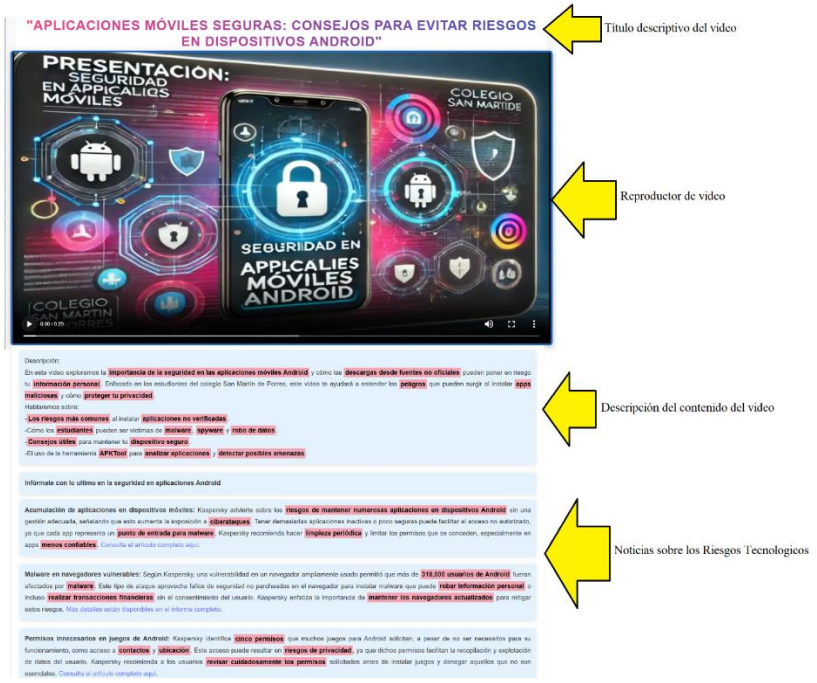
Figura 50

Pantalla de Inicio SMP (Pantalla Principal)

Nota. En esta vista se muestra la información de la institución, también se muestra las noticias y anuncios donde están las funciones de crear, editar y eliminar noticias y anuncios.

Figura 51

Pantalla de Explorando la Seguridad en los Celulares



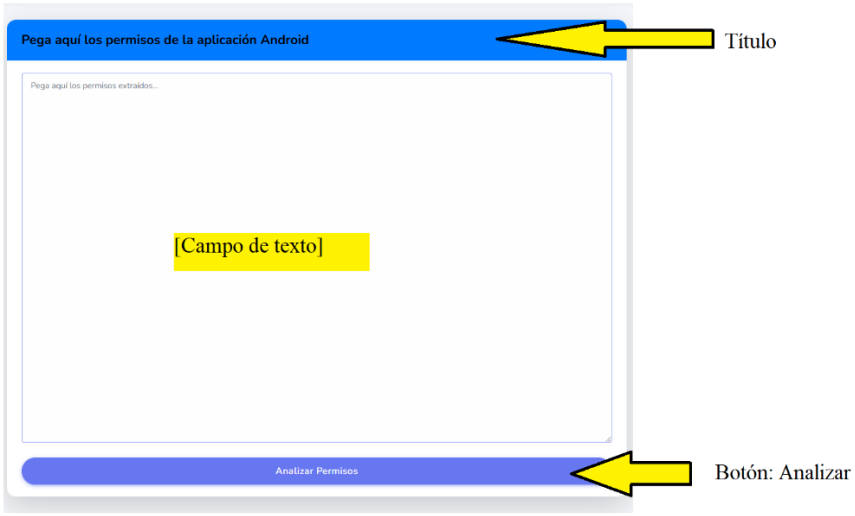
Nota. En esta vista se muestra contenido educativo sobre la seguridad en aplicaciones móviles.

Figura 52

Pantalla de Aprende a Decompilar con "APKTool"

Nota. En esta vista se muestra contenido educativo sobre como decompilar aplicaciones móviles usando la herramienta externa APKTool, también tiene el botón para describir códigos de permisos, lo cual llevara al Asistente de Análisis de Permisos.

Figura 53
Pantalla de Asistente para el Análisis de Permisos



Nota. En esta vista se muestra la función del Asistente donde solo recibe permisos obtenidos del archivo AndroidManifest.xml y el botón de Analizar Permisos.

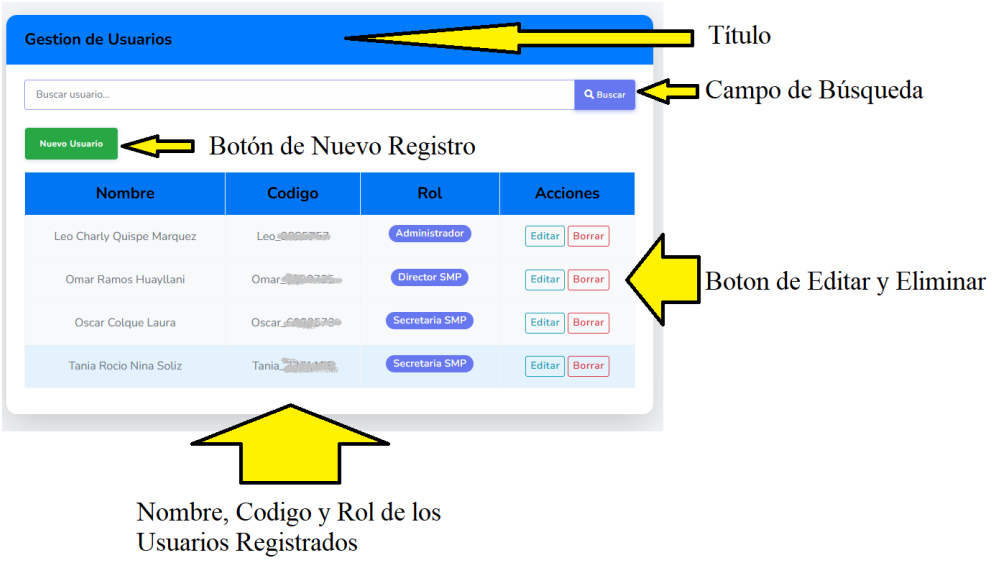
Figura 54
Pantalla de Resultado de Análisis por los Estudiantes

Nota. En esta vista se muestran los resultados del análisis de aplicaciones móviles por parte de los estudiantes de la institución, juntamente con las funciones de buscar, crear nueva interacción, editar y eliminar.

Figura 55
Pantalla de Avisos Importantes PPFF

Nota. En esta vista se muestra los avisos de la institución.

Figura 56
Pantalla de Usuarios Registrados

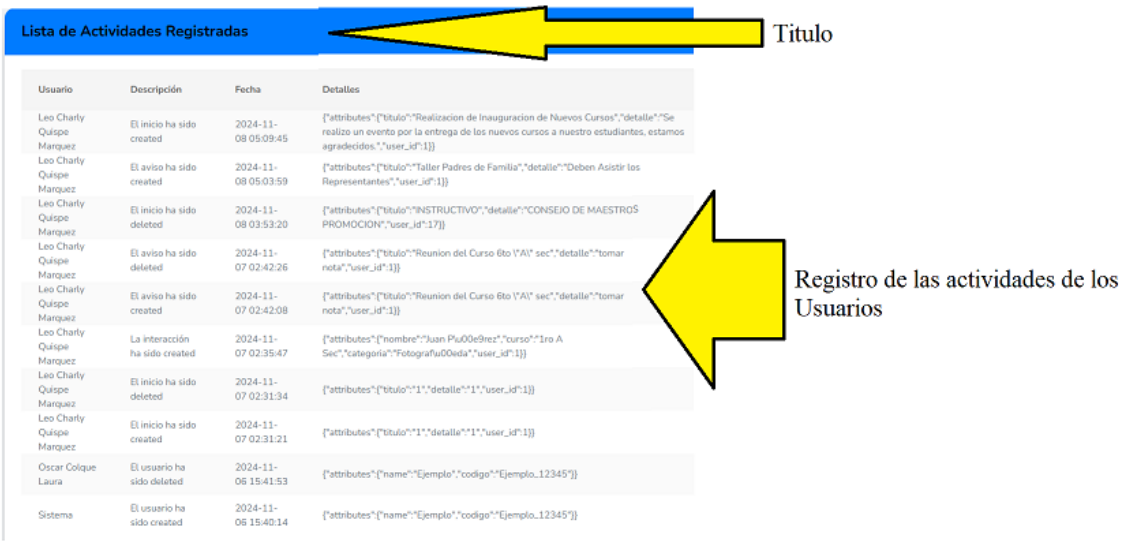


Nota. En esta vista se muestra los usuarios registrado con sus roles respectivos, también con las funciones de buscar, crear nuevo usuario, editar y eliminar.

Figura 57
Pantalla de Gestión de Roles

Nota. En esta vista se muestra los roles registrados, también cuenta con las funciones de crear nuevo rol, editar y eliminar.

Figura 58
Pantalla de Registro de Actividades



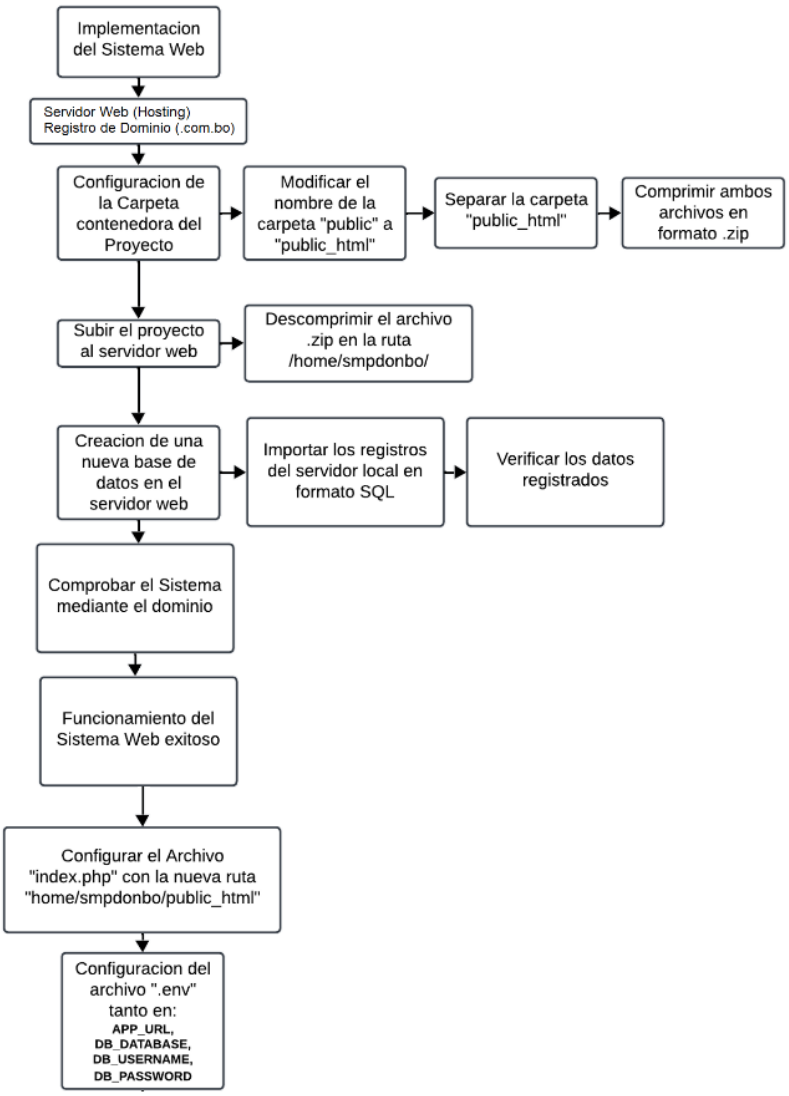
Nota. En esta vista se muestra el registro de las actividades que realiza cada usuario en los diferentes módulos del sistema.

3.2.11. Fase 5: Implementación

3.2.11.1.Diagrama de despliegue

Para garantizar la correcta ejecución del Sistema Web en el servidor web, se llevaron a cabo una serie de pasos fundamentales para migrar el sistema desde el servidor local al servidor web. Con el fin de visualizar claramente el proceso, se elaboró un diagrama que ilustra los pasos involucrados en la subida del sistema web.

Figura 59
Diagrama de despliegue del Sistema Web



Nota. En este diagrama se muestra el proceso y configuración para subir el proyecto al servidor web.

3.2.11.2.Migración al servidor web (hosting) y el dominio (.com.bo)

Para llevar a cabo la migración del Sistema Web desde un servidor local hacia un servidor web (hosting), es fundamental contar con los siguientes recursos:

3.2.11.2.1 Servidor Web (hosting):

Figura 60
Proveedor de servicios de alojamiento web.

Fuente: AvanceHost, 2024, https://www.avancehost.com

3.2.11.2.2 Registro de Dominio (smpdpnbosco.com.bo):

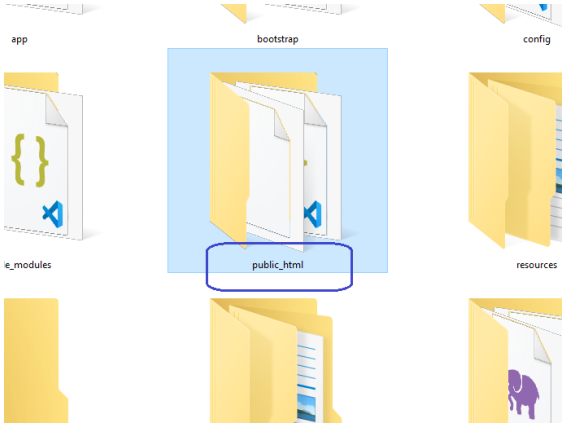
Figura 61
Proveedor de servicios de alojamiento web.

Fuente: WebHosting.com.bo, 2024, https://www.webhosting.com.bo

3.2.11.3.Flujo de Trabajo para la Migración del Sistema Web al Hosting

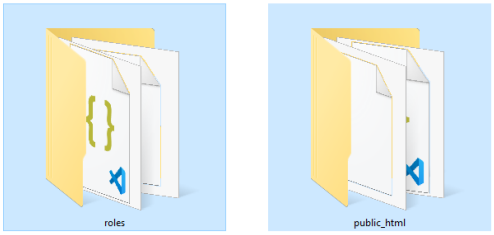
Para el proceso de migración del sistema web, se desarrolló el proyecto con el nombre de “roles” en el cual se desarrolló en el servidor local.

Figura 62
Configuración de la Carpeta del Proyecto: Renombrar y Organizar Archivos



Nota. En esta imagen se muestra la configuración de la carpeta public renombrándolo a public_html.

Figura 63
Carpeta public_html separada de su origen

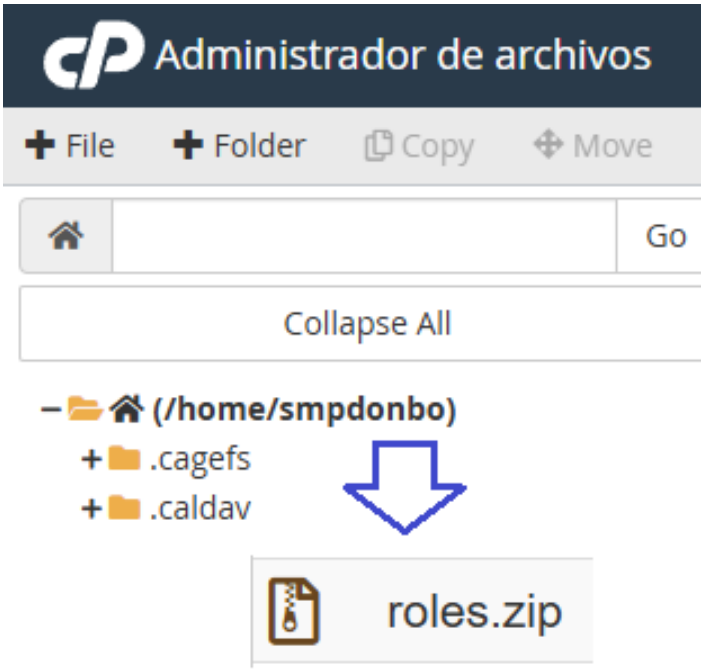


Nota. En esta imagen se muestra el proceso de separar la carpeta public_html fuera de la carpeta principal del proyecto para estructurar adecuadamente los archivos.

Figura 64
Compresión de Archivos del Proyecto

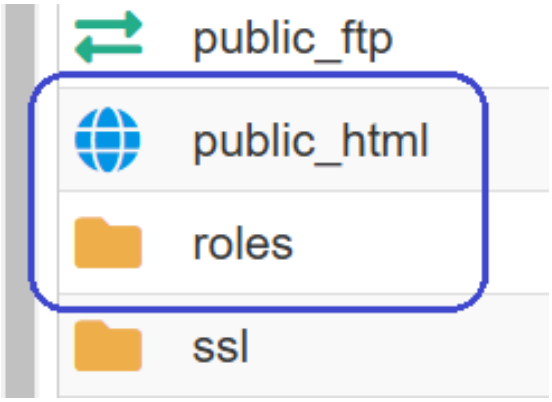
Nota. Una vez organizada la estructura de carpetas, comprimir ambas carpetas (proyecto principal y public_html) en un archivo ZIP para facilitar la transferencia al servidor web.

Figura 65
Migración de la carpeta roles al cpanel del hosting



Nota. Aquí se muestra cómo se subió el archivo comprimido al servidor web (hosting) utilizando un cliente cPanel, ubicándolo en la ruta: /home/smpdonbo/.

Figura 66
Carpetas descomprimidas (roles, public_html)

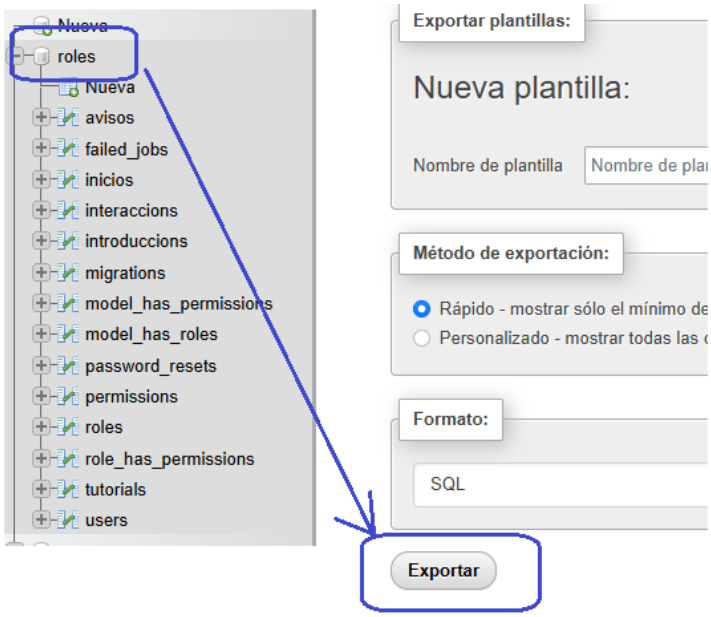


Nota. Aquí se procede a acceder al servidor y descomprimir el archivo ZIP en la misma ruta /home/smpdonbo/ para restaurar la estructura original del proyecto.

Figura 67
Creación de la base de datos y el usuario en el servidor web

Nota. Aquí se crea una nueva base de datos y un Usuario en el servidor web, para almacenar los registros y datos del proyecto.

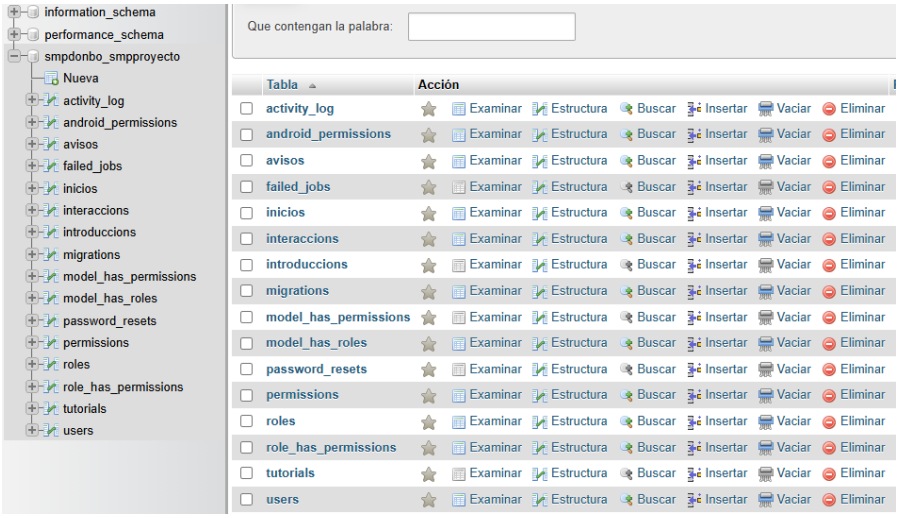
Figura 68
Exportación de los registros en formato SQL del servidor local



Nota. Aquí exportamos los registros principales desde la base de datos local en formato SQL.

Figura 69

Carga del archivo SQL en la base de datos del servidor web



Nota. Aquí se procedió a importar el archivo SQL al servidor web para que se carguen los registros que serán importantes a la hora de ingresar al sistema.

Figura 70

Modificación del archivo index.php con la ruta del servidor web

```php
<?php

use Illuminate\Contracts\Http\Kernel;
use Illuminate\Http\Request;

define('LARAVEL_START', microtime(true));

/*
|--------------------------------------------------------------------------
| Check If Application Is Under Maintenance
|--------------------------------------------------------------------------
|
| If the application is maintenance / demo mode via the "down" command we
| will require this file so that any prerendered template can be shown
| instead of starting the framework, which could cause an exception.
|
*/

if (file_exists('/home/smpdonbo/roles/storage/framework/maintenance.php')) {
    require '/home/smpdonbo/roles/storage/framework/maintenance.php';
}

/*
|--------------------------------------------------------------------------
| Register The Auto Loader
|--------------------------------------------------------------------------
|
| Composer provides a convenient, automatically generated class loader for
| this application. We just need to utilize it! We'll simply require it
| into the script here so we don't need to manually load our classes.
|
*/

require '/home/smpdonbo/roles/vendor/autoload.php';

/*
|--------------------------------------------------------------------------
| Run The Application
|--------------------------------------------------------------------------
|
| Once we have the application, we can handle the incoming request using
| the application's HTTP kernel. Then, we will send the response back
| to this client's browser, allowing them to enjoy our application.
|
*/

$app = require_once '/home/smpdonbo/roles/bootstrap/app.php';

$kernel = $app->make(Kernel::class);

$response = tap($kernel->handle(
    $request = Request::capture()
))->send();

$kernel->terminate($request, $response);
```

Nota. Aquí se procedió a configurar el archivo index.php con la ruta actual que está en el servidor web.

Figura 71
Configuración del archivo .env

Nota. Se procedió a modificar los archivos de configuración del proyecto, como index.php y .env, para asegurar que apunten a la base de datos del servidor web y ajusten las configuraciones de entorno adecuadas (rutas, credenciales de base de datos, etc.).

Figura 72
Prueba del Sistema Web mediante el dominio "smpdonbosco.com.bo "

Nota. Aquí se comprobó el correcto funcionamiento del sistema web accediendo al dominio configurado y verificando que todos los servicios y páginas del sistema se ejecuten correctamente.

CAPITULO IV

CALIDAD, COSTO, SEGURIDAD Y PRUEBAS

INGENIERÍA
DE SISTEMAS
UNIVERSIDAD PUBLICA DE EL ALTO

CAPITULO IV

4.1. INTRODUCCION

Este capítulo aborda los aspectos clave para garantizar el éxito y la viabilidad del sistema web desarrollado para la institución educativa, enfocándose en la calidad, los costos, la seguridad y las pruebas realizadas. Se presentan las estimaciones de costos, detallando tanto los gastos directos como adicionales, y se analiza la gestión de recursos necesarios para completar el proyecto. Además, se describen las medidas de seguridad implementadas para proteger la información y garantizar el acceso solo a usuarios autorizados. Finalmente, se detalla el proceso de pruebas realizadas al sistema para asegurar su correcto funcionamiento, eficiencia y estabilidad bajo diferentes condiciones.

4.2. CALIDAD DEL SISTEMA WEB

La calidad de un sistema web puede evaluarse según diversas dimensiones, y una de las más relevantes es la seguridad. En este contexto, el modelo de McCall se aplica para evaluar diversos atributos como la funcionalidad, fiabilidad, usabilidad, eficiencia en el desempeño, seguridad, y mantenibilidad. Estas características son fundamentales para garantizar que el sistema no solo cumpla con sus requisitos funcionales, sino que también opere de manera eficiente y segura en un entorno de producción.

McCall, en su enfoque de calidad, define estas métricas para asegurar que el sistema esté alineado con las expectativas de los usuarios y las buenas prácticas de desarrollo. A continuación, se presentan las métricas utilizadas para evaluar la calidad del sistema web.

4.2.2. Métrica de Calidad: Funcionalidad

La funcionalidad del sistema es un aspecto clave que cubre varios módulos esenciales para la operación de la aplicación. Estos módulos permiten a los usuarios realizar tareas básicas como registrarse, acceder al sistema, gestionar roles, y crear publicaciones.

Tabla 15

Descripción de Módulos Principales

Modulo	Descripción
Registro	Permite el registro de usuario mediante nombre, código único, contraseña y con verificación CAPTCHA.
Inicio de sesión	Acceso mediante código de usuario y contraseña, con verificación CAPTCHA.
Gestión de Usuarios	Creación, edición y eliminación de usuarios con asignación de roles.
Roles	Creación, edición y eliminación de roles, con asignación de permisos.
Inicio SMP	Publicación de texto, imágenes, y videos con una duración específica.
Explorando la Seguridad en los Celulares	Módulo informativo con campos de texto y videos insertados.
Aprende a Decompilar con “APKTool”	Módulo informativo con campos de texto y videos insertados.
Resultado de Análisis por los Estudiantes	Publicación de texto e imágenes para interacción entre usuarios.
Avisos Importantes PPFF	Permite la publicación de texto, imágenes y videos.
Asistente de Análisis	Inserta texto y analiza permisos al presionar el botón de análisis, proporcionando descripciones de permisos.
Registro de Actividades	Muestra el historial de movimientos de cada usuario en los distintos módulos.
Dashboard	Es una interfaz visual que muestra información del sistema.

Nota. En esta tabla se describe los módulos principales.

Tabla 16

Descripción de Pruebas Funcionales Realizadas

Prueba	Descripción	Resultado
Registro y asignación de roles	Verificación de registro de usuarios y asignación de roles específicos.	Completada exitosamente
Permisos en módulos	Prueba de acceso restringido a módulos según permisos asignados en roles.	Cumple las restricciones
Modificación de publicaciones	Verificación de que los usuarios no puedan modificar publicaciones de otros.	Aviso de restricción activado

Nota. En esta tabla se describe las pruebas funcionales realizadas en el sistema web.

4.2.3. Métrica de calidad: Fiabilidad

La fiabilidad se refiere a la capacidad del sistema para funcionar sin fallos importantes durante su operación. Según McCall, un sistema confiable debe estar disponible y ser capaz de manejar fallos de manera eficiente.

Tabla 17

Descripción de las Métricas de Fiabilidad

Aspecto	Descripción
Disponibilidad	El sistema ha estado en línea el 98% del tiempo desde su implementación.
Frecuencia de errores	No se han reportado errores significativos durante el último mes.

Nota. En esta tabla se muestra las métricas de fiabilidad en el sistema.

4.2.4. Métrica de calidad: Usabilidad

La usabilidad evalúa la facilidad de uso y la eficiencia del sistema desde la perspectiva del usuario. McCall destaca la importancia de un diseño intuitivo y accesible.

4.2.4.1. Opinión de los Usuarios y Resultados de la Encuesta

Se realizó una encuesta breve con los usuarios para evaluar la facilidad de uso y la satisfacción general con el sistema. Los resultados se presentan en la siguiente tabla:

Tabla 18

Evaluación de Usabilidad por los Usuarios

Aspecto Evaluado	Puntuación Promedio (1 a 5)	Comentarios de los Usuarios
Facilidad de uso	4	Los usuarios reportaron una interfaz intuitiva y fácil de navegar.
Facilidad de registro	4.2	Registro y Login sin problemas, pero recomiendan más opciones de ayuda.
Aprendizaje inicial	Los usuarios aprendieron a usar el sistema en menos de 10 minutos.	

Nota. En esta tabla se muestra la evaluación de usabilidad por los usuarios.

4.2.5. Métrica de calidad: Eficiencia en el Desempeño

La eficiencia en el desempeño mide el tiempo de respuesta del sistema bajo diferentes condiciones. McCall considera esencial que los sistemas web tengan un rendimiento adecuado, incluso en redes lentas.

Tabla 19

Descripción de Tiempo de Respuesta y Uso de Recursos

Tipo de red	Intento 1	Intento 2	Intento 3	Promedio	Tasa de éxito (%)
3G Lento	53.2 s	55.0 s	54.5 s	53.5 s	85%
4G Lento	7.5 s	7.5 s	7.5 s	7.5 s	95%
4G Rápido	1.77 s	1.80 s	1.78 s	1.77 s	100%

Nota. En esta tabla se muestra la descripción del uso del sistema en diferentes tipos de red.

4.2.6. Métrica de calidad: Seguridad

La seguridad es una de las características clave según McCall, y es esencial para proteger los datos y asegurar el correcto funcionamiento del sistema.

Tabla 20

Medidas Implementadas

Aspecto de seguridad	Descripción
Autenticación	Se implementa captcha para la autenticación de usuarios, y las contraseñas se encriptan usando bcrypt para protección adicional.
Encriptación	Contraseñas y datos sensibles almacenados en la base de datos usando bcrypt.
Monitoreo de actividad	El sistema registra cada acción del usuario, incluyendo la creación, edición y eliminación de publicaciones y otras actividades en módulos clave.

Nota. En esta tabla se muestra las medidas de seguridad implementadas en el sistema.

4.2.7. Métrica de calidad: Mantenibilidad

La mantenibilidad evalúa la facilidad con la que se pueden realizar cambios y actualizaciones en el sistema. McCall señala que un sistema fácil de mantener es crucial para su longevidad y adaptabilidad.

Tabla 21

Descripción de Métrica de Mantenibilidad

Aspecto de Mantenibilidad	Descripción
Estructura Modular	Código organizado en módulos independientes.
Documentación del Código	Se han incluido comentarios en las funciones y módulos donde se realizaron cambios, describiendo cada acción o ajuste para facilitar comprensión.
Actualizaciones Periódicas	Se planean actualizaciones periódicas cada seis meses, lo que permite mantener el sistema actualizado y adaptable a nuevas necesidades.

Nota. En esta tabla se muestra las métricas de mantenibilidad en el sistema web.

4.3. ESTIMACIÓN DEL COSTO

4.3.1. Modelo COCOMO II

El modelo COCOMO II se utiliza para calcular el esfuerzo requerido para completar un proyecto de software, tomando como base el tamaño estimado del código en **KLOC** (Kilo Líneas de Código). Posteriormente, el esfuerzo se utiliza para calcular tanto el tiempo de desarrollo como el número de personas necesarias para el proyecto.

Estimación del tamaño del software (KLOC)

El tamaño del proyecto se mide en KLOC (Kilo Líneas de Código), que representa la cantidad total de líneas de código que se espera desarrollar. En este proyecto, se ha estimado un tamaño de 8.786 KLOC (que equivale a 8,786 líneas de código). Este valor se obtiene generalmente a partir de la especificación del proyecto o de experiencias pasadas con proyectos similares.

4.3.1.1. Parámetros específicos del proyecto

El modelo COCOMO II ajusta los cálculos según el tipo de proyecto. Dependiendo de la naturaleza del proyecto, se seleccionan diferentes valores para los parámetros, que se extraen de la siguiente tabla:

Tabla 22

Clasificación de Proyecto de Software

Tipo de Proyecto	*a*	*b*	*c*	*d*
Orgánico	*2.4*	*1.05*	*2.5*	*0.38*
Semi-Organizado	*3.0*	*1.12*	*2.5*	*0.35*
Embebido	*3.6*	*1.20*	*2.5*	*0.32*

Nota. En esta tabla se usará el tipo de proyecto “Orgánico”.

4.3.1.2. Factor de ajuste de esfuerzo (EAF)

El EAF (Effort Adjustment Factor) es un parámetro clave que ajusta el esfuerzo estimado en función de varios factores del entorno de desarrollo.

Tabla 23

Tabla para medir el Ajuste de Esfuerzo

Módulo	Muy Bajo	Bajo	Nominal	Alto	Muy Alto	Extra Alto
Fiabilidad requerida del software	0.75	0.88	1.00	1.15	1.40	-
Tamaño de la base de datos	-.	0.94	1.00	1.08	1.16	-
Complejidad del Producto	0.70	0.85	1.00	1.15	1.30	1.65
Restricciones de tiempo de ejecución	-	-	1.00	1.11	1.30	1.66
Restricciones de almacenamiento	-	-	1.00	1.06	1.21	1.56
Volatilidad de la máquina virtual	-	0.87	1.00	1.15	1.30	-
Tiempo de respuesta del ordenador	-	0.87	1.00	1.07	1.15	-
Capacidad del analista	1.46	1.19	1.00	0.86	0.71	-
Experiencia en la aplicación	1.29	1.13	1.00	0.91	0.82	-
Capacidad de los programadores	1.42	1.17	1.00	0.86	0.70	-
Experiencia en S.O. utilizado	1.21	1.10	1.00	0.90	-	-
Experiencia en el lenguaje de programación	1.14	1.07	1.00	0.95	-	-
Prácticas de programación modernas	1.24	1.10	1.00	0.91	0.82	-
Utilización de herramientas de software	1.24	1.10	1.00	0.91	0.83	-
Limitaciones de planificación del proyecto	1.23	1.08	1.00	1.04	1.10	-

Nota. En esta tabla se muestra las medidas tomadas y calculando nos da como resultado EAF=1.7

El EAF tiene en cuenta características específicas del proyecto, como la experiencia del equipo, el entorno de hardware, las restricciones de tiempo y otros factores que pueden influir en la dificultad del proyecto.

El EAF se calcula utilizando la siguiente fórmula:

(2)Ecuación para calcular EAF

$$EAF = EAF_1 * EAF_2 * EAF_3 * \dots * EAF_n \quad (1)$$

Donde cada EAF_n es un subfactor que se determina en función de las condiciones específicas del proyecto. En este caso, se utilizó un valor de EAF de 1.7, el cual refleja ciertas características del equipo y el entorno que hacen que el esfuerzo necesario sea más alto que el valor promedio.

4.3.1.3. Cálculo del esfuerzo (E)

La fórmula general para calcular el esfuerzo es la siguiente:

(3)Ecuación para medir el Esfuerzo (E)

$$E = a * (KLOC)^b * EAF \quad (2)$$

Sustituyendo los valores correspondientes:

$$E = 2.4 * (8.786)^{1.05} * 1.7$$

$$E = 39.96 \frac{personas}{mes}$$

4.3.1.4. Cálculo del tiempo de desarrollo (T)

El tiempo de desarrollo se estima utilizando la fórmula:

(4)Ecuación para encontrar el Tiempo de Desarrollo (T)

$$T = c * (E)^d \quad (3)$$

Sustituyendo los valores correspondientes:

$$T = 2.5 * (39.96)^{0.38}$$

$$T = 10.15\ meses$$

4.3.1.5. Cálculo del número de personas (P)

El número de personas necesarias para realizar el proyecto en el tiempo estimado se calcula como:

(5)Ecuación para calcular el número de Personas (p)

$$P = \frac{E}{T} \quad (4)$$

Sustituyendo los valores:

$$P = \frac{39.96}{10.15}$$

$$P = 3.94\ personas$$

Esto significa que aproximadamente 4 personas se necesitarían para completar el proyecto en el tiempo estimado de 10.15 meses.

4.3.1.6. Cálculo del costo total del proyecto (CTP)

El costo total del proyecto se estima en función del salario mensual de cada persona, el número de personas necesarias y el tiempo de desarrollo. La fórmula para el cálculo del costo es:

(6)Formula para calculo total del proyecto

$$CTP = Salario * Nro\ de\ personas * Tiempo \quad (5)$$

Sustituyendo los valores:

$$CTP = 2500 * 4 * 10$$

$$CTP = 100{,}000\ Bs$$

Este valor representa el costo total del proyecto, cubriendo el esfuerzo humano necesario para completar el desarrollo.

4.3.1.7. Costos Adicionales

- **Hosting**: $49 (aproximadamente Bs. 340).
- **Registro de dominio**: $59 (aproximadamente Bs. 410).

Justificación de Costos Adicionales: El costo de hosting y registro de dominio se obtuvo de proveedores locales y se ajusta al tamaño y las necesidades del proyecto. Estos costos cubren servicios básicos de alojamiento y la compra de un dominio (.com.bo) para asegurar una presencia online profesional.

Tabla 24

Descripción de Costos Adicionales

Costo Adicional	Monto en Dólares (USD)	Monto en Bolivianos (Bs.)
Hosting	$49	Bs. 341.04
Registro de dominio	$59	Bs. 410.54
Total de costos Adicionales	$108	Bs. 751.58

Nota. En esta tabla se muestra el detalle de los costos adicionales como el servidor web y el dominio.

4.3.1.8. Costo total final

Sumamos los costos del hosting y el dominio al costo total del proyecto:

$$Costo\ total\ final = 100{,}00 + 751.58$$

$$Costo\ total\ final = 100{,}751.58\ Bs$$

4.3.2. Conclusión de la Estimación de Costos

El costo total del proyecto con los costos adicionales de hosting y dominio anuales sería de 100,751.58 Bs. Esto incluye el esfuerzo humano necesario para el desarrollo del proyecto, además de los costos de infraestructura como el hosting y el dominio.

4.4. SEGURIDAD

El sistema web implementado para la institución educativa utiliza diversas medidas de seguridad en el sistema, la base de datos y el código para proteger la información y asegurar que solo los usuarios autorizados puedan acceder a funcionalidades críticas.

4.4.1. Seguridad en el Sistema y Base de Datos

4.4.1.1. Autenticación y Autorización

La autenticación de usuarios se realiza mediante un sistema de CAPTCHA en el proceso de inicio de sesión y registro, lo que ayuda a evitar accesos automatizados. Los usuarios registrados en la base de datos son dados de alta con un rol específico que define sus permisos en el sistema. Para la seguridad de las contraseñas, se utiliza el algoritmo de encriptación bcrypt, que asegura la confidencialidad de los datos sensibles.

Los usuarios solo obtienen acceso completo al sistema cuando el administrador les asigna los permisos de acuerdo con sus roles (ej. director, secretario, docente, estudiante). Este proceso permite que los usuarios recién registrados tengan acceso inicial restringido al dashboard (Panel Informativo), donde el administrador valida su acceso y asigna permisos adicionales según el rol especificado.

4.4.1.2. Encriptación de Datos Sensibles

Los datos sensibles, especialmente las contraseñas de los usuarios, se almacenan encriptados en la base de datos utilizando bcrypt. Este método de encriptación protege la información frente a accesos no autorizados y ataques de fuerza bruta.

4.4.1.3. Control de Accesos y Permisos

La gestión de permisos está integrada en el sistema mediante roles que controlan el acceso a diferentes módulos según el cargo del usuario. Por ejemplo, el administrador tiene acceso a todos los módulos, mientras que un estudiante tiene permisos limitados solo para módulos específicos de interacción. Los roles se administran de acuerdo con el cargo del usuario, lo cual garantiza una organización controlada y evita accesos innecesarios.

4.4.1.4. Registro de Actividades

El sistema monitorea y registra las actividades clave de los usuarios en el módulo de registro de actividades. Este registro detalla acciones como la creación, modificación o eliminación de publicaciones, mostrando el nombre del usuario, una descripción de la acción realizada, y la fecha y hora. Este monitoreo facilita la detección de actividades sospechosas y permite rastrear cualquier manipulación de datos en el sistema.

4.4.1.5. Notificaciones de Acceso Restringido

Los usuarios recién registrados tienen un acceso restringido al sistema hasta que el administrador valide su rol y les asigne los permisos correspondientes. En caso de que un usuario intente acceder a un módulo sin los permisos necesarios, el sistema muestra un mensaje de error. Los intentos de acceso a módulos administrativos están restringidos para usuarios sin privilegios, y los roles definidos en el sistema determinan los permisos asignados.

4.4.2. Seguridad en el Código

4.4.2.1. Validaciones del Código

Para evitar errores y accesos no autorizados, se implementaron múltiples validaciones en los controladores, que son esenciales para procesar y recibir datos sin comprometer la seguridad del sistema. Estas validaciones aseguran que la información enviada y recibida en el sistema cumple con los parámetros de seguridad y no genera fugas de datos.

4.4.2.2. Protección Contra Ataques Comunes

El sistema incluye protecciones específicas contra ataques comunes:

- **Inyecciones SQL**: La utilización de Eloquent ORM en Laravel asegura que las consultas de base de datos sean seguras y protegidas contra inyecciones SQL.

- **Cross-Site Scripting (XSS)**: Para prevenir ataques XSS, el sistema emplea la función “{{ }}” en las vistas de Laravel, lo cual escapa cualquier entrada de usuario para que no ejecute scripts maliciosos.

- **CSRF (Cross-Site Request Forgery)**: Los formularios del sistema cuentan con tokens CSRF integrados para evitar solicitudes fraudulentas de otros sitios web.

- **Validaciones Personalizadas**: Se implementaron validaciones específicas en los campos de formularios, como requisitos de longitud y formato en contraseñas, para asegurar la integridad de los datos y evitar entradas no válidas.

4.4.2.3. Ejemplo de Validación y Protección

Se utiliza spatie/laravel-permission, un paquete que permite gestionar permisos y roles de manera segura. Solo los usuarios con los permisos asignados pueden acceder y gestionar módulos específicos. Además, se han configurado middlewares en el archivo web.php, asegurando que únicamente los usuarios autorizados puedan ingresar y navegar en el sistema. Si un usuario intenta iniciar sesión sin las credenciales adecuadas o accede a un módulo sin permisos, se bloquea su acceso y se muestra un mensaje de error.

4.4.2.4. Herramientas de Pruebas de Seguridad

Las herramientas de seguridad implementadas incluyen verificación reCapcha, middleware para filtrar accesos y sesiones de usuario, validaciones personalizadas en los controladores, y mecanismos de verificación de roles con spatie/laravel-permission. Estas herramientas garantizan que solo usuarios autorizados puedan acceder y manipular la información del sistema, manteniendo un entorno seguro.

4.5. PRUEBAS AL SOFTWARE

A continuación, se detalla el proceso de pruebas realizado en el sistema, basado en las características de calidad definidas por el modelo de McCall, particularmente en cuanto a funcionalidad, eficiencia y seguridad del sistema.

4.5.1. Pruebas de caja blanca

Las pruebas de caja blanca, también conocidas como pruebas estructurales, se llevaron a cabo para verificar la lógica interna del sistema y asegurar que los módulos críticos operaran de acuerdo a lo planeado. Estas pruebas se enfocaron principalmente en los módulos relacionados con la autenticación, gestión de usuarios, gestión de roles, y validación de contraseñas, con el objetivo de garantizar la fiabilidad y seguridad del sistema.

4.5.1.1. Pasos para Documentación:

Identificación de Módulos Críticos Evaluados:

- **Autenticación**: Verificación del acceso según roles definidos, garantizando la seguridad del sistema.
- **Gestión de Usuarios**: Control de creación, edición y eliminación de usuarios, asegurando la usabilidad y correcta administración de acceso.
- **Gestión de Roles**: Asignación de permisos a usuarios según sus roles, manteniendo la eficiencia en la administración.
- **Validación de Contraseñas**: Revisión de políticas de seguridad en contraseñas, asegurando que el sistema cumpla con las normativas de seguridad.

Resultados Obtenidos:

- **Error Detectado**: Se identificó un error que permitía a usuarios con permisos limitados acceder al módulo de gestión de usuarios, que debería estar restringido exclusivamente a los administradores. Este error afectaba la seguridad y fiabilidad del sistema.

- **Solución Implementada**: Se añadió la función `@can` en el código para limitar el acceso de dicho módulo exclusivamente a los administradores, reforzando la seguridad del sistema y garantizando que solo los roles apropiados gestionen los usuarios.

4.5.2. Pruebas de caja negra

Las pruebas de caja negra, también conocidas como pruebas funcionales, fueron realizadas para evaluar las funcionalidades del sistema sin considerar la estructura interna del código. Estas pruebas se realizaron desde la perspectiva del usuario final, tomando en cuenta diferentes roles (director, secretarios, docentes y estudiantes), y se evaluaron aspectos de usabilidad, accesibilidad, y seguridad de acuerdo con las restricciones de los roles.

4.5.2.1. Pasos para Documentación:

Funcionalidades Evaluadas:

- **Registro e Inicio de Sesión**: Verificación del acceso seguro de usuarios, implementando mecanismos de autenticación como CAPTCHA y contraseñas encriptadas para garantizar la **seguridad**.

- **Acceso Restringido según Roles**: Evaluación de la efectividad en la limitación de accesos a módulos administrativos, garantizando que solo los usuarios con roles específicos puedan acceder a funciones sensibles.

- **Creación de Publicaciones y Gestión de Contenido**: Pruebas de creación, edición y eliminación de publicaciones para usuarios con permisos apropiados, garantizando la usabilidad en el manejo de contenidos.

Feedback de Usuarios:

- Los usuarios confirmaron que el sistema restringe correctamente el acceso a módulos administrativos según los roles definidos. Por ejemplo, los secretarios no tienen acceso a la gestión de roles, una función exclusiva para los administradores. Sin embargo, tienen la capacidad de gestionar usuarios nuevos y asignarles roles según sus cargos, lo cual refuerza la eficiencia del sistema.

- Los estudiantes tienen acceso solo a los módulos específicos, y cualquier intento de acceder a módulos administrativos muestra un mensaje de error indicando la falta de permisos. Los usuarios consideraron que estas restricciones facilitaban la organización y el flujo de trabajo, mejorando la usabilidad del sistema.

4.5.3. Pruebas de estrés

Las pruebas de estrés se realizaron para medir el rendimiento del sistema bajo condiciones de carga extrema. Para estas pruebas, se utilizó la herramienta Apache JMeter, simulando un entorno con un gran número de usuarios concurrentes para analizar el tiempo de respuesta y la estabilidad del sistema.

4.5.3.1. Pasos para Documentación:

Condiciones de Prueba:

- Se simularon 20 usuarios concurrentes realizando solicitudes a módulos principales del sistema, como el inicio de sesión y la carga de publicaciones.

Resultados de Rendimiento:

- **Tiempos de Respuesta**: La prueba con 20 muestras arrojó un tiempo promedio de respuesta de 790 ms, con una desviación estándar de 213 ms, y el último tiempo de respuesta registrado fue de 592 ms.

- **Estabilidad del Sistema**: Bajo esta carga, el sistema mantuvo su estabilidad sin presentar errores críticos, demostrando que puede manejar un número moderado de usuarios concurrentes sin comprometer el desempeño.

Estas pruebas de estrés muestran que el sistema es robusto bajo condiciones de carga típica y permite planificar futuras mejoras en rendimiento para soportar una mayor cantidad de usuarios.

4.5.4. Pruebas de Accesibilidad

Las pruebas de accesibilidad del sistema se realizaron utilizando la herramienta WAVE (Web Accessibility Evaluation Tool), diseñada para analizar y garantizar la conformidad con las Pautas de Accesibilidad al Contenido Web (WCAG) 2.1. Estas pruebas son esenciales para identificar barreras que podrían limitar el acceso de personas con discapacidades al sistema y para implementar mejoras que aseguren una experiencia inclusiva para todos los usuarios. Los datos obtenidos permitieron clasificar los errores y sugerir acciones correctivas para mitigar los problemas detectados (WebAIM, 2023).

Tabla 25

Resultado de las Pruebas de Accesibilidad

Categoría Evaluada	Conjunto 1	Conjunto 2	Conjunto 3	Conjunto 4	Conjunto 5
Errores	3	5	10	3	18
Errores de Contraste	4	15	21	17	15
Alertas	15	4	3	3	8
Características	21	22	17	17	22
Elementos Estructurales	9	4	4	4	9
ARIA	0	0	0	0	0

Nota. En esta tabla se muestra los resultados de la prueba de accesibilidad al sistema

4.5.4.1. Análisis de Resultados

Errores de Accesibilidad

Los errores reflejan barreras críticas que dificultan la interacción de los usuarios con el sistema. El Conjunto 3, con 10 errores, presentó la mayor cantidad de incidencias, relacionadas principalmente con etiquetas ausentes o incorrectas en elementos interactivos. Estos problemas se corrigieron añadiendo descripciones claras y semánticas adecuadas para garantizar la accesibilidad mediante lectores de pantalla.

Errores de Contraste

El análisis reveló que el Conjunto 4, con 17 errores, requería ajustes significativos en el contraste entre el texto y el fondo. Esto afectaba especialmente a usuarios con discapacidades visuales. Para abordar esta problemática, se implementaron nuevas combinaciones de colores que cumplen con la relación mínima de contraste establecida en las WCAG 2.1 (4.5:1).

Alertas

Aunque las alertas no representan errores críticos, indican áreas del sistema que podrían mejorarse. El Conjunto 1, con 15 alertas, destacó por el uso de elementos redundantes y etiquetas mal definidas. Estos aspectos fueron optimizados eliminando redundancias y ajustando las configuraciones de los elementos afectados.

Características Evaluadas

Las características del sistema, como textos alternativos y formularios accesibles, mostraron un buen desempeño general, con valores entre 17 y 22 características correctamente implementadas. Sin embargo, el Conjunto 3 presentó ciertas inconsistencias, que fueron corregidas mediante una revisión detallada de la implementación semántica.

Elementos Estructurales

La jerarquía de encabezados y la estructura de navegación mostraron variabilidad en su calidad. El Conjunto 2, con solo 4 elementos bien implementados, requirió ajustes significativos para reorganizar los niveles de encabezado (<h1> a <h6>) y mejorar la experiencia de navegación para tecnologías de asistencia.

Implementación de ARIA

No se encontraron problemas en la implementación de ARIA, lo que confirma que los elementos dinámicos del sistema son totalmente compatibles con las tecnologías de apoyo**.**

4.5.4.2. Acciones Correctivas Implementadas

- **Corrección de Errores de Contraste**: Se ajustaron los colores de texto y fondo en todas las áreas identificadas, asegurando que cumplieran con las pautas WCAG 2.1 AA.

- **Optimización de Elementos Semánticos**: Las etiquetas ausentes o mal configuradas se corrigieron para garantizar una experiencia accesible para usuarios de lectores de pantalla.

- **Revisión de la Jerarquía de Encabezados**: Se reorganizó la estructura de los encabezados para reflejar una jerarquía lógica y mejorar la navegación para todos los usuarios.

- **Refinamiento de Características**: Se aseguraron formularios accesibles y botones interactivos con atributos ARIA adecuados.

4.5.4.3. Conclusión

Las pruebas de accesibilidad permitieron identificar errores críticos y áreas de mejora en el sistema, principalmente relacionados con el contraste y la estructura semántica. Las acciones correctivas implementadas no solo garantizaron el cumplimiento con los estándares internacionales, sino que también mejoraron significativamente la experiencia del usuario. Estas pruebas refuerzan el compromiso del proyecto con la inclusión y la accesibilidad universal, asegurando que el sistema sea accesible para todos los usuarios, independientemente de sus habilidades.

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

INGENIERÍA
DE SISTEMAS
UNIVERSIDAD PUBLICA DE EL ALTO

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

Diagnóstico de seguridad en aplicaciones móviles: Se evaluó la seguridad de diversas aplicaciones móviles, analizando permisos y comportamientos. Este proceso permitió identificar aplicaciones con riesgos y generar recomendaciones para un uso más seguro (ver Anexo 5).

Pruebas de seguridad con ingeniería inversa: Utilizando herramientas como APKTool, se realizaron pruebas exhaustivas en las aplicaciones, detectando vulnerabilidades específicas y verificando el cumplimiento de los estándares de seguridad establecidos (ver Anexo 6).

Identificación de permisos sospechosos: Se identificaron aplicaciones que solicitaban permisos peligrosos que podrían comprometer la seguridad del dispositivo. El sistema alerta sobre estos riesgos y promueve una mayor conciencia en los usuarios sobre la importancia de proteger su privacidad (ver Anexo 7).

Verificación de fuentes confiables para descargas: Se constató que descargar aplicaciones solo desde fuentes seguras, como Google Play, ayuda a disminuir los riesgos de malware y proteger los datos del usuario. El sistema orienta a los usuarios sobre cómo elegir fuentes confiables para las descargas (ver Anexo 8).

Diseño de un sistema web didáctico sobre seguridad: Se creó una plataforma educativa que incluye módulos sobre las mejores prácticas de seguridad. Esta plataforma proporciona tutoriales y recursos para ayudar a los usuarios a proteger tanto sus dispositivos como sus datos personales (ver Anexo 9).

5.2. RECOMENDACIONES

Sistema de Registro de Nuevos Usuarios

Se recomienda establecer un procedimiento de registro claro para los nuevos usuarios, que garantice su correcta identificación y acceso al sistema. En lugar de permitir cualquier tipo de código, el sistema debe exigir que los nuevos usuarios utilicen un código único compuesto por su primer nombre y número de cédula de identidad (por ejemplo: Juan_12345678).

- Por qué: Este enfoque asegura que cada usuario sea identificado correctamente desde el momento de su alta y permite una gestión más eficiente y precisa del sistema. Además, contribuye a la seguridad al vincular el código directamente con los datos personales del usuario.

Responsabilidad en la Gestión de Roles y Usuarios

Los usuarios con permisos para gestionar roles y usuarios deben asumir una gran responsabilidad al dar de alta a nuevos usuarios y asignarles roles dentro del sistema.

- Recomendación: Se debe establecer una política clara sobre cómo asignar roles a los usuarios, asegurando que los roles se otorguen de acuerdo con el cargo o nivel de acceso del individuo. Esta asignación debe basarse en criterios específicos, de modo que cada usuario tenga acceso solo a la información y funcionalidades necesarias para su trabajo.

- Por qué: Esto garantizará una gestión ordenada, segura y eficiente de los usuarios, evitando errores que puedan poner en riesgo la integridad del sistema.

Enfoque en la Información y Seguridad

El sistema tiene como propósito principal informar a los usuarios de la unidad educativa sobre temas relacionados con la seguridad en aplicaciones móviles y las novedades dentro de la unidad educativa.

- Recomendación: Desarrollar y mantener una sección dentro del sistema dedicada a noticias de seguridad y actualizaciones relacionadas con la unidad educativa. Las publicaciones deben ser claras, relevantes y fáciles de entender para todos los usuarios.

- Por qué: Esto ayudará a los usuarios a mantenerse informados sobre las mejores prácticas de seguridad, especialmente en un entorno digital, y facilitará una comunicación efectiva dentro de la unidad educativa.

Publicaciones Relevantes y Educativas

Para los usuarios con permisos para crear publicaciones dentro del sistema, es fundamental que las publicaciones sean relevantes, educativas e informativas.

- Recomendación: Las publicaciones deben estar orientadas a proporcionar contenido útil y formativo sobre temas de interés para la comunidad educativa, como actualizaciones de seguridad, nuevas funcionalidades del sistema o noticias importantes sobre la unidad educativa.

- Por qué: Esto asegura que el sistema no solo cumpla con su función de informar, sino que también proporcione valor educativo a los usuarios, aumentando la participación y el compromiso de la comunidad.

Evolución y Futuro del Sistema

Para garantizar que el sistema evolucione de manera adecuada y continúe siendo útil a medida que las necesidades de los usuarios cambian, se recomienda considerar el uso de herramientas similares y modernas para el desarrollo de futuras actualizaciones o proyectos relacionados.

- Recomendación: Evaluar constantemente las tecnologías emergentes que puedan mejorar el rendimiento y la escalabilidad del sistema, así como nuevas herramientas que faciliten la integración de nuevas funcionalidades o servicios.

- Por qué: El uso de tecnologías actualizadas y herramientas avanzadas permitirá que el sistema se mantenga competitivo, eficiente y capaz de adaptarse a las futuras necesidades de los usuarios.

REFERENCIAS BIBLIOGRÁFICAS

A., Pérez; M., Gómez. (2023). Implementación de seguridad y control de acceso en sistemas web educativos. Revista de Ingeniería de Software, 27(1), 95-110.

A., Ruiz. (2019). Web systems and educational transformation. Journal of Educational Technology, 5(3), 120-134.

Alcalá, U. d. (2002). Fundamentos y uso del Lenguaje de Modelado Unificado (UML). Editorial Universitaria de Alcalá.

Arce, P., & Molina, F. (2022). Evaluación de riesgos de ciberseguridad en aplicaciones financieras móviles en Bolivia. Revista Boliviana de Ciberseguridad, 3(4), 55-70.

AvanceHost. (2024). Servicios de hosting y seguridad. Obtenido de https://www.avancehost.com

D., García; S., López. (2023). Acceso restringido en sistemas educativos para la mejora de la seguridad y organización. Journal of Educational Technology, 2(120-132), 19.

Adermann, N., & Boggiano, J. (2022). *A Dependency Manager for PHP*. https://getcomposer.org

Arce, P., & Molina, F. (2022). Evaluación de riesgos de ciberseguridad en aplicaciones financieras móviles en Bolivia. *Revista Boliviana de Ciberseguridad, 3*(4), 55-70.

AvanceHost. (2024). *Servicios de hosting y seguridad*. https://www.avancehost.com

Beal, V. (2020). The importance of web security in information management. *Information Systems Journal, 17*(1), 80-92.

Berners-Lee, T. (1996). World Wide Web: Past, present, and future. *Computer, 29*(10), 69-77.

Berners-Lee, T. (2023). *HTML Living Standard*. https://www.w3.org/TR/html/

Bos, B. (2023). Cascading Style Sheets. https://www.w3.org/Style/CSS/

Dahl, R. (2009). *Node.js® is an open-source, cross-platform JavaScript runtime environment*. https://nodejs.org/

Elizabeth, S. (2022). Uso del celular en el aula y ciberbullying en Bolivia. *Journal of Social Studies, 17*(3), 120-135.

García, D., & López, S. (2023). Acceso restringido en sistemas educativos para la mejora de la seguridad y organización. *Journal of Educational Technology, 2*, 120-132.

Gutiérrez, L., & Fernández, T. (2020). Seguridad en aplicaciones móviles para el transporte urbano en La Paz. *Revista de Estudios Urbanos de Bolivia, 7*(3), 35-50.

He, W., & Freeman, L. A. (2020). Mobile application security: Practices and threats. *Journal of Information Privacy and Security, 16*(4), 235-245.

International Organization for Standardization (ISO) & International Electrotechnical Commission (IEC). (2018). *ISO/IEC 27000:2018 Information technology – Security techniques – Information security management systems – Overview and vocabulary*. ISO/IEC.

Lerdorf, R. (2023). *PHP: A popular general-purpose scripting language especially suited to web development*. https://www.php.net/

López, M., & Díaz, R. (2023). Evaluación de la estabilidad y carga en sistemas educativos digitales. *Journal of Software Engineering, 25*(3), 145-160.

Otwell, T. (2023). *The PHP Framework for Web Artisans*. https://laravel.com

Otto, M., & Thornton, J. (2010). *Build fast, responsive sites with Bootstrap*. https://getbootstrap.com

Pérez, A., & Gómez, M. (2023). Implementación de seguridad y control de acceso en sistemas web educativos. *Revista de Ingeniería de Software, 27*(1), 95-110.

Pérez, R., & Martínez, S. (2022). Cybersecurity challenges in educational settings. *Journal of Cybersecurity and Education, 10*(3), 112-130.

Quiroga, J. (2023). Evaluación de la seguridad de aplicaciones móviles educativas en La Paz. *Revista de Innovación Educativa, 10*(2), 65-80.

Rodríguez, J., & Sánchez, E. (2018). Technological risks in digital education environments. *Journal of Digital Education, 4*(2), 56-68.

Rojo Becerril, D. A. (2020). *Mitigación de riesgos de seguridad en aplicaciones móviles*. Universidad Autónoma de Madrid.

Ruiz, A. (2019). Web systems and educational transformation. *Journal of Educational Technology, 5*(3), 120-134.

Schwabe, D., & Rossi, G. (1995). The object-oriented hypermedia design model (OOHDM). In *Proceedings of the ACM Conference on Hypertext* (pp. 116-128). ACM.

Stair, R. M., & Reynolds, G. W. (2021). *Principles of Information Systems*. Cengage Learning.

Vasquez, C., & Mamani, H. (2021). Uso de aplicaciones móviles para el comercio electrónico en La Paz durante la pandemia. *Revista de Economía y Negocios de Bolivia, 9*(1), 78-92.

WebHosting.com.bo. (2024). *Servicios de hosting y almacenamiento*. https://www.webhosting.com.bo

World Wide Web Consortium (W3C). (2008). *Web Content Accessibility Guidelines (WCAG) 2.0*. https://www.w3.org/TR/WCAG20/

Universidad de Alcalá. (2002). *Fundamentos y uso del Lenguaje de Modelado Unificado (UML)*. Editorial Universitaria de Alcalá.

Android Developers. (2021). *A new standard for mobile app security*. https://android-developers.googleblog.com

APKTool. (2023). *APKTool - Análisis de aplicaciones móviles*. https://github.com/iBotPeaches/Apktool

Cámara Nacional de Comercio de Bolivia. (2022). *Informe sobre costos de desarrollo de software en Bolivia*. https://cncbolivia.org

Organisation for Economic Co-operation and Development (OECD). (2022). *Digital education outlook: Reimagining the future of education*. https://oecd.org

OWASP Foundation. (2023). *Mobile Application Security Verification Standard (MASVS)*. https://owasp.org

OWASP Foundation. (2023). *Mobile security testing guide*. https://owasp.org

Spatie. (2022). *Laravel-permission*. https://github.com/spatie/laravel-permission

Symantec. (2021). *Informe sobre seguridad en dispositivos móviles*. https://symantec.com

United Nations Educational, Scientific and Cultural Organization (UNESCO). (2021). *The role of digital platforms in modern education*. https://unesco.org

United Nations Children's Fund (UNICEF). (2020). *La seguridad digital en la infancia y adolescencia*. https://unicef.org

O'Brien, J. A., & Marakas, G. M. (2019). *Management information systems* (11th ed.). McGraw Hill.

Stair, R., & Reynolds, G. (2021). *Principles of information systems* (14th ed.). Cengage Learning.

BIBLIOGRAFÍA

W., He; L. A., Freeman. (2020). Mobile application security: Practices and threats. Journal of Information Privacy and Security, 16(4), 235-245.

Allen, A. (2024). *Battle ready Laravel*. Independently published.

Clarke, R. A., & Knake, R. K. (2019). *The fifth domain: Defending our country, our companies, and ourselves in the age of cyber threats*. Penguin Press.

Cerra, A. (2022). *The cybersecurity playbook*. Wiley.

Erickson, J. (2008). *Hacking: The art of exploitation* (2nd ed.). No Starch Press.

Finney, G. (2024). *Project zero trust*. Independently published.

Flanagan, D. (2020). *JavaScript: The definitive guide* (6th ed.). O'Reilly Media.

Hand, M. (2023). *Evading EDR: The definitive guide to defeating endpoint detection systems*. Independently published.

Hyppönen, M. (2024). *If it's smart, it's vulnerable*. Wiley.

Kelley, D., & Moyle, E. (2020). *Practical cybersecurity architecture*. Packt Publishing.

Mitnick, K. (2017). *The art of invisibility*. Little, Brown and Company.

Mitnick, K. (2012). *Ghost in the wires: My adventures as the world's most wanted hacker*. Little, Brown and Company.

McPeak, J., & Wilton, P. (2024). *Beginning JavaScript* (5th ed.). Wrox.

Menn, J. (2019). *Cult of the dead cow: How the original hacking supergroup might just save the world*. PublicAffairs.

Singh, S. (2000). *The code book: The science of secrecy from ancient Egypt to quantum cryptography*. Anchor.

Sikorski, M., & Honig, A. (2012). *Practical malware analysis*. No Starch Press.

Stauffer, M. (2021). *Laravel: Up & running* (3rd ed.). O'Reilly Media.

Tatroe, K., MacIntyre, P., & Lerdorf, R. (2020). *Programming PHP: Creating dynamic web pages*. O'Reilly Media.

Welling, L., & Thomson, L. (2016). *PHP and MySQL web development* (5th ed.). Addison-Wesley Professional.

Wilhoit, K., & Opacki, J. (2023). *Operationalizing threat intelligence*. Independently published.

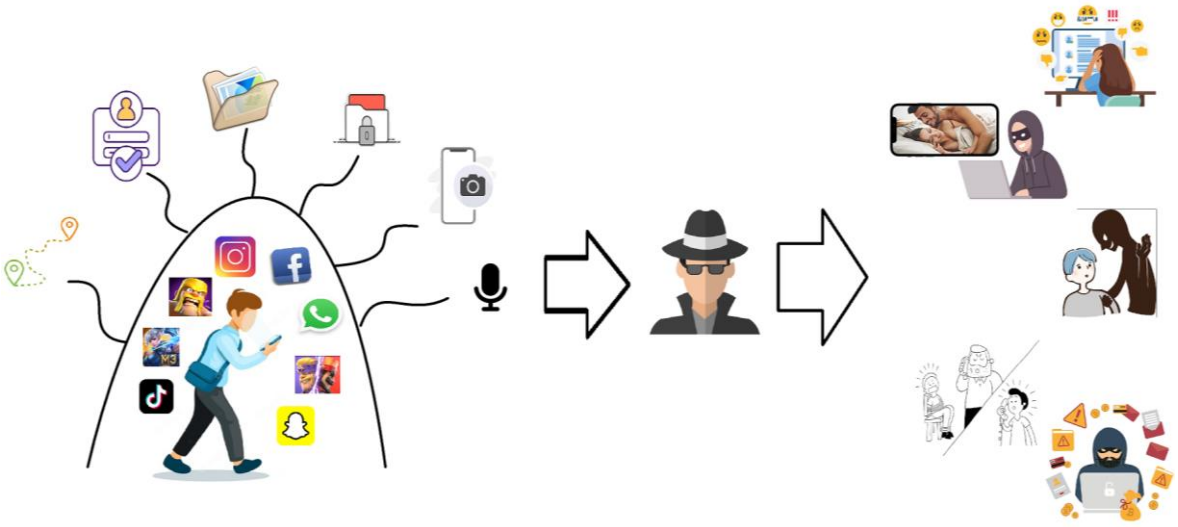
ANEXOS

INGENIERÍA
DE SISTEMAS
UNIVERSIDAD PUBLICA DE EL ALTO

ANEXOS

Anexo 1

Figura 73
Diagrama de la situación inicial de los estudiantes (antes de usar el sistema web)

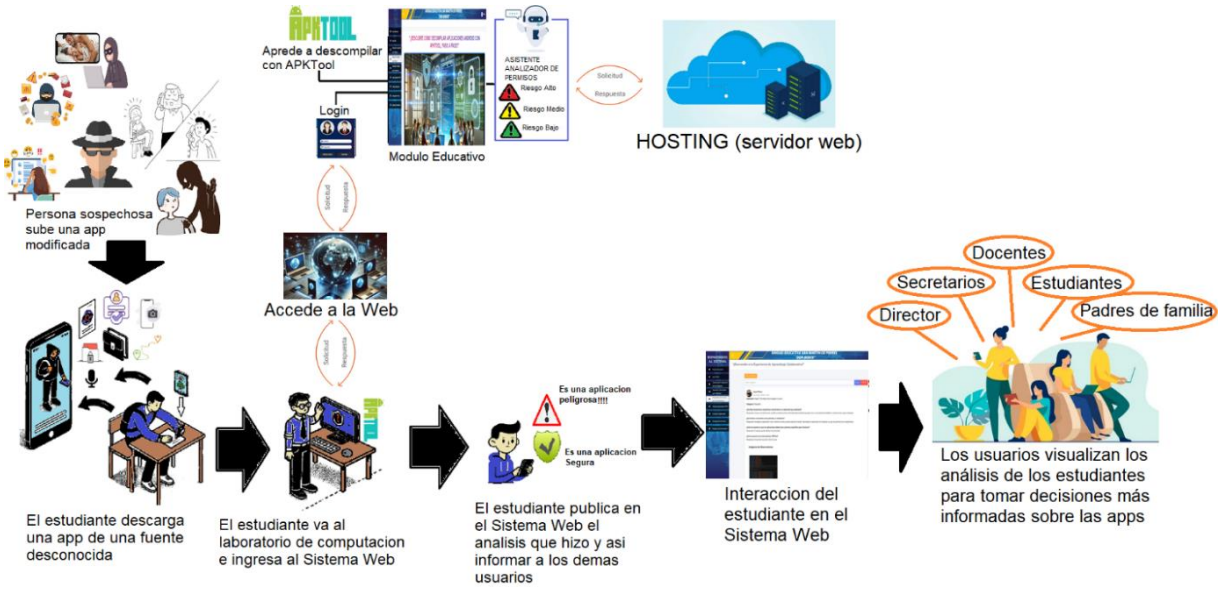


Nota. Esta figura muestra el diagrama que representa cómo los estudiantes, antes de usar el sistema web, eran vulnerables debido a la falta de conocimiento sobre la seguridad en aplicaciones móviles. *Los estudiantes no verificaban los permisos de las aplicaciones y no tomaban medidas para proteger su información personal.*

Anexo 2

Figura 74

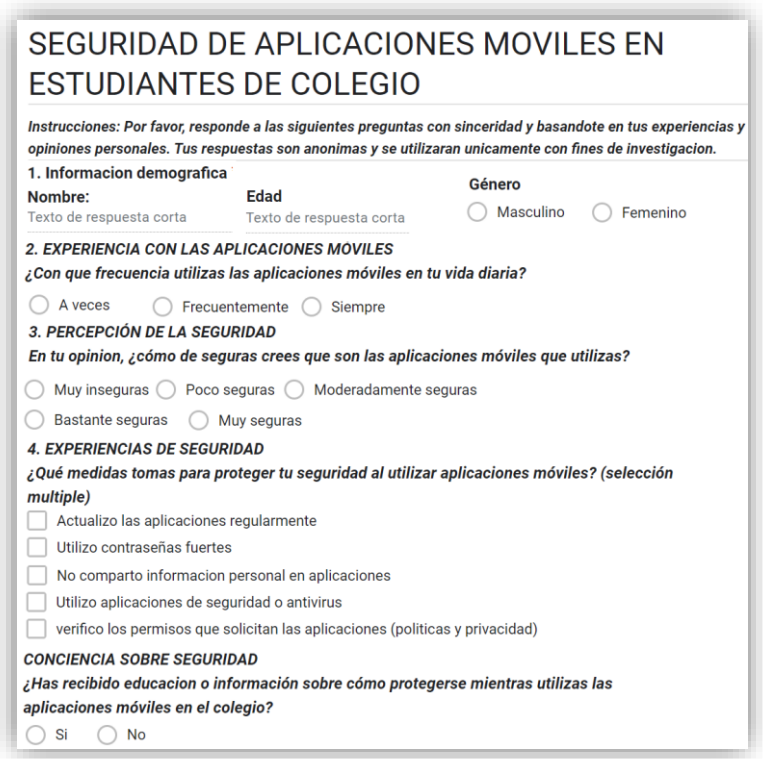
Diagrama posterior a la implementación del sistema web (después de capacitar a los estudiantes)



Nota. En esta figura se visualiza el cambio en el comportamiento de los estudiantes después de haber utilizado el sistema web. El diagrama refleja cómo los estudiantes ahora analizan los permisos de las aplicaciones, toman medidas de seguridad y protegen mejor su información personal al usar aplicaciones móviles.

Anexo 3

Figura 75
Modelo del Formulario en Google Forms



SEGURIDAD DE APLICACIONES MOVILES EN ESTUDIANTES DE COLEGIO

Instrucciones: Por favor, responde a las siguientes preguntas con sinceridad y basandote en tus experiencias y opiniones personales. Tus respuestas son anonimas y se utilizaran unicamente con fines de investigacion.

1. Informacion demografica

Nombre: Texto de respuesta corta

Edad Texto de respuesta corta

Género ○ Masculino ○ Femenino

2. EXPERIENCIA CON LAS APLICACIONES MOVILES

¿Con que frecuencia utilizas las aplicaciones móviles en tu vida diaria?

○ A veces ○ Frecuentemente ○ Siempre

3. PERCEPCIÓN DE LA SEGURIDAD

En tu opinion, ¿cómo de seguras crees que son las aplicaciones móviles que utilizas?

○ Muy inseguras ○ Poco seguras ○ Moderadamente seguras ○ Bastante seguras ○ Muy seguras

4. EXPERIENCIAS DE SEGURIDAD

¿Qué medidas tomas para proteger tu seguridad al utilizar aplicaciones móviles? (selección multiple)

☐ Actualizo las aplicaciones regularmente
☐ Utilizo contraseñas fuertes
☐ No comparto informacion personal en aplicaciones
☐ Utilizo aplicaciones de seguridad o antivirus
☐ verifico los permisos que solicitan las aplicaciones (politicas y privacidad)

CONCIENCIA SOBRE SEGURIDAD

¿Has recibido educacion o información sobre cómo protegerse mientras utilizas las aplicaciones móviles en el colegio?

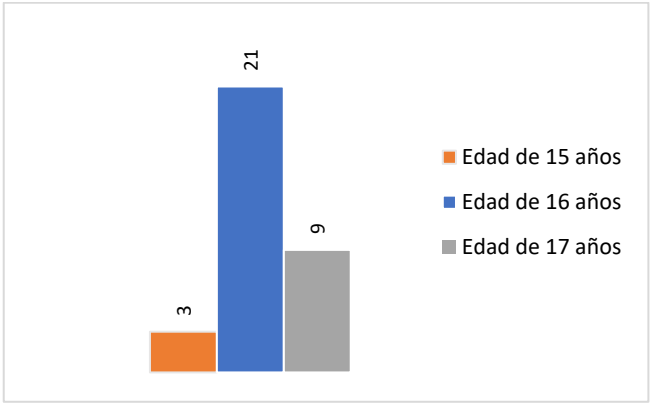
○ Si ○ No

Nota. Formulario utilizado para la encuesta a los estudiantes.

Anexo 4

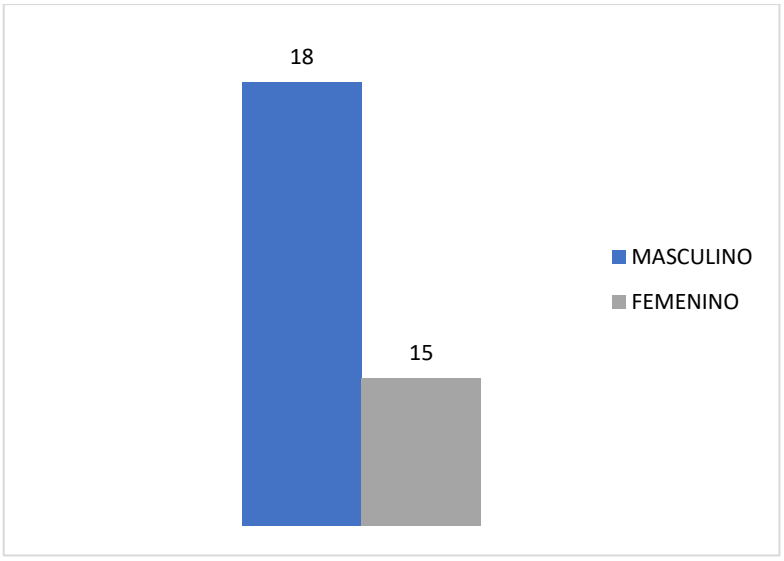
Figura 76

Gráfico: Descripción de la edad de los estudiantes



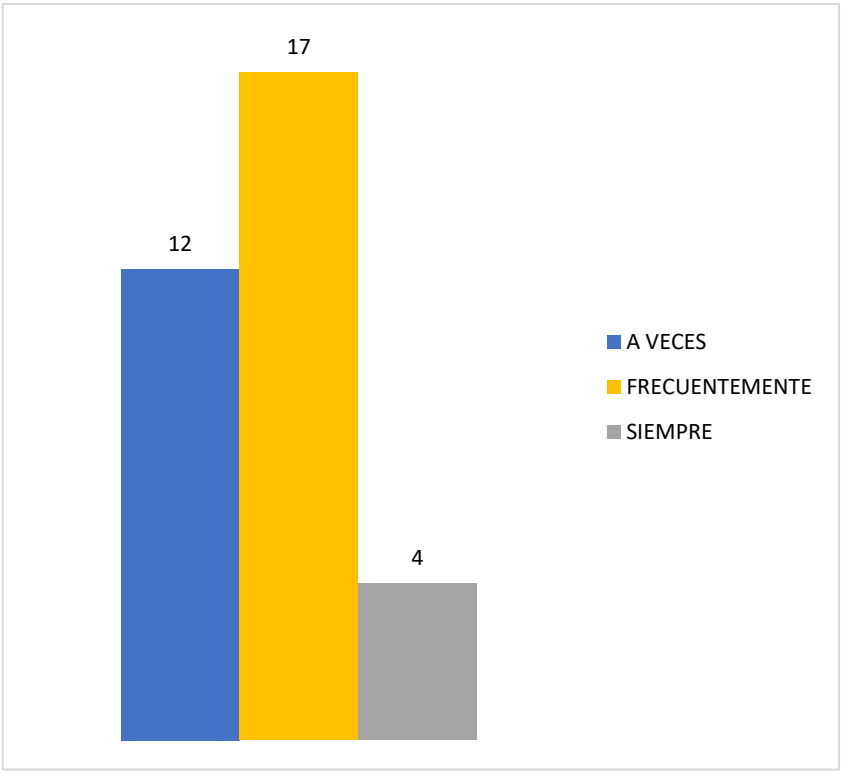
Nota. Este grafico muestra la distribucion de edades entre los estudiantes encuestados.

Figura 77
Gráfico: Distribución de genero de los estudiantes



Nota. Este grafico muestra la distribución del género entre los estudiantes que participaron en la encuesta.

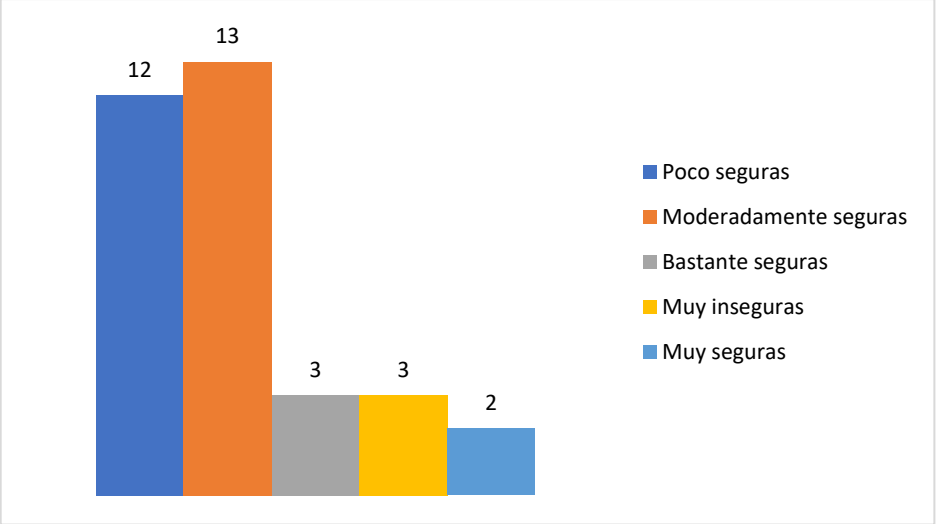
Figura 78
Gráfico: ¿Con que frecuencia utilizas las aplicaciones móviles en tu vida diaria?



Nota. Este grafico muestra el resultado de la frecuencia del uso de las aplicaciones móviles por parte de los estudiantes.

Figura 79

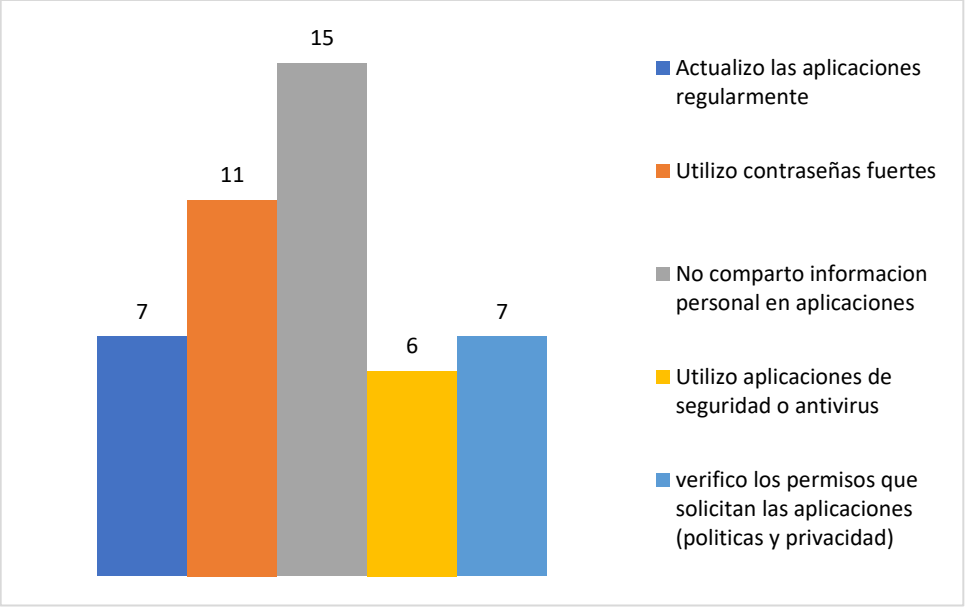
Gráfico: En tu opinión, ¿cómo de seguras crees que son las aplicaciones móviles que utilizas?



Nota. Este grafico muestra el resultado sobre como consideran la seguridad en las aplicaciones móviles por parte de los estudiantes.

Figura 80

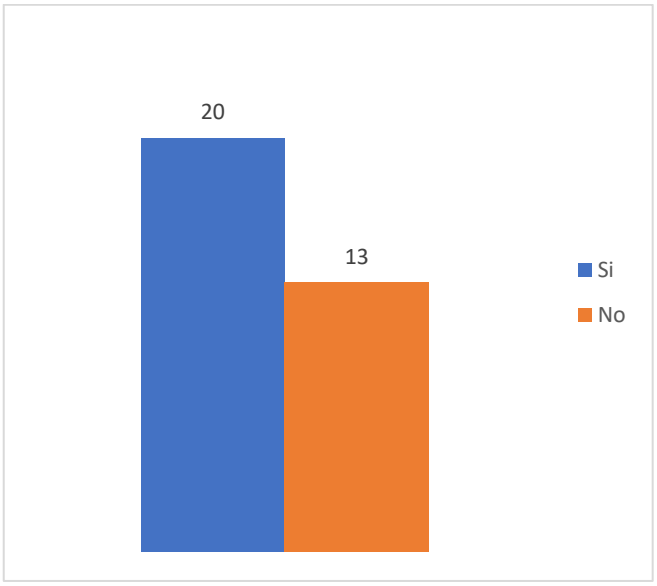
Gráfico: ¿Qué medidas tomas para proteger tu seguridad al utilizar aplicaciones móviles? (selección múltiple)



Nota. Este grafico muestra el resultado sobre como los estudiantes protegen su seguridad al usar las aplicaciones móviles

Figura 81

Gráfico: ¿Has recibido educación o información sobre cómo protegerse mientras utilizas las aplicaciones móviles en el colegio?



Nota. Este grafico muestra los resultados sobre si los estudiantes recibieron información de cómo protegerse sobre el uso de las aplicaciones móviles.

Anexo 5

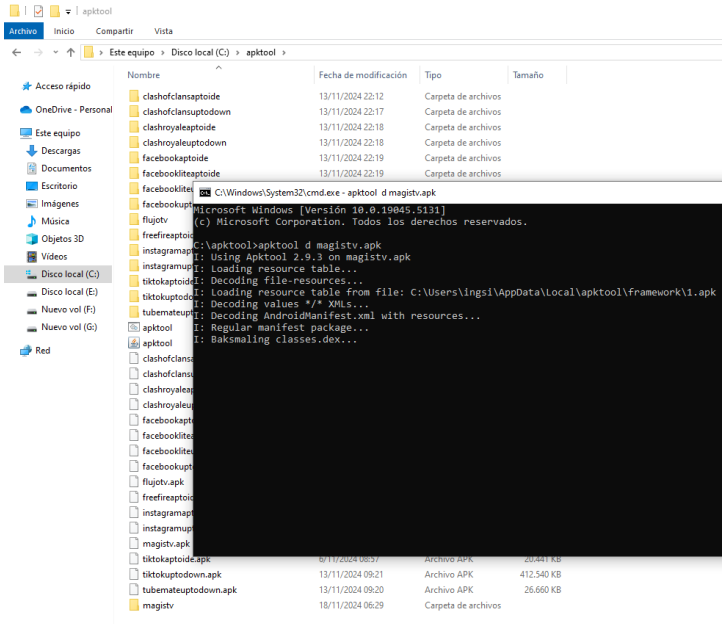
Tabla 26

Análisis de Aplicaciones Externas en Dispositivos Móviles

Implementación	Impacto/Beneficio
Se analizaron aplicaciones móviles de fuentes externas donde aparentemente son gratuitos y están fuera de la Play Store	Se verificaron dispositivos móviles utilizando aplicaciones de fuentes externas, donde se observó una manipulación del sistema operativo que generaba notificaciones inconsistentes e incoherentes con el funcionamiento esperado del dispositivo.

Anexo 6

Figura 82
Decompilación de aplicaciones móviles Android con APKTool

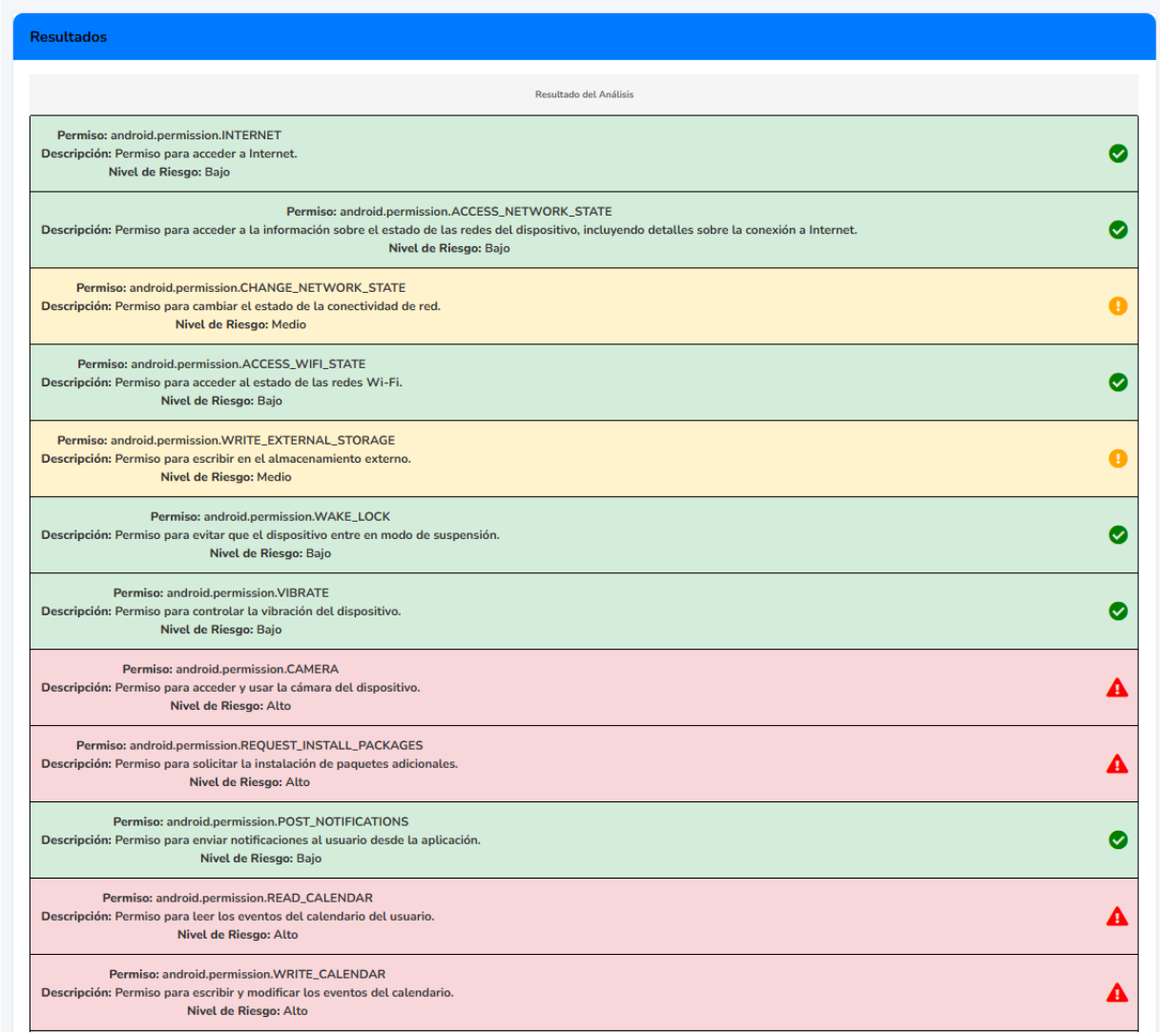


Anexo 7

Figura 83
Estudiantes de la Institución realizando el análisis de aplicaciones Android.

Figura 84

Resultados descriptivos de los permisos encontrados en las aplicaciones midiéndolos según su riesgo



Anexo 8

Tabla 27

Educación a los estudiantes sobre la Seguridad en la Descarga de Aplicaciones

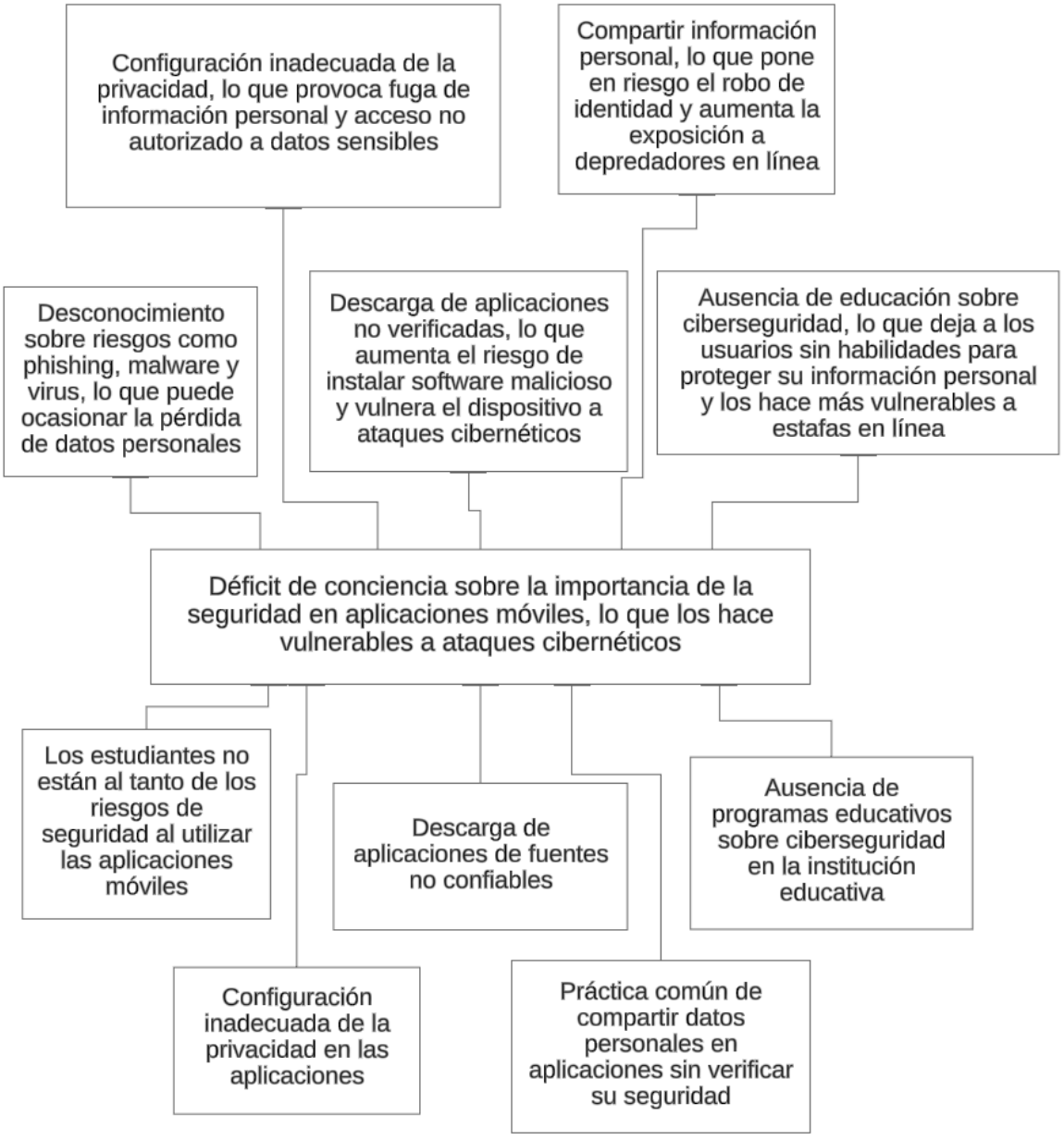
Implementación	Impacto/Beneficio
Se implementaron controles dentro del sistema para educar al usuario sobre la importancia de descargar solo desde fuentes oficiales, brindando recomendaciones y tutoriales para verificar aplicaciones antes de descargarlas.	Esto redujo los riesgos de malware, garantizando la integridad del dispositivo y protegiendo los datos del usuario.

Anexo 9

Figura 85

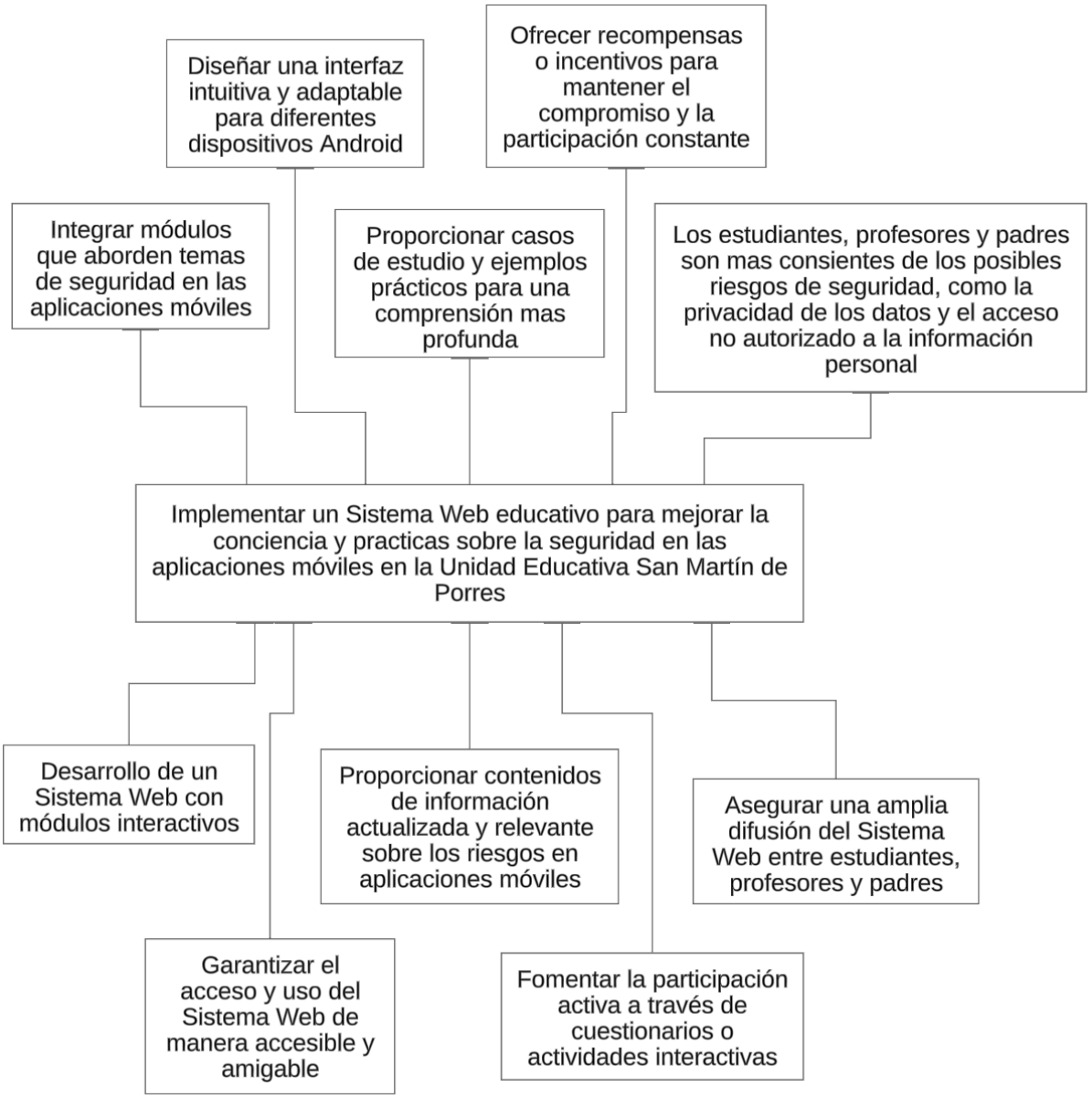
Contenido Educativo sobre la seguridad en las aplicaciones movibles en el Sistema Web

Figura 86
Árbol de Problemas



Nota. El árbol de problemas presenta las causas y efectos relacionados con la falta de conciencia sobre la seguridad en las aplicaciones móviles entre los estudiantes.

Figura 87
Árbol de Objetivos



Nota. El árbol de objetivos ilustra los objetivos a alcanzar para mejorar la seguridad en el uso de aplicaciones móviles por parte de los estudiantes. En este caso, el objetivo principal es aumentar el nivel de conciencia y conocimiento, lo que se logrará mediante actividades educativas y la implementación de prácticas de seguridad.

MANUAL DEL USUARIO

UNIDAD EDUCATIVA SAN MARTIN DE PORRES "DON BOSCO"

Correo electrónico: sanmartindeporresdonbosco@gmail.com

Sitio web: smpdonbosco.com.bo

Versión: 1.0

Let's Start!

INTRODUCCION

DESCRIPCIÓN DEL SISTEMA

Este sistema web permite a los usuarios analizar aplicaciones móviles para identificar permisos sospechosos y mejorar la seguridad de sus dispositivos. Su uso está orientado a estudiantes y personal educativo, y fue creado para proporcionar un entorno seguro que eduque a los usuarios en prácticas de seguridad informática.

PROPÓSITO DEL MANUAL

Este manual fue creado para ayudar a los usuarios a operar el sistema de forma correcta y segura. Proporciona instrucciones detalladas sobre cada módulo, facilitando la navegación y el aprovechamiento de todas las funcionalidades del sistema.

OBJETIVOS

OBJETIVO GENERAL

- Guiar a los usuarios en el uso eficiente del sistema, asegurando que comprendan cómo realizar análisis de seguridad en aplicaciones móviles.

OBJETIVOS ESPECÍFICOS

- Instruir al usuario en los procesos de registro e inicio de sesión.
- Explicar el funcionamiento de los módulos de publicación y análisis de aplicaciones.
- Proporcionar información sobre los permisos de usuario y el acceso restringido según los roles.

REQUISITOS DEL SISTEMA

Requisitos del Navegador

Google Chrome, Mozilla Firefox o navegadores modernos con soporte para HTML5 y JavaScript.

Requisitos de Conexión

- **Conexión a Internet** con una velocidad mínima de **2 Mbps** para asegurar una carga eficiente de contenido como texto, imágenes y algunos videos.
- **Para una mejor experiencia** con videos y contenido multimedia, se recomienda una conexión con **al menos 7.5 Mbps** para que todo cargue de manera rápida y sin interrupciones.
- **Para la mejor calidad** de navegación, especialmente si tu sistema tiene videos de alta calidad, se recomienda una conexión de **10 Mbps o más**.

Compatibilidad de Dispositivos

- **Compatible con ordenadores y laptops**: El sistema está completamente optimizado para usarse en escritorios y computadoras portátiles, ofreciendo la mejor experiencia de usuario.

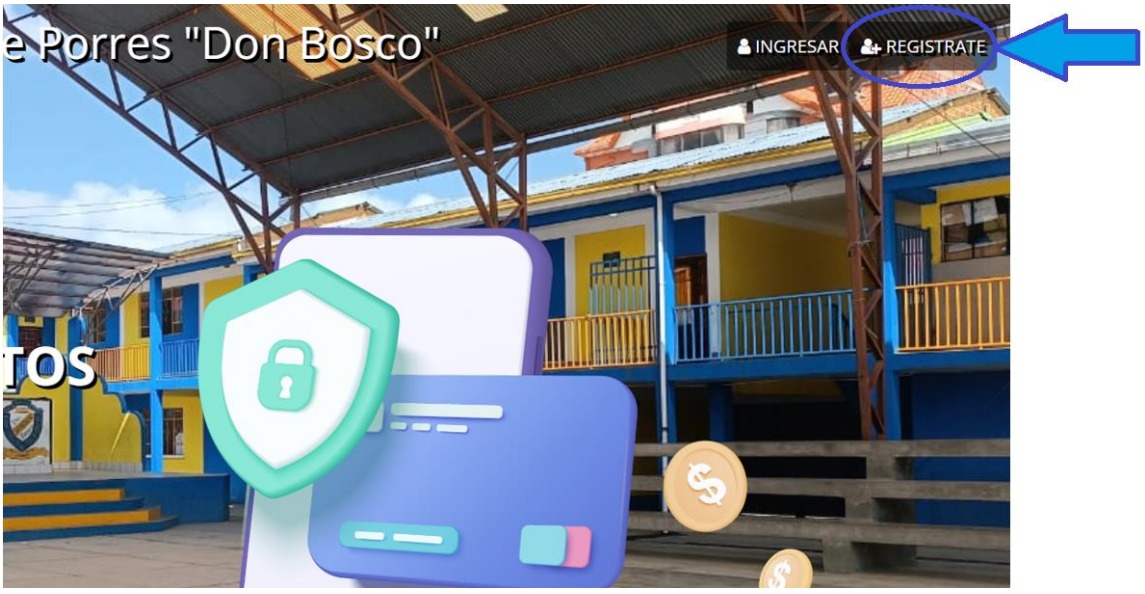
- **Compatible con tablets y dispositivos móviles:** El sistema también es accesible desde tablets y teléfonos móviles, aunque algunas funcionalidades pueden ser limitadas en pantallas más pequeñas.

GUÍA DE OPERACIÓN

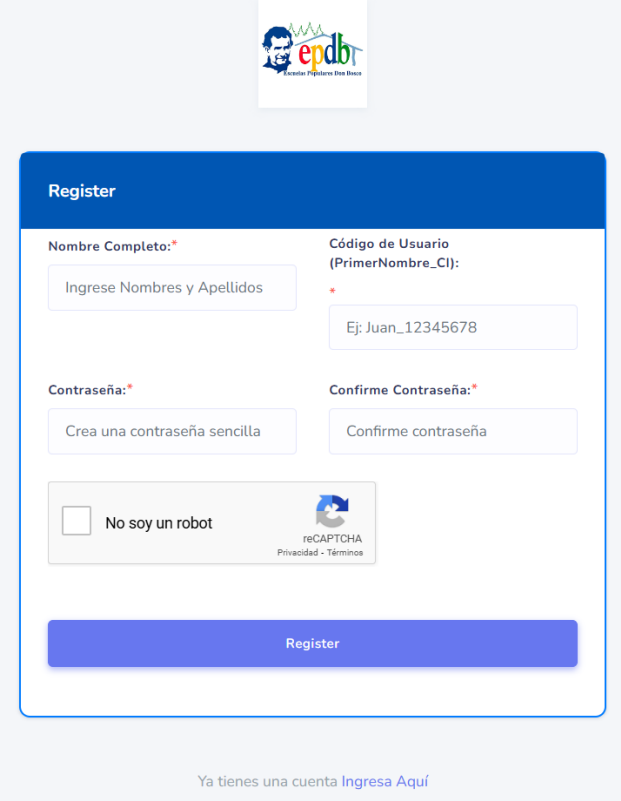
Registro de Usuario

Accede a la página principal del sistema. (https://smpdonbosco.com.bo)

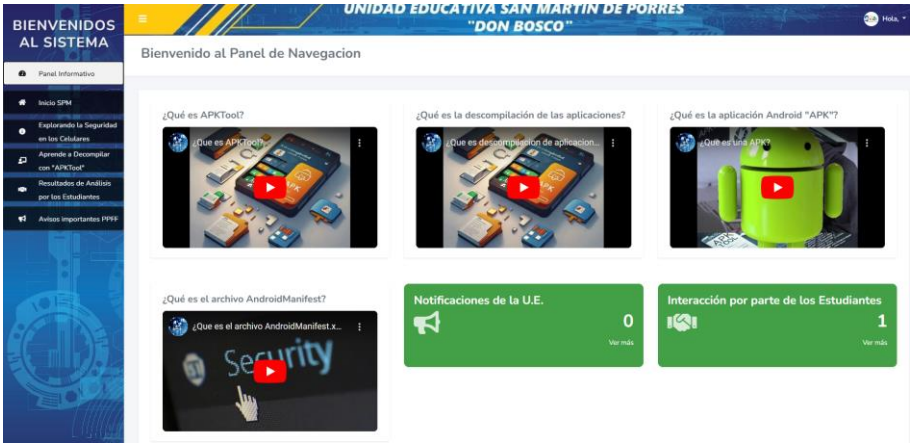
Haz clic en “REGISTRATE”.



Completa los campos requeridos: nombre, código de usuario único, contraseña, y verifica el CAPTCHA.



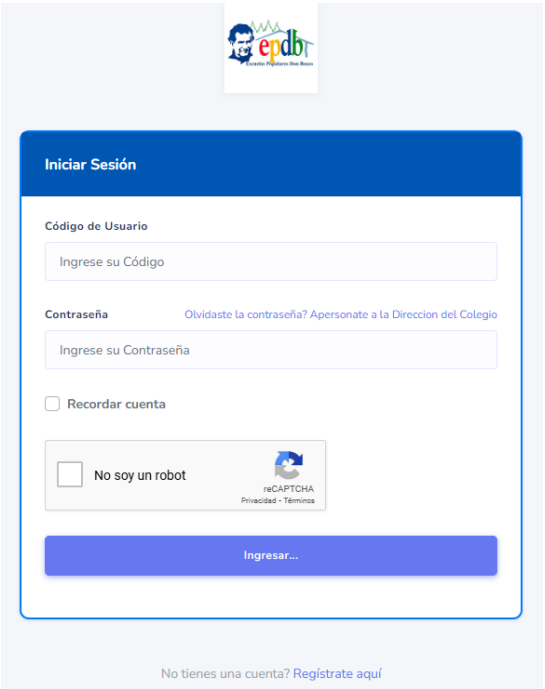
Haz clic en “Register”. Y serás direccionado al Panel Indormativo.



Nota. Si eres un Nuevo Usuario en el sistema, debes dirigirte a la persona que administra el sistema para que te pueda habilitar indicándole tu cargo (Director, Secretario, Docente, Estudiante, Padre de Familia) y así puedas navegar en los diferentes módulos.

Inicio de Sesión

Ingresa tu código de usuario y contraseña.



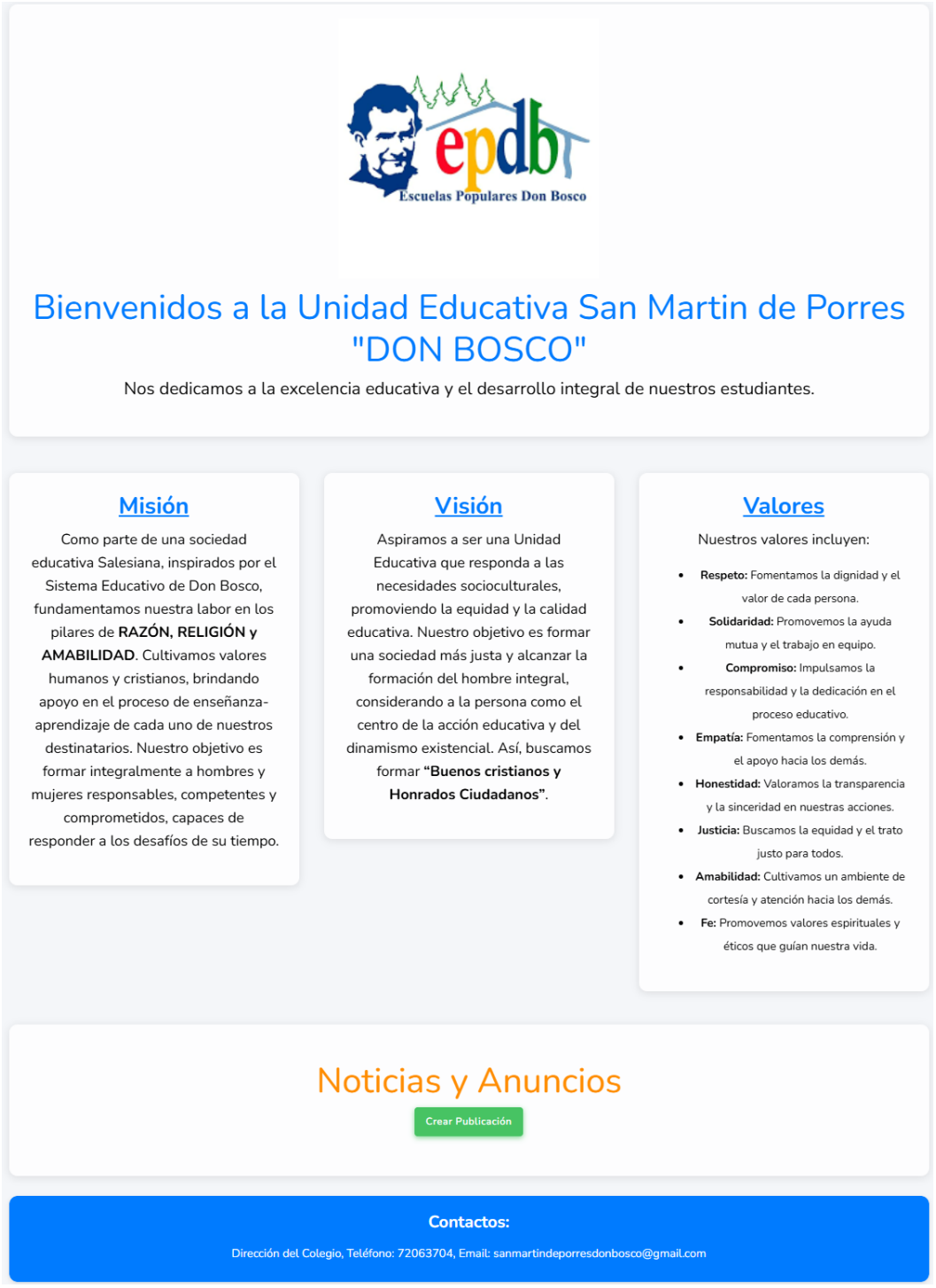
Completa el CAPTCHA y haz clic en “Ingresar…”.

Si los datos son correctos, accederás al Inicio SMP.

Exploración en el Panel Indormativo

El Panel Informativo permite acceder a diferentes módulos según el rol del usuario. El menú lateral muestra las opciones disponibles, como el Inicio SMP, Explorando la Seguridad en los Celulares, Aprende a Decompilar con “APKTool”, Asistente de Análisis, Resultado de Análisis por los Estudiantes, Avisos Importantes PPFF.

Inicio SMP



En este módulo se vera la información de la unidad educativa, y también veras las noticias y anuncios que publicaran el personal administrativo, solo los que tienen permisos podrán publicar contenido.

Explorando la Seguridad en los Celulares

"APLICACIONES MÓVILES SEGURAS: CONSEJOS PARA EVITAR RIESGOS EN DISPOSITIVOS ANDROID"

Descripción:
En este video exploramos la **importancia de la seguridad en las aplicaciones móviles Android** y cómo las **descargas desde fuentes no oficiales** pueden poner en riesgo tu **información personal**. Enfocado en los estudiantes del colegio San Martín de Porres, este video te ayudará a entender los **peligros** que pueden surgir al instalar **apps maliciosas** y cómo **proteger tu privacidad**.
Hablaremos sobre:
- **Los riesgos más comunes** al instalar **aplicaciones no verificadas**.
-Cómo los **estudiantes** pueden ser víctimas de **malware**, **spyware** y **robo de datos**.
- **Consejos útiles** para mantener tu **dispositivo seguro**.
-El uso de la herramienta **APKTool** para **analizar aplicaciones** y **detectar posibles amenazas**.

Infórmate con lo último en la seguridad en aplicaciones Android

Acumulación de aplicaciones en dispositivos móviles: Kaspersky advierte sobre los **riesgos de mantener numerosas aplicaciones en dispositivos Android** sin una gestión adecuada, señalando que esto aumenta la exposición a **ciberataques**. Tener demasiadas aplicaciones inactivas o poco seguras puede facilitar el acceso no autorizado, ya que cada app representa un **punto de entrada para malware**. Kaspersky recomienda hacer **limpieza periódica** y limitar los permisos que se conceden, especialmente en apps **menos confiables**. Consulta el artículo completo aquí.

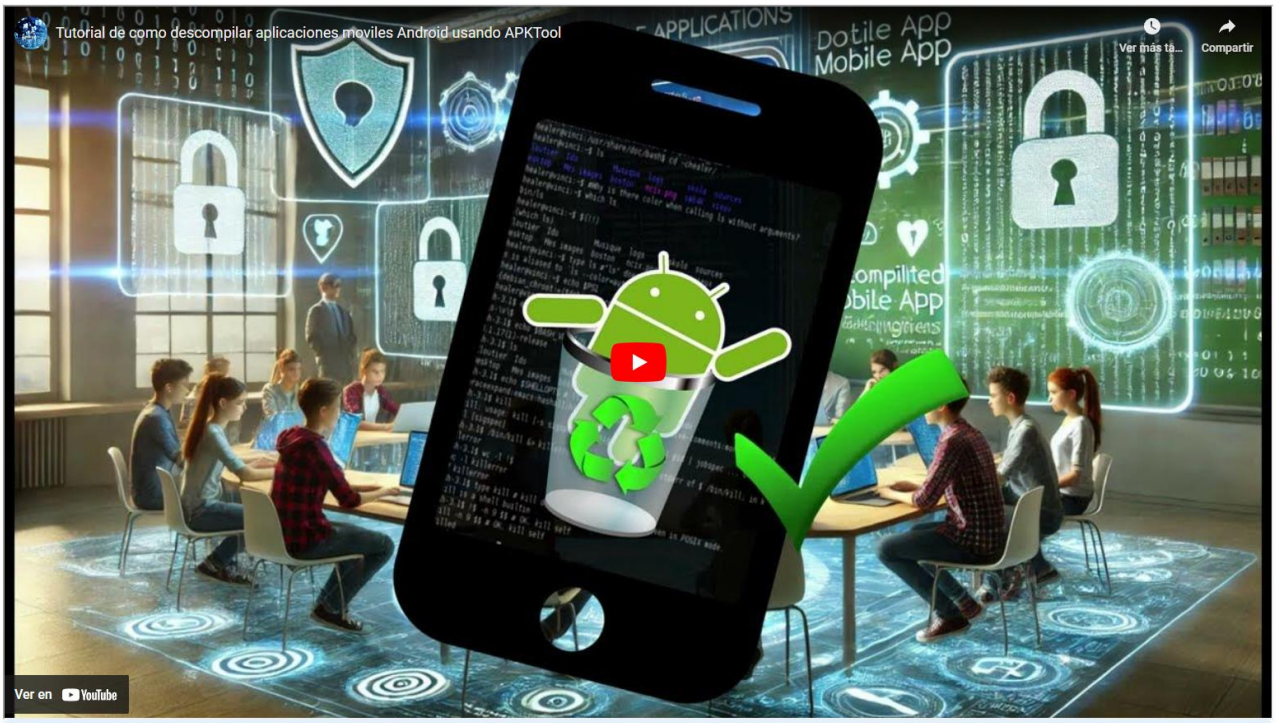
Malware en navegadores vulnerables: Según Kaspersky, una vulnerabilidad en un navegador ampliamente usado permitió que más de **318,000 usuarios de Android** fueran afectados por **malware**. Este tipo de ataque aprovecha fallos de seguridad no parcheados en el navegador para instalar malware que puede **robar información personal** o incluso **realizar transacciones financieras** sin el consentimiento del usuario. Kaspersky enfatiza la importancia de **mantener los navegadores actualizados** para mitigar estos riesgos. Más detalles están disponibles en el informe completo.

Permisos innecesarios en juegos de Android: Kaspersky identifica **cinco permisos** que muchos juegos para Android solicitan, a pesar de no ser necesarios para su funcionamiento, como acceso a **contactos** y **ubicación**. Este acceso puede resultar en **riesgos de privacidad**, ya que dichos permisos facilitan la recopilación y explotación de datos del usuario. Kaspersky recomienda a los usuarios **revisar cuidadosamente los permisos** solicitados antes de instalar juegos y denegar aquellos que no son esenciales. Consulta el artículo completo aquí.

En este módulo veras contenido educativo sobre la seguridad en las aplicaciones móviles y en la descripción veras las informaciones sobre los riesgos tecnológicos.

Aprende a Decompilar con “APKTool”

"¡DESCUBRE CÓMO DECOMPILAR APLICACIONES ANDROID CON APKTOOL, PASO A PASO!"



Descripción del Video

¡Bienvenido a este tutorial! En este video, aprenderemos a investigar aplicaciones móviles y su seguridad, centrándonos en la aplicación Flujo TV. Aunque ofrece acceso gratuito a canales de televisión, películas y series, no está disponible en la Play Store, lo que plantea riesgos potenciales.

A lo largo del video, te guiaré en el proceso de descargar la aplicación en formato APK y usaré APKTool para descompilarla y explorar sus permisos. También analizaremos qué permisos son realmente necesarios y cuáles pueden representar un peligro para tu privacidad y seguridad.

Lo que verás en este video:

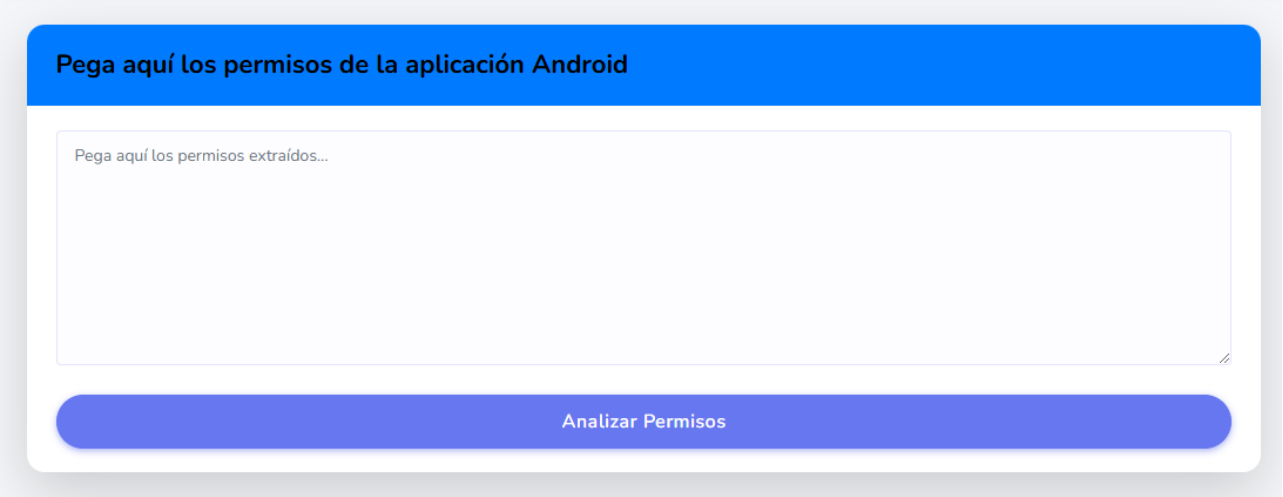
1. Introducción a la seguridad en aplicaciones móviles y la importancia de las descargas desde la Play Store.
2. Presentación de Flujo TV y sus funciones.
3. Proceso de descompilación de la aplicación usando APKTool.
4. Exploración del archivo AndroidManifest.xml para revisar los permisos solicitados.
5. Identificación de permisos peligrosos y recomendaciones de seguridad.

Recuerda, siempre es fundamental revisar los permisos que solicitan las aplicaciones y optar por fuentes confiables. ¡Disfruta del análisis y cuida tu privacidad!
Si quieres instalar APKTool en tu ordenador, ¡no te pierdas el tutorial completo aquí!: Tutorial de Instalación "APKTool"

Explora y Describe Códigos de Permisos

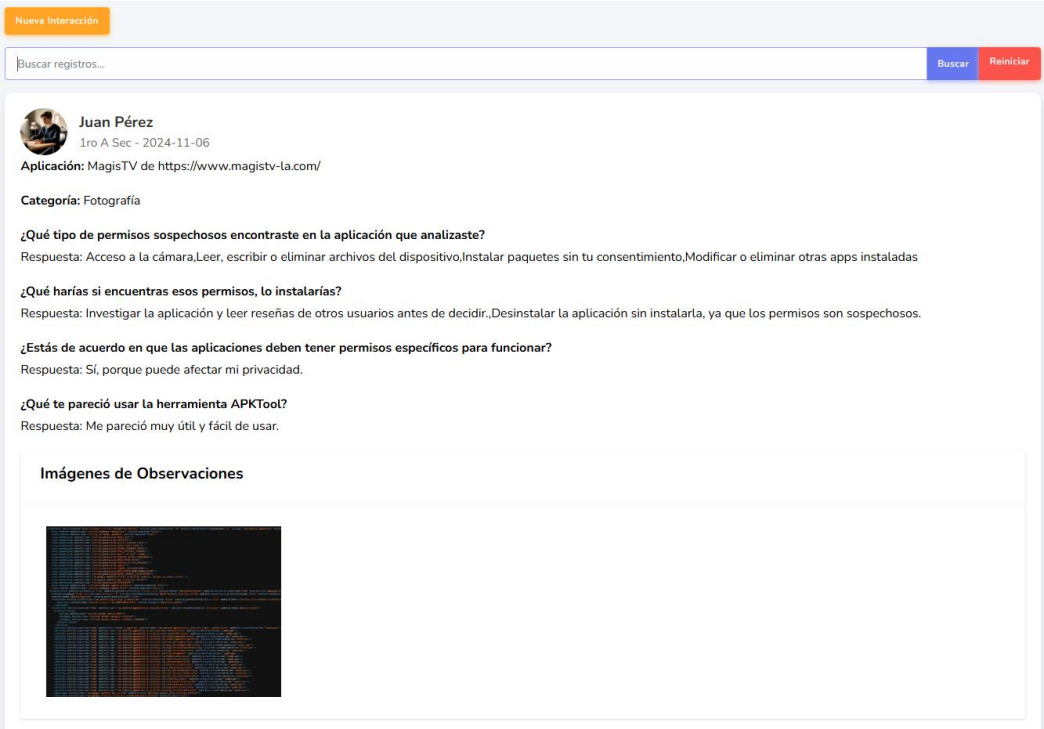
En este módulo veras contenido educativo sobre como descompilar aplicaciones móviles con la herramienta externa “APKTool”, también tendrás acceso al módulo de “Asistente de Análisis” presionando el botón “Explora y Describe Códigos de Permisos”.

Asistente de Análisis



En este módulo podrás ingresar los códigos de los permisos que los obtendrás a través del archivo AdroidManifest.xml, se recomienda ver el tutorial de como decompilar aplicaciones móviles Android y también tendrás que contar con la herramienta APKTool en tu ordenador.

Resultado de Análisis por los Estudiantes



En este módulo veras los resultados del análisis realizado por los estudiantes del colegio, solo los estudiantes o personal con autorización podrá realizar el cuestionario.

Avisos Importantes

En este módulo veras los avisos, comunicados, citaciones de parte del personal administrativo, solo el personal con permisos podrá crear publicaciones

Manual Técnico

UNIDAD EDUCATIVA SAN MARTIN DE PORRES

“DON BOSCO”

Correo electrónico: sanmartindeporresdonbosco@gmail.com

Sitio web: smpdonbosco.com.bo

Versión: 1.0

Let’s Start!

INTRODUCCIÓN

Descripción del Sistema

Este sistema web fue desarrollado en Laravel y permite analizar aplicaciones móviles para proteger la seguridad de los estudiantes en entornos educativos.

Propósito del Manual

Ofrecer a los técnicos instrucciones detalladas para la instalación, configuración, y mantenimiento del sistema.

OBJETIVOS

Objetivo General

Facilitar el mantenimiento y actualización del sistema para asegurar su disponibilidad y seguridad.

Objetivos Específicos

- Proporcionar información detallada sobre la estructura de archivos y bases de datos.
- Instruir sobre la configuración del servidor.
- Explicar las validaciones y medidas de seguridad implementadas en el sistema.

REQUISITOS TÉCNICOS

Requisitos de Hardware

- CPU de 4 núcleos, 4 GB de RAM, y al menos 50 GB de almacenamiento para el servidor.

Requisitos de Software

- Framework Laravel: Versión 8
- Servidor Local: XAMPP 3.2.2 (incluye Apache y Base de Datos MariaDB).
- Gestión de Dependencias: Composer.
- Plataforma de Frontend: Node.js, versión 16.20.2, para gestionar dependencias y ejecutar scripts de desarrollo.
- Framework CSS: Bootstrap 4.6.2 para el diseño responsivo.
- Editor de Código: Visual Studio Code, o cualquier otro editor con soporte para PHP, JavaScript y herramientas de desarrollo.

Instalación del Sistema

Paso 1: Instalación de XAMPP (Servidor Local)

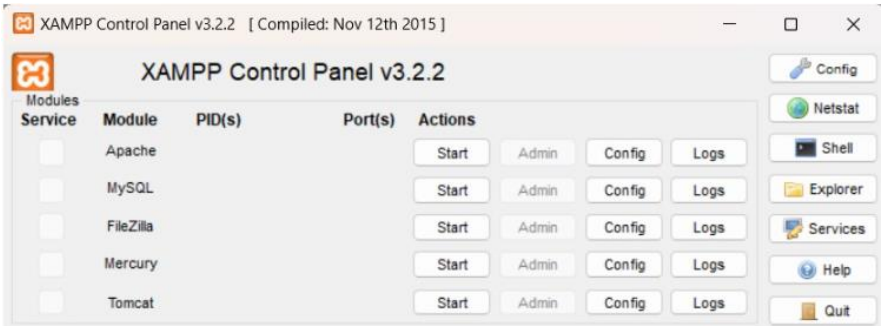
Descargar XAMPP

Visita https://www.apachefriends.org y descarga XAMPP versión 3.2.2 compatible con tu sistema operativo.

Instalar XAMPP

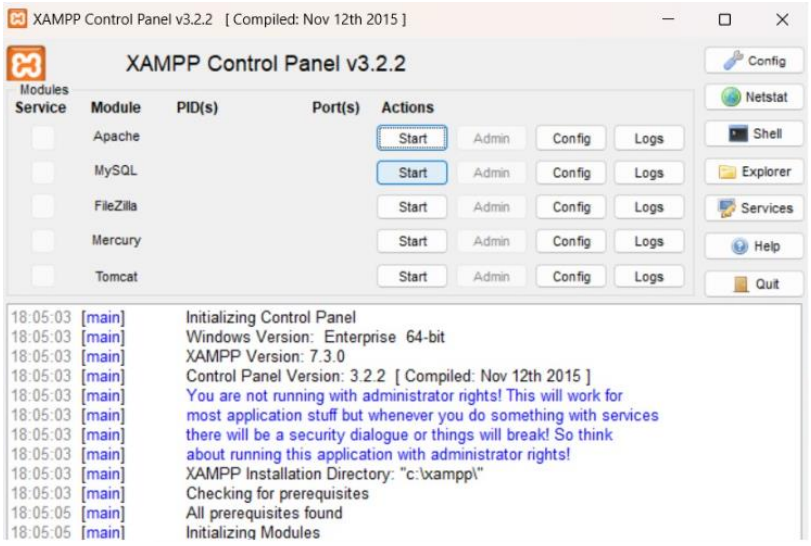
Ejecuta el archivo descargado y sigue las instrucciones en pantalla. Selecciona Apache y MySQL (MariaDB) durante la instalación.

Completa el proceso de instalación y abre el Panel de Control de XAMPP.



Iniciar Apache y MySQL

En el Panel de Control de XAMPP, haz clic en "Start" para iniciar Apache y MySQL. Asegúrate de que ambos servicios están activos antes de continuar.



Paso 2: Instalación de Composer (Gestión de Dependencias de PHP)

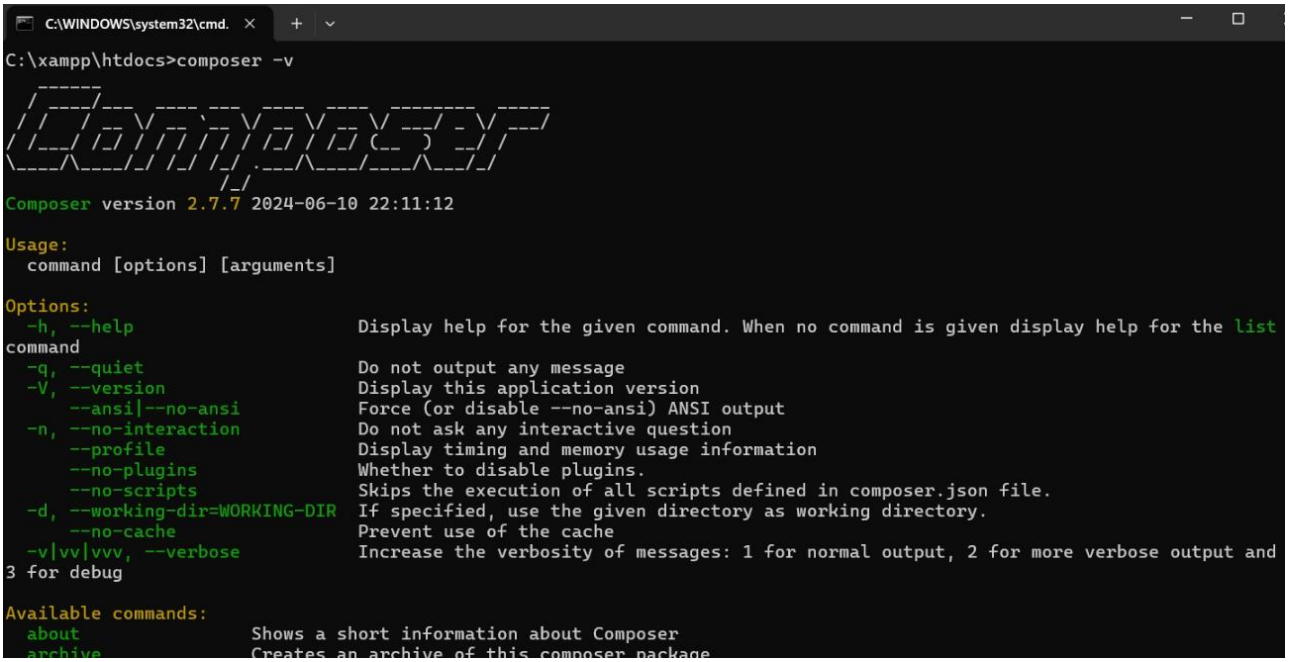
Descargar Composer

Visita https://getcomposer.org y descarga el instalador de Composer para tu sistema operativo.

Instalar Composer

Ejecuta el instalador y sigue las instrucciones en pantalla. Durante la instalación, Composer intentará encontrar automáticamente la instalación de PHP en tu sistema (normalmente en la carpeta de XAMPP).

Completa la instalación. Para verificar que Composer se instaló correctamente, abre una terminal (o símbolo del sistema) y ejecuta: “composer -v”



Instalación de Node.js y npm (Plataforma de Frontend)

Descargar Node.js

Visita https://nodejs.org y descarga la versión 16.20.2 de Node.js compatible con tu sistema operativo. Este instalador también incluye npm, el gestor de paquetes de Node.js.

Instalar Node.js

Ejecuta el instalador de Node.js y sigue las instrucciones en pantalla para completar la instalación.

Verificar la Instalación de Node.js y npm

Para verificar que Node.js y npm se instalaron correctamente, abre una terminal y ejecuta: “node -v” y luego “npm -v”

```
C:\xampp\htdocs>node -v
v16.20.2

C:\xampp\htdocs>npm -v
npm warn cli npm v10.8.2 does not support Node.js v16.20.2. This version of npm supports the following node versions: `
18.17.0 || >=20.5.0`. You can find the latest version at https://nodejs.org/.
10.8.2
```

Instalar Dependencias de Frontend

Desde la carpeta de tu proyecto de Laravel, instala las dependencias de frontend utilizando npm: “npm install”

Instalación de Visual Studio Code (Editor de Código)

Descargar Visual Studio Code

Visita https://code.visualstudio.com y descarga Visual Studio Code para tu sistema operativo.

Instalar Visual Studio Code

Ejecuta el instalador y sigue las instrucciones para completar la instalación.

Abre Visual Studio Code