Facultad de Ingeniería

Carrera de Ingeniería de Software

TRABAJO DE INVESTIGACIÓN PARA OPTAR POR EL GRADO ACADÉMICO DE BACHILLER EN INGENIERÍA DE SOFTWARE

Implementación de mecanismos de seguridad brindados por Android para el desarrollo de aplicaciones móviles bancarias en Lima

Autores

Gómez Vilcapoma, Josué – 1627632

Sumiano Monterrey, Sergui Angelo - 1412661

Asesores

Cota Sencara, David William

Lima, Perú

Diciembre del 2018

Dedicatoria:

Dedicamos de manera muy especial a todos nuestros familiares, principalmente quienes nos apoyaron durante toda nuestra etapa de educación, nuestros profesores empezando desde la primaria hasta la Universidad, ya que nos guiaron en el camino para llegar a ser futuros profesionales exitosos. Nuestros amigos que nos permitieron aprender junto con ellos.

Agradecimientos:

Durante nuestra etapa de educación superior y desarrollo de dicha investigación académica, queremos agradecer a toda persona que nos ayudó desde lo más simple hasta lo más complejo. En primer lugar, nuestra casa de estudios Universidad Tecnológica del Perú por formarnos como estudiantes y futuros profesionales. En segundo lugar, a todos nuestros profesores que dictan cursos referentes a la carrera y profesores directos referentes a Investigación Académica; particularmente agradecemos a los profesores Roxana Quiroz Valenzuela, Eduardo Barriga Altamirano y David Cota Sencara por sus enseñanzas y consejos para el desarrollo de la presente investigación. En tercer lugar, todos nuestros amigos por su ayuda con respecto a la implementación de la investigación. Para terminar, agradecer la colaboración a todos los mencionados por su tiempo.

Resumen

El presente trabajo de investigación nos muestra los principales mecanismos de seguridad que se deben aplicar durante el desarrollo de aplicaciones bancarias para dispositivos Android. En su contenido se explica en un inicio los principales problemas de seguridad que tiene los dispositivos móviles y sus aplicaciones de manera general según varias fuentes de investigación nacionales e internacionales. Junto a estas vulnerabilidades presentamos los mecanismos que se usan para mitigarlas y su relación con los pilares de la información.

También nos centramos en las vulnerabilidades específicas de las aplicaciones bancarias según reportes y estudios en bancos Internacionales y en Latinoamérica, proponiendo medidas para cubrir estos problemas, a través del desarrollo de una demo que emule las actividades de una aplicación bancaria, sobre la cual aplicaremos mecanismos que cubran un cierto número de las vulnerabilidades encontradas.

Dentro del trabajo está incluida estadística que recolectamos de usuarios acerca de seguridad en dispositivos móviles y los resultados de la implementación de los mecanismos de seguridad en la demo de la aplicación bancaria.

Abstract

This research work shows us the main security mechanisms that must be applied during the development of banking applications for Android devices. In its content, the main security problems of mobile devices and their applications are explained in a general way, according to several national and international research sources. Together with these vulnerabilities we present the mechanisms that are used to mitigate them and their relationship with the pillars of the information.

We also focus on the specific vulnerabilities of banking applications according to reports and studies in International banks and in Latin America, proposing measures to cover these problems, through the development of a demo that emulates the activities of a banking application, on which we will apply mechanisms that cover a certain number of the vulnerabilities found.

Within the work is included statistics that we collect from users about security in mobile devices and the results of the implementation of security mechanisms in the demo of the banking application.

INDICE

1. Introducción

Este trabajo de investigación tiene como principal objetivo la implementación de mecanismos de seguridad para el desarrollo de una aplicación móvil bancaria. Dicha investigación nace a partir del acelerado desarrollo de las tecnologías actualmente, sus aplicaciones y los grandes beneficios para las personas; no obstante, dependiendo del rubro en el cual se aplica, este debe tener un mínimo de seguridad. En el rubro de aplicaciones móviles bancarias, estas deben cumplir con ciertos mecanismos de seguridad, puesto que un fallo de seguridad puede llevar a consecuencias significativas por el ente financiero y el cliente.

En el primer capítulo, se aborda una breve introducción acerca del trabajo de investigación, así como los puntos a tratar para el desarrollo de la investigación para llegar a las conclusiones finales.

En el segundo capítulo, se expone la problemática que conlleva el desarrollo de aplicaciones móviles en el sector financiero, puesto que este debe cumplir con ciertos lineamientos de seguridad.

En el tercer capítulo, se detalla los objetivos de la investigación, tanto el objetivo principal y los objetivos secundarios. En el cual se indicará puntos que conlleven a dar seguridad a una aplicación móvil bancaria.

En el cuarto capítulo, se explica detalles técnicos para dar un mejor entendimiento en el desarrollo de la investigación, tales como Sistema Operativo, Seguridad, Aplicaciones Móviles, Seguridad Informática entre otros,

En el quinto capítulo, se presenta un estado de la cuestión que nos ayuda a recolectar información acerca de qué mecanismos de seguridad debe cumplir un aplicativo móvil

de banco, como las acciones a seguir para que un aplicativo no tenga dichas vulnerabilidades que son tanto técnicas como también una cultura de seguridad en las personas. Además, considerando los pilares de la seguridad tales como Autenticación, confidencialidad, integridad, no repudio y disponibilidad.

En el sexto capítulo, se expone la metodología a seguir para implementar mecanismos de seguridad a una aplicación bancaría. En el cual se detalla los mecanismos de seguridad a partir de la investigación del estado de la cuestión y información recolectada, como parte de trabajo de campo, de las aplicaciones moviles bancarias en el Perú con respecto a la seguridad.

En el séptimo capítulo, se implementa la información recolectada en los anteriores 2 capítulos, la cual conlleva el desarrollo de una aplicación bancaria, considerando que mecanismos de seguridad se debe aplicar. Además, se hizo un análisis a la aplicación a partir de la experiencia del usuario.

2. Problemática

La llegada de las aplicaciones móviles en el mundo contemporáneo fue un gran avance para las empresas, puesto que impulsan sus productos o servicios mediante una aplicación móvil que facilita al usuario realizar diversas funciones, no obstante, algo a considerar en cualquier sistema informático es la seguridad. La seguridad en una aplicación móvil para el sector bancario es de mucha importancia, puesto que al mínimo error de seguridad puede conllevar a pérdidas económicas significantes, mala imagen y pérdidas de clientes a la entidad financiera. Se debe poner en énfasis el desarrollo de aplicaciones, pero considerando la seguridad que deba conllevar dicha aplicación, puesto que dependiendo del rubro de la aplicación se requiere un nivel de seguridad

distinto. Para esto, se debe implementar ciertos mecanismos para poder dar seguridad a aplicaciones del rubro bancario.

2.1. Definición del Problema

Identificar cuáles son los mecanismos de seguridad para el desarrollo de una aplicación móvil bancaria.

3. Objetivos

3.1. Objetivo General

Determinar qué mecanismos de seguridad son necesarios para el desarrollo de una aplicación móvil bancaria.

3.2. Objetivos Específicos

- Identificar qué mecanismos de seguridad usan los desarrolladores de aplicaciones móviles
- Encontrar las prioridades en seguridad al desarrollar una aplicación móvil bancaria
- Detectar el nivel de cultura de seguridad informática de los usuarios.
- Identificar las buenas prácticas de los programadores.

4. Marco Teórico

4.1. Introducción

En esta parte del trabajo se detallarán los conceptos y definiciones técnicas que son necesarios para comprender el problema en mención. Entre los conceptos que se explicara son Sistema Operativo, Mecanismos de Seguridad, Encriptación de Datos, Seguridad Informática y Aplicaciones Móviles.

4.2.Sistema Operativo

Para definir los términos Sistema Operativo, la Universidad de Alicante (2015) lo define cómo. “Es el software que se sitúa entre la máquina y los programas. Básicamente su función es administrar recursos del sistema” (p. 2). Es decir, es aquel que se utiliza como software base para que exista una interacción entre el hombre y máquina, y así poder usar todas las funciones disponibles y programas correspondientes.

Así mismo, Aponte y Dávila (2016) mencionan de manera más técnica. “Un Sistema Operativo (SO) es el software básico de una computadora que provee una interfaz entre el resto de programas del computador, los dispositivos hardware y el usuario” (p. 20).

4.2.1. SO Móvil

Por lo que se sabe, en una computadora requiere de un sistema operativo para poder utilizar. Así mismo, los teléfonos inteligentes, dispositivos móviles o muchos de otros nombres que se atribute a dichos aparatos, también requieren de un sistema operativo pero orientado a dispositivos móviles. Salazar (2017) menciona que, los dispositivos móviles distan de las computadoras en ciertos aspectos, por ejemplo, a cada dispositivo móvil por defecto viene un sistema operativo, el cual no es posible cambiar de sistema operativo, puesto que cada dispositivo está orientado a un sistema operativo móvil. Entre ellas tenemos iOS, Android, Windows Phone y Blackberry.

4.2.2. Aplicaciones Móviles

Como se sabes en el mundo actual ya es muy conocido los teléfonos inteligentes o Smartphone, estos dispositivos móviles tiene aplicaciones que viene por defecto u otras que son de descargar. Ramírez (2016), menciona

acerca que las aplicaciones móviles abarcan varios mercados. Es decir, actualmente las apps móviles se desarrollan en muchos ámbitos como por ejemplo juegos, redes sociales entre otros que se pueden acceder desde una aplicación móvil.

4.2.3. iOS

Según la Universidad Nacional del Nordeste (2012). iOS es un sistema operativo móvil desarrollado por la empresa Apple. Una de las características principales del SO móvil es que no se puede instalar en hardware de terceros. Su interfaz está muy orientado al usuario para una interacción amigable y fluida entre usuario y dispositivo.

4.2.4. Android

Según la Universidad Nacional del Nordeste (2012). Define que Android es un sistema operativo Linux para el uso o empleo en dispositivos móviles. Actualmente Google es propietario del sistema operativo. Es decir, es un sistema operativo Linux, pero orientado a dispositivos móviles, por lo cual se infiere que dicho sistema operativo tiene menores características, puesto que lo alberga un hardware de menor tamaño a diferencia de computadoras.

4.3.Seguridad Informática

Quiroz y Marcías (2017) detallan que la seguridad informática tiene dos objetivos. "Mantener al mínimo los riesgos sobre los recursos informáticos, -todos los recursos- y garantizar así la continuidad de las operaciones de la organización (…). El objetivo secundario (…) consiste en garantizar que los documentos, registros y archivos informáticos de la organización mantengan siempre su confiabilidad total." (p. 680-681). En resumen, se basa en proteger todos los recursos informáticos

software o hardware y garantizar la confiabilidad de los documentos digitales de la organización.

4.3.1. Mecanismos de Seguridad

Tirado y otros (2017), explican que la implementación de mecanismos reduce y mitiga los ataques informáticos, en la cual se emplea tecnologías para proteger la información de la empresa y/o organización.

Así mismo, según Oliva (2006), los mecanismos de seguridad vienen a ser servicios que brindan seguridad a un entorno informático, estos pueden ser cifrado, en la cual se utiliza criptografía asimétrica y simétrica, dicho mecanismo se relaciona con la confidencialidad de la información; firma digital, es integrar un aglomerado de datos con una unidad de datos para así salvaguardar la integridad de la información, dicho mecanismo se le atribuye la integridad, autentificación y no repudio.

4.4. Encriptación de Datos

Una definición básica acerca de la encriptación de datos es modificar o alterar la información para que no sea legible para personas que no deben visualizar dicha información. Según Welivesecurity (2014) define como. “Cifrar o encriptar datos significa alterarlos, generalmente mediante el uso de una clave, de modo que no sean legibles para quienes no posean dicha clave. Luego, a través del proceso de descifrado, aquellos que sí poseen la clave podrán utilizarla para obtener la información original. Esta técnica protege la información sensible de una organización, ya que, si los datos cifrados son interceptados, no podrán ser leídos.” (p. 5). En otras palabras, cifrar es convertir una información en otra de tal forma que no sea legible a simple vista, ahora, se menciona acerca de un término clave,

que en palabras no técnicas viene a ser como una contraseña que se usa para poder visualizar dicha información que está cifrada o encriptado.

En la criptográfica o encriptación hay doy términos muy usados como simétrica y asimétrica. Escobar y Moya (2015) definen como. Por un lado, la llave simétrica es aquella que se utiliza una misma clave para poder cifrar y descifrar la información, además, que dicha clave va incluida en los datos, por lo que no hace tan seguro dicho mecanismo. Por otro lado, la llave asimétrica se define dos tipos de llave, una pública y otra privada, dependiendo de cómo se use uno puede ser para cifrar y el otro para descifrar la información. La llave publica es la que se puede compartir con cualquier usuario a diferencia de la clave privada que solo es uso personal del usuario u organización. Es decir, si tengo mi clave publica la puedo compartir a otros para que me envíen mensajes cifrados, y yo al ser el único con la llave privada que puede descifrar dicha información puedo acceder a ver la información enviada.

4.5.Arquitectura de SO

Todo sistema informático tiene definida una arquitectura de software. Caro y otros (2011), definen que la arquitectura es el diseño que tiene un determinado software, que tienen una estructura definida y cada componente o elementos se relacionan entre sí. Se diseña un modelo de arquitectura para uso de un software, cada componente o elemento tiene una determinada función.

5. Estado de la cuestión

El desarrollo de aplicaciones móviles para celulares no has dado diferentes tipos de tecnología para crear aplicaciones útiles y seguras. Estas aplicaciones se han vuelto parte de nuestra vida actualmente, pues apoyan y agilizan muchas de las actividades que realizamos, en el ámbito social, político, comercial, económico, ocio, etc. Por otro lado no todas estas actividades son igual de importantes o críticas para una persona o

empresa, y entre los sectores que más se necesita seguridad para el uso de aplicaciones móviles está el sector financiero, sobre todo en países como Perú en el que todavía no está asentada una cultura de la seguridad informática en los usuario finales. Este tema si bien tiene responsabilidad por parte de ambos protagonistas (usuario y entidad financiera), es esta ultima la que al desarrollar una aplicación móvil deberá asegurarse que sea lo suficientemente segura y al mismo tiempo fácil de usar, pues lo que se pone de por medio en caso de alguna falla es el dinero del usuario y la reputación de la empresa. Entonces lo que vamos a exponer en el siguiente trabajo es un modelo de cómo debe ser implementado una aplicación financiera, de acuerdo a muchos factores, como normas calidad, benchmarking, buenas prácticas, vulnerabilidades, etc. Por último los mecanismos de seguridad que vamos a explicar estarán orientados al sistema operativo Android, pues es el sistema operativo que más se usa en el Perú (Lima) y es al cual debemos enfocarnos para adecuarnos más a la realidad de la sociedad.

Es un hecho que actualmente los dispositivos móviles son parte de la vida cotidiana de las personas y son casi imprescindibles para una persona moderna debido a que simplifican, economizan y hacen veloz muchas de sus actividades, como comunicarse entre ellas, interactuar por redes sociales, hacer pagos y transferencias, ubicar lugares, etc. Estas tecnologías móviles vienen siendo lideradas por dos gigantes de la tecnología, por un lado, Google con el sistema operativo Android y por el otro lado Apple Inc con iOS.

Normalmente el usuario final mide la elección de uno de estos por características como diseño, velocidad, aplicaciones o precios. Pero muchas veces ignoran las características de seguridad de estos dispositivos, lo cual es un error muy común y bastante peligroso, pues en esta era digital los dispositivos móviles se han convertido en el blanco preferido para robar información personal. Aunque sea casi transparente para los usuarios finales,

realmente existen mecanismos bastantes complejos que desarrollan Apple y Google para sus productos móviles, y no solo nos referimos a la seguridad intrínseca del sistema operativo o propias de un modelo de celular, sino también a las herramientas y facilidades que se les da a los desarrolladores para crear aplicaciones móviles seguras, pues al final los consumidores no tiene como objetivo apreciar las características técnicas de un celular como el sistema operativo, las bibliotecas , arquitectura , etc. Sino la funcionalidad que ofrecen las aplicaciones móviles, que usaran para su trabajo, estudio u ocio.

El objetivo de este estado de la cuestión es dar a conocer la situación de la seguridad en dispositivos móviles de los últimos años, ver los mecanismos de seguridad que Android en relación a los pilares de la seguridad de la información: confidencialidad, disponibilidad, integridad, no repudio y autenticación con el fin de brindar una solución que ayude a los desarrolladores.

Palabras Claves: Android, iOS, mecanismos de seguridad móviles, desarrollo de aplicaciones móviles.

5.1.Contextualización

Comencemos entendiendo que es un Smartphone y sus características. Los primeros Smartphone combinaban las funciones de un PDA (Personal Digital Asistant) y un dispositivo móvil. Actualmente entre sus funciones más destacadas tenemos los reproductores de música, navegación por GPS, cámaras digitales compactas, pantallas táctiles, navegadores web, conexión Wi Fi y la capacidad de añadir una gran cantidad de aplicaciones. (Panchal y Chauchan, 2016).Existen muchas más funciones pero estas mencionadas son las básicas.

Como en una computadora que tiene un sistema operativo como base principal el cual se encarga de funciones como la gestión de memoria, gestión procesos, gestión archivos y seguridad, además de servir como programa principal sobre el cual otros programas van a funcionar, los dispositivos móviles dependen de un sistema operativo que determine sus principales funciones (Ahmad,Tebseen,Ahmad y Ahmad 2013) y el modo como se van a comportar al interactuar con otras aplicaciones.

Existen bastantes alternativas en cuanto a sistemas operativos para celulares. Entre los principales sistemas que usan en la actualidad se encuentran: Android (Google),iOS(Apple Inc), Symbian(Nokia), RIM's (BlackBerry), Bada (Samsung) y Windows Phone (Microsoft).(Panchal y Chauchan,2016). Muchas de estas todavía existen, pero no ocupan ni el 1% del mercado y probablemente dejen de usarse.

Analizaremos el sistema operativo Android. Los cuales tenían como se aprecia en la Ilustración 1 para diciembre del 2017 el 81%. (Sahan 2017).

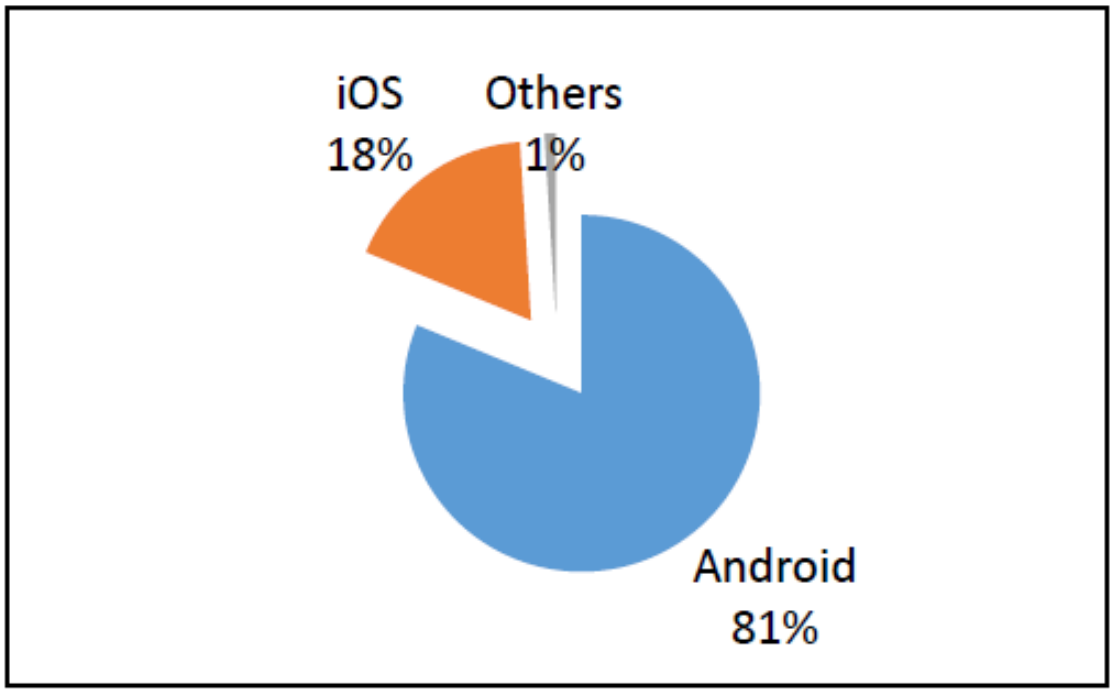


Figura I. Distribución del uso de los Sistemas Operativos.

5.1.1. Sistemas Operativos.

Podemos definir Android de manera básica con lo siguiente: Android es un sistema operativo de código abierto, basado en el kernel de Linux desarrollado inicialmente por Android Inc hasta que fue comprado por Google en el año 2005 y en el año 2007 Google junto a la fundación Open Handset Alliance (un conjunto de compañías de hardware, software y redes) cuyo objetivo era la creación de estándares abiertos para dispositivos móviles, crearon Android (Jamdaade 2016). Probablemente como respuesta a la presentación del iPhone y la necesidad de un sistema operativo móvil no propietario.

Entre las principales empresas que formaban este grupo estaban: Google, HTC, Sony, Intel, Motorola, Samsung y LG. (Panchal 2016). Hay que destacar que ahora Android no solo está en celulares, sino en varios

dispositivos como televisores, relojes, automóviles. Como se aprecia muchas empresas de TI se vieron beneficiadas de la creación de un sistema operativo libre.

En la parte orientada a los desarrolladores Google provee su IDE oficial Android Studio para poder desarrollar aplicaciones móviles para su sistema operativo, esto junto con su SDK, el cual contiene un conjunto de herramientas y APIS. Para la programación se usa los lenguajes JAVA y C, pero no usa el JVM(Java Virtual Machine), sino la máquina virtual Dalvik como intermediario entre las aplicaciones y el hardware.(Panchal 2016). Esta máquina virtual solo es propia de Android, es una característica que hace que una aplicación se ejecute en cualquier modelo de celular Android.

Su naturaleza de software libre le ha dado a Android la posibilidad de apoderarse de gran parte del mercado, esto junto a la alta personalización y la gran cantidad de empresas que pueden hacer uso del sistema operativo para crear productos y software.

Además gran parte de la aceptación y usabilidad de Android es debido al tema de precios y la amplia gama de opciones para elegir por parte del usuario final y a esto le sumamos la gran cantidad aplicaciones en su google play store,y la facilidad y herramientas que brinda a los desarrolladores para poder crear y publicar nuevas aplicaciones móviles.

5.1.2. Seguridad.

Al hablar de tecnología debemos entender que el tema de seguridad no va orientado al hardware, a las redes, la infraestructura, si bien todas estas pueden ser blancos de ataques o cibercrimen, el objetivo principal de estos

ataques es la obtención, modificación o eliminación de la información. La información actualmente es uno de los activos más importantes para las empresas y las personas, pues en esta era digital la exposición de nuestros datos se da voluntaria o involuntariamente con mucha facilidad.

Poder tener acceso a información de una empresa puede ir desde averiguar quiénes son sus socios comerciales, cuáles son sus planes estratégicos de negocio, robo de productos o patentes, etc; y para el usuario final también existe un gran peligro, puesto que se puede obtener datos personales, fotos y videos privados, horarios, lugares donde trabaja, su sueldo, su relación con otras personas, y a diferencia de las empresas donde es muy complicado el robo de dinero, es muy común obtener claves de tarjeta y robar la identidad de los usuarios.

Hablar de la falta de cultura de seguridad de TI no es el punto de este estado de la cuestión, pero cabe aclarar que gran parte de estos ciberataques son exitosos en su mayoría por error o negligencia de las personas, más que por el conocimiento técnico del atacante.

Las tendencias en ataques a la seguridad de la información vienen dadas por el porcentaje de que abarca una tecnología en el mercado, por ejemplo, ya se vio con Microsoft que la mayoría de virus, programas maliciosos están orientados a su sistema operativo Windows por ser el que más se usa. Hay pocos esfuerzos por desarrollar programas maliciosos para OSX y ni hablar de GNU-Linux.

Con el crecimiento exponencial del uso de los Smartphone en todo el mundo, era obvio que las nuevas modalidades de ataques iban ir orientadas a estas

nuevas tendencias, por ello Android ha trabajado y sigue trabajando continuamente en las medidas de seguridad que deben adoptar para evitar ataques de programas maliciosos en sus dispositivos y dar la herramienta necesaria para el desarrollo de aplicaciones seguras.

Existen 5 pilares en los que se basa la seguridad de la información y que usaremos para poder clasificar y medir los mecanismos de seguridad de estos sistemas operativos

Confidencialidad: Propiedad por la cual la información no esté disponible a individuos u organizaciones no autorizadas.

Integridad: Propiedad por la cual se protege la modificación parcial o total de la información.

Disponibilidad: Propiedad de la información de ser accesible para un ente autorizado cuando este lo requiere.

No repudio: Propiedad por la cual no se puede negar la participación de entes en la comunicación o tratamiento de la información.

Autenticación: Proceso por el cual se confirma que un ente es quien dice ser.

5.2.Mecanismos de Seguridad en Android

Para hablar de mecanismos de seguridad debemos empezar definiendo que componentes tienen un rol importante en este tema, por ello hemos identificado 6 características que son las que proporcionan la mayor parte de la seguridad en los SO móviles: La arquitectura del sistema operativo, autenticación, procedencia de aplicaciones, permisos de aplicaciones, aislamiento y encriptación (Jamdaade 2016) Además como medida de clasificación que otros autores no han hecho indicaremos

su relación con los pilares de la seguridad de la información. Con ello dejaremos de lado las comparaciones subjetivas y tendremos unos valores objetivos para medir la seguridad de estos sistemas operativos.

5.2.1. Arquitectura.

Para analizar los mecanismos de seguridad debemos empezar conociendo la estructura interna de estos sistemas operativos, es decir la arquitectura desarrollado en ellos, pues esta nos brinda información importante de su comportamiento y los componentes que estos tienen

En (Panchal 2016) se define la arquitectura Android dividido en 5 capas como lo muestra la Ilustración 2.

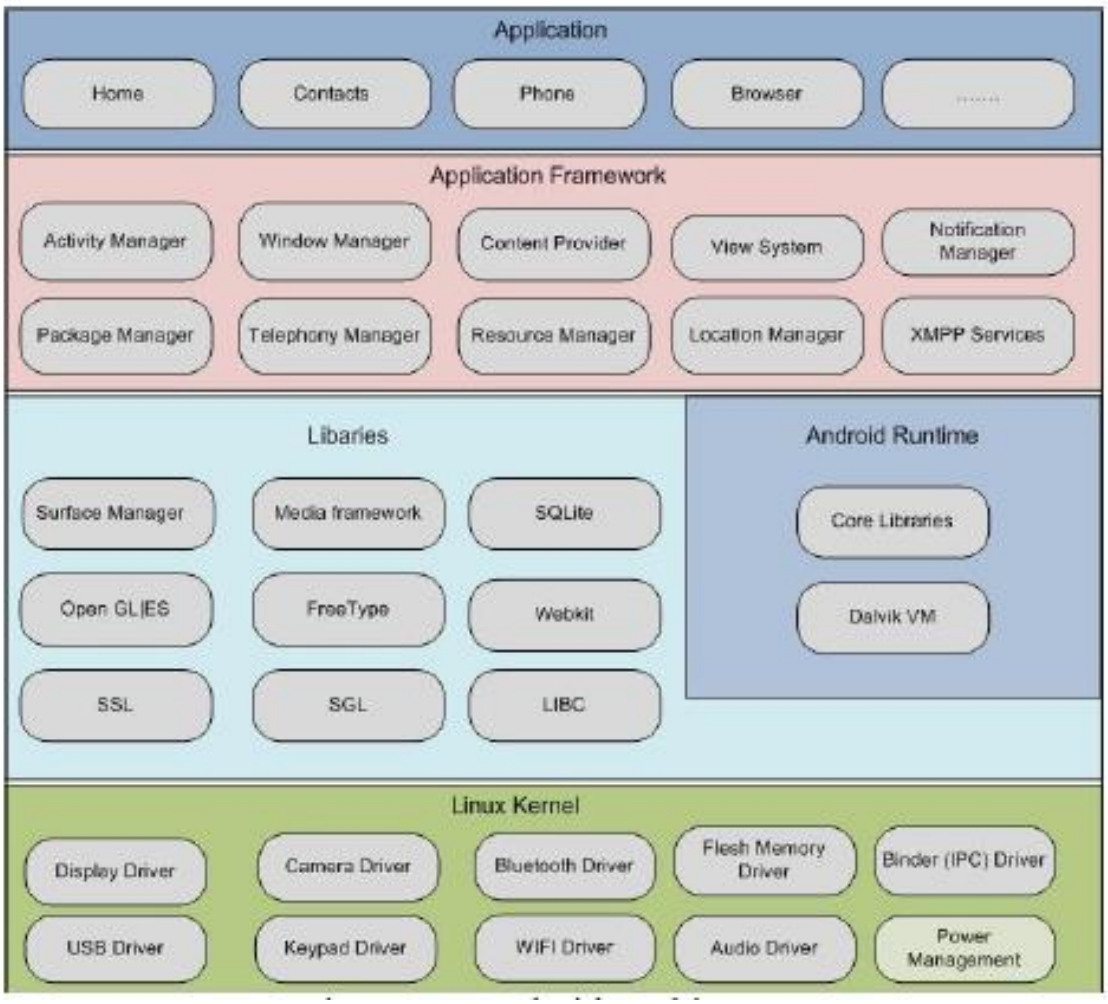


*Figura II***.** Arquitectura del Sistema Operativo Android.

Linux Kernel: La base de la arquitectura Android y controla servicios de sistema como la seguridad, administración de memoria, redes y es una capa entre el hardware y el software.

Bibliotecas: Se encuentran una capa arriba del Linux Kernel. Son bibliotecas nativas que se encargan de funciones como el almacenamiento de base de datos (SQLite), audio y video (Media), la seguridad en internet(SSL), fuentes (FreeType), etc.

Android Runtime: En la misma capa que las bibliotecas se encuentran las bibliotecas core y la máquina virtual Dalvik (DVM). Esta es la encargada de optimizar las aplicaciones Android y permite el desarrollo de aplicaciones usando el lenguaje estándar de programación java.

Framework de Aplicación: Es una capa por encima de las bibliotecas y el Android Runtime, cuenta con componentes y APIS que pueden ser usadas y reutilizadas por otras aplicaciones.

Aplicaciones: Esta capa está por encima de la capa de framework de aplicación, es la capa donde puede ser instalada las aplicaciones del celular.

La arquitectura es un tema muy importante para cualquier desarrollador puesto que al conocerse su estructura se puede escoger como interactuara la aplicación con esta misma. Así será un apoyo al momento de implementar otros mecanismos de seguridad pues sabrás en que capa esta y como se utilizará en el dispositivo.

Si debemos relacionar el análisis anterior con alguno de los pilares de la seguridad de la información este sería la integridad, el poder tener una arquitectura separada por capas y cada una con una funcionalidad independiente nos brindara más seguridad al momento de usar e instalar aplicaciones.

5.2.2. Autenticación de usuario.

La autenticación es la capacidad que se tiene para poder demostrar que un usuario es quien dice ser y poder darle permisos para poder utilizar el sistema.

Empecemos hablando de una encuesta mencionando datos de interés de los usuarios Android presentada en (Mohammed 2017) la cual no muestra la Ilustración 4

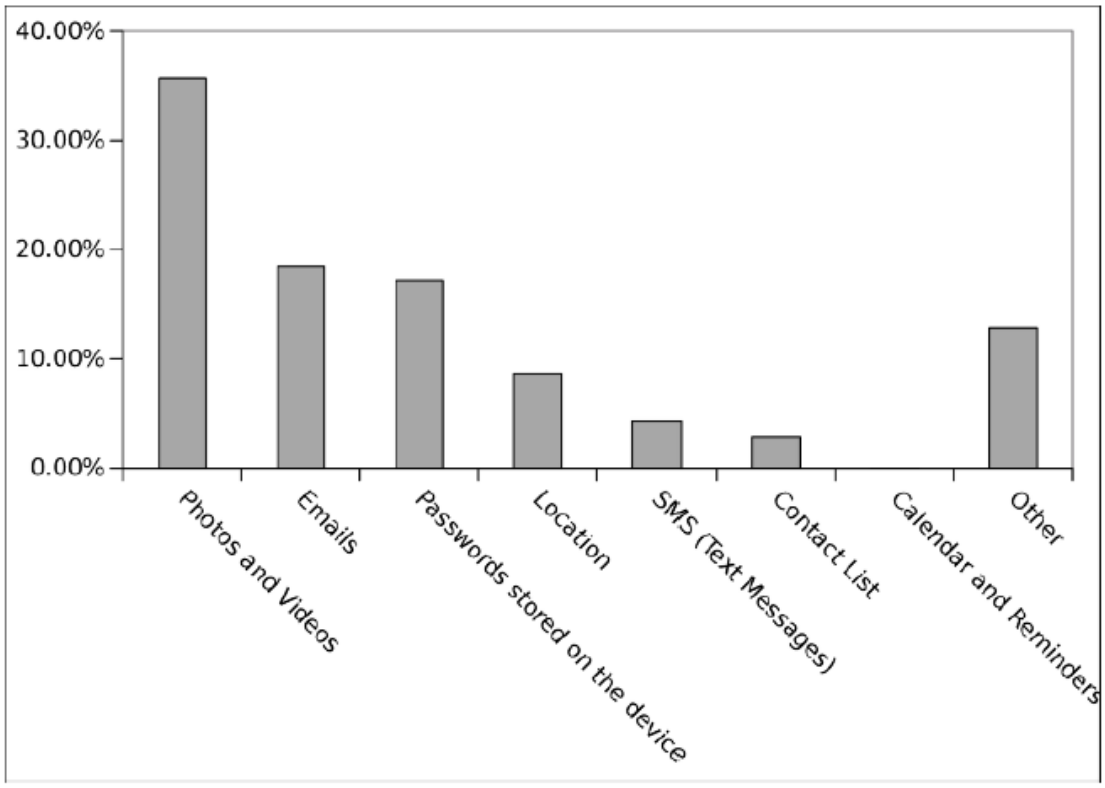


Figura IV. Datos más importantes para un usuario de Android.

La encuesta nos demuestra que la gran mayoría de usuarios tiene como concepto de datos más importantes sus fotos y videos , que desde un punto de vista afectivo se entiende que lo valoren bastante, pero los datos más valiosos deberían ser como prioridad las contraseñas almacenadas en sus dispositivos, pues con las contraseñas se podría acceder a cuentas que provean de imágenes/videos , correos, localizaciones, conversaciones, transacciones, calendarios , nuestras relaciones con otras personas, etc.

Si bien esta encuesta fue hecha a usuarios de Android sería erróneo pensar que estos resultados no se apegan a los usuarios de iOS también, pues al final de cuentas son usuarios finales con poca cultura de seguridad de la información. Claro que esto es un punto de vista pero no creemos que se aleje de la realidad, solo un desarrollador o informático consideraría respuestas distintas.

Como inferencia de estas encuestas podemos decir que el usuario final aunque no considere las contraseñas como un tema prioritario de seguridad necesitamos proveerles con mayor razón mecanismos que aseguren su autenticación para el celular y aplicaciones que usen.

En Android tenemos la característica de bloqueo y desbloqueo como la creación de PIN, patrones de desbloqueo, scanner de huella digital o el Smart lock y las características que permite el desbloqueo mostrando tu cara. Además de poder localizar y borrar tú data remotamente si activas ciertas funciones (Patterson 2017). Aunque debemos mencionar que el Smart lock ha demostrado que se puede engañar el scanner facial con una foto.

También la que añade Hayran (2016) acerca de la geolocalización

Geo-Location es una característica muy útil para ubicar su teléfono en caso de pérdida. Apple como una característica de su sistema operativo y el servicio en línea que lo acompaña proporciona esta característica. Nuestros teléfonos inteligentes llevan consigo una gran cantidad de datos confidenciales que, en las manos equivocadas, pueden utilizarse para el robo de identidad y el fraude. Para esta situación, auto borrar vienen a ayudar. Si su teléfono es robado o se pierde, puede borrar sus datos confidenciales personales de su teléfono. Cuando esta característica habilitada 10 intentos fallidos de código de acceso borrará automáticamente todos los datos en el dispositivo.

Para Android, no hay una solución nativa. Pero hay aplicaciones de terceros en el mercado.

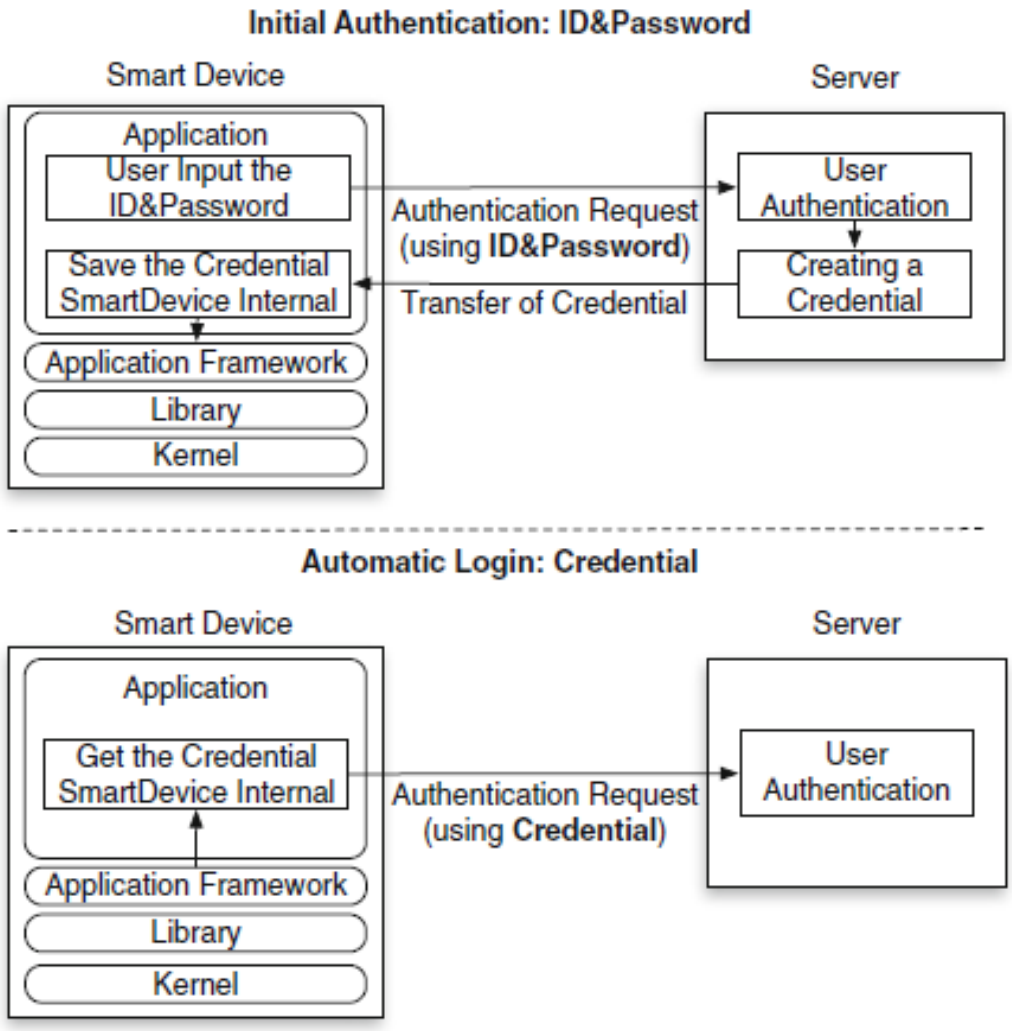


Figura V. Login automático usando credenciales.

Por otro lado, debemos abarcar otra problemática, la cual es el control que se tiene de las cuentas de usuario, específicamente el autologin. En Android y iOS la mayoría de aplicaciones proveen un login automático, esta característica da acceso sin la necesidad de poner repetidamente la contraseña luego de la verificación inicial. Es decir, la autenticación inicial se ve reemplazada por una credencial de usuario como se aprecia en la Ilustración 5. Si las credenciales son copiadas. Esto dejaría al atacante poder acceder a la cuenta de la víctima, estos ataques son llamados clonación de credencial (Choi 2016). De esta manera, queda en tu celular una credencial almacenada en memoria.

A esta información le podemos sumar que los sistemas operativos, tienen una partición específica donde guardan los datos de credenciales. Las aplicaciones de Android usan APIS para las credenciales, que como muestra la Tabla 1 el lugar de almacenamiento y método depende de la API (Choi 2016).

Tabla 1

Métodos de manejo de credenciales según la API

API	Location	Method
AccountManager	/system	database
SharedPreferences	/data	xml
SQLite	/data	database

Luego de identificar el API un atacante puede proceder de la siguiente manera. Sabiendo la localización de estas credenciales puede acceder al lugar exacto donde se localiza esa credencial y extraerla. La ubicación dependerá de la API como se ve en la Tabla 2.

Tabla 2

Localización de credenciales de aplicaciones SNS en dispositivos móviles

App	Location	Version
Google Account	/data/system/accounts.db	4.4.2-937116
Twitter	/data/system/accouts.db	5.1.0
Facebook	/data/data/com.facebook.katana/ databases/prefs.db	17.0.0.23.16
Kakaostory	/data/data/com.kakao.story/ database/kakao_story.db	1.9.1
Cyworld	/data/data/com.btb.minihompy/ shared_prefs/SKCOMMS_ACCOUNTDATA.xml	3.2.0
Nateon	/data/data/Uxpp.UC/ shared_prefs/nateon_login.xml	2.4.8

Como consecuencia y ya el atacante con estos conocimientos, el atacante obtiene las credenciales y la inserta en su propio dispositivo, cuando abra la

aplicación usara la credencial robada para darle acceso a la cuenta robada. Y a partir de este punto solicitar y buscar toda la información que ese cuenta tenga el servidor. Por ello, se debe hacer esfuerzos para que no ocurra la duplicación de credenciales especialmente en dispositivos Android (Choi 2016).El flujo de robo de credenciales puede observarse en la Ilustración 6.

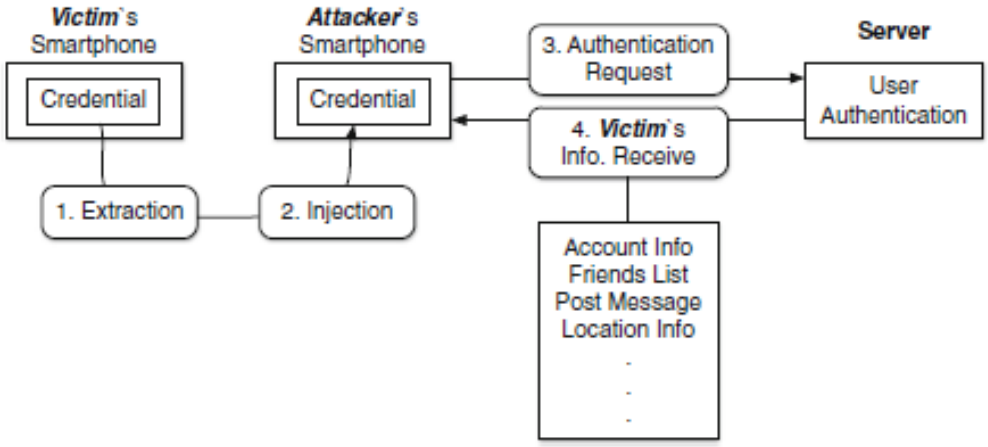


Figura 6. Diagrama de un ataque de clonación de credencial.

Apodarse de estas credenciales sobre todo en aplicaciones que manejan información sensible como las bancarias puede ser muy perjudicial. Este problema en específico se puede solucionar al evitar dejar archivos de la aplicación en lugares fáciles de acceder o también mediante el uso de una autenticación de 2 pasos, sumándole a la credencial algo que solo el usuario posea físicamente.

5.2.3. Procedencia de Aplicación.

Por procedencia de aplicación nos referimos a los mecanismos de estos sistemas operativos para controlar las aplicaciones móviles, tanto para el uso del usuario final como para el desarrollador.

Para comenzar observemos el crecimiento la tienda de aplicaciones de Android o Play Store el año 2017

En junio de 2017, Google Play Store lanzó más de 3.000.000 de aplicaciones de Android y más de 80 mil millones las aplicaciones se han descargado. En 2017, Google I / O descubrió que tenían más de 2 mil millones activos Usuarios de Android por mes, a diferencia del año anterior, con una cifra de aproximadamente 1.5 billones de usuarios activos. A partir de 2008 hasta el presente, Android ha tenido muchas actualizaciones que se han mejorado gradualmente su funcionamiento sistema al agregar nuevas funciones y corregir errores en anteriores versiones. Cada nueva versión lleva el nombre de un postre en orden alfabético: Cupcake 1.5; Donut 1.6; Eclair 2.0; Froyo 2.2 Gingerbread 2.3; Honeycomb 3.0; Ice Cream 4.0; Jelly Bean 4.1; KitKat 4.4; Lollipop 5.0; Marshmallow 6.0; Nougat 7.0 y la última versión de Oreo 8.0. (Larareska, 2017,pp 166-117)

La gran cantidad de aplicaciones generadas por Android y la misma cantidad de desarrolladores que tiene de su lado es el resultado de su modelo de negocio open source, lo que brinda bastantes facilidades para explotar las habilidades de los desarrolladores al momento de crear aplicaciones móviles y el gran abanico de posibilidades que se les da a los usuarios para escoger el programa que consideren mejor.

La tendencia libre de Android nos muestra un entorno más flexible propio de su modelo de negocios open source. En Android las personas deben crear su cuenta de desarrollador como en Apple. Google no requiere de una firma de los desarrolladores, quienes pueden crear tantas firmas como quieran sin

la verificación de Google. Esto aumenta la flexibilidad de desarrollo y debido al gran número de aplicaciones, hace más sencillo la publicación de la misma. Aunque esto puede dejar que atacantes usen el nombre de marcas conocidas para disfrazar sus aplicaciones además de poder publicar el APK de la aplicación (un instalador) en cualquier web para que el usuario la baje, aun siendo de dudosa procedencia. (Hayran, 2016).

Por ello, el uso de aplicaciones bancarias debería ser bajas solo desde la Play Store y comprobando que lo haya desarrollado el banco oficial. El uso de aplicaciones de terceros para facilitar o personalizar estas aplicaciones puede ser un punto de vulnerabilidad, debido a que podrían estar robando información cada vez que uses tu aplicación bancaria.

5.2.4. Permisos de Aplicación.

Como se vio en puntos anteriores, tanto Android como iOS tienen una arquitectura que no permite que las aplicaciones tengan acceso a los demás componentes, estos se ejecutan en un entorno aislado dentro de la arquitectura para así no comprometer los datos del usuario y componentes.

Por un lado, en Android una de las características de seguridad que se resalta son acerca de los permisos, los cuales accede a servicios de todo el sistema operativo y por ende conlleva a un hueco de seguridad. Dichos permisos por default están denegados, con el fin de no tener un impacto negativo. Para dar dichos permisos y acceder a recursos como llamadas de teléfono, cámara, entre otros; se debe de especificar los permisos para que se autorice. A parte de especificar que permisos requiere dicha aplicación, el usuario final es quien decide que permisos otorgar y eso se corrobora en cada instalación de

una aplicación móvil de la plataforma Android, la cual te declara los permisos requeridos por la aplicación para poder instalar dicha aplicación. Así mismo, existen firma de aplicaciones, la cual tiene como finalidad identificar al autor de la aplicación para así tener una relación de confianza entre aplicaciones, es decir, ciertas firmas de aplicaciones tienen permisos a ciertos recursos de las aplicaciones, estos tienen Niveles de protección como Normal, Dangerous, Signature y SignatureOrSystem. Para solicitar al usuario permisos se declara en el archivo AndroidManifest.xml con la etiqueta <user-permisión>, en el cual con el atributo name se especifica a que recurso se quiere acceder. (Madero, 2013) Como se explicó AndroidManifest.xml es la que gestiona los permisos relacionados con la aplicación.

Con lo mencionado, Android centraliza sus permisos con los recursos, con otras aplicaciones y aplicaciones de terceros hacia la misma aplicación en un solo archivo que viene a ser AndroidManifest.xml. Un punto a recalcar es que antes de acceder a recursos del sistema, requieren del permiso del usuario final ya sea como permisos o privilegios, queda en criterio del usuario final que permisos o privilegios otorgar. También hay que aclarar que las aplicaciones móviles deben ser seguras para cuidar sus datos que se alojan en la aplicación, si no, también se utiliza para robar información. Es decir, hay aplicaciones de dudosa procedencia que requieren varios permisos o privilegios y un usuario sin cultura de la seguridad otorga los permisos o privilegios sin saber que sus datos están siendo expuestos a terceras personas.

Lo descrito se relaciona con la integridad de la información, porque la información del usuario será comprometida en medida que este otorgue permisos o privilegios, las buenas practicas del desarrollador al no comprometerla información del usuario con demasiados permisos y privilegios.

Nuestras aplicaciones bancarias pueden ser bastante seguras pero no hay que olvidar que están alojadas en un dispositivo que contiene otras aplicaciones, las cuales con los permisos adecuados pueden intervenir en los ficheros de otras aplicaciones o espiar tus actividades si se les da irresponsablemente el permiso que se solicita.

5.2.5. Encriptación de Datos

Uno de los puntos a considerar entre la comparativa entre Android y iOS, es acerca sobre la encriptación o cifrado de datos. De por medio sabemos que de terminadas aplicaciones se necesita credenciales de acceso, poniendo como ejemplo, redes sociales, aplicaciones de banco, cuentas de correo entre otros.

Por un lado, en la plataforma Android por lo visto en puntos anteriores, aísla sus aplicaciones dentro de la arquitectura. Google (2018), menciona acerca del Sistema Android Keystore que permite al desarrollador almacenar credenciales en un contenedor que es seguro a nivel de sistema operativo. La función de dicho sistema va orientada a proteger credenciales contra un uso no autorizado, no permite la extracción fuera del dispositivo evitando acceder al proceso de Aplicación y permite el uso autorizado a las credenciales siempre y cuando se especifique en el código.

Con respecto al uso autorizado de claves, Google (2018) permite que al momento de generar o importar las llaves el programador pueda especificar el uso autorizado. Así mismo, al momento de importar o generar las claves las autorizaciones no se pueden modificar. Dicha funcionalidad de seguridad, no compromete a que durante la ejecución o uso de las claves se comprometa a cambiar las autorizaciones.

Android Keystore, se divide en 3 clases, Google (2018) indica que para usar la función de keystore, se emplean las clases KeyStore, KeyPairGenerator o keyGenerator. Estas clases se introducen a partir del API 18 (Android 4.3). Dichas clases se emplean hasta la actualidad con la última versión de Android.

La clase KeyStore, es un punto de instalación para guardar claves y certificados criptógrafos. Así mismo, keyPairGenerator sirve para generar dos claves públicas y privadas, y para generar se usa el método de fábrica getInstance (Google, 2018). También, la clase KeyGenerator genera claves secretas (simétricas).

Cada clase tiene una determinada función, la KeyStore es para almacenamiento, la KeyParGenerator para generar pares de llaves públicas y privadas, y por ultimo KeyGenerator para las llaves simétricas.

Para el acceso a determinadas claves o su uso, Google (2018) implementa una medida de seguridad que se pueda acceder a dicha clave siempre y cuando el usuario se autentifique mediante un patrón, PIN, contraseña y/o huella dactilar. Por ejemplo, dicha seguridad se ve implementada en

aplicaciones bancarias que te solicitan a parte de tu cuenta, una clave de seguridad de aplicación o la huella dactilar.

Por último, para el envío y recepción de datos entre cliente y servidor, Google (2018) usa TLS y SSL para comunicaciones seguras, lo cual es necesario para que la información no se vea comprometida frente ataques informáticos. Pero el uso de TLS y SSL no asegura una conexión segura, está atenta a que los certificados de seguridad no están actualizados, esto debido que ciertas aplicaciones se conectan a servidores de terceros que no necesariamente están actualizados sus certificados. Esto resulta un inconveniente, debido a que el servidor externo no está bajo uso del desarrollador. Para dicho problema, se usa las Autoridades de Certificación (CA) que funciona como un intermediario que emite un certificado al servidor con el fin de tener actualizado los certificados digitales y se adjunta la clave privada, dicho certificado el cliente puede verificar que el servidor cuenta con certificados actualizados para su uso en una red de conexión segura. Pero esto también trae un inconveniente menor, que es garantizar que el certificado enviado al servidor, se envié al servidor correcto.

Continuando con la gestión de llaves, la KeyChain sirve para almacenar contraseñas, certificados y otros datos menores. Para esto, cuando una aplicación desee acceder a un llavero o clave, lo pueda realizar sin que otra aplicación no autorizada pueda acceder a dicha clave. Se tiene dos tecnologías, una de ellas es Certificate, Key, and Trust Services que su función es gestionar las claves públicas, privadas y simétricas, dichos servicios una aplicación puede acceder para crear certificados, recuperar información de un certificado, entre otros; el otro es Keychain Services que

solo gestiona llaveros, la cual puede agregar, eliminar y editar los elementos que posee el llavero (Google, 2018). Con respecto a gestionar llaves se tiene dos tecnologías, una que se encarga de gestionar claves públicas y privadas, y la otra las colecciones de llaveros.

Siguiendo con los puntos, la comunicación segura que usar Apple en una red lo compone con los protocolos SLL y TLS. Dichos protocolos se usan para comunicaciones a través de una red o conexiones de red TCP/IP, se usa para validar datos y evitar que la información este atenta a modificaciones de ataques. Para aprovechar dichas tecnologías, a nivel alto se puede usar él envió de data a través de una dirección URL HTTPS, a nivel bajo el API CFNetwork para negociar conexiones SSL o TLS, sirve para crear, envío y recepción de mensajes de manera segura por una red, y se puede usar Secure Transport API (Apple, 2018). Para el envío de datos por una red, se menciona el uso de dos protocolos TLS y SSL, así mismo, se menciona dos tecnologías como CFNetwork y Secure Transport.

Con lo visto tanto en Android y iOS, ambos hacen uso de llaves públicas y privadas, encriptación simétrica y asimétrica, el uso de protocolos como TLS y SSL para el envío y recepción de datos entre cliente- servidor. En Android maneja con 3 clases principales ya mencionadas y IOS los nombra tecnologías que tiene varias para casos en específicos. Ahora con respecto a los protocolos TLS y SSL, en Android se diferencia que tiene un servicio de Autorización de Certificados que soluciona el problema de que el servidor externo tenga certificaciones desfasadas que conllevaría una vulnerabilidad a diferencia de iOS que proporciona claves públicas y privadas, además de, conexiones por medio de TLS y SSL, con respecto a la encriptación de datos

del usuario, ambos tiene un nivel de seguridad a servicio del desarrollador similares, con la diferencia de un mejor manejo de Android con los servidores de terceros.

Más que todo, como ya se vio, el desarrollador tiene múltiples herramientas para dar una óptima y eficiente seguridad a una aplicación tanto iOS y Android, todo va por un tema de buenas prácticas, conocimientos y experiencias del desarrollador.

Ahora si lo relacionamos con los pilares de la seguridad, los 5 puntos (Confidencialidad, Integridad, Disponibilidad, No repudio, y Autentificación) se muestra en la encriptación de datos, puesto que la propiedad de la información solo está disponible a la persona autorizada, solo a ella, la integridad de la información no se ve afectada por el uso de protocolos de seguridad y las claves simétricas y asimétricas, disponibilidad de información, solo accesible por la persona autorizada.

5.2.6. Revisión de Software

Por un lado, Android en sus aplicaciones móviles antes de aceptarlos en Google Play pasa por una revisión final. Google (2018), cuenta con un programa denominado App Security Improvement, la cual es un servicio para los desarrolladores de aplicaciones Android, consiste en mejorar la seguridad de las aplicaciones de estos. Dicho programa tiene la finalidad de dar sugerencias, recomendaciones, buenas practicas, entre otros para que los desarrolladores puedan dar una mejor seguridad a sus aplicaciones. Es decir, Android proporciona dicho servicio para mejorar la seguridad.

El programa en mención, funciona de la siguiente manera:

Antes de aceptar cualquier aplicación en Google Play, la examinamos en busca de seguridad, incluidos posibles problemas de seguridad. También reexaminamos continuamente las más de un millón de aplicaciones en Google Play para detectar amenazas adicionales. Si su aplicación está marcada por un posible problema de seguridad, le notificaremos de inmediato para ayudarlo a solucionar el problema rápidamente y ayudar a mantener seguros a sus usuarios. Le enviaremos alertas utilizando tanto el correo electrónico como Google Play Console, con enlaces a una página de soporte con detalles sobre cómo mejorar la aplicación. (Google, 2018)

El programa ya mencionado, examina las aplicaciones de Google Play en busca de fallos de seguridad, al encontrar uno notifica al desarrollador mediante correo y si es necesario una página soporte para ayudarlo en el problema de seguridad de la aplicación.

5.3.Seguridad en Aplicaciones Bancarias

5.3.1. Resultados de Accenture

El comportamiento del usuario final y las buenas prácticas de desarrollo seguro son esenciales para la seguridad de los datos, tanto para su almacenado como su transmisión, sobre todo en aplicaciones de banca móvil.

De acuerdo con un análisis de NowSecure, el 35 por ciento de las comunicaciones enviadas por dispositivos móviles son no cifradas y el dispositivo promedio se conecta a 160 direcciones IP únicas diariamente. Además, se estima que el 43% de los usuarios no usan un patrón de bloqueo código PIN.

Aplicaciones, infraestructura, acceso a vulnerabilidades, datos confidenciales y el aumento de puntos de conexión de red, son desafíos adicionales en un entorno móvil. (Accenture, 2018)

Los porcentajes anteriores fueron tomados en EEUU, lo cual nos sirve como una referencia debido a que al ser un país desarrollado, sobre todo en el rango de TI nos muestra el grado de falta de cultura de seguridad que existe, y que probablemente se repita o agrave en países en vías de desarrollo como lo es el Perú.

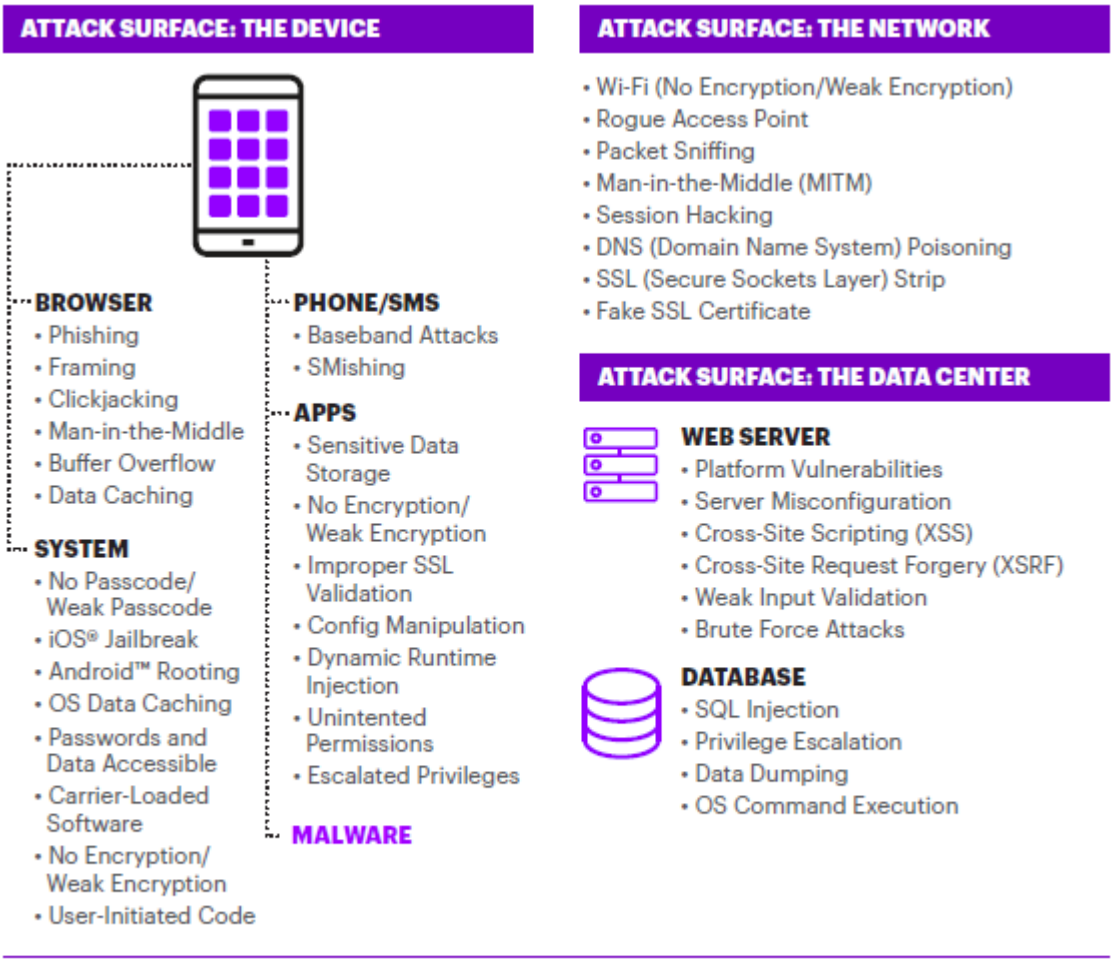
El artículo anterior menciona al cifrado, y Esset describe de la siguiente manera

Cifrar o encriptar datos significa alterarlos, generalmente mediante el uso de una clave, de modo que no sean legibles para quienes no posean dicha clave. Luego, a través del proceso de descifrado, aquellos que sí poseen la clave podrán utilizarla para obtener la información original.

Esta técnica protege la información sensible de una organización, ya que, si los datos cifrados son interceptados, no podrán ser leídos (Esset, 2017).

El cifrado de datos es una técnica n fundamental en los sistemas informáticos, desde contraseñas almacenadas hasta las comunicaciones, en el caso específico de celulares y aplicaciones bancarias la información transmitida y el uso de redes wifi hace más fácil la obtención de datos sensibles que enviamos, como nuestras transferencias y claves de seguridad. El cifrado de claves nos asegura que solo el receptor autorizado pueda leer nuestra información, mientras cualquier intruso que la robe solo obtenga datos desordenados y sin sentido.

Al momento de crear una aplicación bancaria, esta deberá tener como máxima prioridad la seguridad, aun por encima del diseño o comodidad de la misma, debido a que la información sensible que maneja pasa por 3 puntos en la cadena tecnológica donde se pueden explotar vulnerabilidades: el dispositivo, la red y el centro de datos.



Source: "Secure Mobile Development Best Practices," NowSecure, 2016. Access at: https://www.nowsecure.com/ebooks/secure-mobile-development-best-practices/

Como se puede apreciar hay muchos posibles riesgos en el flujo tecnológico , de las 3 partes la más segura son los data center, pues cuenta con la mayor infraestructura de seguridad en la mayoría de los casos, luego están la redes y finalmente como punto más flojo del flujo los dispositivos móviles, los cuales si nos orientamos a aplicaciones bancarias esta pueden verse

afectadas por temas de Rooting, contraseñas débiles, almacenamiento de información sensibles en lugares inadecuados , privilegios de otras aplicaciones y certificados SSL inapropiados.

El estudio de vulnerabilidades descrito de ACCENTURE fue realizado a finales del 2016 en 30 aplicaciones bancarias en iOS y Android en EEUU y revelo que de las 465 pruebas completadas para aplicaciones bancarias que se ejecutan en Android ™ OS, 44 o el nueve por ciento tenía problemas de baja seguridad; 48 o 10 el por ciento tenía problemas de seguridad medios; y 10 o dos por ciento problemas de seguridad de alto nivel.

Desarrollar y dar seguridad a aplicaciones móviles financieras contempla no solo el hecho de encriptar información, también todo el siglo de desarrollo de software ,así como las buenas prácticas , con el fin de evitar caer en las vulnerabilidades más conocidas que se presentan en una plataforma móvil, como lo explica Accenture en el siguiente cuadro, que contiene el tipo de análisis hecho, el sector al que pertenece el problema, el problema como tal, el impacto y el porcentaje de aplicaciones que cuentan con esta

vulnerabilidad.

	#	Analysis	Section	Issue	Impact	CVSS*	%
Android™	1	Dynamic	Permissions	World-Writable Files	High	7.7	33%
	2	Dynamic	Network	Broken SSL and Sensitive Data in Transit (with Encryption)	High	7.4	13%
	3	Dynamic	Permissions	Writable Executables	High	7.7	7%
	4	Static	Code	Obfuscation	Medium	N/A	60%
	5	Static	Code	SecureRandom	Medium	5.5	73%
	6	Static	Code	Dynamic Code Loading	Medium	4.3	33%
iOS®	7	Dynamic	Network	Cookie "HttpOnly" Tag	Medium	5.3	40%
	8	Dynamic	Network	Cookie "Secure" Tag	Medium	5.3	54%
	9	Dynamic	Network	TLS Traffic with Sensitive Data	Low	1.6	80%
	10	Static	Network	App Transport Security	Low	N/A	60%

Como primer punto, Archivos de escritura globales de sobre escritura: Con esto se refiere a los archivos globales que creados al momento de usar una aplicación y que pueden, existe el riesgo que otras aplicaciones tengan los permisos necesarios para sobrescribir o acceder a archivos. En el estudio realizado el por Accenture el 33% de las aplicaciones bancarias poseen archivos que podían ser modificados por otras aplicaciones.

Como segundo punto, Revisión SSL inadecuada: Significa que una aplicación no tiene un certificado de validación adecuado o que no hacían una verificación de nombre de host, lo cual deja la oportunidad de realizar un ataque de hombre en el medio, esto se refiere a poder interceptar los datos

a mitad de su envió, verlos y alterarlos sin que su emisor y receptor legales se den cuenta. El 13% de las aplicaciones en Android tenían este problema.

Como tercer punto, Archivos de ejecución de sobre escritura: Por si solos no generan un problema, pero combinado con otras vulnerabilidades pueden dar on control remoto de tu aplicación, el 7 % de las aplicaciones Android tiene este problema.

Como cuarto punto, Ofuscación: El código fuente no está ofuscado, es decir a través de ingeniera inversa es fácil lograr saber cómo funciona la aplicación a nivel de desarrollador, lo que puede dar oportunidad a encontrar puertas traseras o bugs a explotar, en Android se identificó esta vulnerabilidad en el 60% de las aplicaciones.

Como quinto punto, Seguridad Aleatoria: Las aplicaciones que usan el JCA(Arquitectura de Criptografía de Java), para generar claves o firmas usan un generador de números pseudo aletorios (PNRG) puede que no reciban valores encriptados fuertes por una mala inicialización, también tiene el problema las apps que invocan directamente el SSL PNRG. El 73% de las apps en Android tenían este problema.

Como sexto punto, Carga de Código Dinámico: Al momento de implementar una aplicación que especifica que recursos y bibliotecas deben ser cargadas por defecto al momento de iniciar la aplicación. Normalmente dentro del APK se cargan estas bibliotecas, pero también se debe especificar qué cosas no deben cargar sin que haya un pedido o una interacción en específico con la aplicación. Dejar métodos y librerías cargadas por defecto

sin necesidad de usarse es una mala práctica y un riesgo para la aplicación. El 33% de las aplicaciones en Android tenían este problema.

Como séptimo punto, Cookie HttpOnly: Al momento de crear un cookie y ponerle el httpOnly nos aseguramos que el buscador solo deje usarlo en servidor y no del lado cliente, evitando un ataque de XSS o porJavaScript.

Como octavo punto, Cookie Secure Tag: Si esta bandera esta como verdadero se asegura que el navegador solo envie el cookie si se está usando un canal de transmisión segura. Asi se evita que el cookie se envie sin encriptar.

Para finalizar, Transporte de data sensible en TLS y SSH: Se debe evitar que información muy sensible pase por estos canales como Usuario, Contraseña, Coordenadas, Dirección Wifi MAC, IMEI sin una autentificación pinning (más información: https://blog.elevenpaths.com/2013/08/certificate-pinning-el-que-el-como-y-el.html)

Basados en estos datos podemos observar que desde el punto de vista del desarrollo hay muchos factores que se pueden implementar para mejorar la seguridad de las aplicaciones móviles, estas no requieren de una gran infraestructura tecnológica para aplicarse y son más bien responsabilidad del desarrollar implementarlas y de las instituciones y usuarios exigirlas.

Un gran defecto en las prácticas de desarrollo en general es dejar variables o código por defecto, esto como ya se explicó hace bastante expuesta a la aplicación, ya sea por ser una vulnerabilidad conocida o por la facilidad de aplicarse ingeniería inversa en ella. También como explica Accenture “Estándares y marcos, como el examen de las instituciones financieras

federales Consejo (FFIEC) y el Instituto Nacional de Estándares y Tecnología (NIST), proporcionan orientación sobre el empleo de múltiples factores autenticación para aumentar la seguridad de aplicaciones de banca móvil".

También debemos mencionar que los mecanismos de autenticación de las aplicaciones móviles bancarias deben tener mayores mecanismo de autenticación, en este caso aún por encima de la comodidad del usuario se debe poner su seguridad, así también lo explica Accenture en su informe

"Muchas instituciones financieras que requieren autenticación de múltiples factores para la atención al cliente las aplicaciones web aprovechan SMS (Corto Servicio de mensajes) tecnología a través de fuera de banda comunicación. SMS y otras formas de la tecnología de comunicación fuera de banda son inherentemente inseguros y pueden ser comprometidos por un atacante experto. En base a nuestro análisis y observaciones, autenticación de múltiples factores hace en línea banca más segura al reducir la exposición para la mayor amenaza individual para tomar posesión de la cuenta, phishing y credenciales de cuenta mal aprobadas."

Según Rojas (2016), en una revisión del software de varias aplicaciones móviles Android del sector bancario en Chile, identifico varios problemas de seguridad de dichas aplicaciones. Entre ellos, uso indebido de los permisos o privilegios de la aplicación, es decir, que la aplicación acceda a recursos como sistema de archivos, contactos entre otros; no implementar time outs para los cierre de sesión o mal implementados; que la información usada se almacene en texto plano o visible para cualquier persona; también,

las peticiones enviadas al servidor se usa HTTPS como protocolo de transporte seguro, pero con una mala configuración es inseguro. Así mismo, el uso de malas prácticas debido a los pocos conocimientos en seguridad de los desarrolladores da como resultado aplicaciones inseguras, como en este caso, una aplicación móvil Android insegura.

Ahora si con la información mencionada lo relacionamos con los mecanismos ya antes mencionado, se debe poner mayor énfasis en la encriptación de la información, permisos de aplicación, manejo de información personal del usuario y la forma como se envía la información al servidor.

Si hemos hablado acerca de qué mecanismos de seguridad se debe aplicar para implementar una aplicación móvil bancaria segura, debemos considerar el ámbito en el cual dicha aplicación debe ser desarrollada. Con respecto a ámbito se refiere que versión de Android actualmente tiene un mayor nivel de seguridad. Según Google (2018), Android 7.0 Nougat (API 24) es la versión de Android más segura, mejora su arquitectura para brindar varios niveles de seguridad para la protección de datos. Cifra todos los datos usando una clave de cifrado de 128 bits; así mismo, la clave de cifrado va ligado a la credencial de la pantalla de bloqueo (huello o patrón), para solo así el usuario propietario del móvil pueda descifrar los datos.

Así mismo, Panda Security (2016) describe que dicha versión es la más segura, añade al Sistema Operativo un nuevo sistema de cifrado, mejora para acceder al servicio de VPN, ya no permite el uso de permisos compartidos entre aplicaciones con el fin de alterar información sensible del dispositivo.

Con lo descrito por Android y Panda Security, la versión Nougat de Android viene a ser la más segura entre las versiones que tiene. Incorpora un nuevo

5.4.Conclusiones

Como conclusiones podemos decir que los 5 mecanismos de seguridad descritos por Jamdaade (2016): Arquitectura, autenticación, procedencia de aplicaciones, permisos de aplicaciones, aislamiento son los apropiados a estudiarse para medir las seguridades de las aplicaciones móviles, aunque Mohammed (2017) a través de la encuesta que realiza se puede concluir que es necesario que el usuario tenga una buena cultura de la seguridad de la información para estar realmente protegido. Podríamos de decir que Jamdaade olvido de contemplar este factor no técnico al momento de seleccionar los mecanismos, pues el usuario final es el que usara el producto, más allá de las medidas de seguridad que puedan usar los desarrolladores o compañías.

Así mismo, como también el usuario final debe tener cierta cultura de seguridad de la información y buenas hábitos en seguridad en el uso de sus aplicaciones y dispositivos móviles. Los desarrolladores deben tener buenas prácticas al momento de desarrollar una aplicación móvil cualquiera.

Para terminar, una aplicación móvil segura implica:

- Cultura de seguridad de los usuarios
- Buenas practicas del desarrollador

Con respecto a las buenas practicas del desarrollador:

- Realizar un Login seguro con los filtros o validaciones necesarias para determinar si el sujeto dice quien dice ser.

- El uso de protocolos de seguridad para envío y recepción de datos por web, como HTTPS y SSL.

- Encriptar la información sensible del usuario (password) usando algoritmos de Encriptamiento seguro.
- Toda entrada de datos en la app, se debe validar lo ingresado.
- Generar token de seguridad dinámico para realizar una transferencia.
- Para envío de información sensible (password) usar un teclado aleatorio.
- Considerar la versión de SO Android con mayor nivel de seguridad.
- Ocultar parcialmente información del usuario.
- Ocultar contraseña cuando se muestre.
- No declarar permisos innecesarios.
- Considerar la red a la que se conecta y si el dispositivo tiene root.

6. Metodología de la investigación

6.1. Definición de Tecnologías

Para la presente implementación, y con lo descrito en el estado de la cuestión, la siguiente lista muestra las herramientas y/o tecnologías, para la implementación de una aplicación Android para el sector bancaria, con las funciones básicas que debe tener dicha aplicación, considerando la seguridad sobre otros aspectos.

6.2. Requerimientos de Seguridad

Como cualquier implementación de cualquier sistema, se detalla los requerimientos de sistema que requiere dicha implementación. A continuación, se enumerará todos los requerimientos relacionados con la seguridad de la aplicación móvil, solamente todo relacionado con la seguridad. Los requerimientos enlistados se tomaron en

consideración lo investigado en el estado de la cuestión y el análisis de las aplicaciones móviles bancarias en Lima, Perú.

- La aplicación no debe permitir ejecutarse en un dispositivo con root. (REQ1)
- La aplicación debe permitir ejecutarse solo en red de datos. (REQ2)
- La aplicación debe ocultar con el carácter especial "*" la contraseña. (RE3)
- La aplicación debe enviar la contraseña de forma encriptado al WebServices. (REQ4)
- La contraseña debe guardarse de forma encriptado en la Base de Datos. (REQ5)
- La aplicación debe ocultar parcialmente con el carácter especial "*" toda información sensible del usuario. (REQ6)
- La aplicación debe realizar peticiones mediante protocolos seguros HTTPS y SSL. (REQ7)
- La aplicación solo se debe declarar los permisos de celular necesarios para su uso. (REQ8)
- La aplicación debe usar un teclado numérico aleatorio para el ingreso de contraseña. (REQ9)
- La aplicación solo debe ejecutarse con versión Android 7.0 (REQ10)
- La aplicación solo debe realizar una transferencia bancaria con un token dinámico. (REQ11)

6.3. Análisis App Móvil – Lima

A continuación detallaremos un análisis realizado a 3 aplicaciones bancarias, en el cual rescataremos mecanismos de seguridad usados y a la vez las carencias en seguridad que tiene. El análisis solo se limita a lo observado con el uso de dicha aplicación, ya sea en iOS o Android, dicho análisis será cortejado con los

requerimientos de seguridad que se realizó en el punto anterior. Así mismo, cualquier requerimiento que no se contempló en el trabajo de investigación

6.3.1. Interbank

Comparación con respecto a los requerimientos.

	Si	No	Falta Información
REQ1		X	
REQ2		X	
REQ3	X		
REQ4			X
REQ5			X
REQ6	X		
REQ7			X
REQ8	X		
REQ9		X	
REQ10		X	
REQ11	X		

Por un lado, se resalta el uso de DNI, Clave Web y Número de Tarjeta, en caso de no optar por la Clave Web, adiciona detector de huella para poder iniciar sesión desde la cuenta bancaria. Así mismo, cualquier transacción se realiza por medio de un Token que envía la clave al número de celular afiliado.

Por otro lado, usa teclado default propio del sistema operativo el cual genera una vulnerabilidad en caso que se esté guardando todo lo registrado por el teclado.

6.3.2. BCP

Comparación con respecto a los requerimientos.

	Si	No	Falta Información
REQ1	X		
REQ2		X	
REQ3	X		
REQ4			X
REQ5			X
REQ6	X		
REQ7			X
REQ8			X
REQ9	X		
REQ10		X	
REQ11	X		

Por un lado, se resalta que usa un login con dos pasos, ingresa solo el número de tarjeta y una clave web. Así mismo, posee un teclado aleatorio para información como contraseña.

Por otro lado, usa un Token virtual para la confirmación de cualquier operación.

6.3.3. Scotiabank

Comparación con respecto a los requerimientos.

	Si	No	Falta Información
REQ1		X	
REQ2		X	
REQ3	X		
REQ4			X
REQ5			X
REQ6	X		
REQ7			X
REQ8			X
REQ9	X		
REQ10			X
REQ11	X		

Por un lado, un punto a resaltar de Scotiabank, es su Login con varias validaciones; solicita Número de tarjeta, Clave Pin, validar imagen y fecha de cumpleaños. Es decir, solicita al usuario varias credenciales para saber que es quien dice ser.

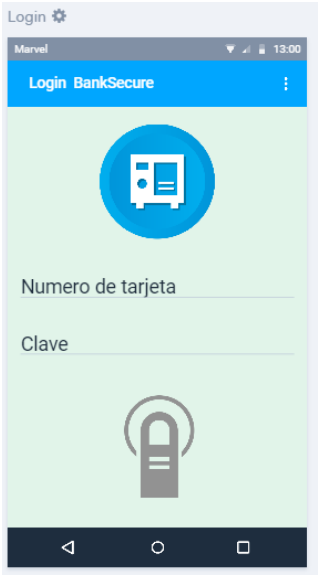
Por otro lado, como cualquier operación requiere de una validación como un Token para realizar dicha transacción.

6.4.Prototipos Iniciales

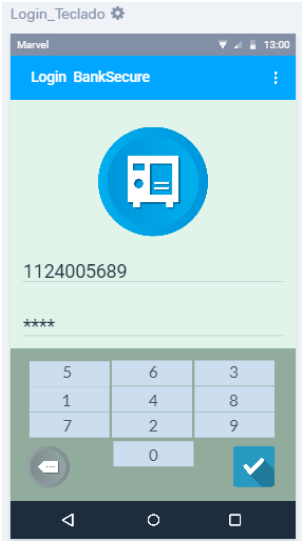
Para el desarrollo de prototipos del diseño inicial y no el cual será el diseño final, usaremos una herramienta Online MarvelApp para que diseño de prototipos de la aplicación móvil.

6.4.1. Login

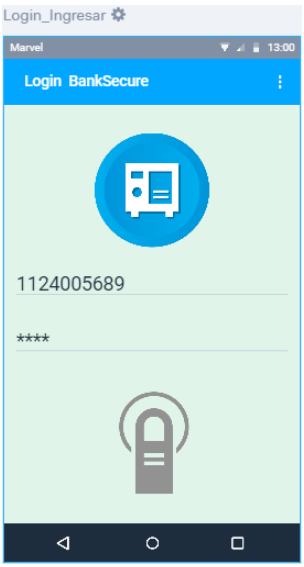
La interfaz de login de la cuenta bancaria deberá tener consideraciones de seguridad que en las siguientes imágenes se explicara. La siguiente imagen es un login simple donde solicita el número de cuenta y una clave que no es el patrón que se introduce en los cajeros automáticos.



Por motivos de seguridad no se debe usar el teclado por default del dispositivo móvil o un tercero, si no se debe diseñar uno para la introducción de la clave y el número de cuenta bancaria, así como se muestra en la siguiente imagen. Así mismo, el teclado numérico que se muestra debe ser en orden aleatorio y no uno normal que empiece desde arriba del 9 hasta el 0.

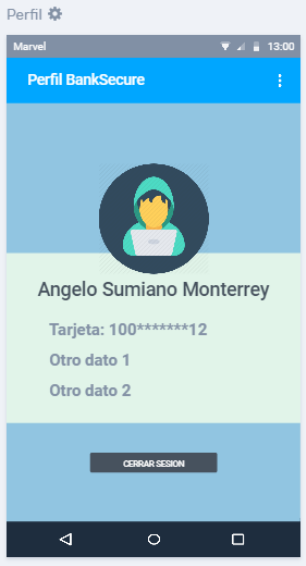


Así mismo, otro medio de autentificarse con la cuenta bancaria es usando la huella digital, pero solo está habilitado para dispositivos que tengan soporte el uso de huella digital.



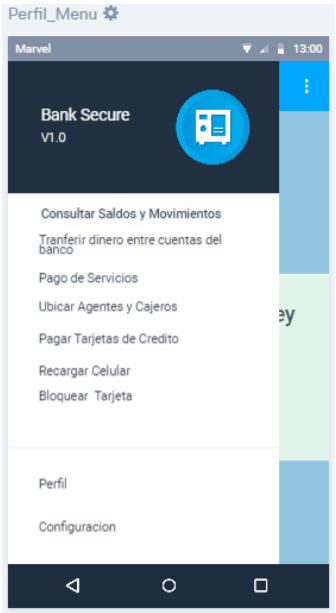
6.4.2. Perfil

La interfaz principal que es de Perfil, muestra datos de la persona pero ocultando parte de dichos datos con '*' por motivos de seguridad.



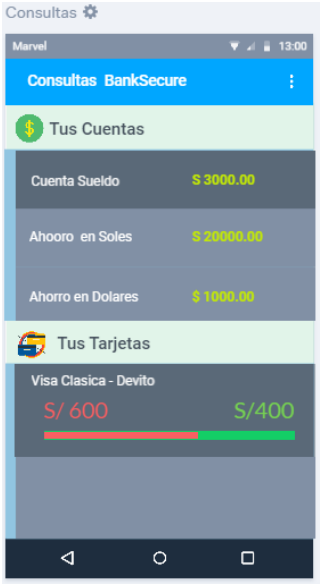
6.4.3. Menú

El menú de las operaciones que puede realizar una persona con cuenta bancaria por el dispositivo móvil, como se muestra tiene las operaciones que realiza cualquier banco acá en el Perú.



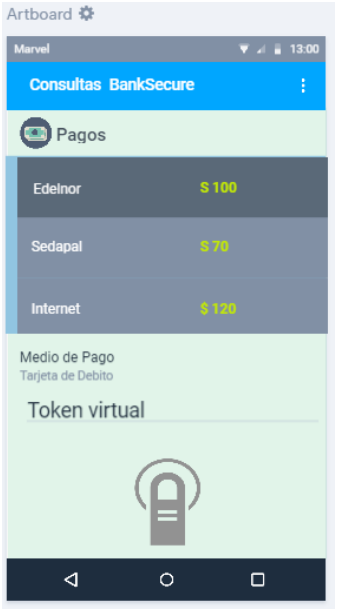
6.4.4. Consulta Estado de Cuenta

Aquí solo son consulta de saldo de la cuenta, tanto en soles como en dólares.



6.4.5. Transferencias y/o Operaciones

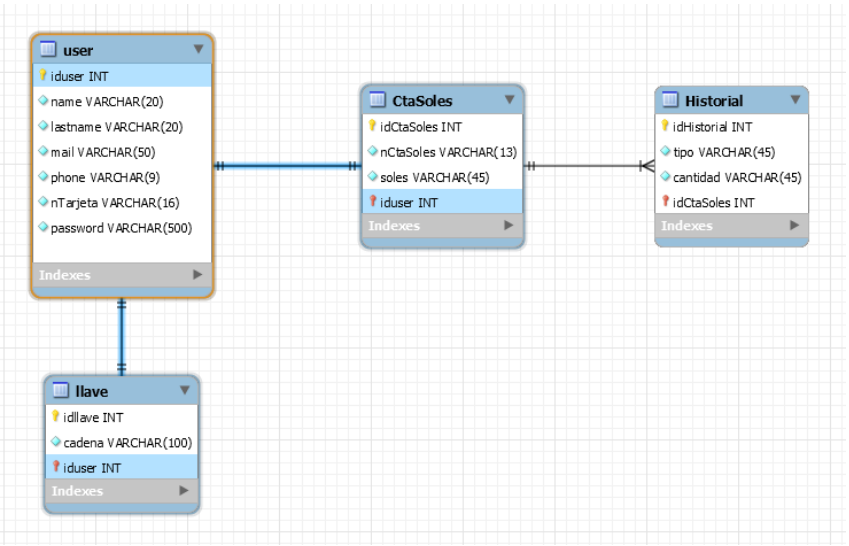
En esta interfaz, se realiza un ejemplo de una transacción u operación bancaria, el cual se requerirá una llave token para poder completar dicha operación. En este caso, una clave que se enviará al celular vinculado a la cuenta.



7. Resultados y/o Propuesta de Solución

7.1.Modelo de Base de Datos Básico

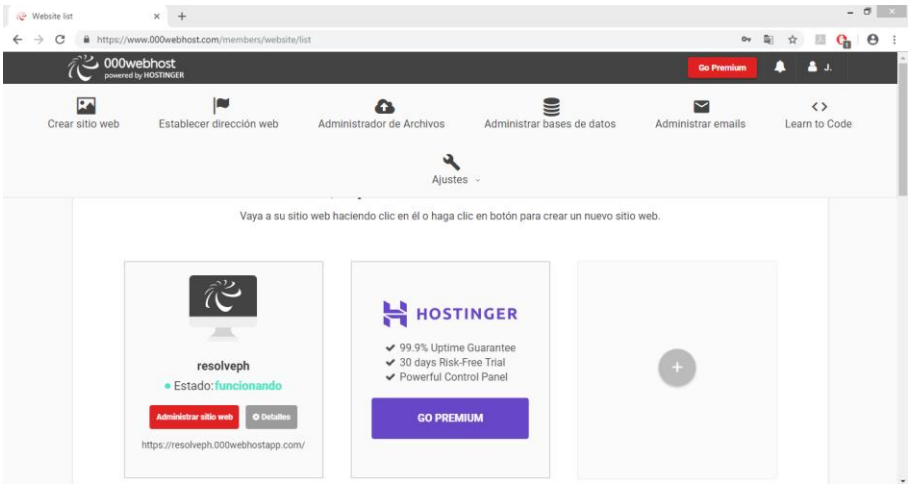
La siguiente imagen muestra el modelo de base de datos de la aplicación.



Al solo considerar implementar una aplicación móvil bancaria con respecto a la seguridad, se diseñó un modelo de base de datos bancaria básico que contenga lo básico con información de una persona con cuenta bancaria.

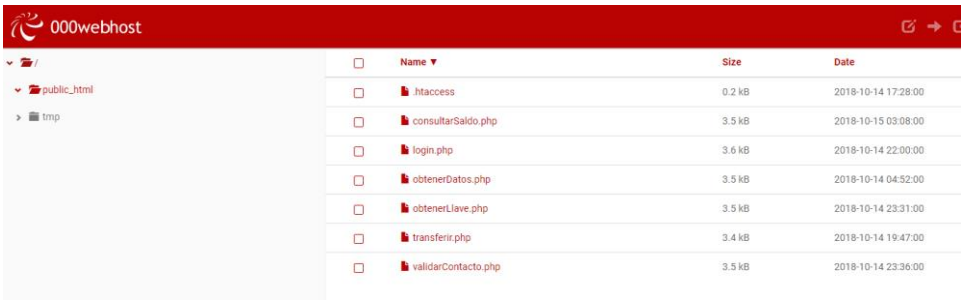
7.2.Web Service

Para la Web Service y base de datos se usó el servicio gratuito de Hostinger “000webhost”.



7.3.EndPoint

La siguiente imagen se visualiza los endpoint que responderán a las peticiones del cliente (Aplicación Android) y dependiendo retornara un valor o no según se requiera.



7.4.Base de Datos

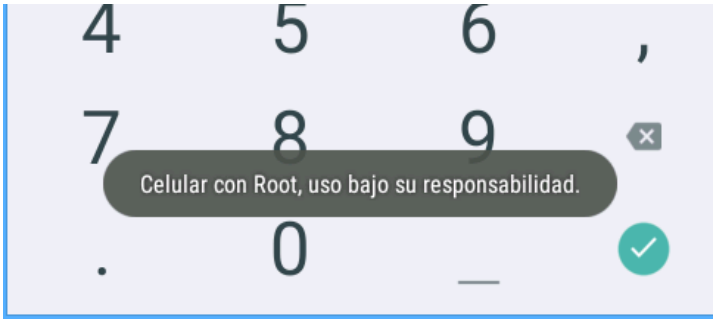
La siguiente imagen muestra las tablas insertadas en la base de datos gratuita de Hosting, usa la base de datos MariaDB.

7.5.Implementación de Mecanismos de Seguridad

En las siguientes ítems, de detalla la implementación de cada mecanismo de seguridad en una aplicación móvil bancaria modelo.

7.5.1. REQ1

El requerimiento 1, se valida si el celular o dispositivo móvil Android tiene o no Root. Con esto detectamos que dicha vulnerabilidad implica un hueco de seguridad que expone de una manera u otra la información de la aplicación móvil.

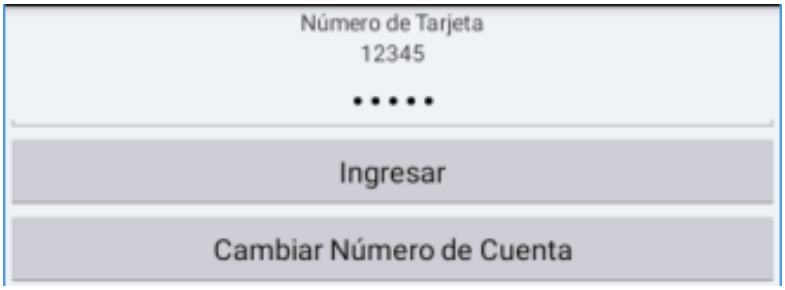


7.5.2. REQ2

El requerimiento 2, se valida el medio por donde se está accediendo a internet, es decir, se le informa al usuario si se conecta por wifi, red de datos o no tiene acceso a internet. En un ambito real, se recomienda no acceder a redes wifi no seguras, por lo que solo se debe acceder por red de datos del celular. Así mismo, en la aplicación solo notificamos al usuario con el fin de que pueda usar la aplicación, pero ya en un caso real, se recomienda restringir.

7.5.3. REQ3

El requerimiento 3, lo más común y que se ha vuelto estándar es oculta lo escrito en la contraseña, en la imagen se muestra que oculta la información de la contraseña.



7.5.4. REQ4 y REQ5

Tanto el requerimiento 4 y 5, es acerca de encriptar la contraseña, enviar a la base de datos y almacenarla. Para el encriptar la contraseña, usamos 2 parámetros:

- Contraseña ingresada por el usuario.
- Llave de seguridad.

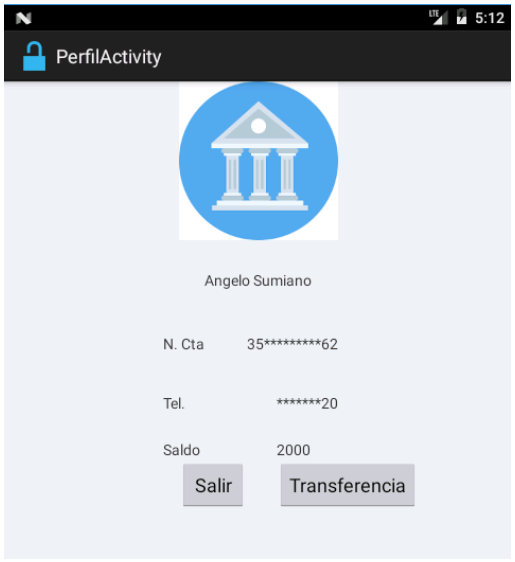
Con dicho parámetros se genera la contraseña encriptado para almacenar en la base de datos.

password
pDEoQM9zneB9yrfhU4RNLg==
pDEoQM9zneB9yrfhU4RNLg==

idllave	cadena	iduser
1	2422	1
2	2422	2

7.5.5. REQ6

El requerimiento 6, consta de ocultar parcialmente información del usuario con el fin de no tener legible por si otra persona está espiando u otra cosa.



7.5.6. REQ7

El requerimiento 7, consta del uso de protocolos seguros para el envío y recepción de datos desde la base de datos, se usa tanto HTTPS y SSL.

```
ruta="https://resolveph.000webhostapp.com/login.php";
```

```
SSLContext sc = SSLContext.getInstance("SSL");
```

7.5.7. REQ8 y REQ10

Los requerimientos 8 y 10, son acerca de la versión segura den android (7.0, API24) y el uso adecuado de permisos de aplicación. En este caso usamos el API24 Nougat, puesto que es la versión con mayor nivel de seguridad en Android y solo declaramos el permiso de internet y la red a la que se accede.

AppBancaria Manifest

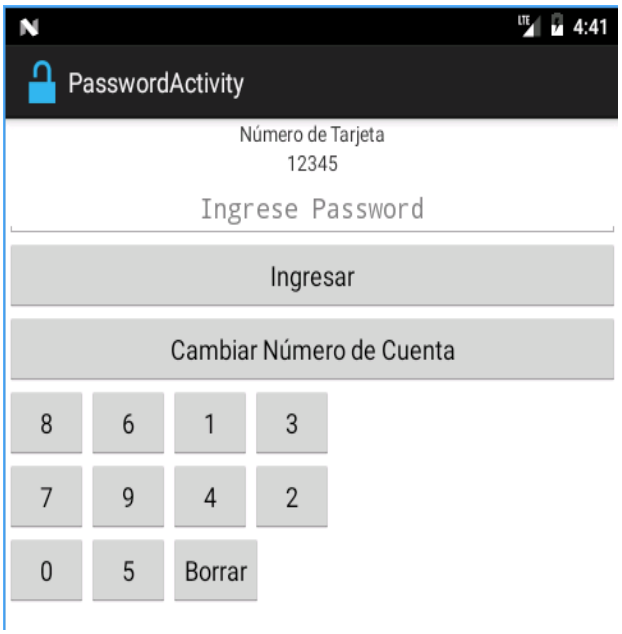
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.appbancaria"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" /> />

    <uses-sdk
        android:minSdkVersion="24"
        android:targetSdkVersion="24" />
```

7.5.8. REQ9

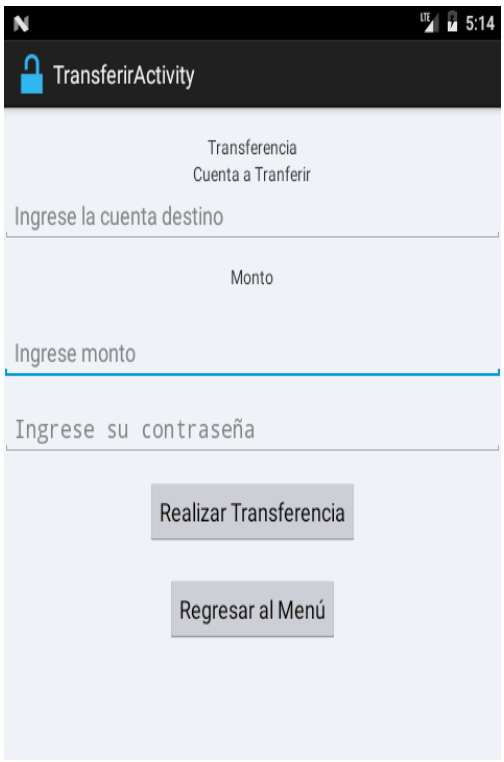
El requerimiento 9, se implementa un teclado aleatorio de seguridad al usuario con el fin de tener un mayor nivel de seguridad, puesto que los teclados por default muchas veces almacenan todo lo que se introdujo y esto viene a ser una vulnerabilidad.



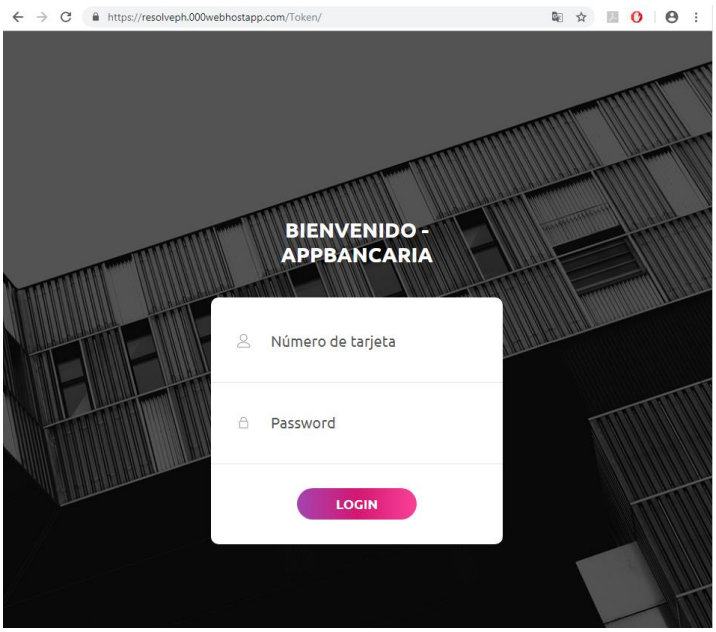
7.5.9. REQ11

En el requerimiento 11, para realizar una transferencia bancaria se requiere de un token virtual, dicho token se puede implementar de diversas maneras,

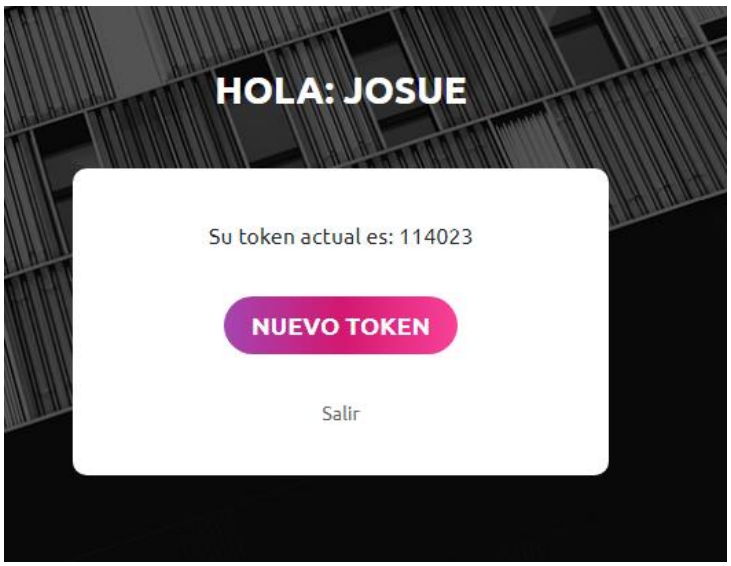
en dicho caso se usó una página web para la generación de claves aleatorios para el token y así poder realizar transferencias. Cada vez que se usa un toque, este se actualiza, para así tener un mejor nivel



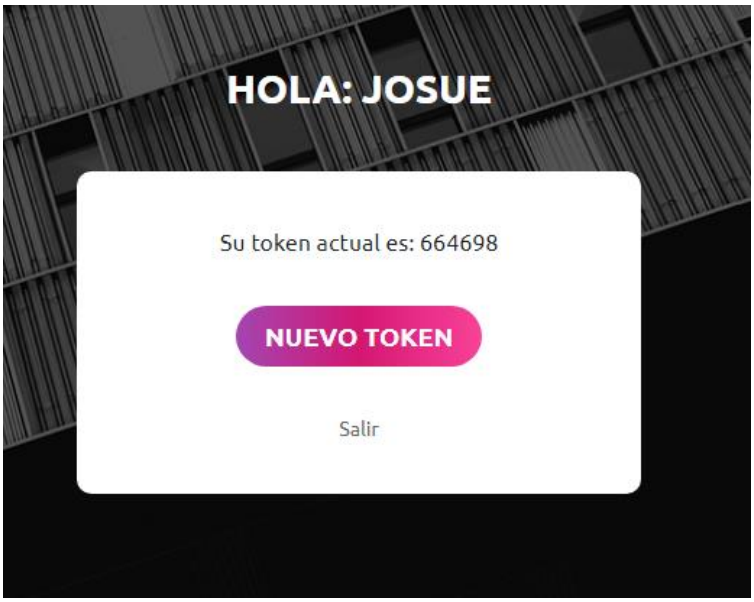
Mediante la página web, el usuario entra con su número de tarjeta y su clave web, para visualizar su actual token y si requiere, poder cambiarlo.



El usuario visualiza su actual Token virtual, y si desea puede actualizar dicho Token.

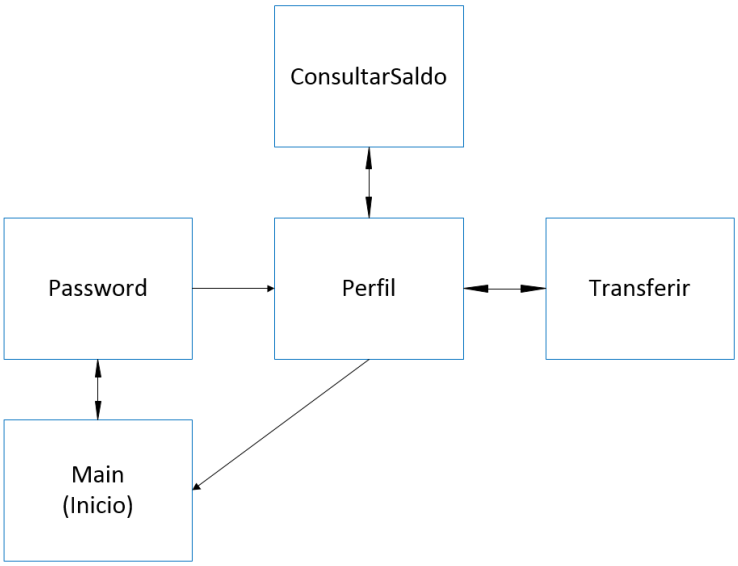


Si el usuario desea, puede actualizar su token manualmente, al realizar una operación bancaria, el token se actualiza automáticamente.



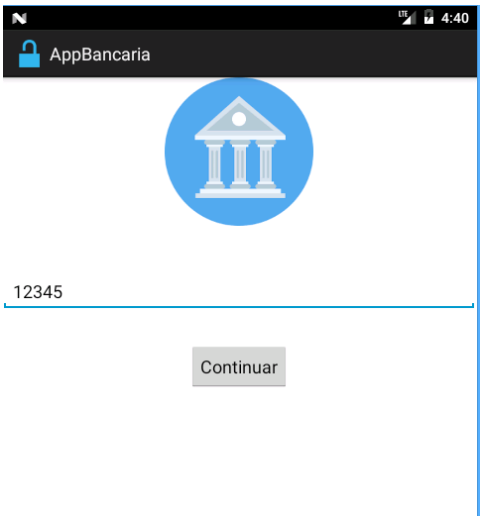
7.6.Flujo de la Aplicación Móvil

La actividad inicial es MainActivity. La siguiente imagen muestra el flujo de la aplicación móvil.



7.6.1. MainActivity

Dicha actividad solo comprende el ingreso del número de tarjeta del usuario. Para fines de pruebas solo se usa un número simple en número de tarjeta. El botón continuar me dirige a la actividad PasswordActivity.



En caso de que no ingrese nada la aplicación notifica que no ingreso nada. Así mismo, si el número de tarjeta es incorrecto.

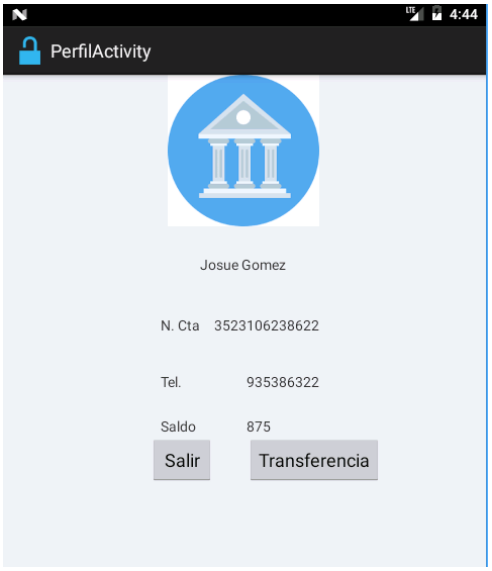
7.6.2. PasswordActivity

En dicha actividad para el ingreso de datos de la contraseña se usa un teclado aleatorio numérico. El botón Ingresar valida si la cuenta es correcta y dirige a la actividad PerfilActivity. El botón Cambiar Número de Cuenta regresa a la actividad anterior MainActivity para ingresar otro número en caso se ingresó mal.

Si no ingresa nada, le sale un mensaje diciendo ingrese password, en caso de que ingreso mal la contraseña, le notifica al usuario contraseña error.Solo podrá intentar 3 veces la contraseña, luego de eso se bloquea la cuenta.

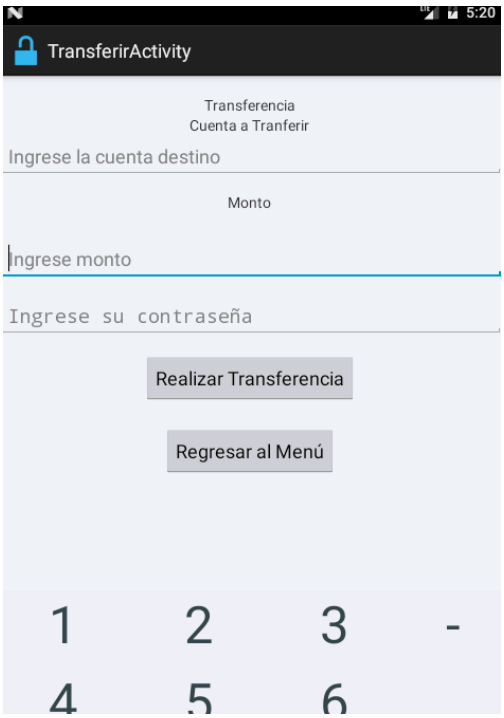
7.6.3. PerfilActivity

En dicha actividad muestra información acerca de la cuenta bancaria de un usuario. Muestra el botón Salir en caso de cerrar sesión o ingresar con otra cuenta. El botón Transferir redirige a la actividad TransferirActivity para realizar una transacción bancaria.



7.6.4. TransferirActivity

En dicha actividad se ingresa cuenta bancaria destino, el monto a enviar y el token virtual dinámico. Con el botón Realizar Transferencia se hace la operación bancaria. Con el botón Regresar al Menú redirige a PerfilActivity



Se tiene diferentes casos en esta actividad:

- Notifica si algún campo está vacío.
- Notifica si la cuenta destino es la misma que se está usado.
- Notifica si la cuenta destino no existe.
- Notifica si el monto excede los fondos de la tarjeta
- Notifica si el Token ingresado es incorrecto.

7.6.5. ConsultaSaldoActivity

En dicha actividad, el usuario puede consultar las operaciones que se han hecho en torno de su cuenta de saldo.

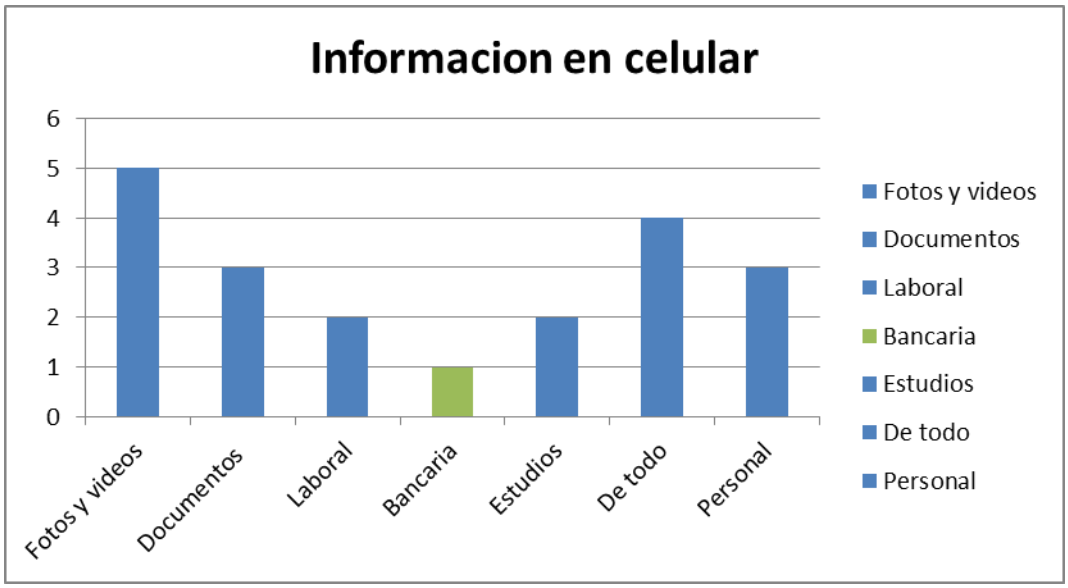
7.7. Resultados de las encuestas

7.7.1. Usuario

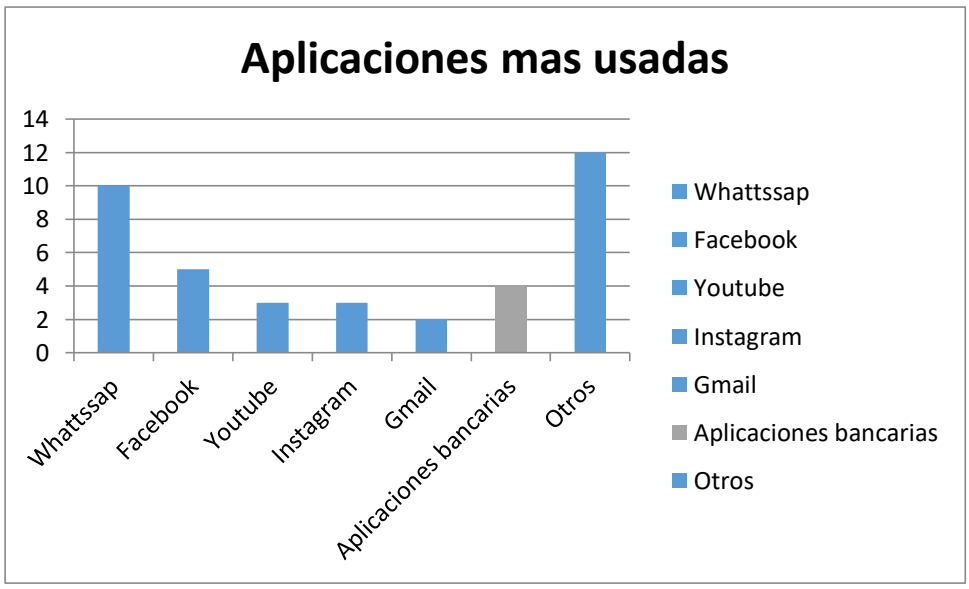
Ruta del documento Excel completo:

Información en celular	
Tipo de información en celular	**Conteo**
Fotos y videos	5
Documentos	3
Laboral	2
Bancaria	1

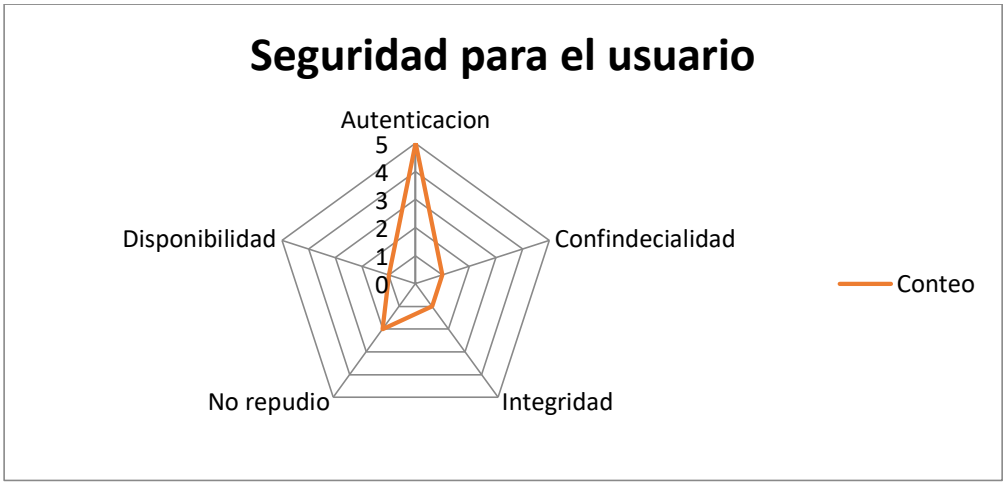
Estudios	2
De todo	4
Personal	3



Aplicaciones más usadas	
Aplicaciones bancarias	**Conteo**
Whattsaap	10
Facebook	5
Youtube	3
Instagram	3
Gmail	2
Aplicaciones bancarias	4
Otros	12



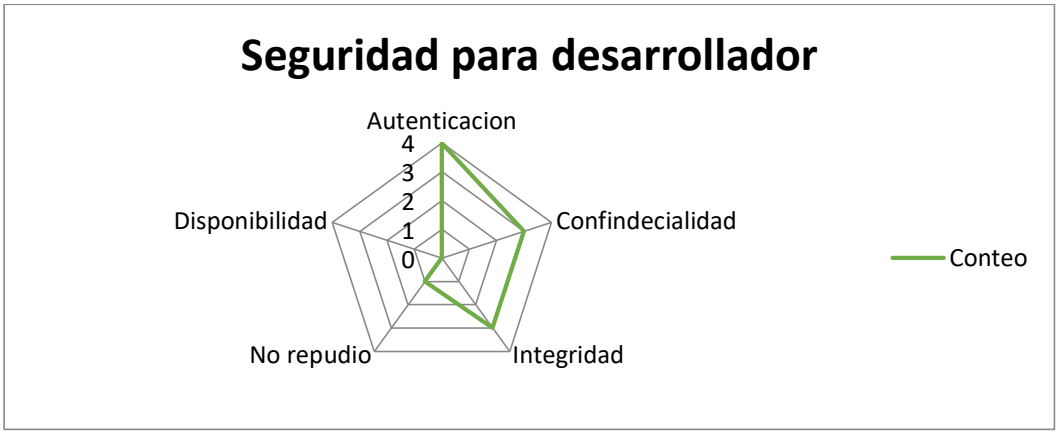
Pilar de Seguridad	Conteo
Autenticación	5
Confidencialidad	1
Integridad	1
No repudio	2
Disponibilidad	1



7.7.2. Desarrollador:

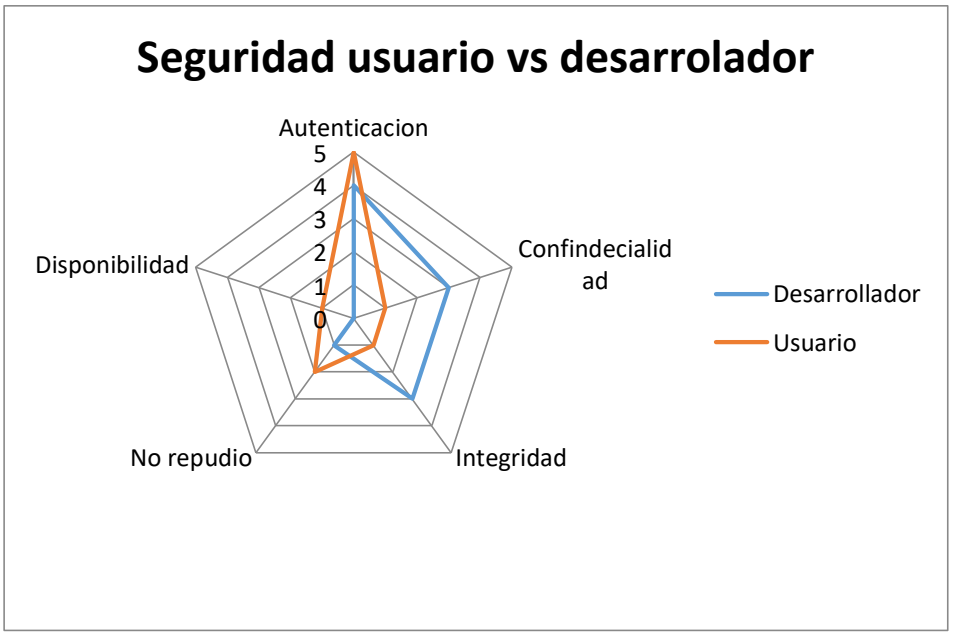
Ruta del documento Excel completo:

Pilar de Seguridad	Conteo
Autenticación	4
Confidencialidad	3
Integridad	3
No repudio	1
Disponibilidad	0



Comparación usuario vs Desarrollador

Pilar de Seguridad	Desarrollador	Usuario
Autenticacion	4	5
Confindecialidad	3	1
Integridad	3	1
No repudio	1	2
Disponibilidad	0	1

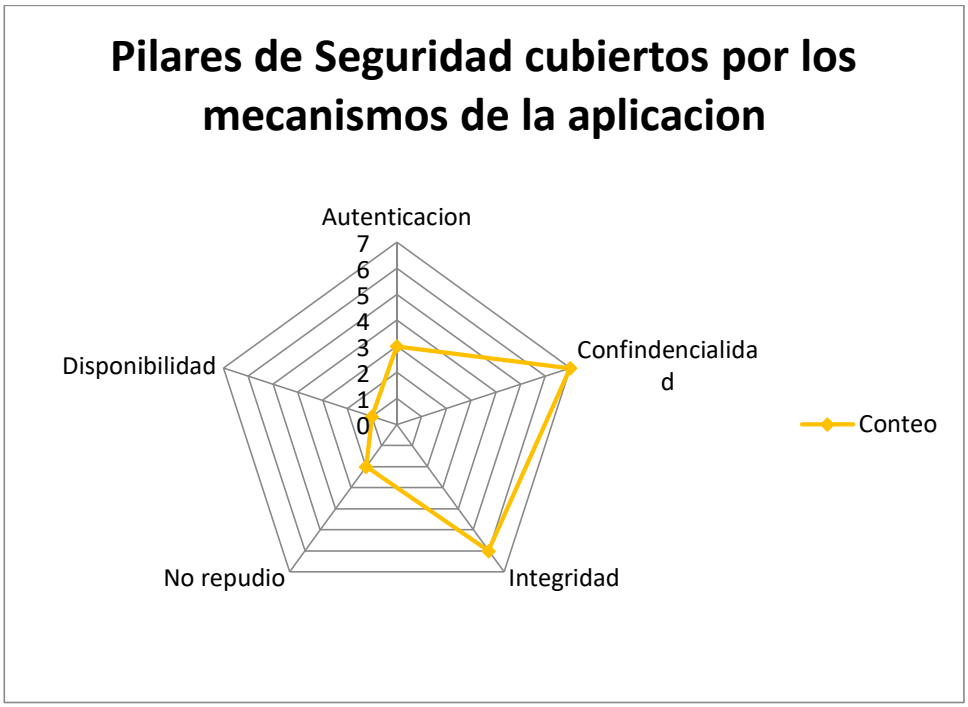


7.8.Pilares de Seguridad cubiertos por los mecanismos de la aplicación

Ruta del documento Excel completo:

Pilares de Seguridad de Información	Conteo
Autenticación	3
Confidencialidad	7

Integridad	6
No repudio	2
Disponibilidad	1



7.9.Resultados Finales

Los mecanismos de seguridad actuales tienden a cubrir más el tema de autenticación, descuidando otros pilares de la seguridad de la información que son igual de importantes aunque transparentes para el usuario final

El control de permisos adecuados, el uso de protocolos seguros con https y ssl, y la encriptación de datos en la base de datos son la base fundamental para el desarrollo de una aplicación web bancaria

La autenticación de los usuarios deberá ser de 2 o 3 capas para asegurar tanto el inicio de sesión como las transacciones que se deban hacer en la aplicación.

8. Conclusiones

Para concluir, con la recolección de información del estado de la cuestión, encuesta virtual y presencial, se determinó y/o identifico cuáles son aquellos mecanismos mínimos que debe tener una aplicación móvil de naturaleza bancaria. Entre las cuales lo que mayor prioridad se tiene, abarca la autentificación de un usuario, en el cual se desarrolla una forma en el que el usuario es quien dice ser; el Encriptamiento de datos de entrada y salida de la aplicación móvil. Dejando de lado la parte del desarrollo de la aplicación, de nada sirve programar una aplicación si el usuario final no tiene una cultura en la seguridad de la información, esto conlleva a un hueco de seguridad. Así mismo, las buenas practicas del desarrollador, puesto que una variable mal declarada, un método inseguro u otro, tienden a ser explotados por terceros para robar la información.

En sí, para desarrollar una aplicación bancaria considerando la seguridad se debe cumplir 3 ítems importantes:

- Buenas prácticas del desarrollador.
- Cultura de seguridad de la información del usuario final.
- Implementación de mecanismos de seguridad.

9. Recomendaciones

Desarrollar la aplicación con el uso de buenas prácticas como ITIL para una mejora en los procesos y desarrollo

Feedback de necesidades a futuro de los bancos que trabajen con aplicaciones móviles

Desarrollar mecanismos que permitan cubrir por igual los 5 pilares de la seguridad de la información

Tener un mayor conocimiento y experiencia de buenas prácticas en el desarrollo de aplicaciones móviles.

Abarcar otros ítems que acompaña el desarrollo de aplicaciones bancarias, como por ejemplo: servidor de base de datos, tipo de conexiones (VPN, Proxy, etc.) entre otros.

10. Cronograma

CRONOGRAMA DE TRABAJO																			
FASES Y ACTIVIDADES																			
ITEMS	SEMANA CLASES	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	F
I. Restructurar Estado de la Cuestión																			
1	Modificar Tema del Trabajo	x																	
2	Modificar Objetivo General y Especificos		x																
3	Adecuar Estado de la Cuestión con el Tema del Trabajo			x	x	x													
4	Entrega del Primer Avance						x												
II. Implementación del Trabajo de Investigación																			
1	Definir de Tecnologias				x														
2	Desarrollar Prototipos					x													
3	Desarrollo de Aplicación Móvil						x	x	x	x									
4	Documentación de lo realizado																		
5	Entrega del Segundo Avance (100%)										x								
III. Redacción de Informe																			
1	Desarrollar Entrevistas									x	x	x							
2	Analisis o feedback de la implemetación del app												x	x					
3	Documentación de lo realizado													x					
4	Entrega del Tercer Avance														x				
VI. Etapa Final																			
1	Levantar Observaciones															x			
2	Realizar feedback															x	x		
4	Entrega Trabajo Final																	x	

11. Bibliografía

Choi, J., Na, G., y Jeong, Y. (2016), *Hardware-assisted credential management scheme for preventing private data analysis from cloning attacks. Multimedia Tools & Applications*,75 Issue 22, 14833-14848.

Google (2018). *App security improvement program*. California, EU.: Google. Recuperado de https://developer.android.com/google/play/asi?hl=es-419

Google (2018). *KeyGenerator*. California, EU.: Google. Recuperado de https://developer.android.com/reference/javax/crypto/KeyGenerator?hl=es-419

Google (2018). *KeyPairGenerator*. California, EU.: Google. Recuperado de https://developer.android.com/reference/java/security/KeyPairGenerator?hl=es-419

Google (2018). *Keystore*. California, EU.: Google. Recuperado de https://developer.android.com/reference/java/security/KeyStore?hl=es-419

Google (2018). *Seguridad con HTTPS y SSL*. California, EU.: Google. Recuperado de https://developer.android.com/training/articles/security-ssl?hl=es-419

Google (2018). *Sistema Android keystore*. California, EU.: Google. Recuperado de https://developer.android.com/training/articles/keystore?hl=es-419

Gunasekera, S. A. (2012). *Android apps security*. New York, EU: Apress.

Hayran, A., Igdeli, M., y Gemci, C. (2016) *Security evaluation of IOS and Android*. International Journal of Applied Mathematic, Electronics and Computers,4 ,258-261

Jamdaade, K., Mr. Akshay, Khairmode, A., y Kamble, S. (2016). *A Comparative study between Android & iOS*. International Journal of Current in Engineering & Research(UCTER),2 Issue 6,495-501.

Lazareska, L., y Jakimoski, K. (2017). *Analysis of the advantages and disadvantages of Android and iOS systems and converting applications from Android to iOS platform and viceversa. American Journal of Software Engineering and Applications* ,6(5),116-120.

Madero, C. (2013). *Controles y seguridad bajo entorno Android* (proyecto fin de carrera). Universidad Carlos III de Madrid, Madrid, España.

Mohammed, A. (2017). *Android users privacy awareness survey*. International Journal of Interactive Mobile Technologies,11 Issue 3,130-144.

Olivares, S. y Gonzales, J. (Ed.). (2016) *La generación Z y los retos del docente*. Nayarit, México: Editorial Ecorfan.

Panchal, P., y Chauchan, A. (2016) *Google Android OS Vs. Apple iOS*. International Journal of Advanced in Engineering, Science & Technology, 3 Issue 5,822-828.

Patterson, B. (2017). *6 easy ways to keep your Android phone secure*. PCWorld, 35 Issue 6, 39-43.

Sahan, A. (2017) *Android v/s iOS - the unceasing battle*. International Journal of Computer Applications 180(3) 23-26.

Seung-hwan, J., Hee-suk, S., y Jin, K. (2016). *Research on android malware permission pattern using permission monitoring system*. Multimedia Tools & Applications, 75 Issue 22, 14807-14817.

Universidad Alicante (2013). *Publicación en app store y distribución Ad Hoc*. Alicante, ES.: Universidad Alicante. Recuperado de http://www.jtech.ua.es/dadm/restringido/serv-ios/sesion06-apuntes.pdf

Universidad Alicante (2013).*Servicios y herramientas en iOS*. Alicante, ES.: Universidad Alicante. Recuperado de http://www.jtech.ua.es/dadm/restringido/serv-ios/wholesite.pdf

ANEXO 1

FICHA DE TRABAJO DE INVESTIGACION EN OPCION AL GRADO DE INGENIERIA DE SISTEMAS E INFORMATICA

1) **Título del Trabajo:** Comparación de los mecanismos de seguridad brindados por .NET Compact, Framework y MSA para el desarrollo de aplicaciones móviles
2) **Nombres de los alumnos posibles a participar:** 02 personas
3) **Tiene perspectiva la investigación de tener continuidad para la investigación:** si
4) **Palabras claves:** sistemas operativos móviles, .Net Compact Framework , Mofira, SOAP.
5) **Posibles Tutores**:

Objetivos de la Tarea de Investigación:

6) La idea central del texto no es crear un manual de cómo implementar mecanismo de seguridad en ambas plataformas, el texto busca dar una visión general de cómo cada una de la plataformas en cuestión atacan la temática de la seguridad de la información, que mecanismos proponen para garantizar la misma, y posteriormente realizar una comparación desde un punto de vista crítico de estos items.
7) **Componentes de la tarea de Investigación:**
    - HTTP pluggable protocol
    - Validación de las entradas de usuario
    - La seguridad de las comunicaciones
    - VPN
8) **Observaciones y recomendaciones al alumno:**
9) **Alumno o alumnos que seleccionaron esta tarea:**
10) **Fecha de Asignación de la tarea:**

Firma del alumno **Firma del profesor asesor del curso de Investigación**
Gianncarlo Gómez Morales

Cursos que se relacionan para el trabajo de investigación:

- Análisis de Riesgos de Tecnología de Información
- Métodos y Medios para la protección de la información.

ANEXO 2