Pontificia Universidad Católica del Ecuador

Facultad De Ingeniería

Escuela de Sistemas

TEMA:

Desarrollo de una aplicación prototipo para la detección temprana de códigos QR maliciosos: Una perspectiva de seguridad móvil

AUTOR:

ERICK SEBASTIAN VEGA MUELA

TRABAJO PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE SISTEMAS Y COMPUTACIÓN

QUITO, JUNIO – 2023

DEDICATORIA

A mi abuelita, mi abuelito, mi primo y a mi tía que siempre estuvieron presentes.

AGRADECIMIENTO

A mis Padres que me apoyado en todos mis estudios, a mi hermana cuya presencia constante ha sido un gran respaldo en mi vida, agradezco igualmente a mi amada familia, mis entrañables amigos, y a mi perro Jeiko.

RESUMEN

La presente tesis se enfoca en la creación de un prototipo de aplicación móvil segura para dispositivos Android con códigos QR. El proyecto utiliza una metodología basada en la creación de prototipos para analizar exhaustivamente los requisitos de seguridad.

Durante el desarrollo, se emplea Android Studio, se aplican arquitecturas limpias y se gestiona adecuadamente los recursos para lograr los objetivos en un tiempo razonable. Se aplican metodologías de investigación cuantitativa y aplicada, junto a una metodología de desarrollo iterativo que permiten llevar a cabo un análisis exhaustivo y una implementación efectiva de la aplicación. En el desarrollo de la investigación se destaca la metodología de prototipos empleada, la clasificación de actividades y requisitos, el diseño de la interfaz de usuario, las funcionalidades de la aplicación y el diseño de la base de datos. Estos aspectos resultan cruciales para la creación de un prototipo sólido y funcional.

En función de evaluar la efectividad y la seguridad del prototipo desarrollado se implementan las pruebas funcionales, no funcionales y de carga, obteniéndose resultados satisfactorios.

Palabras Clave: Códigos QR, Seguridad, Aplicativo móvil, Metodología de prototipos

ÍNDICE

ÍNDICE DE FIGURAS, GRÁFICOS Y TABLAS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO I: INTRODUCCIÓN

1. Marco de referencia

En este capítulo se expone el marco de referencia, cuyo objetivo es contextualizar el estudio sobre la detección temprana de códigos QR maliciosos desde una perspectiva de seguridad móvil. La seguridad móvil se ha convertido en una preocupación en todo el mundo debido al incremento en el uso de dispositivos móviles. Según el informe anual de una reconocida empresa de investigación de mercado conocida como Statista, se prevé que la cantidad de usuarios con teléfonos móviles inteligentes a nivel mundial llegue a los 7.33 mil millones en 2023, un 5.3% más desde los 6.95 mil millones en 2021. Además, la cada vez mayor sofisticación de los ataques de malware dirigidos a estos dispositivos también ha aumentado la necesidad de mejorar la seguridad móvil.

En este sentido, se resalta la relevancia de disponer de herramientas de detección eficaces que permitan identificar y mitigar los riesgos asociados a los dispositivos móviles. Según un informe de Latam Kaspersky (2023), en el año 2022 se detectaron 1,6 millones de instaladores de software malicioso para dispositivos móviles, lo cual implica un incremento del 30% en comparación con los datos registrados en el año previo. Además, se estima que para el año 2023 se incrementará la cantidad de usuarios de dispositivos móviles, lo que aumentará el riesgo de ataques de malware móvil.

En este contexto, los códigos QR han adquirido una gran popularidad ya que se han convertido en una herramienta sumamente popular para acceder a la información en línea de forma ágil y sencilla. Sin embargo, también pueden ser utilizados por ciberdelincuentes para redireccionar a los usuarios a sitios web con intenciones maliciosas o para descargar malware en sus dispositivos móviles. Según Qrcode-tiger (2023), se estima que la cantidad de escaneos de códigos QR se prevé que alcance los 99.5 millones de escaneos para el año 2025.

Por lo tanto, el desarrollo de un prototipo de una aplicación para la detección temprana de códigos QR maliciosos es un tema de gran relevancia en el campo de la seguridad móvil. Esto podría ayudar a proteger a los usuarios de dispositivos móviles contra las amenazas de malware y mejorar la seguridad en línea en general.

1.1. Justificación

Los códigos QR, son ampliamente reconocidos como códigos de respuesta rápida, han surgido como una opción revolucionaria al código de barras convencional, ofreciendo una solución más dinámica y eficiente. De acuerdo con un informe de Insider Intelligence (2022), se prevé que la cantidad de usuarios de códigos QR en los Estados Unidos. aumente un 5,3% en 2021 debido a la necesidad de formas de pago sin contacto y de menor interacción física a causa de la pandemia de COVID-19. Este informe también pronostica que, para 2022, el número de escaneos de códigos QR a nivel mundial alcanzará los 5.300 millones de usuarios, lo cual implica un incremento del 2,7% en comparación con el número de escaneos en 2021. Además, se estima que la cantidad de usuarios de códigos QR continúe creciendo en los próximos años. Los códigos QR permiten descargar aplicaciones y archivos con un solo escaneo, además de acceder rápidamente a páginas web.

Hoy en día, muchas empresas utilizan códigos QR como una estrategia de marketing para mejorar sus ventas y adaptarse a la tecnología. Sin embargo, a diferencia de los enlaces web convencionales, los códigos QR no se pueden analizar antes de escanearlos, lo que los convierte en una potencial amenaza de seguridad. Además, estos códigos también pueden ser utilizados para llevar a los usuarios a sitios fraudulentos con la intención de apropiarse de datos personales y financieros de manera ilícita, lo que representa un riesgo de phishing.

Según la empresa de seguridad informática Kaspersky (2022), es crucial que los usuarios sean cautelosos al escanear códigos QR y siempre verifiquen la autenticidad del sitio web antes de proporcionar información confidencial.

Aunque cualquiera puede crear y distribuir un código QR para que lo escaneen miles de personas, es fundamental tener precaución y comprobar la autenticidad de las páginas web previamente a ingresar cualquier dato personal o financiero con el propósito de evitar caer en estafas que imitan a sitios web oficiales y exponer la integridad de la información personal y financiera a posibles amenazas. Es importante tener en cuenta que, en muchas ocasiones, las personas se percatan de que sus datos han sido utilizados de manera malintencionada demasiado tarde.

Esto puede dar lugar a la merma significativa de valiosas sumas de dinero o incluso en su involucramiento en actividades delictivas. Ante este escenario, es necesario considerar el desarrollo de una innovadora aplicación móvil que posibilite identificar de manera precisa la presencia de la procedencia confiable de un código QR escaneado.

La creación de un prototipo de aplicación móvil se justifica como una medida adicional de seguridad para proteger la información. Si bien existen varias aplicaciones en el mercado que analizan los URLs en busca de archivos y URLs sospechosas para detectar virus y malware, la aplicación móvil propuesta se centraría específicamente en la detección de códigos QR maliciosos y en un potencial fallo de seguridad que permite instalar aplicaciones APK en dispositivos Android de manera sencilla, solo con escanear un código QR. La principal diferencia entre la aplicación móvil propuesta y otras aplicaciones en el mercado radica en su enfoque específico en los códigos QR maliciosos. Además, su objetivo es demostrar que podemos defendernos de aplicaciones que buscan engañarnos para obtener información de forma fraudulenta. Por lo tanto, la creación de esta aplicación móvil representa un avance importante en la seguridad de la información.

1.2. Planteamiento del problema

Según el sitio web QRQuestion (2020), los códigos QR vieron la luz en el año de 1994, cuando la empresa japonesa Denso Wave, una destacada filial del grupo automovilístico Toyota, dio forma a esta brillante invención. Un ingeniero de la empresa desarrolló un ingenioso sistema de codificación bidimensional con puntos cuadrados en una paleta de colores que alternan entre el blanco y el negro, lo cual permitía una gestión más eficiente y una mayor capacidad de almacenamiento de información que los códigos de barras tradicionales. Tal como se describe en la página web de El Universo (2020), En Ecuador, el empleo de códigos QR se ha vuelto cada vez más común, debido a su facilidad de uso y múltiples aplicaciones en diferentes ámbitos, incluyendo el marketing y la publicidad.

Sin embargo, esta facilidad de uso también ha generado un riesgo importante para los usuarios, ya que se ha detectado la existencia de códigos QR maliciosos que pueden instalar aplicaciones APK en dispositivos Android de manera sencilla, sin que el usuario sea consciente de ello. Este potencial fallo de seguridad representa un riesgo para salvaguardar la privacidad y asegurar la protección de la información confidencial de los usuarios. Esto justifica la necesidad de desarrollar soluciones que permitan detectar y prevenir la presencia de códigos QR maliciosos.

Además, los códigos QR también pueden ser utilizados en ataques de phishing, ya que los usuarios pueden ser redirigidos a páginas web engañosas cuidadosamente concebidas con el propósito malicioso de apoderarse de datos personales y financieros confidenciales. De acuerdo con la empresa de seguridad informática Kaspersky (2022), los códigos QR maliciosos pueden ser utilizados para llevar a los usuarios a sitios fraudulentos con el fin de robar información confidencial. Es por eso por lo que es importante que los usuarios sean cautelosos al escanear códigos QR y siempre verifiquen la autenticidad de la página web previo a divulgar cualquier dato personal o financiero.

En función a la problemática planteada se propone la siguiente pregunta principal:

- ¿Cómo se puede desarrollar una aplicación móvil que permita detectar y prevenir la presencia de códigos QR maliciosos y resguardar la confidencialidad y protección de los datos de los usuarios al momento de escanear códigos QR en dispositivos Android?

Y las siguientes preguntas secundarias:

- ¿Cuáles son los principales riesgos asociados a la presencia de códigos QR maliciosos en dispositivos Android?
- ¿Qué características debe tener una aplicación móvil para detectar códigos QR maliciosos de manera efectiva?
- ¿Cuáles son los recursos necesarios para el desarrollo y mantenimiento de la aplicación móvil para la detección de códigos QR maliciosos en dispositivos Android?

1.3. Objetivo General

Crear una app móvil para dispositivos Android que permita detectar y prevenir la presencia de códigos QR maliciosos y resguardar la confidencialidad y protección de los datos de los usuarios al momento de escanear códigos QR.

1.4. Objetivos Específicos

- Identificar los principales riesgos asociados a la presencia de códigos QR maliciosos en dispositivos Android.
- Planificar y crear una app móvil que tenga la capacidad de identificar el origen confiable de un código QR escaneado, centrándose en la detección de códigos QR maliciosos y en un potencial fallo de seguridad que permite instalar aplicaciones APK en dispositivos Android de manera sencilla, solo con escanear un código QR.
- Proporcionar al usuario información útil sobre el código QR escaneado y alertar si se trata de un sitio malicioso, con el objetivo de resguardar tanto los datos personales como los financieros de los usuarios.
- Investigar y elaborar un registro acerca de los virus, incluyendo la información relevante y malware que pueden estar presentes en los códigos de respuesta rápida (QR) para mejorar la eficacia de la aplicación móvil en la detección de amenazas.

1.5. Antecedentes

En relación con el tema de investigación presentado, es importante destacar que la utilización de códigos QR se ha popularizado en los últimos años, y se ha visto una creciente distribución como una forma fácil y rápida para acceder a la información. Sin embargo, esto también ha llevado a un aumento en la distribución de malware y en la realización de ataques de phishing. Muchas empresas lanzan al mercado aplicaciones de códigos QR sin ninguna utilidad extra, por lo que estas aplicaciones, por lo general, no protegen al usuario contra el malware.

McAfee ha informado que algunos ciberdelincuentes ya han utilizado esta vulnerabilidad en los códigos QR de Android para distribuir malware a través de ellos. Por lo tanto, es importante que los usuarios tengan precaución al escanear códigos QR de fuentes desconocidas y que se aseguren de que sus dispositivos Android cuenten con software de seguridad actualizado (Boletín de seguridad de Android, 2023). Esta vulnerabilidad ha sido encontrada en la última versión y posteriores de los dispositivos Android (12), y consiste en que al apuntar la cámara hacia el código QR, se puede instalar una aplicación maliciosa sin necesidad de analizar otro elemento que no sea el código QR. Además, el enlace que dirige a esta aplicación maliciosa puede estar alojado en Google Drive. Aunque este fallo de seguridad ha sido analizado en varios dispositivos Android, aún no hay una actualización oficial para corregirlo. Si algún malware logra explotar esta vulnerabilidad, los usuarios pueden enfrentar problemas significativos.

Según Gupta (2023), en su página web explica que "Dado que cualquier tipo de malware o enlaces de phishing en los códigos QR plantea riesgos de seguridad significativos tanto para las empresas como para los consumidores, se deben considerar medidas de seguridad estrictas para mitigar el riesgo". Es decir, nunca se eliminará la amenaza, sino que se deben tomar medidas para mitigar los riesgos de seguridad en respecto a los códigos QR. Por lo tanto, en la presente investigación nos centraremos en dar relevancia e importancia a varios de los virus que pueden afectar a este sistema operativo, además de la importancia de una

aplicación prototipo para la protección y detección de los códigos de respuesta rápida (QR) maliciosos dado el aumento del uso de estos códigos QR en la sociedad actual. Todo ello con el objetivo de mejorar la seguridad móvil de todos los usuarios.

1.6. Alcance

El presente trabajo de titulación contempla el desarrollo de una aplicación móvil, que permitirá leer y alertar sobre los códigos QR que pueden generar peligro, y una funcionalidad de escáner de documentos. Se busca mejorar la seguridad y eficiencia en la detección de códigos QR, permitiendo a los usuarios tener mayor control y protección sobre la información que comparten.

El tiempo para desarrollar esta aplicación es de 3 meses y contará con las siguientes funcionalidades:

- Mis Documentos: Esta funcionalidad permite escanear documentos, CI, libros y otros elementos.
- Lector de Códigos QR: Esta funcionalidad permite identificar códigos QR maliciosos.
- Generador de códigos QR: Esta funcionalidad permite a los usuarios crear códigos QR.
- Política de Privacidad: Esta funcionalidad permite a los usuarios leer la política de privacidad de la aplicación a través de una vista web.
- Compartir Aplicación: Esta funcionalidad permite a los usuarios compartir la aplicación con otros.
- Califíquenos: Esta funcionalidad permite a los usuarios valorar la aplicación.
- Cambiar Tema: Esta funcionalidad permite a los usuarios cambiar el tema de la aplicación seleccionando un nuevo tema de una lista de opciones.

1.7. Organización del trabajo

El trabajo se organiza de la siguiente manera: el Capítulo II aborda aspectos teóricos relacionados con el hardware de los smartphones, las vulnerabilidades y virus, los códigos QR, las herramientas de desarrollo, el diseño de la interfaz de usuario y la metodología de prototipos. En el Capítulo III se presenta la metodología de desarrollo del plan de tesis, que incluye enfoques de investigación cualitativa y aplicativa, así como la metodología de desarrollo de software. Además, el Capítulo IV detalla el proceso de desarrollo de la investigación, haciendo hincapié en la metodología de prototipos, las funcionalidades de la aplicación, las mejoras de seguridad en el escaneo de códigos QR, el desarrollo de interfaces y el diseño de la base de datos. Por último, el Capítulo V aborda la implementación de la aplicación y presenta los resultados obtenidos.

CAPÍTULO II: FUNDAMENTACIÓN TEÓRICA

2. Marco Teórico

2.1. Generalidades

Se han seleccionado los temas necesarios para la elaboración de este capítulo, con el objetivo de proporcionar una descripción detallada del hardware presente en los smartphones, incluyendo la cámara, los sensores y el almacenamiento. Además, se abordan temas relacionados con el software, como los sistemas operativos móviles Android y iOS. También se examinan aspectos de seguridad móvil, las vulnerabilidades en los smartphones y los tipos de virus más comunes, como el phishing y el malware. Asimismo, se realiza un análisis específico de la seguridad en la plataforma de descarga de aplicaciones para dispositivos móviles, Google Play Store y se exploran los códigos QR. En este capítulo, también se presentan herramientas de desarrollo como Android Studio, además del desarrollo de la interfaz de usuario y la metodología de creación de prototipos. En resumen, el marco teórico que se presenta ofrece una base sólida para el estudio de la protección en dispositivos móviles y la evaluación de riesgos en relación con los códigos QR.

2.2. Hardware: Smartphones

Según Moes (2020), el hardware se refiere a los elementos tangibles y físicos de un dispositivo, aquellos perceptibles a través de los sentidos. Es decir, incluye todos los componentes tangibles que podemos encontrar en una computadora o dispositivo electrónico, como la placa madre, CPU, GPU, memoria RAM, disco duro y fuente de alimentación. En el caso de dispositivos móviles, la evolución de los componentes electrónicos ha llevado a una modernización del proceso de producción, que integran varios componentes principales en un solo chip, los elementos que contribuyen al funcionamiento adecuado de los dispositivos de manera excelente en un espacio reducido. Además, se deben considerar otros componentes como la batería, sensores, cámara, puertos y antenas.

De acuerdo con Provazza (2019), un dispositivo móvil o teléfono inteligente se define como un aparato que ofrece múltiples funcionalidades avanzadas y un alto rendimiento

computacional. Estos dispositivos han evolucionado con el tiempo, incorporando cada vez mayores capacidades de cómputo y conectividad. Además de realizar y recibir llamadas de voz por diferentes medios, también permiten enviar mensajes de texto. Los smartphones disponen de una pantalla táctil que utiliza diferentes tecnologías, dependiendo del fabricante, lo que nos permite interactuar de manera sencilla con una diversidad de aplicaciones extensa y servicios disponibles en las distintas tiendas de aplicaciones digitales.

2.2.1. Cámara

La funcionalidad de la cámara destaca como una de las características primordiales de un teléfono móvil, ya que brinda a los usuarios la capacidad de capturar imágenes y grabar videos de excelente resolución. En los dispositivos móviles disponibles en el mercado, las cámaras se encuentran en la parte frontal y trasera, y en algunos casos, pueden tener múltiples cámaras dependiendo del fabricante. La nitidez de la imagen guarda una relación directa con la lente de la cámara, que enfoca la luz en el sensor de imagen. En los smartphones actuales, existen lentes gran angular, teleobjetivo y macro que permiten un enfoque y acercamiento precisos para capturar imágenes detalladas.

2.2.2. Sensores

En los smartphones, los sensores tienen una amplia variedad de aplicaciones y usos en la tecnología. Por ejemplo, tenemos sensores de movimiento como el giroscopio y el acelerómetro que se utilizan para aplicaciones de realidad aumentada, juegos, brújula, entre otros. También contamos con un sensor de huellas dactilares que proporciona una autenticación única del usuario. Otro sensor importante es el GPS, el cual está diseñado para transmitir información a los satélites y generar una señal para triangular la posición exacta del usuario. Además, contamos con sensores de luz ambiental cuya función principal es detectar la cantidad de luz en el ambiente y modificar las opciones de brillo disponibles en la pantalla del dispositivo. Estos sensores mejoran la experiencia del usuario al interactuar con el mundo físico de manera innovadora.

Según Arredondo (2015), "*Los sensores integrados en los dispositivos móviles tienen como principal acometido el de recopilar información importante para el funcionamiento en general, pero también para ciertas aplicaciones y, en definitiva, el sistema operativo*". Es importante destacar que los sensores pueden recopilar información que puede involucrar datos acerca de la localización del dispositivo, el movimiento y la orientación de este, entre otros aspectos, lo cual puede ser utilizado por diversas aplicaciones para ofrecer servicios personalizados al usuario.

2.2.3. Almacenamiento

El almacenamiento en los dispositivos móviles se refiere a la capacidad interna de almacenamiento que tiene el dispositivo. Aquí se almacenan todos los datos y archivos necesarios para su correcto funcionamiento, como documentos, archivos, aplicaciones, vídeos y música.

Existen dos medios principales de almacenamiento en los activos móviles: la memoria flash NAND y la memoria RAM. La primera se utiliza para almacenar los archivos que estarán disponibles incluso si apagamos el dispositivo, ya que es un tipo de almacenamiento no volátil. En cambio, la memoria RAM se utiliza como un tipo de almacenamiento de acceso aleatorio para el almacenamiento temporal de datos y para la ejecución rápida de aplicaciones en el sistema.

A lo largo de los años, la habilidad para almacenar datos en los dispositivos móviles ha variado y actualmente puede oscilar entre 16 GB y más de 1 TB. Sin embargo, el precio de los dispositivos aumenta proporcionalmente con el almacenamiento que ofrecen. La mayoría de los smartphones modernos tienen la opción de aumentar la capacidad de almacenamiento mediante la utilización de una tarjeta de memoria microSD, cuyo precio dependerá de su capacidad de almacenamiento y velocidad de transferencia de datos en operaciones de lectura y escritura.

La velocidad de acceso y escritura es crucial, dado que influye en la agilidad al abrir aplicaciones, realizar tareas y guardar cambios realizados. Una de las tecnologías que está presente en la actualidad es UFS 3.0, la cual ofrece una velocidad de lectura y escritura mucho más rápida en comparación con otras tecnologías del mismo medio.

2.3. Software: Sistemas Operativos móviles (Android y iOS)

Según el sitio Computer Hope (2021), el software se define como "un conjunto de instrucciones que permiten al usuario interactuar con un ordenador, su hardware o realizar tareas. Sin software, la mayoría de los ordenadores serían inútiles". Esta definición resalta la importancia fundamental del software en la informática moderna, ya que posibilita que los usuarios realicen diversas operaciones en sus ordenadores, como enviar correos electrónicos, acceder a Internet, procesar datos y generar documentos. En pocas palabras, el software es un componente necesario para el funcionamiento de los ordenadores; sin él, la mayoría de las operaciones diarias que realizamos en nuestros ordenadores serían imposibles.

Android se configura como un sistema operativo diseñado especialmente para su implementación en dispositivos móviles, tabletas, relojes inteligentes, televisores inteligentes y otros dispositivos. Lanzado en 2008, se fundamenta en el núcleo de Linux, el cual se encarga de gestionar los recursos de hardware, incluyendo la unidad central de procesamiento (CPU) y los periféricos de entrada y salida. Al estar basado en Linux, los desarrolladores pueden crear versiones estables, seguras y eficaces del sistema operativo, lo que ha permitido crear versiones personalizadas para diferentes fabricantes.

Android se destaca como uno de los sistemas operativos móviles más ampliamente adoptados, siendo accesible en una amplia variedad de dispositivos alrededor del mundo. Es conocido por su alta capacidad para gestionar aplicaciones y ejecutar múltiples procesos en segundo plano. La integración de varios servicios proporcionados por Google ha beneficiado a los usuarios, lo que permite un desarrollo de aplicaciones ilimitado y una distribución fácil a través de la tienda oficial Google Play Store.

Además, Android se ha extendido a otros tipos de dispositivos, como electrodomésticos parlantes y televisores inteligentes, gracias a su flexibilidad y personalización. La popularidad de Android en dispositivos móviles ha llevado a una mayor adopción en otros tipos de dispositivos, como sistemas de información y entretenimiento en vehículos. Su capacidad para adaptarse a diferentes necesidades y dispositivos lo hace una elección inteligente para diversos dispositivos electrónicos de consumo y vehículos automotores.

En 2007, Apple necesitaba un sistema operativo móvil para su nuevo iPhone y creó iOS. Desde entonces, los dispositivos móviles han evolucionado en aparatos versátiles y prácticos, ofreciendo diversas funcionalidades y los usuarios han demandado nuevas funcionalidades.

En lugar de utilizar un sistema operativo ya existente, Apple optó por desarrollar su propio sistema operativo. En 2005, Apple compró Safari Software y contrató a algunos de sus mejores desarrolladores para trabajar en un pequeño proyecto. El equipo elegido se centró en crear un sistema operativo fácil y sencillo que pudiera aprovechar cada uno de los componentes con los que contaba el iPhone en ese momento.

El modelo original de iPhone se lanzó en 2007 con el sistema operativo iOS 1.0, que incluía tecnologías pioneras de la época, como una pantalla táctil y un control por gestos integrado. Apple ha seguido invirtiendo en investigación y en sus componentes para mejorar el mercado y ha actualizado su sistema operativo con nuevas características de vanguardia, como una tienda de aplicaciones segura y óptima, asistentes de voz como Siri y el sistema de pagos Apple Pay.

iOS se caracteriza por su facilidad de uso y alta seguridad, y la popularidad de los dispositivos ha ido creciendo conforme a sus nuevas características implementadas. Apple, como una gran compañía, se centra en lanzar actualizaciones de forma regular para el sistema operativo iOS, mejorando su rendimiento y agregando nuevas características y funciones que pueden ser relevantes para los usuarios.

2.4. Seguridad móvil

Según la definición de NortonLifelock (2023), la seguridad móvil se refiere a la salvaguardia de dispositivos como smartphones, tablets, laptops y otros aparatos capaces de acceder a una red contra amenazas y vulnerabilidades que pueden provenir tanto de virus y malware internos como de fuentes externas, como la red a la que se conectan. Es importante destacar que la seguridad móvil no se limita a la protección del propio dispositivo, sino que también se refiere a la protección de la información almacenada en el dispositivo. Por lo tanto, es fundamental proteger los dispositivos móviles de posibles amenazas internas y adicionales para salvaguardar la confidencialidad y seguridad de los datos almacenados en los mismos.

2.5. Vulnerabilidades en Smartphones

Según Norton Antivirus (2023), los dispositivos móviles son susceptibles a múltiples amenazas, como el malware, spyware, phishing y otros peligros, que pueden infiltrarse en el equipo mediante el uso de aplicaciones instaladas, conexiones de red sin protección y sitios web y correos electrónicos que carecen de medidas de seguridad adecuadas. En pocas palabras, esta referencia destaca las múltiples amenazas a las que se enfrentan los dispositivos móviles. Además, un estudio de McAfee (2021) destaca que las posibles debilidades de seguridad presentes en los sistemas operativos utilizados en dispositivos móviles Android, las cuales podrían permitir a los atacantes acceder a la información del usuario sin su conocimiento. Es fundamental entender estas vulnerabilidades para tomar medidas de seguridad adecuadas.

2.6. Virus: Phishing, Malware

Según la definición de phishing proporcionada por Kaspersky IT Encyclopedia (2020), en su sitio web oficial, se entiende por phishing "un tipo de fraude en Internet que busca obtener las credenciales de un usuario mediante el engaño. Esto implica el robo de contraseñas, números de tarjetas de crédito, datos de cuentas bancarias y otra información confidencial" En otras palabras, el phishing es un engaño que puede hacer que los usuarios pierdan sus contraseñas y datos personales en línea.

De acuerdo con la definición de Belcic (2023), el malware se refiere a cualquier software o código informático invasivo que tiene como objetivo dañar, infectar o acceder a sistemas informáticos. Existen diferentes tipos de malware, y cada uno afecta a los dispositivos de manera distinta, pero su principal objetivo es comprometer la protección y la confidencialidad de los sistemas informáticos. En resumen, el malware es un programa malicioso que busca dañar los sistemas informáticos o los dispositivos. En ocasiones, el malware tiene como objetivo recopilar información financiera y confidencial, lo que puede resultar en extorsiones, fraudes y robos de identidad.

2.7. Análisis de seguridad de la tienda de aplicaciones móviles (Google Play Store)

El análisis de seguridad es una práctica esencial para garantizar que los sistemas y datos estén protegidos contra amenazas, y esto también aplica para la tienda de aplicaciones móviles de Google, conocida como Google Play Store. En el caso específico de esta plataforma, el análisis de seguridad es crucial debido a la gran cantidad de aplicaciones que están disponibles para descargar y que pueden potencialmente representar un riesgo para la seguridad de los usuarios.

En este sentido, Google realiza un análisis de seguridad exhaustivo de todas las aplicaciones que se encuentran en su tienda, con el fin de identificar cualquier posible vulnerabilidad o riesgo que puedan representar. Tal como señala Android Developer (2022), "utilizamos una variedad de técnicas para detectar y prevenir el abuso de las aplicaciones en Google Play, incluyendo algoritmos de aprendizaje automático y análisis estático y dinámico".

Además, Google también cuenta con un programa llamado "Google Play Protect", que es una herramienta de seguridad que escanea automáticamente las aplicaciones que se implementan en los dispositivos móviles con el propósito de detectar cualquier tipo de amenaza. De acuerdo con Android Source (2022), “Google Play Protect realiza escaneos diarios de más de 100 mil millones de aplicaciones para asegurarse de que todo lo que se descargue sea seguro”.

La evaluación de seguridad es una práctica esencial para asegurar la protección de los usuarios en la tienda de aplicaciones móviles de Google, y la empresa ha implementado diversas herramientas y programas para lograr este objetivo. Es importante destacar que también es responsabilidad de los usuarios estar informados y tomar precauciones al momento de descargar y utilizar aplicaciones en sus dispositivos móviles.

2.8. Códigos QR: Códigos QR en nuestra sociedad Actual

En la sociedad actual, los códigos QR han cobrado gran importancia. Estos códigos, formados por cuadrados y líneas en una estructura en 2D, representan una forma de código de barras que permiten acceder, almacenar y distribuir grandes cantidades de información. Al escanearlos con la cámara de un dispositivo portátil o una tableta, se tiene la capacidad de acceder a dicha información, lo que ha permitido la extensa disponibilidad en distintos ámbitos como la publicidad y los inventarios. De acuerdo con Calvo (2023) afirmó lo siguiente:

> El uso de códigos QR puede marcar la diferencia y ayudar a un negocio a despegar, aumentar las ventas, fidelizar a sus clientes o alcanzar cualquier otro objetivo propuesto. Ofrecen miles de posibilidades si se ejecutan correctamente como parte de una buena estrategia de marketing. Además, no son discriminatorios y pueden ser utilizados tanto por empresas grandes como por PYMES.

En otras palabras, el uso de estos códigos puede impulsar a los negocios para obtener más clientes y oportunidades en el mercado. Aunque este artículo se centra en la funcionalidad y ventajas de los códigos QR en el marketing, en realidad, estos códigos pueden ser aplicados en muchas otras áreas de la sociedad actual.

2.8.1. Actividades ilícitas con códigos QR

Los códigos QR pueden ser utilizados de manera ilícita para redirigir a los usuarios a páginas maliciosas, descargar malware o solicitar información personal. Es crucial estar alerta sobre el uso de estos códigos y asegurarnos de que nos dirijan a sitios web legítimos.

Además, debemos verificar la fuente del código antes de escanearlo, ya que a menudo se encuentran códigos QR en lugares públicos como buses de transporte urbano, señales de tránsito, menús de restaurantes, entre otros.

De acuerdo con Alcaraz (2022), una de las actividades ilícitas más comunes es descargar archivos perjudiciales en el dispositivo de la persona afectada. Por ejemplo, muchos bares y restaurantes utilizan códigos QR para que los usuarios descarguen un archivo PDF con el menú o instalen una aplicación para realizar pedidos. En este y otros contextos similares, los atacantes podrían fácilmente alterar el código QR para engañar al usuario e instalar una aplicación fraudulenta. Esto facilita la distribución ilícita de aplicaciones que recopilan datos de los usuarios.

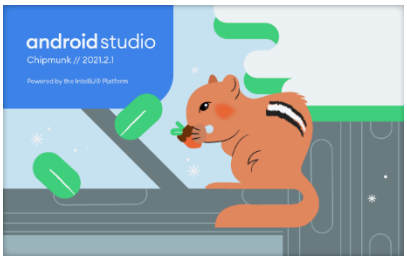
2.9. Herramientas de Desarrollo: Android Studio

Como afirma Nielfa (2020), Android Studio se configura como un entorno de desarrollo integrado (IDE) basado en la plataforma IntelliJ IDEA, que brinda diversas capacidades para la creación de aplicaciones destinadas al sistema operativo Android. Además de incluir un editor de códigos eficiente, este IDE cuenta con un sistema de compilación adaptable, un emulador de alta velocidad y herramientas para detectar inconvenientes de compatibilidad, desempeño y experiencia de usuario. También es compatible con lenguaje C++, NDK y la plataforma Google Cloud. En síntesis, Android Studio es una herramienta que optimiza la eficiencia de los desarrolladores al proveer una variedad de funciones valiosas.

En cuanto a sus ventajas, Android Studio permite simular diferentes dispositivos, lo que facilita la ejecución de pruebas y la creación de aplicaciones para distintos tamaños de pantalla y hardware. También permite ver diferentes partes de código al mismo tiempo, lo que aumenta la eficiencia y facilita la tarea de los desarrolladores. La versión de Android Studio utilizada en el desarrollo de este proyecto se ilustra en la Figura 1.

Figura 1.

Versión utilizada para el desarrollo de esta aplicación en el entorno de Android Studio: Chipmunk// 2021.2.1



Nota. Adaptado de *Versión de Android Studio 2021.2.1* [Captura de pantalla], por Developer Android, 2022 (https://shorturl.at/aHPQ4). CC BY 2.0

Para programar en Android Studio, es posible utilizar diferentes sistemas operativos como Windows, Mac o Linux. En la tabla 1, tabla 2 y tabla 3 se presentan los requisitos necesarios para trabajar en cada uno de estos entornos.

Tabla 1

Tabla de Requisitos de sistema para instalar Android Studio en Windows

	Requisitos Mínimo	**Requisitos Recomendados**	**Requisitos Alto Rendimiento**
CPU	2da generación: Intel Core i3/i5/i7/i9 (Sandy Bridge)	6ta generación: Intel Core i3/i5/i7/i9 (Skylake)	11va generación: Intel Core i3/i5/i7/i9 (Tiger Lake)
RAM	4 GB	8 GB	12 GB
OS	Windows 7 (edición de 32 y 64 bits).	Windows 8/10 (edición de 32 y 64 bits).	Windows 8/10 (edición de 32 y 64 bits).
Almacenamiento	2 GB HDD	4 GB SDD	10 GB SSD
Resolución(píxeles)	1280x800	1440x900	1920x1080
Versión de Java	Java Development Kit (JDK) versión 8	Java Development Kit (JDK) versión 8	Java Development Kit (JDK) versión 8

Nota. La tabla representa los requisitos para usar Android Studio en Windows.

Tabla 2

Tabla de Requisitos de sistema para instalar Android Studio en MAC

	Requisitos Mínimo	**Requisitos Recomendados**	**Requisitos Alto Rendimiento**
CPU	Procesador Intel Core i5-2520M	Procesador Intel Core i5-6267U	Procesador Intel Core i5-1038NG7
RAM	8 GB	8 GB	12 GB
OS	MacOS® 10.14 Mojave (versión 10.14)	MacOS® 11.0 Big Sur (versión 11.0)	MacOS® 12.0 Monterey (versión 12.0)
Almacenamiento	8 GB HDD	8 GB SDD	10 GB SSD
Resolución(píxeles)	1280x800	1440x900	1920x1080
Versión de Java	Java Development Kit (JDK) versión 8	Java Development Kit (JDK) versión 8	Java Development Kit (JDK) versión 8

Nota. La tabla representa los requisitos para usar Android Studio en MAC.

Tabla 3

Tabla de Requisitos de sistema para instalar Android Studio en LINUX

	Requisitos Mínimo	**Requisitos Recomendados**	**Requisitos Alto Rendimiento**
CPU	2da generación: Intel Core i3/i5/i7/i9 (Sandy Bridge)	6ta generación: Intel Core i3/i5/i7/i9 (Skylake)	11va generación: Intel Core i3/i5/i7/i9 (Tiger Lake)
RAM	8 GB	8 GB	12 GB
OS	Ubuntu 18.04 LTS (versión 18.04)	Ubuntu 20.04 LTS (versión 20.04)	Ubuntu 21.04 (versión 21.04)
Almacenamiento	8 GB HDD	8 GB SDD	10 GB SSD
Resolución(píxeles)	1280x800	1440x900	1920x1080
Versión de Java	Java Development Kit (JDK) versión 8	Java Development Kit (JDK) versión 8	Java Development Kit (JDK) versión 8

OS: (DeveloperAndroid, 2023) *"Cualquier distribución Linux de 64 bits compatible con Gnome, KDE o Unity DE; GNU C Library (glibc) 2.31 o posterior."*

2.10. Diseño de la interfaz de Usuario

La interfaz de usuario, comúnmente conocida como GUI (Graphical User Interface en inglés), es una herramienta ampliamente utilizada en nuestro día a día. La encontramos en dispositivos móviles, tabletas y nos permite realizar diversas acciones, como realizar transacciones de pago, entre otras funcionalidades.

Tal como señala Albornoz (2014), La interfaz de usuario desempeña un papel crucial en la competitividad del producto, ya que, si el usuario no puede completar una acción o no comprende la secuencia de pasos necesarios, el producto no tendrá éxito. Es importante tener en cuenta que el diseño de la interfaz gráfica no debe considerarse como una tarea secundaria, ya que tiene un impacto significativo en el rendimiento y la aceptación de una aplicación. En resumen, si el usuario no puede realizar la acción deseada, no encuentra lo que necesita o no le agrada el diseño de la interfaz, el producto fracasará.

2.11. Metodología de Prototipos

La metodología de prototipado es una estrategia ampliamente empleada en el proceso de desarrollo de productos para generar versiones iniciales y exploratorias del mismo. Estos prototipos son una versión simplificada del producto final que permiten a los desarrolladores y clientes evaluar el diseño, la funcionalidad y la usabilidad del producto antes de la producción en masa. Los prototipos se crean rápidamente y se modifican en función de las retroalimentaciones del cliente, esta técnica facilita una mayor optimización en el proceso de desarrollo del producto, lo cual conlleva a una mayor eficacia en su implementación. De acuerdo con Denise & Carmen (2021), "La estrategia de prototipado se encuentra vinculada con la idea de mejora constante y el enfoque de ciclo de Deming, que implica un proceso iterativo centrado en la concepción, implementación, evaluación y ajuste de un plan”. En resumen, la metodología de prototipos es una herramienta útil para los desarrolladores que buscan una forma eficiente y eficaz para desarrollar productos que cumplan con las necesidades y expectativas de los usuarios finales.

CAPÍTULO III: METODOLOGÍA

3. Metodología de desarrollo del plan de tesis

Para lograr una adecuada organización de los distintos entregables del proyecto de desarrollo, es necesario emplear una metodología eficaz en la creación de software.

3.1. Investigación Cualitativa

En el desarrollo de esta tesis, es fundamental comprender las amenazas relacionadas con este tipo de malware. Para ello, se llevará a cabo una entrevista con un experto en seguridad informática que pueda proporcionar información detallada sobre las últimas técnicas utilizadas por los hackers para insertar códigos QR maliciosos en dispositivos Android, así como las consecuencias para los usuarios si escanean un código QR malicioso. Se puede revisar el Anexo A para obtener más información.

Además, el experto nos ha brindado información valiosa sobre las mejores prácticas para proteger a los usuarios de estos riesgos y sobre las características que debería tener una aplicación móvil para detectar y prevenir la presencia de códigos QR maliciosos. A través de esta entrevista, se busca obtener información directa acerca de los desafíos que los usuarios de dispositivos Android enfrentan a diario en cuanto a la seguridad informática, así como las soluciones innovadoras que se están desarrollando para abordar estos riesgos. En definitiva, la entrevista con el experto en seguridad informática será una herramienta clave para asegurar el éxito del proyecto y proporcionar una solución efectiva con el objetivo de salvaguardar la confidencialidad y resguardar la integridad de los datos, garantizando así la privacidad y seguridad de la información de los usuarios al momento de escanear códigos QR.

3.2. Investigación Aplicativa

Durante la fase de Investigación Práctica, se realizará un análisis exhaustivo de la eficacia de la aplicación móvil creada para detectar códigos QR potencialmente dañinos. El objetivo de esta evaluación es analizar la precisión de la aplicación en la identificación de códigos QR maliciosos.

Para llevar a cabo esta evaluación, se realizarán pruebas de la aplicación móvil en diferentes escenarios y se analizarán los resultados obtenidos.

La precisión de la aplicación móvil en la detección de códigos QR maliciosos se puede evaluar utilizando diferentes métodos, algunas de las técnicas más frecuentemente empleadas son las siguientes:

1. Pruebas de laboratorio: se pueden realizar pruebas de laboratorio en las que se utilizan códigos QR maliciosos conocidos y se comprueba si la aplicación móvil detecta y alerta al usuario sobre la presencia de estos códigos maliciosos. Estas pruebas se realizan en un entorno controlado y permiten medir con precisión la efectividad de la aplicación.
2. Comparación con otras aplicaciones: se puede comparar la precisión de la aplicación desarrollada con otras aplicaciones similares en el mercado que tienen la capacidad de detectar códigos QR maliciosos.

La aplicación proporcionará información útil sobre el código QR escaneado y alertará si se trata de un sitio malicioso, asegurando la salvaguardia de la confidencialidad y protección de la información de los usuarios al realizar el escaneo de códigos QR.

En resumen, la Investigación Aplicativa se enfocó en el análisis de la eficacia de la aplicación móvil en términos de rendimiento y resultados obtenidos en la detección de códigos QR maliciosos, analizando su precisión en diferentes escenarios y asegurando la salvaguardia de la confidencialidad y resguardo de los datos personales de los usuarios al momento de interactuar con la aplicación al momento de escanear códigos QR.

3.3. Metodología de desarrollo de software

La metodología de prototipado es una aproximación que se basa en el enfoque iterativo del desarrollo de software, destacando la importancia de crear versiones preliminares o prototipos de la aplicación con el fin de obtener comentarios anticipados por parte de los usuarios y clientes.

El objetivo principal es identificar y corregir problemas de usabilidad, funcionalidad y rendimiento del software antes de su lanzamiento.

En este caso, la metodología de prototipos se aplicará para el desarrollo de una aplicación que detecte y prevenga la presencia de códigos QR maliciosos. El equipo de desarrollo trabajará en estrecha colaboración con los usuarios y clientes para crear prototipos de la aplicación y recibir retroalimentación de ellos. Las iteraciones se centrarán en la mejora de la funcionalidad y la eficacia del software.

La metodología de desarrollo iterativo permitirá al equipo de desarrollo adaptarse rápidamente a los cambios del proyecto y crear una aplicación de alta calidad. La implementación de pruebas de integración ayudará a garantizar que la aplicación sea robusta y funcione correctamente. La metodología de prototipado es una estrategia efectiva para desarrollar una aplicación de gran utilidad y calidad, diseñada para satisfacer las necesidades de los usuarios y prevenir la presencia de códigos QR maliciosos.

3.3.1. Modelo que aplica

Según el sitio web Startup_Guide_IONOS (2020), la metodología de desarrollo iterativo se compone de cuatro etapas clave: planificación, diseño, construcción y evaluación. Cada fase se divide en múltiples iteraciones, en las cuales se crean versiones del software cada vez más refinadas y mejoradas. La metodología de prototipos se enfoca en crear prototipos o modelos iniciales de la aplicación para obtener retroalimentación temprana de los usuarios y clientes, lo que permite identificar y corregir problemas de usabilidad, funcionalidad y rendimiento del software antes de su lanzamiento.

CAPÍTULO IV: DESARROLLO DE LA INVESTIGACIÓN

4. Desarrollo de la Metodología de Prototipos

El propósito de este capítulo es brindar una descripción del proceso de desarrollo del aplicativo móvil mediante el uso de la metodología de prototipos. A continuación, se presentarán las etapas y fases del proceso de desarrollo de software, desde la identificación de requisitos hasta la implementación de la aplicación. También se abordarán las responsabilidades de los miembros del equipo de desarrollo. Este enfoque permitirá cumplir con los objetivos planteados y lograr una aplicación móvil de alta calidad que satisfaga las necesidades de los usuarios.

4.1. Asignación de Roles

En la metodología de prototipos, resulta fundamental asignar roles a los involucrados en el proyecto. La asignación de roles se detalla en la tabla 4.

Tabla 4

Tabla de Roles

Rol	Responsable
Líder del proyecto:	Ing. Herrera Nelson
Diseñador:	Erick Vega
Desarrollador:	Erick Vega
Tester:	Erick Vega

4.2. Creación de Área del Trabajo

Se empleará una herramienta en línea, específicamente la plataforma de prototipado Asana, para colocar los diseños y prototipos. Esto permitirá al equipo visualizar y evaluar la interfaz de usuario de manera efectiva. Asana se destaca por su facilidad de uso y su capacidad de colaboración en tiempo real. En la figura 2 se muestra la creación de este tablero.

El Área de trabajo consta de 3 partes:

- Pendiente de revisión • En revisión • Finalizados

Figura 2

Creación del tablero en Asana

Nota. Tablero de Asana para el control de desarrollo.

4.3. Clasificación de Actividades Kanban

Una vez que se hayan definido las tareas a realizar, se utilizarán etiquetas de colores en el campo "Actividades" de Asana para identificar qué tareas corresponden a cada etapa del proceso de desarrollo. Por ejemplo, las tareas relacionadas con "Prototipos" se pueden etiquetar con color naranja, mientras que las tareas de "Prototipado de la Interfaz de Usuario" pueden tener una etiqueta verde. La Figura 3 muestra una vista general de esta organización, lo cual facilitará una mejor organización y seguimiento de las actividades en el flujo de trabajo.

Figura 3

Clasificación de Actividades

Nota. Etiquetas con colores para un mejor control de actividades.

4.4. Clasificación de Requerimientos

Identificar y establecer los requerimientos necesarios para desarrollar el prototipo es fundamental. En este caso, se aplicó el enfoque del análisis comparativo de software para recopilar los requisitos específicos de aplicaciones capaces de escanear códigos QR y documentos académicos.

Para recopilar estos requisitos, se llevó a cabo un análisis comparativo de software, el cual brinda información valiosa para identificar los requerimientos necesarios en el desarrollo del prototipo. Al examinar y probar diversas aplicaciones, se logró elaborar una lista de requerimientos que deben cumplirse para el escaneo de códigos QR y documentos académicos. Estos requisitos servirán como base para el diseño del prototipo y la posterior implementación de la solución.

La Tabla 5 presenta un conjunto de aplicaciones que han servido como guía para la construcción y la recopilación de requisitos de la aplicación funcional.

En la Tabla 6 se han recuperado algunas de las características más importantes relacionadas con la interfaz de usuario, basadas en diversas aplicaciones disponibles en el mercado.

La Tabla 7 recopila información sobre la compatibilidad y actualización de las distintas aplicaciones, con el fin de realizar un control basado en los dispositivos disponibles.

Tabla 5

Aplicaciones elegidas para su análisis y sus funcionalidades.

App	Google Lens	QR Code Scanner Reader Xiaomi	Camera Xiaomi	Camera Huawei	Camera Samsung	Google Camera
Versión	16.0.16130.20128	13.22202.17	4.3.004180.5	11.0.1.350	13.0.01.7	8.8.224.514217832.10
Funcionalidades	-Escaneo de códigos QR y códigos de barras. -Reconocimiento de objetos, lugares, plantas y animales. -Escaneo y conversión de tarjetas de presentación y notas. -Escaneo de códigos QR de WiFi y eventos. -Edición de imágenes escaneadas y resaltado de texto.	- Escaneo de códigos QR -Soporte para diferentes códigos QR. - Escaneo de documentos. -Compartir Información escaneada:	-Modo manual. -Modo de grabación de video. -Modo de cámara frontal. -Modo de escaneo de códigos QR. -Modo de escaneo de documentos.	-Modo foto estándar. -Modo de escaneo de documentos. -Modo documento. -Modo de detección de objetos. -Modo de escaneo. -Modo profesional.	-Escaneo de códigos QR con la cámara. -Reconocimiento de texto para escanear documentos y convertirlos en texto editable. -Detección de bordes para escanear documentos y recortarlos automáticamente. -Ajuste de color, brillo y contraste para mejorar la legibilidad de los documentos escaneados.	-Escaneo de códigos QR y códigos de barras. -Detección automática de código QR. -Escaneo de documentos y fotos de documentos. -Recorte y ajuste de la imagen del documento. -Edición de documentos con opciones de eliminación de sombras. -Almacenamiento de documentos escaneados en la galería de imágenes del dispositivo. -Compartir documentos escaneados a través de aplicaciones de mensajería o correo electrónico.

Nota. La tabla presenta información esencial de distintas aplicaciones que se analizaron para la obtención de requerimientos.

Tabla 6

Características remarcables de la Interfaz de usuario.

App	Google Lens	QR Code Scanner Reader Xiaomi	Camera Xiaomi	Camera Huawei	Camera Samsung	Google Camera
Interfaz de usuario	Botón de exploración en la parte inferior de la interfaz.	Escaneo rápido y preciso de códigos QR y de barras.	Captura de fotografías y videos de excelente calidad.	Selección de modo de cámara mediante desplazamiento lateral.	Captura de fotos y videos en alta resolución y calidad.	Captura rápida de fotos y videos de alta calidad.
	Cámara para escanear documentos y códigos QR.	Generación de códigos QR personalizados.	Acceso rápido a la cámara a través del botón de inicio.	Botón de captura para tomar fotos y grabar videos.	Configuración de modo de disparo y filtros de imagen.	Control manual de exposición y enfoque para ajustes precisos.
	Visualización de información relacionada con el contenido escaneado.	Acceso rápido a información escaneada.	Visualmente atractiva: diseño moderno y limpio.	Funciones avanzadas de edición de fotos.	Control manual de ajustes de enfoque y exposición.	Muestra información los niveles de exposición de la imagen.
	Identificación de texto en documentos escaneados.	Acceso rápido a configuraciones y ajustes avanzados.	Personalizable adaptable a las preferencias del usuario.	Enfoque táctil en la pantalla.	Visualización en tiempo real de la imagen a capturar.	Zoom óptico para acercar y alejar imágenes.
	Conversión de imágenes a archivos PDF.	Códigos QR en imágenes almacenadas en el dispositivo	Detección de escenas y ajustes de configuración.	Visualización de la línea de cuadrícula para la composición de la imagen.	Controles de cámara fácilmente accesibles y personalizables.	Diseño actualizado con elementos visuales modernos y elegantes.
	Acceso rápido a funciones principales con un solo toque.	Interfaz intuitiva con fácil acceso a funciones adicionales	Acceso a todas las funciones y ajustes de la cámara.	Acceso directo a la galería de fotos y videos guardados.	Acceso rápido a la galería de imágenes capturadas.	Acceso directo a Google Lens para identificar textos.

Nota. Esta tabla muestra características más importantes que se identificaron para la interfaz de usuario.

Tabla 7

Compatibilidad de la aplicación con los distintos dispositivos y Fecha de la última actualización.

App	Google Lens	QR Code Scanner Reader Xiaomi	Camera Xiaomi	Camera Huawei	Camera Samsung	Google Camera
Compatibilidad	Samsung Galaxy S, Note, A, J, M y Tab líneas de dispositivos Google Pixel, Pixel 4a, Pixel 4a (5G), Pixel 5 LG G, Q series Motorola Moto G, E, X, Z y One series Sony Xperia X, XZ, XA, L y M series Xiaomi Mi, Redmi y Poco series Huawei Mate, P, Y y Nova series OnePlus 7, 7T, 8, 8T, 9, 9 Pro	Xiaomi Mi 11, Mi 11 Lite, Mi 11 Ultra, Mi 10, Mi 10T, Mi 10T Pro, Mi 10i, Mi Note 10 Xiaomi Redmi Note 10, Redmi Note 10 Pro, Redmi Note 9, Redmi 9, Redmi 9 Power, Redmi 9A, Redmi 9C Xiaomi Poco X3, Poco X3 Pro, Poco F3	Xiaomi Mi 11, Mi 11 Lite, Mi 11 Ultra, Mi 10, Mi 10T, Mi 10T Pro, Mi 10i, Mi Note 10 Xiaomi Redmi Note 10, Redmi Note 10 Pro, Redmi Note 9, Redmi 9, Redmi 9 Power, Redmi 9A, Redmi 9C Xiaomi Poco X3, Poco X3 Pro, Poco F3	Huawei P40, P40 Pro, P40 Pro+, P30, P30 Pro, P20, P20 Pro-Huawei Mate 40, Mate 40 Pro, Mate 40 Pro+, Mate 30, Mate 30 Pro, Mate 20, Mate 20 Pro-Huawei Nova 8, Nova 8 Pro, Nova 7, Nova 7 Pro, Nova 6, Nova 6 SE	Samsung Galaxy S21, S21 Plus, S21 Ultra, S20, S20 Plus, S20 Ultra, S10, S10 Plus, S10e Samsung Galaxy Note 20, Note 20 Ultra, note 10, Note 10 Plus Samsung Galaxy A71, A51, A50, A31, A21s, A20	Google Pixel, Pixel 2, Pixel 3, Pixel 4 Samsung Galaxy S10, S10+, S10e, S9, S9+, S8, S8+ OnePlus 7, 7 Pro, 6, 6T, 5, 5T Xiaomi Pocophone F1, Redmi Note 7, Redmi Note 8 Pro, Mi 9 LG G7 ThinQ, V30, V40 Huawei P30 Pro, Mate 20 Pro, Mate 30 Pro
Fecha última Actualización	23 de marzo de 2022	26 de enero de 2022	18 de marzo de 2022	9 de marzo de 2022	22 de marzo de 2022	7 de diciembre de 2021

Nota. Esta tabla muestra varios dispositivos en los cuales está disponible las distintas aplicaciones y su fecha de actualización.

En base a las tablas presentadas los requisitos identificados son los siguientes:

Requisitos críticos:

- Pantalla de Inicio.
- Registro de Documentos Escaneados y reconocer códigos QR maliciosos de forma eficiente y confiable.
- Protección de los datos del usuario escaneados y creados mediante medidas de seguridad adecuadas.
- Almacenamiento de Información de documentos escaneados, tarjetas, etc., en la base de datos interna.
- La aplicación debe proporcionar opciones para compartir los documentos escaneados de manera segura.

Requisitos deseables:

- Barra de Búsqueda de Documentos.
- La aplicación debe permitir escanear documentos, incluyendo agregar notas y edición de documentos.
- Despliegue de una side Activity para generar códigos QR de texto, SMS y URL.
- La aplicación debe permitir al usuario ver la política de privacidad de la aplicación y calificar la aplicación.
- Identificar texto en imágenes escaneadas.

Requisitos Opcionales:

- La aplicación debe ser rápida y eficiente al escanear códigos QR y crear documentos.
- La aplicación debe ser compatible con diferentes dispositivos y versiones de Android.

- La aplicación debe tener opciones de personalización adicionales para la interfaz de usuario, como cambiar el tema.

4.5. Requerimientos Generales

Los requisitos generales del aplicativo se detallan en la Tabla 8 y estos deben ser considerados al momento de definir las funcionalidades que se van a desarrollar.

Tabla 8

Requerimientos generales del aplicativo

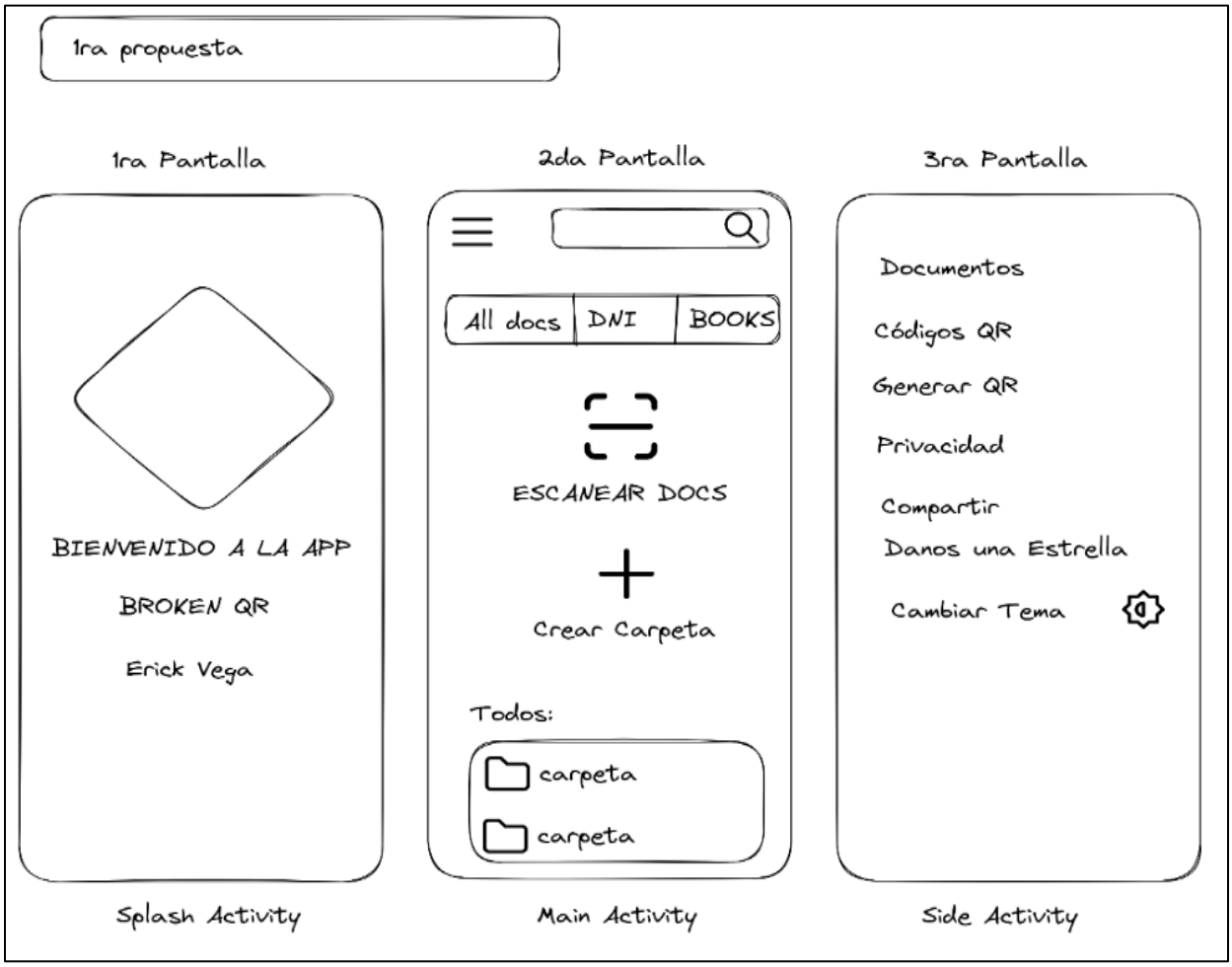
1	Se solicita la implementación de una aplicación móvil que permita escanear documentos, CI, libros y otras imágenes.
2	Se solicita que la aplicación tenga un lector de códigos QR que sea capaz de identificar códigos QR maliciosos.
3	Se solicita la inclusión de un generador de códigos QR en la aplicación, que permita a los usuarios crear sus propios códigos QR.
4	Se solicita la implementación de una funcionalidad de política de privacidad en la aplicación, que permita a los usuarios leer la política de privacidad. Dicha política de privacidad deberá estar en un servicio gratuito de alojamiento y se podrá acceder por la web.
5	Se solicita la implementación de una función de compartir aplicación, que permita a los usuarios compartir la aplicación con otros usuarios.
6	Se solicita la inclusión de una función de valoración en la aplicación, que permita a los usuarios dar su opinión y valorar la aplicación.
7	Se solicita la implementación de una función para cambiar el tema de la aplicación, que permita a los usuarios personalizar la apariencia y funcionalidad de esta.
8	Se solicita que la aplicación cuente con una función de compartir, que permita a los usuarios compartir con otros los documentos escaneados.

4.6. Prototipado de la Interfaz de Usuario

Para el prototipo de la interfaz se usó la herramienta: Excalidraw, esta herramienta nos permite tener un diseño inicial de la aplicación móvil además se tomó de TablerIcons.com varios de los iconos que se usaron para tener una idea inicial se puede observar con detalle en la Figura 4 y en la Figura 5.

Figura 4

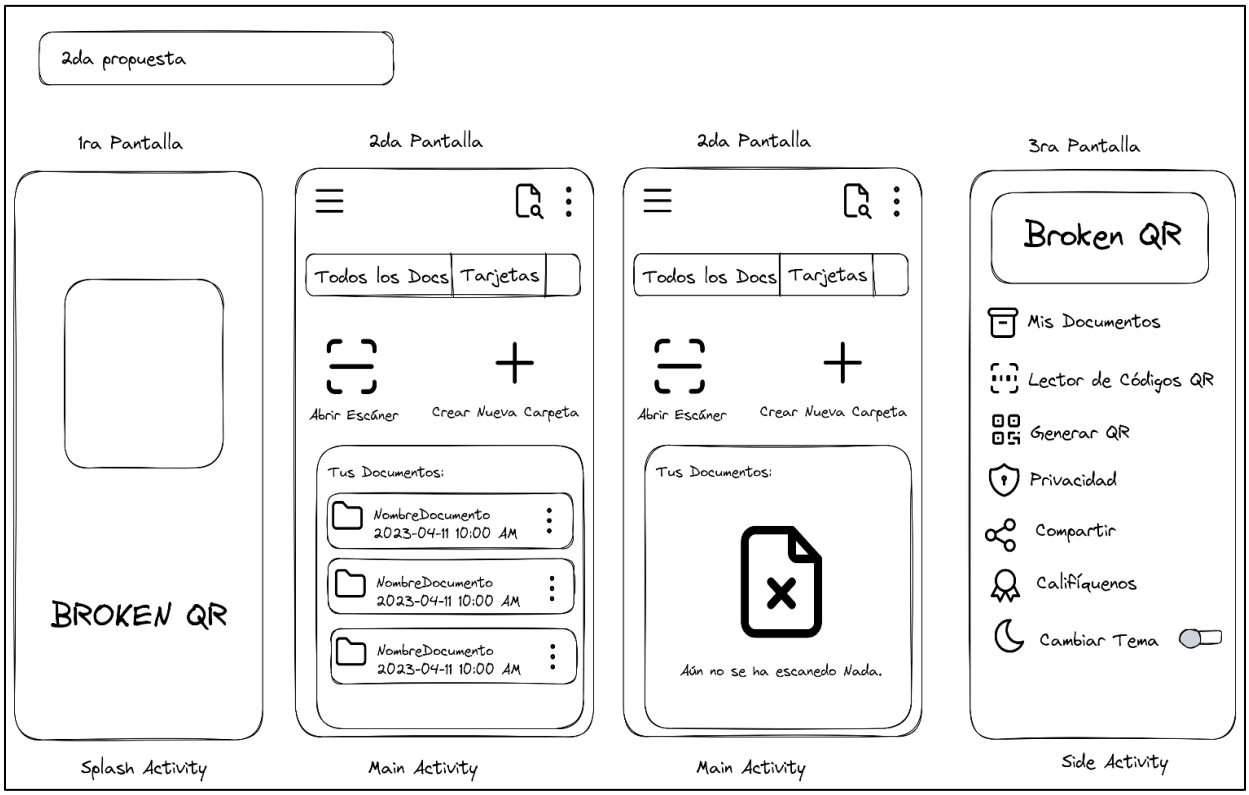
Diseño del prototipo de la Interfaz de Usuario Propuesta 1



Nota. El gráfico representa una de las propuestas en un diseño preliminar.

Figura 5

Diseño del prototipo de la Interfaz de Usuario Propuesta 2



Nota. El gráfico representa una de las propuestas en un diseño preliminar que se usó para el desarrollo de esta aplicación.

Se han desarrollado los primeros diseños de los prototipos de las pantallas iniciales, con el objetivo de resaltar las principales pantallas que captarán la atención del usuario. Para lograr esto, se ha buscado una estética visual atractiva y clara, utilizando iconos fácilmente reconocibles. En la primera propuesta (Figura 4), se incluyeron varios elementos llamativos, pero esta interfaz solo sirvió como un boceto inicial para mejorar y perfeccionar la segunda propuesta (Figura 5). En esta última versión, se han incorporado elementos de la propuesta anterior, pero con un diseño más pulido y una arquitectura de aplicación más definida.

4.7. Funcionalidades de la Aplicación

El aplicativo se divide en funcionalidades clasificadas en dos grupos: las ejecutadas por el sistema, y las realizadas por el cliente. Estas funcionalidades se detallan en la Tabla 9 y la Tabla 10.

Tabla 9

Funcionalidades Ejecutadas por el sistema

Funcionalidad:	Nombre:	Descripción:
FS1	Escaneo de documentos	El sistema permitirá escanear documentos, CI, libros y otros elementos utilizando la cámara del dispositivo móvil.
FS2	Identificación de códigos QR maliciosos	El sistema incluirá un lector de códigos QR que identificará códigos QR maliciosos para proteger la seguridad del usuario.
FS3	Almacenamiento de documentos escaneados	La aplicación utilizará el almacenamiento local del dispositivo para guardar los documentos escaneados.
FS4	Editar Fotos	La aplicación permite a los usuarios editar y retocar las imágenes de los documentos escaneados utilizando una biblioteca de edición de fotos integrada en la aplicación. Se pueden ajustar la calidad y legibilidad del documento escaneado con brillo, saturación, contraste, exposición, recorte y otros efectos visuales.
FS5	Escaneo de documentos	El sistema permitirá escanear documentos, CI, libros y otros elementos utilizando la cámara del dispositivo móvil.

Nota. La tabla presenta las funcionalidades de una aplicación móvil de manera sistemática.

Tabla 10

Funcionalidades Ejecutadas por el Cliente

Funcionalidad:	Nombre:	Descripción:
FC1	Escanear Documentos	El usuario podrá escanear documentos, CI, libros y otros elementos utilizando la cámara del dispositivo móvil.
FC2	Lector de códigos QR	El usuario podrá escanear códigos QR utilizando la cámara del dispositivo móvil.
FC3	Generador de códigos QR	El usuario podrá crear sus propios códigos QR utilizando la aplicación.
FC4	Lectura de política de privacidad	El usuario podrá leer la política de privacidad a través de la funcionalidad implementada en la aplicación.
FC5	Compartir aplicación	El usuario podrá compartir la aplicación con otros usuarios a través de distintos medios.
FC6	Valoración de la aplicación	El usuario podrá dar su opinión y valorar la aplicación a través de una función implementada en la aplicación.
FC7	Personalización de la apariencia	El usuario podrá cambiar el tema de la aplicación con un botón.
FC8	Compartir documentos escaneados	El usuario podrá compartir los documentos escaneados utilizando la función de compartir implementada en la aplicación.

Nota. La tabla presenta las funcionalidades de una aplicación móvil que pueden ser ejecutadas por el cliente.

4.7.1. Mejora de Seguridad en el escaneo de Códigos QR

Con el objetivo de garantizar una experiencia de escaneo más segura, se han implementado medidas adicionales en la actividad de escaneo de códigos QR en Android. Estas mejoras se han realizado utilizando las bibliotecas ZXingScannerView y OkHttp.

En primer lugar, se han realizado modificaciones en el código para introducir una verificación de seguridad adicional en los enlaces escaneados. Antes de abrir el enlace en el navegador, se lleva a cabo una verificación utilizando la biblioteca OkHttp. Durante este proceso, se examinan los encabezados de seguridad de la respuesta obtenida, en particular el encabezado “Strict-Transport-Security”. La presencia de estos encabezados se considera un indicador de seguridad en el enlace escaneado.

En función de los resultados de esta verificación, se muestra un mensaje informativo al usuario. Si se detecta la presencia de encabezados de seguridad, se indica que el enlace parece ser seguro y se procede a abrirlo en el navegador. Por el contrario, si no se encuentran encabezados de seguridad o se detectan anomalías, se recomienda al usuario no abrir el enlace debido a posibles riesgos de seguridad.

Estas mejoras tienen como finalidad verificar los enlaces antes de abrirlos en el navegador, con el propósito de prevenir la exposición a sitios web potencialmente maliciosos y proteger la privacidad y seguridad de los usuarios.

4.8. Desarrollo Interfaces

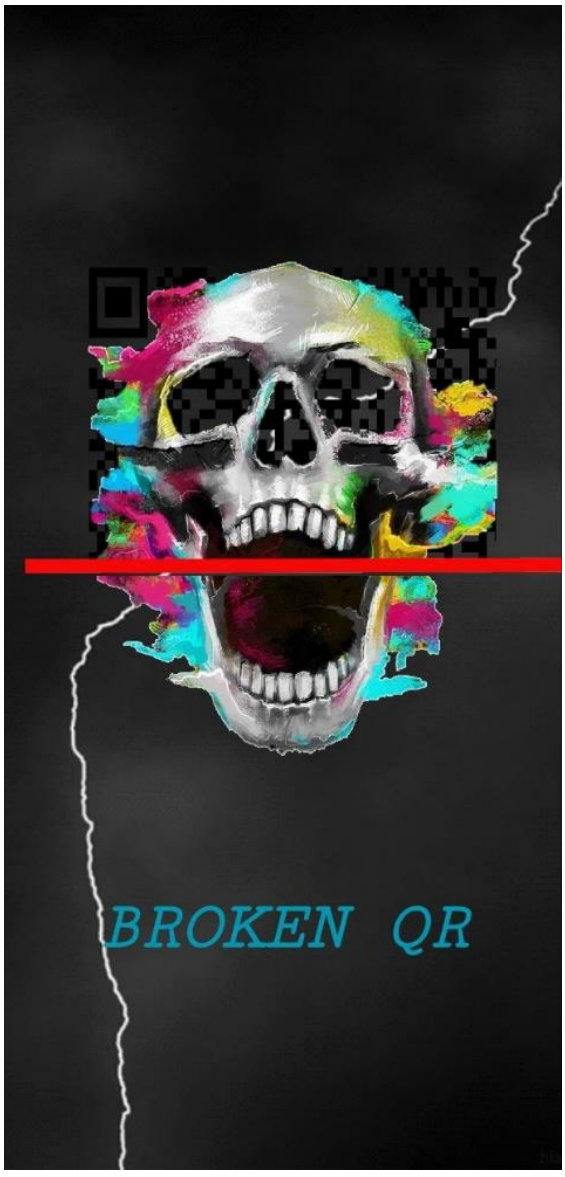
En la etapa de desarrollo de interfaces, se trabaja en la creación y diseño de las diferentes pantallas y elementos visuales que conforman la aplicación. Cada interfaz tiene como objetivo proporcionar al usuario una experiencia intuitiva, atractiva y funcional.

4.8.1. Pantalla de Inicio

La pantalla de inicio es la interfaz principal que el usuario encuentra al iniciar la aplicación. Su diseño y contenido se han cuidado especialmente para captar su atención de manera efectiva y mantener su interés desde el primer momento. Para lograr esto, se ha utilizado un GIF y una imagen llamativa que generan impacto visual. En la Figura 6 se observa una vista preliminar de esta pantalla.

Figura 6

Pantalla de Inicio

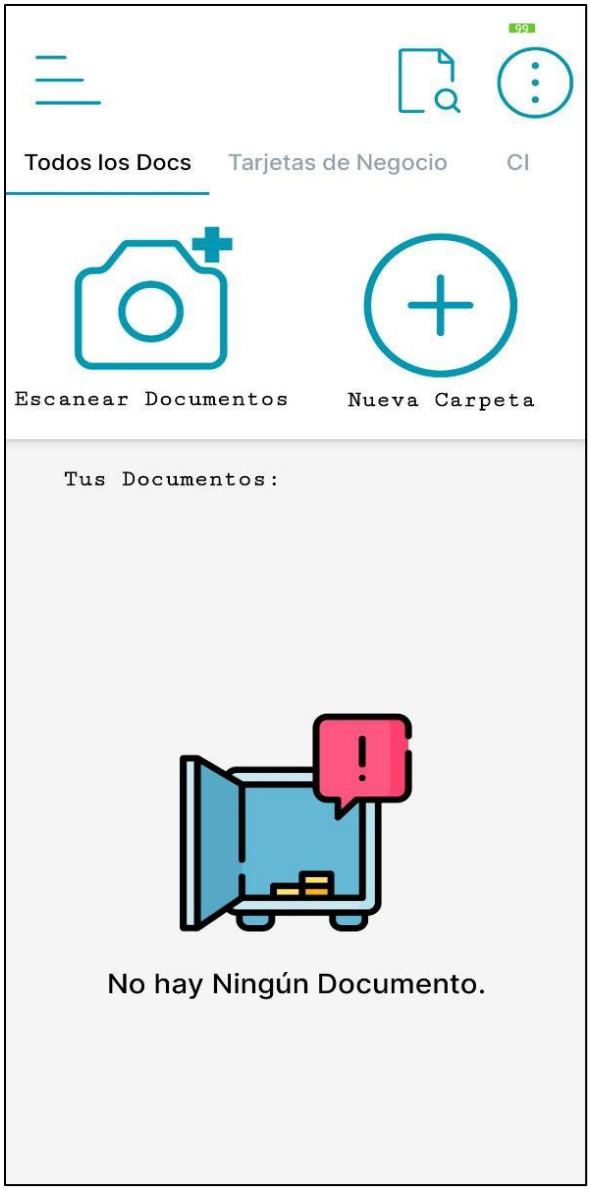


4.8.2. Pantalla Principal

Esta pantalla representa la segunda interfaz que el usuario visualiza al iniciar la aplicación, y en ella se mostrarán todos los documentos escaneados. En la Figura 7, se ven botones que permiten al usuario buscar, ordenar y compartir documentos y cambiar la vista de visualización. Además, se ha incluido un botón que despliega las actividades adicionales que se pueden realizar dentro de la aplicación.

Figura 7

Pantalla Principal

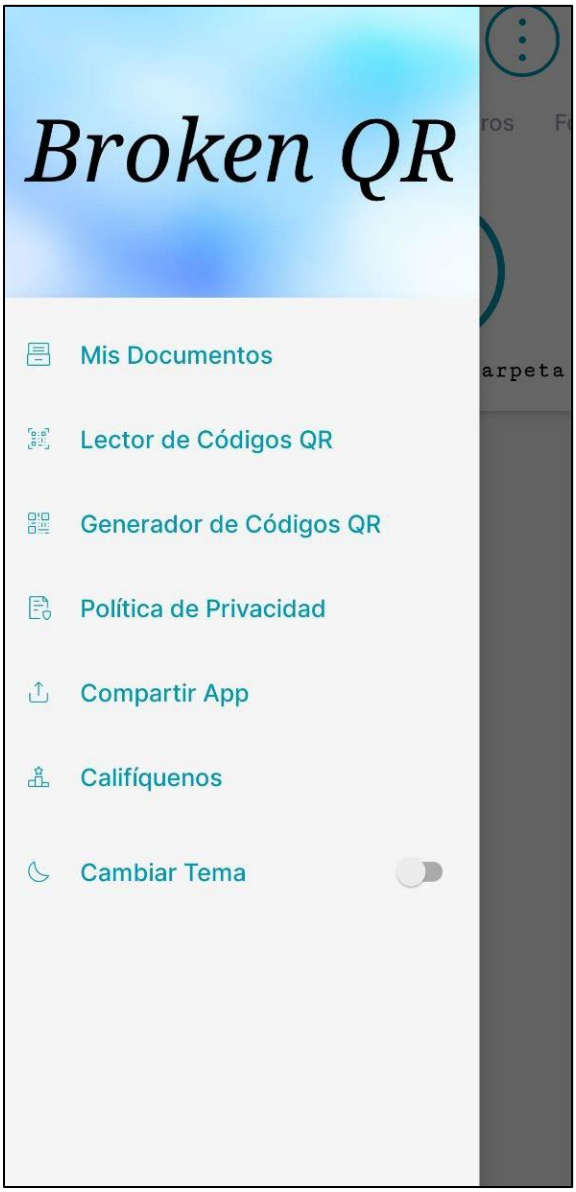


4.8.3. Pantalla Actividades Adicionales

La Pantalla de Actividades Adicionales se muestra cuando el usuario presiona un icono ubicado en la pantalla principal. Aquí se presentan varias funcionalidades que pueden ser seleccionados por el usuario. En la Figura 8 se presenta una representación funcional de esta pantalla.

Figura 8

Pantalla Actividades Adicionales

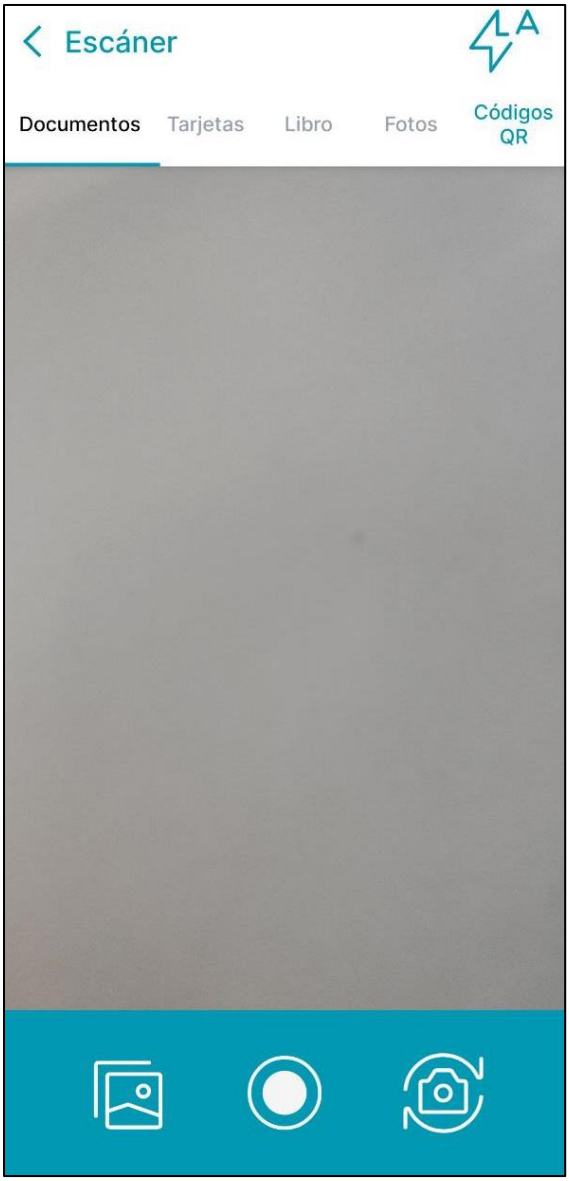


4.8.4. Pantalla del Escáner

En esta pantalla podemos encontrar el escáner de documentos, tarjetas, libros, fotos y códigos QR en la Figura 9 tenemos una vista general.

Figura 9

Pantalla del Escáner

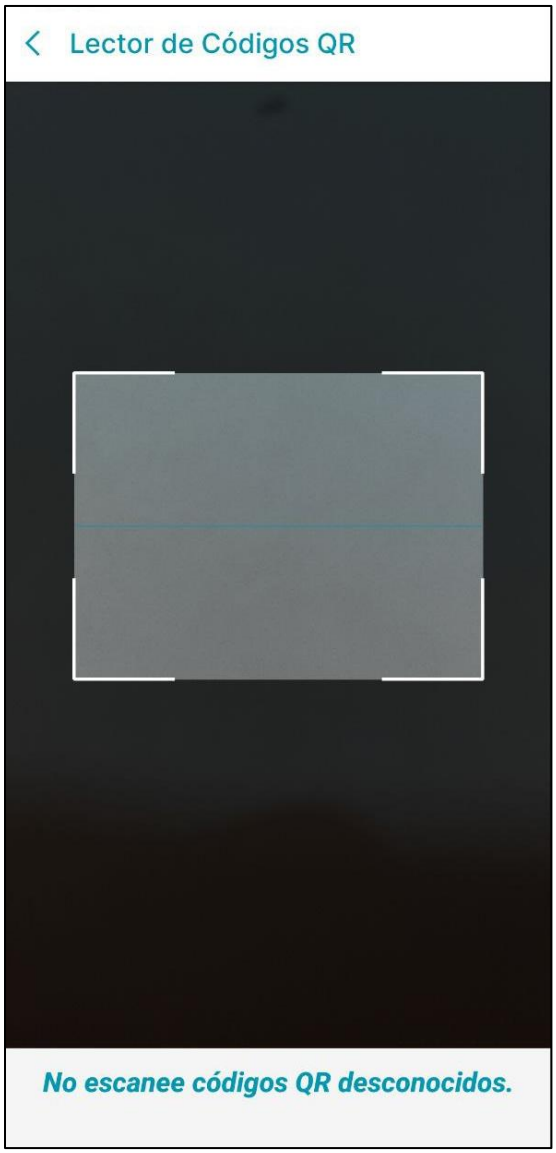


4.8.5. Lector de Códigos QR

En esta pantalla, se encuentra el lector de códigos QR junto con una sección que ofrece consejos disponibles dentro de la aplicación. En la Figura 10 se puede apreciar una vista funcional del lector.

Figura 10

Pantalla Lector de Códigos QR



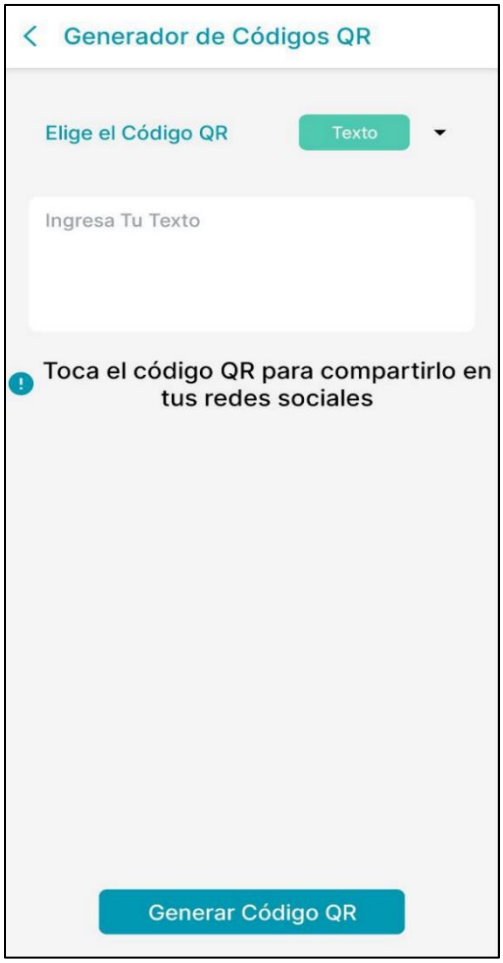
En esta situación, la aplicación exhibirá recomendaciones importantes para la exploración de códigos QR cada 5 segundos. Estos consejos son fundamentales para garantizar una correcta lectura de los códigos y asegurar una experiencia óptima para el usuario. Se puede revisar el Anexo B, Figura 23 para obtener más información.

4.8.6. Generador de Códigos QR

En esta pantalla, se encuentra el generador de códigos QR, que crea diferentes tipos de códigos QR. Además, si se desea compartir el código QR en alguna red social, simplemente se debe tocar el código QR. En la Figura 12 se puede apreciar una vista funcional de la pantalla para generar Códigos QR.

Figura 11

Pantalla Generador de Códigos QR

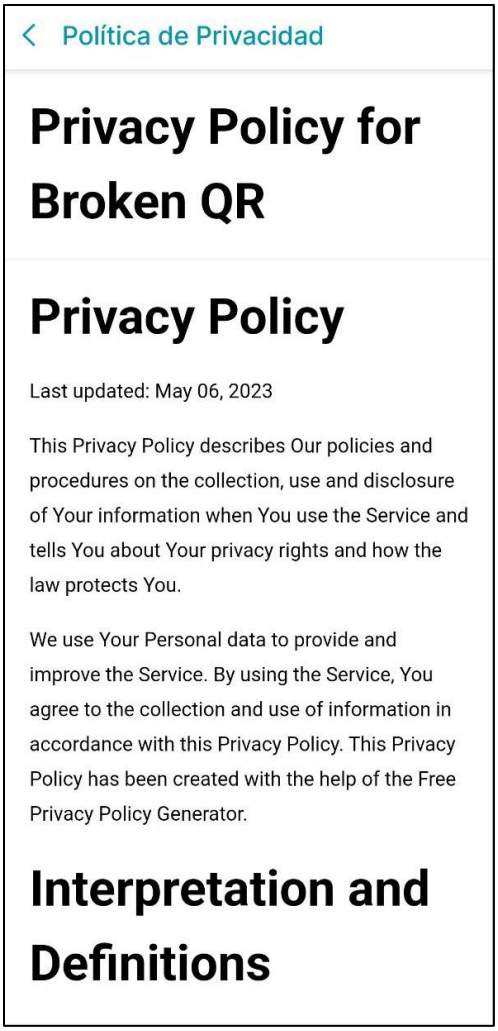


4.8.7. Política de Privacidad

En esta pantalla, se presenta la política de privacidad de la aplicación Broken QR, actualizada hasta el 6 de mayo de 2023. En la Figura 13 se puede apreciar una vista funcional de la pantalla sobre las políticas de privacidad para esta aplicación.

Figura 12

Pantalla Política de Privacidad para Broken QR



4.9. Diseño de la Base de Datos

El diseño de la base de datos es crucial en cualquier aplicación. En este sentido, se ha desarrollado la tabla "all docs" para su utilización durante el escaneo de documentos. Para obtener más información al respecto, se puede consultar el Anexo C, Figura 24 y tabla 16.

La tabla 16 almacena información relacionada con los documentos individuales de un grupo. Entre los datos que se registran se encuentran el nombre de la imagen, la nota asociada a la imagen y la ruta de acceso a la imagen.

CAPÍTULO V: IMPLEMENTACIÓN

5. Implementación de la aplicación

La ejecución de una aplicación es una etapa crucial en cualquier proyecto de desarrollo de software. En el contexto de la aplicación móvil creada para la identificación y mitigación de códigos QR maliciosos en dispositivos Android, la implementación se enfoca en transformar el diseño y los requisitos de la aplicación en un producto operativo y preparado para su utilización. Durante este periodo, se realiza la codificación, la combinación de distintos elementos y la ejecución de rigurosas pruebas para asegurar la excelencia y eficacia de la aplicación.

Es importante resaltar que la ejecución cumple una función esencial en el cumplimiento de los objetivos particulares y generales del proyecto, así como en la protección de la confidencialidad y resguardo de los usuarios al escanear códigos QR.

5.1. Implementación de la tesis

La implementación de la tesis constituye una de las fases más significativas dentro del proceso de investigación, ya que en este momento se materializan todas las ideas y soluciones propuestas en la teoría. En este contexto, el propósito es realizar de manera eficiente la concepción y elaboración de la aplicación móvil orientada a la identificación y mitigación de códigos QR maliciosos en dispositivos Android, y evaluar su efectividad en la identificación de los riesgos asociados. Para alcanzar dicho objetivo, se harán variadas pruebas rigurosas que verificarán la calidad y funcionalidad de la aplicación, garantizando simultáneamente la salvaguardia de la confidencialidad y resguardo de los usuarios al efectuar el escaneo de códigos QR.

5.2. Pruebas funcionales

Dentro del proceso de desarrollo de software, las pruebas de funcionalidad juegan un rol esencial al asegurar que el resultado definitivo se ajuste a los requerimientos y especificaciones establecidas.

Estas pruebas se enfocan en verificar el adecuado rendimiento y desempeño de las funciones y atributos del software, garantizando de esta forma su correcto funcionamiento conforme a lo esperado. Son una etapa clave que asegura la calidad y la correcta implementación del software en el cumplimiento de su propósito.

En este sentido, las pruebas funcionales son una parte integral del proceso de calidad del software, ya que permiten identificar errores y problemas en el software antes de que se implemente.

5.3. Pruebas no funcionales

En el desarrollo de software, las pruebas no funcionales son igualmente importantes que las pruebas funcionales, ya que evalúan aspectos críticos del software que no están directamente relacionados con su funcionamiento operativo. Estas pruebas no funcionales abordan áreas clave como la eficiencia, seguridad, usabilidad y compatibilidad del software. Son elementos fundamentales para garantizar que el producto cumpla con los criterios y requisitos definidos en relación con el rendimiento, seguridad, experiencia del usuario y compatibilidad con diversas plataformas y dispositivos.

5.3.1. Pruebas de carga

Las pruebas de carga juegan un papel esencial en el proceso de desarrollo de aplicaciones y sitios web, ya que permiten analizar la capacidad de respuesta de un sistema frente a situaciones de carga intensa. Estas pruebas simulan el comportamiento de múltiples usuarios accediendo al sistema a la vez, con el fin de identificar posibles cuellos de botella, problemas de rendimiento y otros inconvenientes que podrían afectar la experiencia del usuario. En este apartado, se abordarán los aspectos clave a considerar en las pruebas de carga, así como las herramientas y técnicas más utilizadas en este proceso. El objetivo es garantizar un rendimiento óptimo del sistema y brindar una experiencia fluida al usuario en condiciones de uso intensivo.

5.4. Resultados

Se utilizó la tabla 12 para las pruebas funcionales, la tabla 13 para las pruebas no funcionales y la tabla 14 para realizar pruebas de carga. Estas pruebas permiten evaluar si la aplicación presenta fallos o si existen acciones que puedan afectar su funcionamiento y la experiencia del usuario. Las tablas se crearon con el propósito de determinar la respuesta del sistema después de llevar a cabo una acción específica. La utilización de estas pruebas proporciona una estructura sistemática para registrar y analizar los resultados obtenidos, lo cual facilita la detección de posibles fallos y la toma de decisiones para mejorar tanto el rendimiento como la calidad del software.

Tabla 11

Pruebas Funcionales

Caso de prueba	Descripción	Entrada	Resultado esperado	Resultado real	¿Pasa la prueba?
CP-001	Verificar si se puede instalar la aplicación.	El archivo de instalación de la aplicación (APK)	La aplicación se instala correctamente.	**Figura 15**	Pasó
CP-002	Verificar si la aplicación se puede abrir.	El ícono representativo de la aplicación en el dispositivo	La pantalla principal de la aplicación se muestra sin errores.	**Figura 16**	Pasó
CP-003	Verificar si se puede escanear un código QR válido.	Código QR válido.	El contenido del código QR se muestra correctamente en la pantalla.	**Figura 17**	Pasó
CP-004	Verificar si la aplicación maneja correctamente un código QR Malicioso.	Código QR inválido.	Se muestra un mensaje de alerta indicando que el código QR escaneado no es válido.	**Figura 18**	Pasó
CP-005	Verificar si la aplicación maneja correctamente un código QR Normal.	Código QR válido.	Se muestra un mensaje de explicando que el código QR es válido.	**Figura 19**	Pasó

Nota. Se muestran las pruebas funcionales que se realizaron.

Tabla 12

Pruebas No Funcionales

Caso de prueba	Tipo de Prueba	Resultado esperado	Resultado real	¿Pasa la prueba?
CPNF-001	Compatibilidad	Capacidad de la aplicación para funcionar en diferentes dispositivos con diferentes versiones de Android.	**Tabla 14**	Pasó
CPNF - 002	Seguridad	Capacidad de la aplicación para funcionar en diferentes dispositivos con diferentes parches de Seguridad.	**Tabla 14**	Pasó
CPNF - 003	Disponibilidad	Capacidad de la aplicación para estar disponible para los usuarios en todo momento	**Figura 20**	Pasó

Nota. Se muestran las pruebas no funcionales que se realizaron.

Tabla 13

Pruebas de Carga

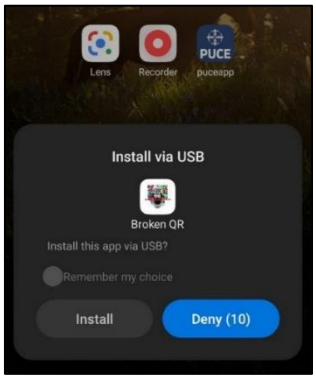
Caso de prueba	Acción realizada	Resultado esperado	¿Pasa la prueba?
CPC-001	Escaneo de múltiples códigos QR consecutivamente.	La aplicación debe tener la capacidad de gestionar la carga de trabajo y completar el procesamiento de cada código QR en un tiempo inferior a 5 segundos.	Pasó
CPC -002	Escaneo de un código QR pequeño y digitalización de un documento pequeño (1 página)	La aplicación debe ser capaz de escanear y procesar el código QR, así como digitalizar el documento en menos de 5 segundos.	Pasó
CPC -003	Escaneo de múltiples códigos QR consecutivamente y digitalización de múltiples documentos	La aplicación debe ser capaz de gestionar eficientemente la carga de trabajo, procesando cada código QR y digitalizando cada documento en menos de 15 segundos.	Pasó

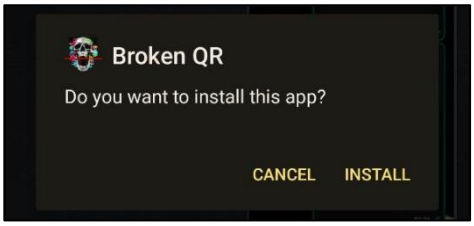
Nota. Se muestran las pruebas de carga que se realizaron en el aplicativo.

La Figura 15 muestra la instalación de la aplicación en un dispositivo Android, con la presentación de 2 cuadros de diálogo adaptados a la versión de Android 12.

Figura 13

Instalación de la Aplicación

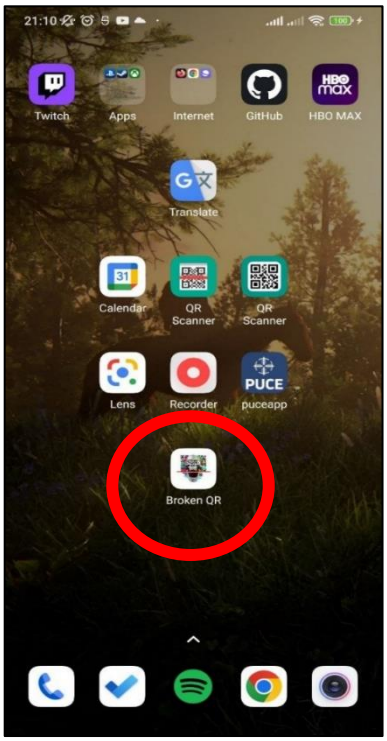




En la Figura 16 muestra el icono en un dispositivo Android listo para su uso. Para iniciar la aplicación, simplemente se debe tocar el icono.

Figura 14

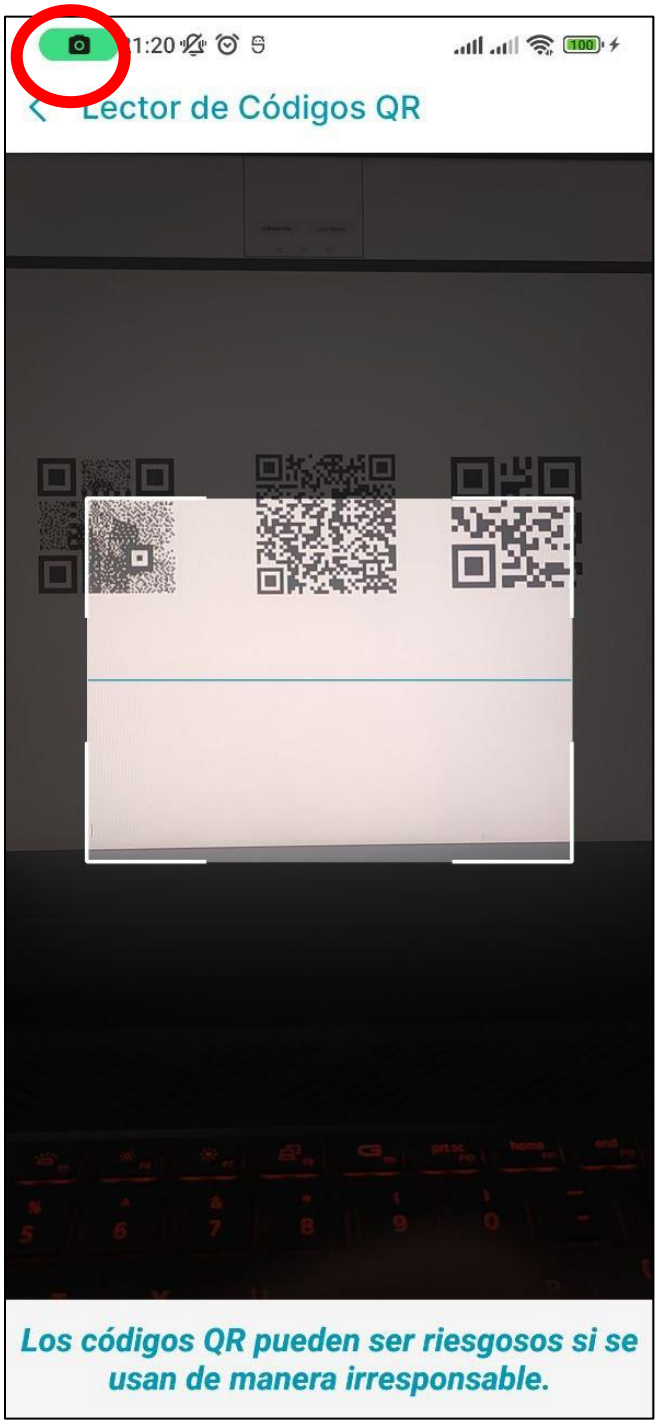
Se muestra el Icono Funcional



En la Figura 17 se muestra la pantalla funcional en un escenario real, destacando la funcionalidad de la cámara del dispositivo y el icono en el sistema Android listo para su utilización.

Figura 15

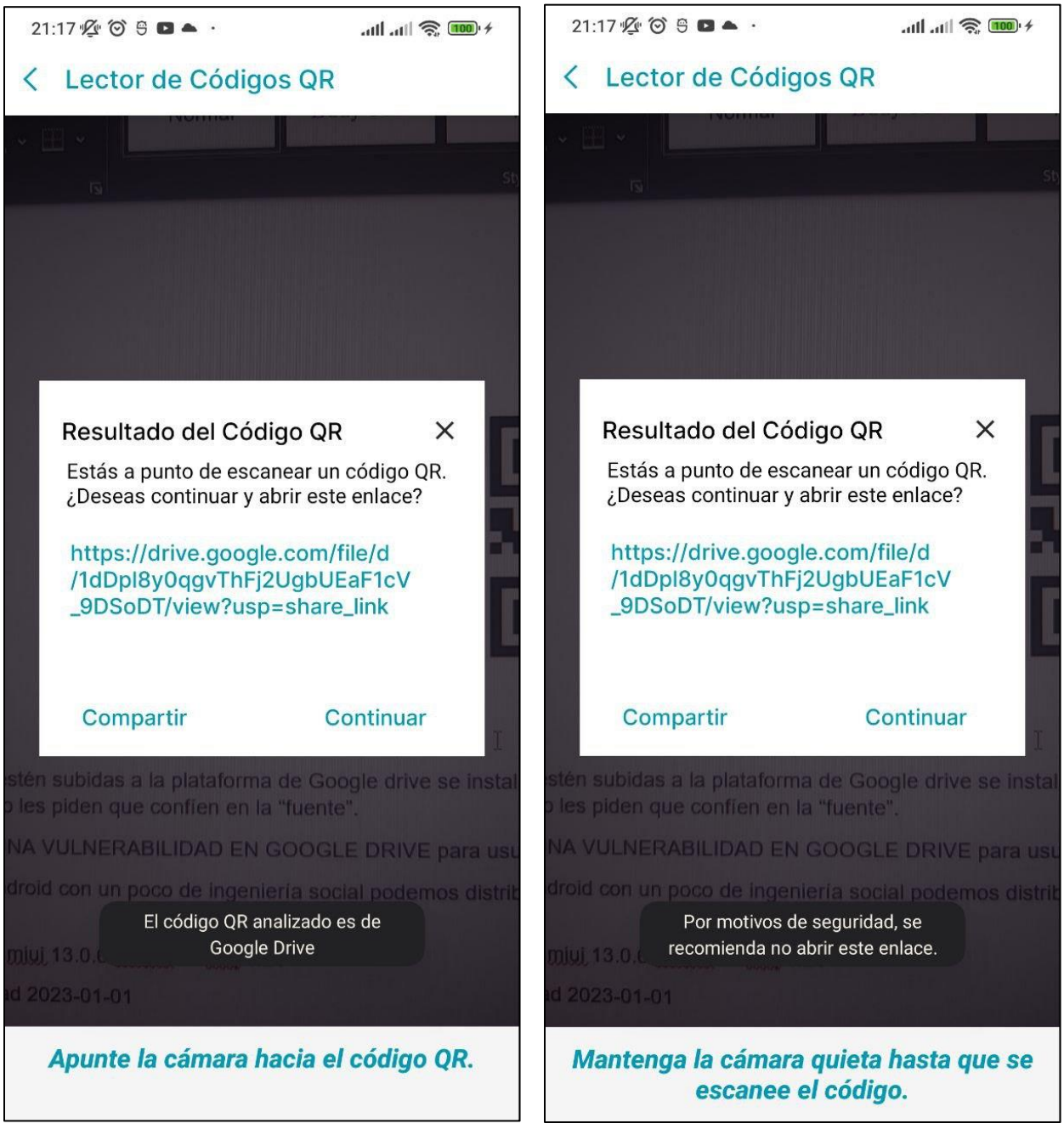
Pantalla Lector de Códigos QR con códigos de prueba.



La Figura 18 muestra la pantalla de seguridad y un cuadro de diálogo al escanear un código QR malicioso en un escenario real. Se resalta la funcionalidad de seguridad y se muestra un mensaje de precaución sobre el enlace escaneado.

Figura 16

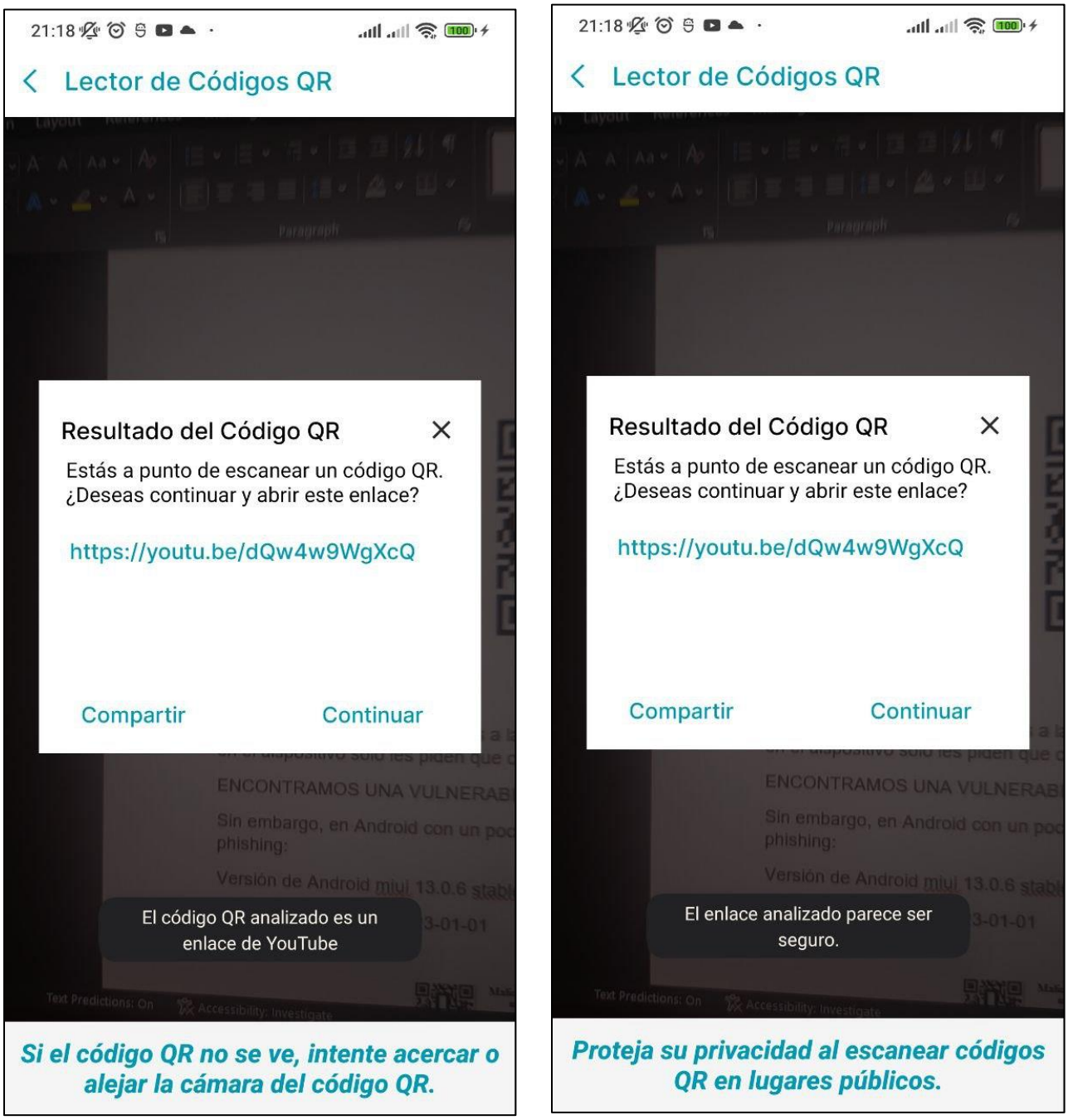
Pantallas de Seguridad Cuando se Escanea un Código QR malicioso



La Figura 19 muestra la pantalla de seguridad y un cuadro de diálogo al escanear un código QR normal que lleva a un enlace de YouTube en un escenario real. Se destaca la función de seguridad y se muestra un mensaje informativo sobre el enlace escaneado.

Figura 17

Pantallas de Seguridad Cuando se Escanea un Código QR normal



La tabla 15 muestra los dispositivos en los cuales se ha instalado la aplicación, junto con la versión de Android y el parche de seguridad correspondiente. Esto se ha llevado a cabo con el objetivo de proteger dichos dispositivos frente a posibles amenazas de seguridad.

Tabla 14

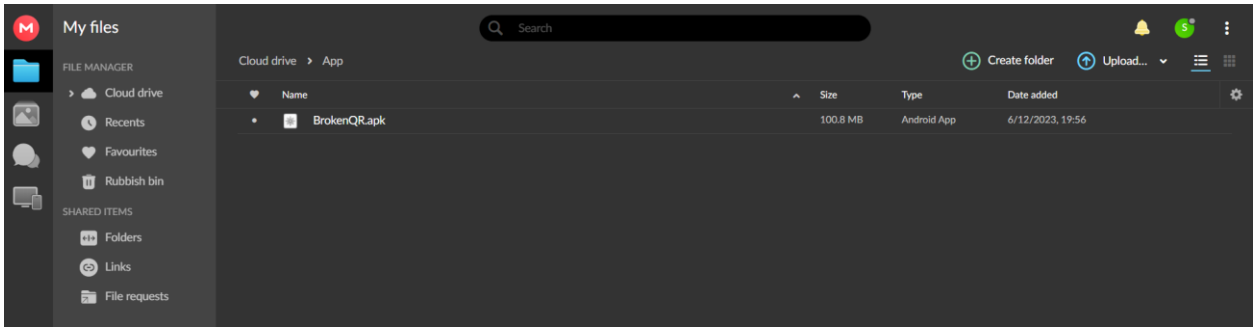
Se presenta los dispositivos que se instaló la Aplicación

Dispositivo Escaneado	Versión de Android	Versión de parche de seguridad Android
Tecno pova 4	Android 11.	2023-02-01
Xiaomi Mi Note 10 Lite	Android 10.	2023-01-01
Xiaomi Mi Note 10 Lite	Android 9 Pie.	2023-01-01
Huawei mate 20 pro	Android 10.	2021-05-10
Samsung Galaxy S20	Android 10.	2022-08-01
Google Pixel 6 pro	Android 13.	2023-02-05

La Figura 20 muestra la pantalla de la plataforma de almacenamiento en la nube donde se encuentra alojada la aplicación. En este caso, se ha elegido mega.nz debido a sus funciones de seguridad.

Figura 18

La aplicación estará disponible en el servicio de almacenamiento en la nube de Mega.nz.



5.4.1. Instalación

La apk del aplicativo está disponible en mega.nz para su descarga. En ese momento, se trabajó en el desarrollo de la aplicación en Android Studio. Cabe destacar que esta aplicación forma parte de un proyecto de titulación, lo que significa que los derechos de esta son propiedad exclusiva de la Pontifica Universidad Católica del Ecuador, y no estará disponible en ninguna tienda de aplicaciones móviles.

La aplicación ha sido diseñada para ser compatible con dispositivos que utilicen el sistema operativo Android y tengan una versión mínima con un API de 30 o superior. Es importante destacar que, si es necesario, hay que habilitar la opción que permite instalar aplicaciones provenientes de fuentes desconocidas.

5.4.2. Códigos QR usados en la investigación

Para las pruebas correspondientes, se utilizan dos códigos QR. El código QR mostrado en la Figura 21 contiene un enlace a un vídeo disponible en la plataforma de YouTube. Por otro lado, el código QR de la Figura 22 corresponde a una aplicación alojada en la plataforma de Google Drive. Ambos códigos QR se emplearon para evaluar el rendimiento y la funcionalidad de la aplicación en distintos escenarios.

Figura 19

Código QR de un Video de Youtube

Figura 20

Código QR de Aplicación para pruebas

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- Se logró desarrollar una aplicación móvil exitosa para dispositivos Android que cumple con los objetivos establecidos de detectar y prevenir códigos QR maliciosos, garantizando la confidencialidad y protección de los datos de los usuarios al escanear códigos QR. Los resultados obtenidos fueron consistentes con las expectativas, lo que demuestra la efectividad de la aplicación en abordar el problema planteado.
- Los hallazgos teóricos y metodológicos de esta investigación indican que el desarrollo de una aplicación móvil dedicada a la detección temprana de códigos QR maliciosos es una solución efectiva para combatir amenazas de seguridad en dispositivos móviles. Estos hallazgos tienen implicaciones significativas en la seguridad móvil y el uso seguro de códigos QR.
- Es fundamental que los usuarios adopten precauciones al escanear códigos QR y verifiquen la autenticidad de los sitios web antes de proporcionar información confidencial, se recomendaba evitar caer en estafas que imitaran sitios web oficiales y pusieran en riesgo la seguridad de la información personal y financiera.
- El desarrollo de un prototipo de aplicación móvil para la detección temprana de códigos QR maliciosos fue un tema relevante en la seguridad móvil, ya que puede ayudar a proteger a los usuarios de dispositivos móviles contra las amenazas de malware y mejorar la seguridad en línea de manera general.

Recomendaciones

- Los usuarios deben tomar medidas de seguridad adicionales al escanear códigos QR, como verificar la autenticidad del sitio web antes de proporcionar información confidencial.
- Las empresas deben ser cautelosas al utilizar códigos QR como estrategia de marketing y tomar medidas para garantizar la seguridad de los usuarios.
- Es necesario usar herramientas de detección eficaces que permitan identificar y mitigar los riesgos asociados con los dispositivos móviles, como una aplicación prototipo para la detección temprana de códigos QR maliciosos.
- Los gobiernos y las organizaciones relacionadas con la seguridad deben aumentar la conciencia pública sobre los riesgos asociados con los códigos QR y promover prácticas seguras para su uso.
- Se recomienda incorporar una sección en la aplicación que permita realizar estadísticas a partir de las encuestas de opinión de los usuarios que utilizan la plataforma. Una manera de hacerlo podría ser a través de la sección "Califíquenos" ya existente en la aplicación, donde se invita a los usuarios a responder preguntas específicas sobre su experiencia con la aplicación. Luego, se pueden utilizar esos datos para obtener información valiosa que ayude a mejorar la aplicación.

BIBLIOGRAFÍA

Albornoz, M. C. (20 de Octubre de 2014). *Diseño de interfaz gráfica de usuario*. http://sedici.unlp.edu.ar/: http://sedici.unlp.edu.ar/handle/10915/41578

Alcaraz, J. (29 de Enero de 2022). *Tenga cuidado con los códigos QR: esto le pueden hacer los estafadores*. elColombiano: https://www.elcolombiano.com/tecnologia/como-se-puede-suplantar-un-codigo-qr-BE16472164

Almagro, C. A. (Diciembre de 2011). *Universidad de Granada.* Departamento de Lenguajes y Sistemas Informáticos: https://lsi.ugr.es/curena/doce/lp/tr-11-12/lp-c01-impr.pdf

angular.io. (2020). *angular.io.* https://angular.io/docs

ARREDONDO, J. L. (11 de Mayo de 2015). *cincodias.elpais.com*. ¿PARA QUÉ SIRVE CADA UNO DE LOS SENSORES QUE TIENE TU MÓVIL?: https://cincodias.elpais.com/cincodias/2015/05/11/lifestyle/1431341623_109997.html#:~:text=Los%20sensores%20integrados%20en%20los,en%20definitiva%2C%20el%20sistema%20operativo.

Auth0® Inc. (2013 - 2020). *https://jwt.io/.* https://jwt.io/introduction/

Axessnet. (06 de 12 de 2020). *AXESSNET*. https://axessnet.com/como-funciona-el-internet-via-satelite-enlace-satelital/

Belcic, I. (2023, Marzo 15). *Avast Software*. c-malware: https://www.avast.com/c-malware

Bootstrap Community. (2020). *getbootstrap.com.* https://getbootstrap.com/docs

C. Xia, G. Y. (2009). Efficient Implement of ORM (Object/Relational Mapping) Use in J2EE Framework: Hibernate. *Efficient Implement of ORM (Object/Relational Mapping) Use in*

J2EE Framework: Hibernate. Wuhan, Hubei, China: International Conference on Computational Intelligence and Software Engineering.

Calvo, L. (02 de Febrero de 2023). *GoDaddy*. es.godady.com: https://es.godaddy.com/blog/que-es-un-codigo-qr-y-como-funciona/

Computer Hope. (16 de Agosto de 2021). *Computer Hope*. www.computerhope.com: https://www.computerhope.com/jargon/s/software.htm

Denise , S., y Carmen , G. (15 de Marzo de 2021). *https://freed.tools*. Prototipo: qué es y para qué sirve: https://freed.tools/blogs/ux-cx/prototipo

Developer Android. (24 de Octubre de 2022). *developer.android.com*. Recomendaciones sobre seguridad de apps: https://developer.android.com/topic/security/best-practices?hl=es-419

Developer Android. (2023, Marzo 13). *Android Studio Electric Eel | 2022.1.1*. developer.android.com: https://developer.android.com/studio/releases#logcat

DeveloperAndroid. (2023, Marzo 13). *Download and install Android Studio*. developer.android.com: https://developer.android.com/codelabs/basic-android-kotlin-compose-install-android-studio#5

Eduardo Polo Ortega, F. J. (2015). *Servicios de red e Internet.* Madrid, España: RA-MA Editorial.

G., E. C. (18 de 08 de 2011). *slideshare.net.* https://es.slideshare.net/edisoncoimbra/71-redes-por-satlite-sh

GilJin Yang, B. C. (01 de 2014). *Research Gate.* ww.researchgate.net/publication/298642533_Implementation_of_HTTP_live_streaming_for_an_IP_camera_using_an_open_source_multimedia_converter

Grasa, J. M. (10 de 17 de 2017). Acceso a Internet vía satélite. En J. Mora, *Guías de Tecnología fácil* (p. 24). Madrid: Asociación española ingenieros de telecomunicación. http://www.coitaoc.org/files/estudios/tecnologia_facil_7aba8393.pdf

Graydon, M. &. (7 de August de 2019). *'Connecting the unconnected': a critical assessment of US satellite Internet services.* SAGE JOURNALS: https://doi.org/10.1177/0163443719861835

Grijalva, N. (15 de 10 de 2012). *blogspot*. http://sofware1nathalygrijalva.blogspot.com/2012/10/modelo-espiral.html

Guniganti, R. &. (2013). A Comparison of RTMP and HTTP Protocols with respect to Packet Loss and Delay Variation based on QoE. *semanticscholar.org*.

Gupta, D. (2023, Marzo 2). *beaconstac*. blog.beaconstac.com: https://blog.beaconstac.com/2021/06/qr-codes-exploitation/

Gutiérrez, J. J. (12 de 05 de 2014). *Qué es un framework web.* http://www. lsi. us. es/~ javierj/investigacion_ficheros/Framework

I. Fette, A. M. (December de 2011). *Internet Engineering Task Force (IETF) .* https://www.hjp.at/doc/rfc/rfc6455.html

INSIDER INTELLIGENCE. (2022, Febrero 11). *QR Codes Forecast and Trends 2022*. https://www.insiderintelligence.com/content/qr-codes-forecast-trends-2022: https://www.insiderintelligence.com/content/qr-codes-forecast-trends-2022

Instituto Nacional de Estadísticas y Censos. (2019). *www.ecuadorencifras.gob.ec.* https://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas_Sociales/TIC/2019/201912_Principales_resultados_Multiproposito_TIC.pdf

Joskowicz, J. (10 de 02 de 2008). Reglas y prácticas en eXtreme Programming. Universidad de Vigo, 22. https://d1wqtxts1xzle7.cloudfront.net/31398587/xp_-_jose_joskowicz-with-cover-page-v2.pdf?Expires=1625441610&Signature=HZfoDu6RCpoB-dKMMuLRNmZRaiz0cWUrcjbndtRyjECrK33QWDAGtlNDg1Cnw9kvQJ9Psul9gXX8CRJculpI5KBhJgUMc~bIqu72mdVo6cpvqyy3-XejGZUvukkePQRHmxpb-Ddq

Kaspersky IT Encyclopedia. (2020, Enero 03). *What is phishing?* encyclopedia.kaspersky.com: https://encyclopedia.kaspersky.com/knowledge/what-is-phishing/#:~:text=Phishing%20is%20a%20type%20of,details%20and%20other%20confidential%20information.

KASPERSKY.ES. (07 de Julio de 2022). *KASPERSKY DAILY*. Ataques phishing mediante Código QR a usuarios de QQ: https://www.kaspersky.es/blog/phishing-qr-code-attack-on-qq-users/27357/

Lagatree, K. (2006). Keep it Together. En K. Lagatree, *Keep It Together: 200+ Tips, Tricks, Lists, and Solution for EverydayLife* (p. 432). Random House Reference.

Latam kaspersky. (28 de Febrero de 2023). *Comunicados de prensa*. Kaspersky descubre 200,000 nuevos instaladores de troyanos bancarios para dispositivos móviles; el doble que en 2021: https://latam.kaspersky.com

Ley De Comercio Electrónico, Ley 67 (Congreso Nacional 17 de 05 de 2002).

Ley Orgánica De Comunicación, 22 (Legislativo 25 de 06 de 2013).

Maria. (24 de 04 de 2019). *instalacionestk.com*. https://www.instalacionestk.com/conoce-las-ventajas-y-desventajas-del-internet-satelital/

Maza, M. Á. (2012). *javascript Certificado de profesionalidad.* Innovación Y Cualificación.

Moes, T. (03 de Enero de 2020). *SofwareLab.org*. SofwareLab: https://softwarelab.org/es/que-es-hardware-y-software-definicion-y-diferencias/

Muñoz, J. (25 de 07 de 2006). *maestrosdelweb.* www.maestrosdelweb.com/intersatelite/

Nielfa, J. S. (22 de Junio de 2020). *Android Studio: El entorno de desarrollo oficial de Android*. scoreapps.com: https://scoreapps.com/blog/es/android-studio/

NortonLifelock. (2023, Marzo 17). *us.norton.com*. us.norton.com: https://us.norton.com/blog/mobile

Oracle Corporation and/or its affiliate. (2020). *What is MySQL.* https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html

Ordoñez, J. L. (2009). *Videoconferencia.* Mexico: Alfaomega Grupo Editor.

Pérez Ibarra, S. G. (17 de 06 de 2021). *Red de Universidades con Carreras en Informática.* http://sedici.unlp.edu.ar/handle/10915/120476

Pérez, M. (2011). *Microsoft SQL Server 2008 R2. Motor de base de datos y administración.* RC Libros.

Provazza, A. (01 de Octubre de 2019). *www.techtarget.com*. DEFINITION: https://www.techtarget.com/searchmobilecomputing/definition/smartphone

PUCE. (10 de 07 de 2017). *LA ESTACIÓN CIENTÍFICA YASUNÍ*. http://www.yasuni.ec

Qrcode-tiger. (21 de Febrero de 2023). *www.qrcode-tiger.com*. Pronóstico de códigos QR para 2023: ¿Llegaron los códigos QR para quedarse?: https://www.qrcode-tiger.com/es/qr-code-forecast#QR_codes_are_here_for_the_long_run_experts_say

Real Academia Española. (2021). REAL ACADEMIA ESPAÑOLA.

Red5. (06 de 12 de 2020). *Red5pro.com*. Red5.org: http://red5pro.com/

Richard, A. (2017). Can Weather Affect Satellite Internet? *Hearst Newspapers*, 1. Can Weather Affect Satellite Internet?: http://smallbusiness.chron.com/can-weather-affect-satellite-internet-26822.html

Sampieri, R. F. (2014). Definiciones de los enfoques cuantitativo y cualitativo, sus similitudes y diferencias. En C. F. Roberto Hernández Sampieri. RH Sampieri, Metodología de la Investivación.

Source Android. (06 de Febrero de 2023). *source.android.com*. Android Security Bulletin—February 2023: https://source.android.com/docs/security/bulletin/2023-02-01

StartupGuideIONOS. (17 de Agosto de 2020). *Modelo en espiral: el modelo para la gestión de riesgos en el desarrollo de software*. www.ionos.mx: https://www.ionos.mx/startupguide/productividad/modelo-en-espiral/

Taylor Otwell. (2011-2020). *Laravel.* https://laravel.com/

The PHP Group. (2001-2020). *php.net.* https://www.php.net/manual/es/intro-whatis.php

Turner, A. (2018, Septiembre 10). *bankmycell*. How Many People Have Smartphones Worldwide (May 2023): https://www.bankmycell.com/blog/how-many-phones-are-in-the-world#:~:text=According%20to%20Statista%2C%20in%202023,of%20that%20year%27s%20global%20population.

Yolanda Martínez, S. d. (06 de 2015). *Triplemente marcadas: Desconexiones comunicativas en la Amazonia sur ecuatoriana.* Cuenca: Universidad de Cuenca. Triplemente marcadas: Desconexiones comunicativas en la Amazonia sur ecuatoriana: http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0719-367X2015000100007

GLOSARIO DE TÉRMINOS

SDK: Un kit de desarrollo de software (SDK) consiste en un conjunto de herramientas y programas suministrados por proveedores de hardware y software que los desarrolladores podían utilizar para crear aplicaciones adaptadas a plataformas particulares.

JDK: El conjunto de herramientas de desarrollo de Java (JDK) se considera como la base fundamental para la construcción de todas las aplicaciones dirigidas a la plataforma Java.

Gradle: La herramienta facilita la construcción y desarrollo de aplicaciones de software, al mismo tiempo que simplifica y automatiza el proceso de prueba, entrega y distribución del software.

XML: (Extensible Markup Language) en Android es un lenguaje de marcado ampliamente utilizado para definir la interfaz de usuario de una aplicación.

AVD: (Android Virtual Device) es un emulador de dispositivos virtuales el cual nos ayuda a probar y depurar las aplicaciones que se desarrolla, esto se realiza para poder desarrollar y probar en un dispositivo virtual sin necesidad de tener un dispositivo físico.

APK: (Android Application Package) es un formato de archivo ampliamente utilizado por Android para la instalación y distribución de aplicaciones en su ecosistema. Este archivo contiene todos los componentes necesarios para la instalación de una aplicación en un dispositivo Android. El archivo .apk se encuentra comprimido y contiene diversos archivos como metadatos que describen la aplicación.

SQLite: SQLite es una tecnología de código abierto que brinda la posibilidad de almacenar datos en dispositivos con recursos limitados de manera práctica y eficaz.

IDE: (Integrated Development Environment) o su equivalente en español, Entorno de Desarrollo Integrado, es una solución de software que ofrece las herramientas y funciones necesarias para facilitar el desarrollo de aplicaciones.

Java: Java es un lenguaje de programación orientado a objetos, ampliamente utilizado en una variedad de dispositivos en todo el mundo. Es utilizado para desarrollar aplicaciones, sistemas operativos móviles, software empresarial y numerosos programas reconocidos. Con su capacidad multiplataforma, Java ha dejado su huella en miles de millones de dispositivos en todo el mundo, impulsando la funcionalidad de aplicaciones, sistemas operativos de smartphones y software empresarial.

ANEXOS

Anexo A: Entrevista

Entrevista con el experto en seguridad informática:

Pregunta: ¿Cuáles son los principales riesgos asociados a la presencia de códigos QR maliciosos en dispositivos Android?

Experto: Los códigos QR maliciosos pueden ser utilizados por hackers para dirigir a los usuarios a sitios web maliciosos o para descargar malware en sus dispositivos Android. Si el usuario escanea un código QR malicioso, esto podría dar lugar a la pérdida de datos personales, financieros o confidenciales, así como a la toma de control del dispositivo. Por lo tanto, es fundamental que los usuarios estén informados sobre cómo detectar y prevenir la presencia de códigos QR maliciosos.

Pregunta: ¿Cuáles son las características que debería tener una aplicación móvil para detectar y prevenir la presencia de códigos QR maliciosos?

Experto: Una aplicación efectiva para detectar y prevenir la presencia de códigos QR maliciosos debería tener características como el análisis en tiempo real de los códigos QR escaneados, la verificación de la autenticidad del sitio web al que se dirige el código QR y la identificación de patrones de comportamiento sospechosos en la actividad del usuario. También debería alertar al usuario si se detecta un código QR malicioso y proporcionar información detallada sobre los riesgos asociados al código QR escaneado.

Pregunta: ¿Cuáles son las mejores prácticas que los usuarios pueden seguir para protegerse de los riesgos relacionados con códigos QR maliciosos?

Experto: Para protegerse de los riesgos relacionados con códigos QR maliciosos, los usuarios deben seguir ciertas mejores prácticas, como evitar escanear códigos QR de fuentes desconocidas, verificar la autenticidad del sitio web al que se dirige el código QR antes de

ingresar información personal o financiera y mantener actualizado el software de seguridad de su dispositivo Android. También es recomendable utilizar una aplicación móvil confiable para escanear códigos QR y seguir las recomendaciones de seguridad del fabricante del dispositivo.

En cuanto a la entrevista con el experto en seguridad informática, cabe destacar que éste ha preferido mantener su nombre en anonimato para proteger su identidad y evitar posibles repercusiones. Sin embargo, sus conocimientos y experiencia en el campo de la seguridad informática han sido de gran valor para el proyecto. Agradecemos profundamente su colaboración y compromiso en brindar información de calidad sobre los riesgos asociados a códigos QR maliciosos y las mejores prácticas para proteger a los usuarios de estos riesgos.

Anexo B: Código Fuente Java Consejos

Figura 21

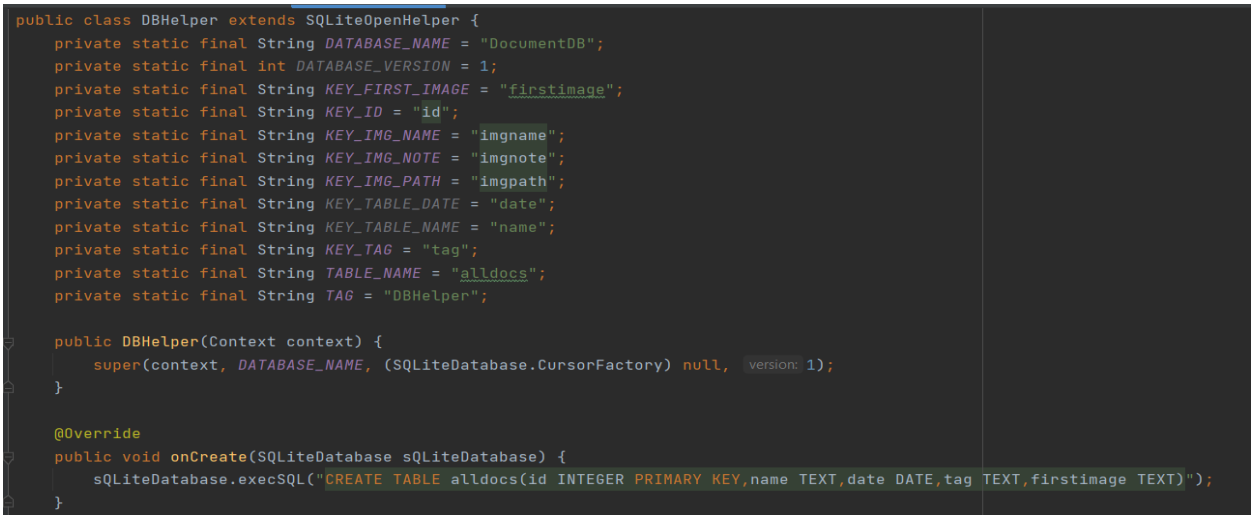
Fragmento de Código en java que presenta consejos para escanear códigos QR

```
String[] texts = {
        "Apunte la cámara hacia el código QR.",
        "Mantenga la cámara quieta hasta que se escanee el código.",
        "No escanee códigos QR desconocidos.",
        "Escanea solo códigos QR de fuentes confiables.",
        "No proporcione información personal.",
        "Los códigos QR pueden ser riesgosos si se usan de manera irresponsable.",
        "Ten cuidado con lo que compartes.",
        "No escanee códigos QR manipulados.",
        "Desconfíe de los códigos QR que ofrecen premios o regalos gratuitos.",
        "Si el código QR no se ve, intente acercar o alejar la cámara del código QR.",
        "Proteja su privacidad al escanear códigos QR en lugares públicos.",
        "Evite mantener la cámara demasiado cerca del código QR.",
        "No escanee códigos QR que hayan sido dañados o desgastados.",
        "Revise políticas de privacidad de sitios web y apps asociados con los códigos QR."
};
final int[] currentIndex = {0};
Handler handler = new Handler();
Runnable runnable = new Runnable() {
    @Override
    public void run() {
        textSwitcher.setText(texts[currentIndex[0]]);
        currentIndex[0] = (currentIndex[0] + 1) % texts.length;
        handler.postDelayed( r: this,  delayMillis: 5000); // Cambia el tiempo en milisegundos para mostrar cada texto.
    }
};
```

Anexo C: Código Fuente Base de Datos SQlite

Figura 22

Creación de la tabla alldocs para su uso en Android Studio



```
public class DBHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "DocumentDB";
    private static final int DATABASE_VERSION = 1;
    private static final String KEY_FIRST_IMAGE = "firstimage";
    private static final String KEY_ID = "id";
    private static final String KEY_IMG_NAME = "imgname";
    private static final String KEY_IMG_NOTE = "imgnote";
    private static final String KEY_IMG_PATH = "imgpath";
    private static final String KEY_TABLE_DATE = "date";
    private static final String KEY_TABLE_NAME = "name";
    private static final String KEY_TAG = "tag";
    private static final String TABLE_NAME = "alldocs";
    private static final String TAG = "DBHelper";

    public DBHelper(Context context) {
        super(context, DATABASE_NAME, (SQLiteDatabase.CursorFactory) null, version: 1);
    }

    @Override
    public void onCreate(SQLiteDatabase sQLiteDatabase) {
        sQLiteDatabase.execSQL("CREATE TABLE alldocs(id INTEGER PRIMARY KEY,name TEXT,date DATE,tag TEXT,firstimage TEXT)");
    }
```

Nota. Modelo del componente en java en la creación de la base de datos.

Se presenta el primer diseño del modelo de base de datos SQLite

Tabla 15

Tabla para almacenar documentos en SQlite.

Column name	Data Type	Constraint
Id	INTEGER	Primary Key
Column name	TEXT	
name	DATE	
date	TEXT	
tag	TEXT	
firstimage	INTEGER	

Nota. En esta tabla se registra información sobre todos los grupos, incluyendo el nombre del grupo, la fecha de creación, la etiqueta asociada y la ruta hacia la primera imagen del grupo.