



UNIVERSIDAD NACIONAL DE LOJA

ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO
RENOVABLES

CARRERA DE INGENIERÍA EN SISTEMAS

**“DISEÑO E IMPLEMENTACIÓN DE BANCA MÓVIL
PARA LA COOPERATIVA DE AHORRO Y CRÉDITO
DE LA MICROEMPRESA FORTUNA DE LA CIUDAD
DE LOJA, CON ACCESIBILIDAD DESDE
DISPOSITIVOS ANDROID”**

TESIS DE GRADO PREVIA A
LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN SISTEMAS.

AUTORA:

Natali del Cisne Cruz González.

DIRECTOR:

Ing. Alex Vinicio Padilla Encalada, Mgs.

**Loja – Ecuador
2014**



CERTIFICADO DEL DIRECTOR

Ingeniero.

Alex Vinicio Padilla Encalada, Mgs.

DOCENTE DE LA CARRERA DE INGENIERÍA EN SISTEMAS DE LA UNIVERSIDAD NACIONAL DE LOJA.

CERTIFICA:

Que el presente trabajo de titulación elaborado previo a la obtención del Título de Ingeniera en Sistemas, titulado **"DISEÑO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO FORTUNA DE LA CIUDAD DE LOJA, CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID"**, realizado por la egresada Natali del Cisne Cruz González, cumple con los requisitos establecidos por las normas generales para la graduación en la Universidad Nacional de Loja, tanto en aspectos de forma como de contenido.

Por lo tanto autorizo proseguir los trámites legales para su presentación y defensa.

Loja, 21 de Noviembre de 2014



Ing. Alex Vinicio Padilla Encalada, Mgs.
DIRECTOR DE TESIS



AUTORÍA

Yo, Natali del Cisne Cruz González declaro ser autora del presente trabajo de tesis y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de la presente tesis en el Repositorio Institucional-Biblioteca Virtual.

Autor: Natali del Cisne Cruz González

Firma: 

Cédula: 1104048770

Fecha: 20 de Noviembre del 2014



CARTA DE AUTORIZACIÓN

Yo, Natali del Cisne Cruz González declaro ser autora de la tesis titulada: "**DISEÑO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO FORTUNA DE LA CIUDAD DE LOJA, CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID**", como requisitos para optar al grado de: **Ingeniera en Sistemas**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional;

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja a los 21 días del mes de Noviembre del dos mil catorce.

Firma:

Autor: Natali del Cisne Cruz González

Cédula: 1104048770

Dirección: Loja (Los Ahorcados y Carlos Román).

Correo Electrónico: natycruz1187@gmail.com ; ncruz@coopfortuna.fin.ec

Teléfono: 2572077

Celular: 0986702896

DATOS COMPLEMENTARIOS

Director de Tesis: Ing. Alex Vinicio Padilla Encalada, Mgs.

Tribunal de Grado: Ing. Henry Patricio Paz Arias, Mgs.
Ing. Mario Andrés Palma Jaramillo, Mgs.
Ing. Franco Hernán Salcedo López, Mgs.



AGRADECIMIENTO

Gratitud total a la Universidad Nacional de Loja por dar la oportunidad de cursar por esta magnífica carrera, por su formación y preparación para el desenvolvimiento profesional, a los docentes de la carrera de Ingeniería en Sistemas por los conocimientos impartidos, al Ing. Alex Padilla director del presente Trabajo de Titulación, supo guiar, corregir y plasmar sus conocimientos a lo largo del desarrollo del proyecto.

A la Cooperativa de Ahorro y Crédito de la Microempresa Fortuna por la accesibilidad de información y apoyo constante.

LA AUTORA



DEDICATORIA

El llegar a este punto de mi vida es por la bendición de Dios a quien dedico este nuevo triunfo ya que él ha sabido guiar mi camino y darme las fuerzas necesarias para salir adelante pese a las falencias cometidas, por permitir que mis padres y hermanos me acompañen y apoyen en cada lucha y cada logro obtenido, son ellos quienes son el motor que impulsa mi vida, a mis hermanos por su apoyo y ayuda desde inicios de mis estudios.

A mis tías, mis amigas y compañeros de trabajo quienes han sabido escuchar, apoyar y permitir vivir esta vida con un sentido de alegría y amor.

A quienes lideran la Cooperativa Fortuna por su apoyo y confianza depositada y a todas aquellas personas quienes han aportado con un pedacito de su vida y conocimientos para llegar a concluir un sueño que empezó ya hace mucho tiempo.

Para todos ustedes dedico mi tesis en agradecimiento por ser parte de mis sueños y mi vida en sí.

Natali del Cisne.



CESIÓN DE DERECHOS

Natali del Cisne Cruz González, autora intelectual del presente trabajo de titulación, autorizo a la Universidad Nacional de Loja, al Área de Energía, las Industrias y los Recursos Naturales no Renovables y por ende a la carrera de Ingeniería en Sistemas hacer uso del mismo en lo que estime conveniente.

Natali del Cisne Cruz González



a. Título

“DISEÑO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO DE LA MICROEMPRESA “FORTUNA” DE LA CIUDAD DE LOJA, CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID”.



b. Resumen

El Diseño e Implementación de Banca Móvil para la Cooperativa de Ahorro y Crédito de la Microempresa "Fortuna" con accesibilidad desde dispositivos Android tiene como propósito un servicio personalizado para acceso desde cualquier lugar y hora, ser un nuevo canal de fidelización que permita satisfacer las necesidades del consumidor.

El trabajo de titulación presenta a la ciudadanía lojana, especialmente a los clientes de la COAC Fortuna, un sistema móvil que permitirá a cada cliente el monitoreo, verificación y control constante de las transacciones bancarias realizadas por su persona o terceros exclusivamente sobre los productos asociados al cliente.

El sistema móvil consta de los módulos de Captaciones Vista, Captaciones Plazo y Cartera, los mismos que estarán en el aplicativo móvil que permitirán realizar consultas de los servicios descritos posteriormente.

La metodología que se aplicó en el desarrollo del sistema móvil fue XP (Programación Extrema), para la ingeniería de requisitos se utilizó la guía de Ingeniería del Software, UML (Proceso Racional Unificado) para modelar la accesibilidad del sistema, para la creación de las aplicaciones y gestión de la información en el servidor se utilizó el lenguaje de programación NetBeans WSDL, para el diseño de las interfaces se apoyó en tecnología como Eclipse SDK Android, el plugin ADT y emuladores para testear la aplicación y como gestor de base de datos Oracle 11g.

Como resultado de aplicar estas tecnologías surge la "Aplicación de Banca Móvil Fortuna" que proporcionan a los clientes de la COAC Fortuna los servicios de: Consulta de Cuentas, Información de Cuenta, Movimientos de Cuenta, Consulta de Pólizas, Información de Póliza, Movimientos de Póliza, Consulta de Créditos, Información de Crédito, Movimientos de Crédito, Información de Tarjeta de Débito y Bloqueo de Tarjeta de Débito, Seguridad: Cambiar clave banca móvil y Servicio de afiliación: Descargar App Móvil.



EXECUTIVE SUMMARY

The Design and Implementation of Mobile Banking for Cooperativa de Ahorro y Crédito de la Microempresa "Fortuna" with access from Android devices aims personalized service for access from any place and time, to be a new channel loyalty that would meet the needs consumer.

The paper presents the titration lojana citizenship, particularly customers of COAC Fortuna, a mobile system to allow each client monitoring, verification and ongoing monitoring of banking transactions conducted by them or others solely on products associated with client.

The mobile system consists of modules Vista Deposits, Term Deposits and Accounts, the same that will be in the mobile application that allow querying of services described below.

The methodology applied in the development of the mobile system was XP (Extreme Programming), for requirements engineering was used the guide "Software Engineering, UML (Rational Unified Process) is used to model the availability of system for the creation of applications and information management on the server was used language Netbeans WSDL programming language, for interfaces designing leaned on technology as Eclipse Android SDK, the ADT plugin and emulators to test the application and as manager of Oracle Database 11g.

See Accounts, Account Information, Account Movements, Consulation Polices, Information: As a result of applying this technology the "Mobile Banking Fortuna" to provide customers the COAC Fortuna services arises Policy Movements Policy, Credit Check, Credit Information, Credit Transactions, Debit Card Information and Debit Card Lock, Security : Change key mobile banking and service membership: Download Mobile App.



Índice de Contenidos

CERTIFICADO DEL DIRECTOR.....	II
AUTORÍA	III
CARTA DE AUTORIZACIÓN.....	IV
AGRADECIMIENTO	V
DEDICATORIA.....	VI
CESIÓN DE DERECHOS	VII
a. Título	VIII
b. Resumen	IX
c. Introducción.....	17
d. Revisión de Literatura	19
Capítulo I: MARCO REFERENCIAL	19
1. Aplicativos Móviles.....	19
1.1. Diferencias entre aplicaciones y web móviles	19
1.2. Características que diferencian las aplicaciones de las webs.....	20
1.3. Etapas del desarrollo de un aplicativo	20
1.4. Aplicaciones en Dispositivos Móviles	22
1.5. Aplicaciones Orientadas a la Inclusión Financiera	26
1.6. Ranking de Sistemas Operativos más usados	27
1.7. Sistema Operativo Android	29
1.7.1. Características y especificaciones.....	30
1.7.2. Arquitectura del Sistema Operativo Android.....	30
1.7.3. Máquina Virtual de Android.....	32
Capítulo II: TECNOLOGÍAS UTILIZADAS.....	33
2. Aplicaciones	33
2.1. Web Service WSDL	33
2.2. Eclipse SDK Android	36
2.3. Modelo Vista Controlador.....	42
2.4. Gestor de Base de Datos.....	43
e. Materiales y Métodos	45
f. Resultados.	47
1. ¿Porqué escoger una aplicación Nativa?	47
2. ¿Por qué Android?	48



3.	Análisis de Requisitos	50
3.1.	Elicitación de Requisitos	50
3.2.	Análisis de Requisitos	52
3.3.	Especificación de Requisitos	55
3.4.	Modelo de Dominio	69
3.5.	Modelo de Casos de uso	71
4.	Diseño e Implementación de los Requisitos	73
4.1.	Prototipado Inicial.....	73
4.1.1.	Prototipado Inicial del Aplicativo Banca Móvil.....	73
4.2.	Descripción de los Casos de Uso	80
4.2.1.	Prototipado Final de la aplicación Banca Móvil Fortuna	80
4.3.	Diagrama de Clases.....	109
4.4.	Diseño de la base de datos.....	111
4.5.	Diagrama de Paquetes	114
4.6.	Implementación	116
3	Validación y Verificación	137
g.	Discusión.....	153
1.	Desarrollo de la Propuesta Alternativa	153
2.	Valoración Técnica Económica y Ambiental.....	156
2.1.	Valoración técnica-económica	156
h.	Conclusiones.....	159
i.	Recomendaciones	160
j.	Bibliografía	161
k.	Anexo	162



Índice de Figuras

Figura 1: Ejemplo de Apps disponibles en el AppStore.....	19
Figura 2: El diseño «responsivo» se adapta dependiendo del dispositivo donde es visualizado.....	20
Figura 3: Lenguajes y Herramientas de desarrollo por Sistema Operativo.....	23
Figura 4: Comportamiento de Aplicaciones Nativas.....	24
Figura 5: Comportamiento de Aplicaciones Web.	25
Figura 6: Comportamiento de Aplicaciones Híbridas.....	26
Figura 7: Banca Móvil	27
Figura 8: Ranking de Sistemas Operativos más usados en el 2013	28
Figura 9: Ranking de Sistemas Operativos más usados en el 2014	29
Figura 10: Logotipo de Android.....	30
Figura 11: Arquitectura Android	31
Figura 12: Pasos para consumir un servicio	34
Figura 13: Ejemplo de estructura de un WSDL	34
Figura 14: Entorno de Eclipse.....	37
Figura 15: Plugin Android Developer Tools.....	38
Figura 16: Instalación del SDK de Android para Eclipse.	38
Figura 17: Emulador de BM Fortuna.....	39
Figura 18: SDK Manager	39
Figura 19: Guía para desarrolladores	40
Figura 20: Modelo Vista Controlador de Android	42
Figura 21 Comparativo de App Nativa y Web App.....	47
Figura 22: Principales sistemas operativos para dispositivos móviles.....	48
Figura 23: Estructura general del proyecto "Banca Móvil"	58
Figura 24: Modelo de Dominio de la aplicación BM Móvil	70
Figura 25: Diagrama de CU Registro de Ingreso	71
Figura 26: Diagrama de CU Consulta de Cuentas	71
Figura 27: Diagrama de CU Consulta de Pólizas	72
Figura 28: Diagrama de CU Consulta de Créditos	72
Figura 29: Diagrama de CU Tarjeta de Debito	73
Figura 30: Ingreso al sistema.....	74
Figura 31: Registro	74
Figura 32: Menú Principal	75
Figura 33: Cuentas	75
Figura 34: Información Cuentas.....	76
Figura 35: Movimiento Cuentas	76
Figura 36: Pólizas	77
Figura 37: Información Pólizas.....	77
Figura 38: Movimientos Póliza	78



Figura 39: Crédito	78
Figura 40: Información de Préstamo	79
Figura 41: Tarjeta de débito	79
Figura 42: Tarjeta de débito	80
Figura 43: Tarjeta de débito	80
Figura 44 Diagrama de Clases	110
Figura 45: Diseño de Base de Datos	112
Figura 46 Diagrama de Paquetes	114
Figura 47: Arquitectura de la aplicación BM Fortuna.....	116
Figura 48: Arquitectura de la aplicación BM Fortuna.....	117
Figura 49: Fragmento de Código de Orientación Horizontal	122
Figura 50: Alineación de Componentes	123
Figura 51: Atributo Android_id	123
Figura 52: Atributo Android_Layout	124
Figura 53: Atributo ImageButton	124
Figura 54: Evento OnItemClickListener.....	125
Figura 55: Permisos de Funcionalidades	126
Figura 56: Fichero AndroidManifest	126
Figura 57: Objeto Com.Registro.	127
Figura 58: Clase Registro y Método setContentView.	127
Figura 59: Ventanas Emergentes.	128
Figura 60: Estructura del Proyecto carpeta RES.....	129
Figura 61: Estructura del Proyecto carpeta DRAWABLE.	129
Figura 62: Estructura de la Carpeta Layout.....	130
Figura 63: Estructura de Carpeta Valúes.	131
Figura 64: Import de BD de Producción de COAC Fortuna.....	132
Figura 65: Subida de BD de COAC Fortuna	133
Figura 66: Conexión a la BD de la COAC Fortuna.....	133
Figura 67: Tabla de Registro de Móvil.	135
Figura 68: Tabla de Históricos.	135
Figura 69: Diagrama de Despliegue.....	136
Figura 70: Test de Pruebas Unitarias.....	138
Figura 71 : Test JUnit.	138
Figura 72: Compatibilidad de Equipos 1.	139
Figura 73 : Compatibilidad de Equipos 2.	140
Figura 74: Compatibilidad de Equipos 3.	140
Figura 75: Compatibilidad de Equipos 4.	140
Figura 76: Compatibilidad de Equipos 5.	141
Figura 77: Compatibilidad de Equipos 6.	141
Figura 78: Compatibilidad de Equipos 7.	141
Figura 79: Log de Pruebas de Carga.	142



Índice de Tablas

TABLA I: DESCRIPCIÓN DE LA ESTRUCTURA DE UN WSDL.....	35
TABLA II: CÓDIGO CONEXIÓN A LA BASE DE DATOS.....	35
TABLA III: API DE ANDROID	41
TABLA IV: REQUISITOS PRELIMINARES	50
TABLA V: MATRIZ DE INTERACCIÓN.....	53
TABLA VI: PERSONAL INVOLUCRADO N°1	56
TABLA VII: PERSONAL INVOLUCRADO N°2	56
TABLA VIII: PERSONAL INVOLUCRADO N°3	56
TABLA IX: PERSONAL INVOLUCRADO N°4	56
TABLA X: DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS.....	56
TABLA XI: REFERENCIA DE LA RESOLUCIÓN DE LA JUNTA BANCARIA SBS.....	57
TABLA XII: CARACTERÍSTICAS DEL CLIENTE	59
TABLA XIII: PANTALLA DE REGISTRO DEL APLICATIVO BANCA MÓVIL.....	81
TABLA XIV: DESCRIPCIÓN DEL CASO DE USO REGISTRO.....	82
TABLA XV: PANTALLA DE AUTENTICACIÓN DEL APLICATIVO BANCA MÓVIL.....	83
TABLA XVI: DESCRIPCIÓN DEL CASO DE USO AUTENTICACIÓN	84
TABLA XVII: PANTALLA DE MODIFICAR CLAVE DEL APLICATIVO BANCA MÓVIL	85
TABLA XVIII: DESCRIPCIÓN DEL CASO DE USO MODIFICAR CLAVE	86
TABLA XIX: PANTALLA DE CONSULTA DE CUENTAS DEL APLICATIVO BANCA MÓVIL	87
TABLA XX: DESCRIPCIÓN DEL CASO DE USO CONSULTA DE CUENTAS.....	88
TABLA XXII: PANTALLA DE INFORMACIÓN DE CUENTA DEL APLICATIVO BANCA MÓVIL....	89
TABLA XXII: DESCRIPCIÓN DEL CASO DE USO INFORMACIÓN DE CUENTA.....	90
TABLA XXIII: PANTALLA DE MOVIMIENTO DE CUENTA BANCA MÓVIL	91
TABLA XXIV: DESCRIPCIÓN DE CASO DE USO MOVIMIENTO DE CUENTA	92
TABLA XXV: PANTALLA DE CONSULTA DE PÓLIZAS DEL APLICATIVO BANCA MÓVIL.....	93
TABLA XXVI: DESCRIPCIÓN DEL CASO DE USO CONSULTA DE PÓLIZAS	94
TABLA XXVII: PANTALLA DE INFORMACIÓN DE PÓLIZA DEL APLICATIVO BANCA MÓVIL ...	95
TABLA XXVIII: DESCRIPCIÓN DEL CASO DE USO INFORMACIÓN DE PÓLIZA.....	96
TABLA XXIX: PANTALLA DE MOVIMIENTO DE PÓLIZA DEL APLICATIVO BANCA MÓVIL.....	97
TABLA XXX: DESCRIPCIÓN DEL CASO DE USO MOVIMIENTOS DE PÓLIZA.....	98
TABLA XXXI: PANTALLA DE CONSULTA DE CRÉDITO DEL APLICATIVO BANCA MÓVIL.....	99
TABLA XXXII: DESCRIPCIÓN DE CASO DE USO CONSULTA DE CRÉDITOS.....	100



TABLA XXXIII: PANTALLA DE INFORMACIÓN DE CRÉDITO DEL APLICATIVO BANCA MÓVIL	101
TABLA XXXIV: DESCRIPCIÓN DEL CASO DE USO INFORMACIÓN DE CRÉDITO.....	102
TABLA XXXV: PANTALLA DE MOVIMIENTO DE CRÉDITO DEL APLICATIVO BANCA MÓVIL	103
TABLA XXXVI: DESCRIPCIÓN DEL CASO DE USO MOVIMEINTOS DE CRÉDITO.....	104
TABLA XXXVII: PANTALLA DE ADMINISTRAR TARJETA DE DÉBITO DEL APLICATIVO BANCA MÓVIL.....	105
TABLA XXXVIII: DESCRIPCIÓN DEL CASO DE USO ADMINISTRAR TARJETA DE DÉBITO ..	106
TABLA XXXIX: PANTALLA DE BLOQUEAR TARJETA DE DÉBITO DEL APLICATIVO BANCA MÓVIL	107
TABLA XL: DESCRIPCIÓN DEL CASO DE USO BLOQUEAR TARJETA DE DÉBITO	108
TABLA XLI: DESCRIPCIÓN DE DIAGRAMA DE CLASES.....	111
TABLA XLII: DESCRIPCIÓN DEL DISEÑO DE LA BASE DE DATOS.....	113
TABLA XLIII: CÓDIGO DE MODELO DE CUENTAS.....	118
TABLA XLIV: CÓDIGO DE VISTA DE ACTIVITY_POLIZAS	118
TABLA XLV: CÓDIGO DE CONTROLADOR DE INFORMACION_CUENTA.....	119
TABLA XLVI: MÉTODOS DE WEBSERVICE.....	134
TABLA XLVII: DESCRIPCIÓN DE TIPOS DE CONSULTA	136
TABLA XLVIII: CÓDIGO DE PRUEBA DE STRESS ACTIVITYMANAGER	143
TABLA XLIX: CÓDIGO DE PRUEBA DE STRESS FORTUNABM.APK.....	143
TABLA L: CÓDIGO DE PRUEBA DE STRESS CLASE PRINCIPAL.....	144
TABLA LI: CÓDIGO DE PRUEBA DE STRESS CLASE PRINCIPAL.....	144
TABLA LII: PRUEBA DE VALIDACIÓN RECURSOS	145
TABLA LIII: ESTRUCTURA DEL PLAN DE PRUEBAS DE VALIDACIÓN	145
TABLA LIV: ESCENARIOS PARA LAS PRUEBAS DE VALIDACIÓN.....	146
TABLA LV: DISEÑO DE LAS PRUEBAS DE VALIDACIÓN	147
TABLA LVI: VALORACIÓN ECONÓMICA DE RECURSOS HUMANOS	157
TABLA LVII: VALORACIÓN ECONÓMICA DE RECURSOS TÉCNICOS	157
TABLA LVIII: VALORACIÓN ECONÓMICA DE RECURSOS MATERIALES.....	158
TABLA LIX: VALORACIÓN ECONÓMICA DE RECURSOS DE SERVICIOS	158
TABLA LX: TOTAL DE RECURSOS	158



c. Introducción

En la actualidad, la COAC Fortuna consta de una única oficina denominada Matriz la misma que se encuentra ubicada en el lado Norte de la ciudad de Loja, calles Bolívar e Imbabura esquina, cuenta con un total de más de 8000 clientes activos desde sus inicios del 2005, los servicios que actualmente presta a la ciudadanía son apertura de cuentas, pólizas, tarjetas de débito, créditos, pago de pensiones, dispensario médico, ahorros programados, entre otros beneficios.

Sin embargo, los dirigentes de la COAC Fortuna han tomado a consideración los pedidos solicitados por los clientes para tener disponibilidad de monitorear su información bancaria, realizar la gestión de su información de manera personalizada con tecnología que se adapte a la situación actual y sin necesidad de acudir a la Matriz de la cooperativa tomando en cuenta el tiempo que les demora acercarse a las oficinas principales.

Tomando en consideración los problemas antes descritos se ha creído conveniente aportar con el desarrollo de una herramienta enfocada a mecanismos de uso diario como los dispositivos móviles. Según la revista Redes & Telecom de España se indica que para "el 2013 y 2014 el número de teléfonos móviles en el mundo alcanzara más de 2.000 millones de unidades", mientras que según estudios realizados por Cisco se marcó que en "2016 habrá más teléfonos móviles que personas en el mundo, estimándose que para el 2016 la población mundial alcanzara la marca de los 7,300 millones de habitantes, según la ONU". De hecho años atrás, Steve Jobs predijo que la era post PC había comenzado y que la penetración de la Tablet y teléfonos inteligentes sería cada vez más grande [1].

Por ende, la mejor alternativa para la COAC Fortuna es el desarrollo de una herramienta que permita conexión desde dispositivos móviles a su información bancaria, de esta manera se da cumplimiento a los objetivos planteados con anterioridad.

Se aplicó la ingeniería de requisitos que permitió determinar las necesidades y prioridades del sistema móvil. El diseño e implementación de los requisitos en base a la



metodología ágil seleccionada para la aplicación móvil permitieron la comunicación del Servidor de Aplicaciones, Servidor de Base de datos y los dispositivos móviles, validar y verificar el correcto funcionamiento del sistema móvil a través de las pruebas de valoración que permitieron tener la seguridad de la estabilidad del aplicativo y considerando la constante evolución del Sistema Financiero de nuestro país se genera una base de conocimientos necesaria para la continuidad y adaptabilidad de nuevas reformas o procesos de la cooperativa, necesarias para continuar ya sea con el complemento del aplicativo como para las reformas constantes que el sistema financiero impone sobre las aplicaciones móviles.

La documentación inicia con el resumen del trabajo de Titulación que contiene los aspectos más relevantes del proyecto realizado, una introducción que detalla a breves rasgos el contenido del informe como el marco teórico que está dividido en dos capítulos, el primero aborda el ámbito donde se ejecuta el proyecto y la descripción del sistema, el segundo capítulo permite conocer sobre las tecnologías utilizadas para el desarrollo del sistema móvil, posteriormente permite revisar los métodos y técnicas que se aplicaron y metodología empleada para el desarrollo del sistema. Como último punto y de gran importancia se plasman los resultados del proyecto que contiene la ingeniería de requisitos, diseño e implementación, validación y pruebas. Todo el contenido del informe permite la discusión de resultados, conclusiones, recomendaciones y anexos, en este capítulo se define toda la implementación, herramientas y conceptos necesarios para llegar al punto final de la aplicación banca móvil.

d. Revisión de Literatura

Capítulo I: MARCO REFERENCIAL

En el presente capítulo se plasmarán los principales conceptos que abarcan el desarrollo del proyecto de titulación, desde los más generales hasta los más específicos para su completo entendimiento.

1. Aplicativos Móviles

Una aplicación móvil o también llamada Apps es un programa software que se puede descargar y al que puede acceder directamente desde un teléfono o desde algún otro aparato móvil.

Actualmente hay aplicaciones de todo tipo, forma y color y los nuevos modelos de negocio hicieron de las aplicaciones algo rentable, tanto para desarrolladores como para los mercados de aplicaciones, como App Store, Google Play y Windows Phone Store [2].

En fig. 1 se puede observar un ejemplo de Apps disponibles en el AppStore.



Figura 1: Ejemplo de Apps disponibles en el AppStore

1.1. Diferencias entre aplicaciones y web móviles

Las aplicaciones comparten la pantalla del teléfono con las webs móviles, pero mientras las primeras tienen que ser descargadas e instaladas antes de usar, a una web puede accederse simplemente usando Internet y un navegador; sin embargo, no todas pueden verse correctamente desde una pantalla generalmente más pequeña que la de un ordenador de escritorio.

Las que se adaptan especialmente a un dispositivo móvil se llaman «web responsivas» observar fig. 2, ya que se puede pensar en ellas como un contenido que toma la forma del contenedor, mostrando la información según sea necesario.

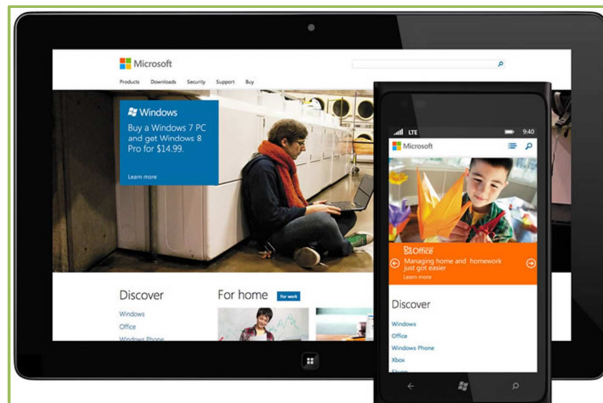


Figura 2: El diseño «responsivo» se adapta dependiendo del dispositivo donde es visualizado

1.2. Características que diferencian las aplicaciones de las webs

- Las aplicaciones pueden verse aun cuando se está sin conexión a Internet, además, pueden acceder a ciertas características de hardware del teléfono como los sensores, capacidades que actualmente están fuera del alcance de las webs.
- Una aplicación ofrece una mejor experiencia de uso, evitando tiempos de espera excesivos y logrando una navegación más fluida entre los contenidos.
- Una App nativa es similar a un programa de ordenador instalado en un dispositivo móvil.
- Una web App es una web diseñada para su visualización en dispositivos móviles gracias a los compiladores que haría las veces de navegador.

1.3. Etapas del desarrollo de un aplicativo




Se detalla las etapas de desarrollo de un aplicativo orientadas al presente proyecto de titulación.



1. Conceptualización

El resultado de esta etapa se la definió en el Análisis de Requisitos (Ver sección de Resultados), aquí se plasmó una idea de aplicación, que tiene en cuenta las necesidades y problemas de los usuarios. La idea responde a una investigación preliminar y a la posterior comprobación de la viabilidad del concepto.



Entregables

-  Ideación
-  Investigación
-  Formalización de la idea

2. Definición

En este paso del proceso se describe con detalle a los usuarios para quienes se diseñará la aplicación (Ver Especificación de Requisitos, documento IEEE Std. 830-1998), se define la metodología que se aplica (Ver sección Materiales y Métodos), y se conciben las bases de la funcionalidad del aplicativo, lo cual determinará el alcance del proyecto y la complejidad de diseño y programación de la App.




Entregables

-  Definición de usuarios
-  Definición funcional

3. Diseño

En la etapa de diseño se llevan a un plano tangible los conceptos y definiciones anteriores, primero en forma de bosquejo, que permiten crear los primeros prototipos para ser probados con usuarios, y posteriormente, en un diseño visual acabado que será provisto al desarrollador, en forma de archivos separados y pantallas modelo, para la programación del código (Ver Diseño e Implementación de los Requisitos, sección de Prototipado Inicial).

Entregables

-  Bosquejo
-  Prototipos
-  Diseño visual



4. Desarrollo

En esta etapa se da vida a lo definido en la etapa de diseño y se crea la estructura sobre la cual se apoyará el funcionamiento de la aplicación. Una vez que existe la versión inicial, se corrige errores funcionales para asegurar el correcto desempeño de la App.





Entregables

-  Programación del código

5. Publicación

La aplicación es finalmente puesta a disposición de los usuarios en Play Store. Luego de este paso trascendental se realiza un seguimiento a través de analíticas, estadísticas y comentarios de usuarios, para evaluar el comportamiento y desempeño de la App, corregir errores, realizar mejoras y actualizarla en futuras versiones.

Entregables

-  Lanzamiento
-  Seguimiento
-  Actualización
-  Tipos de aplicaciones según su desarrollo

1.4. Aplicaciones en Dispositivos Móviles

Las aplicaciones móviles son programas software capaces de ejecutarse en dispositivos inteligentes y pueden ser descargados desde la web o en las tiendas App. De acuerdo a los tipos de aplicaciones tenemos aplicaciones nativas, aplicaciones web y aplicaciones híbridas, que se detallan a continuación [3].

1.4.1. Aplicaciones Nativas

Las aplicaciones nativas son aquellas que han sido desarrolladas con el software que ofrece cada sistema operativo, llamado Software Development Kit o SDK. Así, Android, iOS y Windows Phone y las aplicaciones nativas se diseñan y programan específicamente para cada plataforma, en el lenguaje utilizado por el SDK.

La fig. 3 representa los lenguajes y herramientas de desarrollo por cada Sistema Operativo.








				
Lenguajes	Obj-C, C, C++	Java (C, C++)	Java	C#, Vb .Net, etc
Herramientas	Xcode	Android SDK	BB Java Eclipse Plug-in	Visual Studio, Windows Phone, Dev Tools
Archivos Ejecutables	.app	.apk	.cod	.xap
Tiendas App	Apple Itunes	Google Play	BlackBerry App World	Windows PhoneMarket

Figura 3: Lenguajes y Herramientas de desarrollo por Sistema Operativo



1.4.1.1. Características

-  Rápida ejecución sobre el dispositivo que está corriendo, esto debido a que interactúan directamente con el sistema operativo que los aloja.
-  Empleo de todos los componentes de hardware disponible en los dispositivos como sensores, GPS, cámara, etc.

1.4.1.2. Ventajas

-  Total acceso que puede tener la aplicación sobre los componentes de hardware del dispositivo.
-  No requieren conexión web para ser ejecutadas.

1.4.1.3. Desventajas

-  Dependencia del dispositivo, aplicación y sistema operativo.
-  La aplicación nativa debe ser aprobada por la compañía dueña del Sistema Operativo para poder ser publicada y accesible al público.

La fig. 4 muestra el comportamiento de una aplicación nativa.

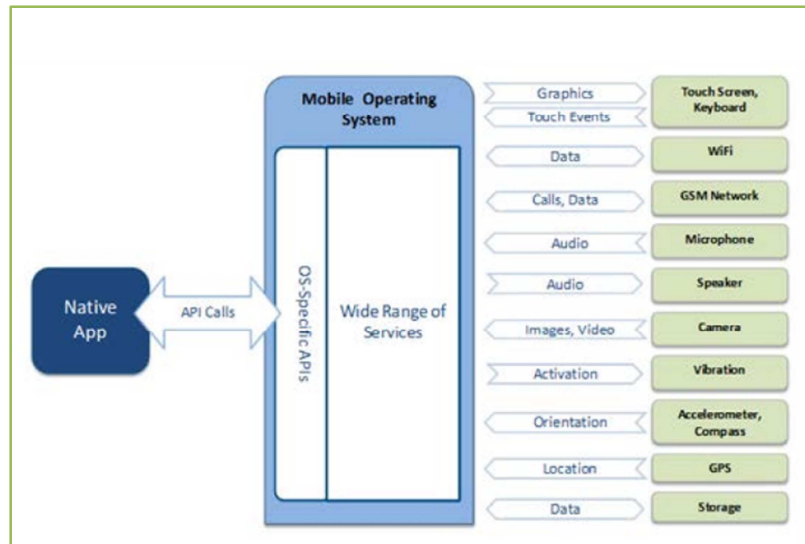





Figura 4: Comportamiento de Aplicaciones Nativas.

1.4.2. Aplicaciones Web

La base de programación de las aplicaciones web, también llamadas WebApps es el HTML, conjuntamente con JavaScript y CSS.

En este caso no se emplea un SDK, lo cual permite programar de forma independiente al sistema operativo en el cual se usará la aplicación. Por eso, estas aplicaciones pueden ser fácilmente utilizadas en diferentes plataformas sin mayores inconvenientes y sin necesidad de desarrollar un código diferente para cada caso particular, no necesitan ser descargadas ni estar pendiente de las actualizaciones.

1.4.2.1. Ventajas

-  Las aplicaciones web se pueden ejecutar en múltiples dispositivos, evitando así las complejidades de tener que crear varias aplicaciones.
-  El proceso de desarrollo es más sencillo.
-  No se necesita la aprobación de ningún fabricante.

1.4.2.2. Desventajas

- Los accesos al teléfono son muy limitados.
- Aplicaciones no se pueden vender en las tiendas App.

La fig. 5 muestra el comportamiento de una aplicación web.

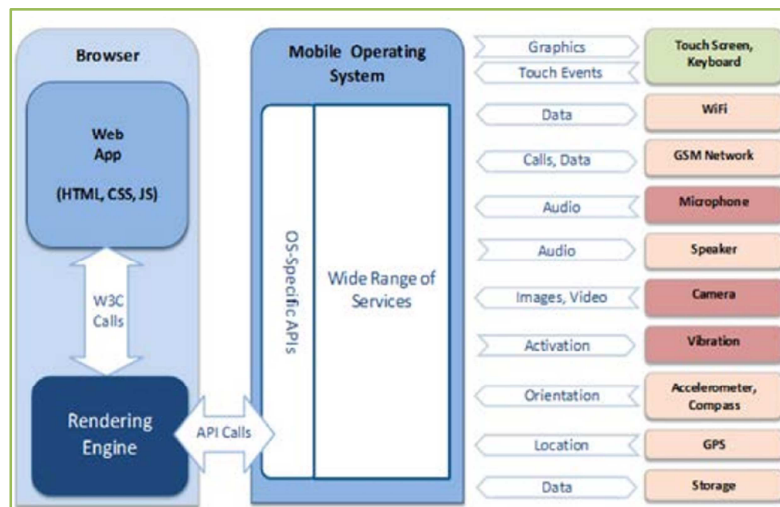


Figura 5: Comportamiento de Aplicaciones Web.

1.4.3. Aplicaciones Híbridas

Este tipo de aplicaciones es una especie de combinación entre las dos anteriores. La forma de desarrollarlas es parecida a la de una aplicación web, usando HTML, CSS y JavaScript y una vez que la aplicación está terminada, se compila o empaqueta de forma tal, que el resultado final es como si se tratara de una aplicación nativa.

Esto permite casi con un mismo código obtener diferentes aplicaciones, por ejemplo, para Android y iOS, y distribuirlas en cada una de sus tiendas. A diferencia de las aplicaciones web, estas permiten acceder, usando librerías, a las capacidades del teléfono, tal como lo haría una App nativa.

Las aplicaciones híbridas, también tienen un diseño visual hay formas de usar controles y botones nativos de cada plataforma para apegarse más a la estética propia de cada una.

1.4.3.1. Ventajas

- Pueden ser instaladas o utilizadas en cualquier Sistema Operativo.

1.4.3.2. Desventajas

- El proceso de desarrollo para este tipo de aplicaciones es más complicado ya que debe combinar aplicaciones nativas y web.
- Estas aplicaciones no se pueden vender en las tiendas App.

La fig.6 muestra el comportamiento de una aplicación híbrida.

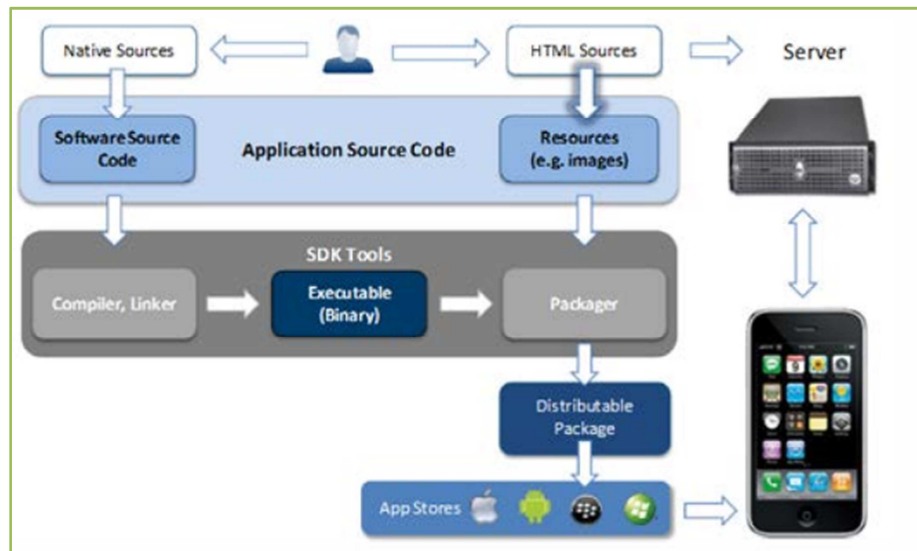


Figura 6: Comportamiento de Aplicaciones Híbridas.

1.5. Aplicaciones Orientadas a la Inclusión Financiera

La industria de los servicios financieros personales se encuentra inmersa en una serie de cambios drásticos: tecnologías móviles, de nube y sociales. La innovación parece estar presente en todos los ámbitos, en el de la banca empresarial esa innovación es la banca móvil.

Los analistas del sector de banca de particulares definen a la banca móvil como "plataformas que permiten al cliente acceder a servicios financieros", en los que emplean

tres tecnologías principales en la banca móvil: SMS, navegador móvil y aplicaciones personalizadas [4].

La fig.7 muestra una aplicación Banca Móvil.



Figura 7: Banca Móvil

La banca móvil ofrece nuevas formas de relación con el cliente, como la captación y conservación o las ventas cruzadas y dirigidas, es parte del modelo de negocio actual y se ofertan como un componente más junto a sus otros canales y persiguen dos objetivos:

- Incrementar el nivel de fidelización del cliente.
- Reducir gastos de operación por medio de una estructura unificada.

Los beneficios de la banca móvil incluyen:

- Migración de las transacciones del cliente al canal menos costoso.
- Complementa el modelo de banca centrada en el cliente.
- Ayuda a los bancos a generar nuevas propuestas de valor y fuentes de ingreso.

1.6. Ranking de Sistemas Operativos más usados

Para poder determinar el principal sistema operativo en el que se debe desarrollar la aplicación móvil para la COAC Fortuna, se ha tomado en consideración el estudio realizado en la Universidad Española Cardelan Herrera, la misma que realiza cada año un estudio de los principales sistemas operativos móviles más usados, en donde se describe lo siguiente:

Año 2013

En la fig. 5 el sistema operativo iOS registra una pérdida de casi un 10% de terreno, pasando del 63,19% en Abril 2012 a un 54,91% en Febrero 2013. En cambio Android ha ganado parte del terreno perdido por iOS, pasando de un 19,27% a un 25,65% en Febrero. Java ME se mantiene, mientras que Symbian muestra un crecimiento inesperado después de un 2012 de bajas. Blackberry desciende de un 2.04 a un 1.39%. Windows Phone es la clara amenaza, sobre todo para BlackBerry. Para otras versiones de Sistema Operativo, esta todo bastante repartido [5].

La fig. 8 ilustra de forma gráfica el Ranking de los Sistemas Operativos más usados en el 2013.

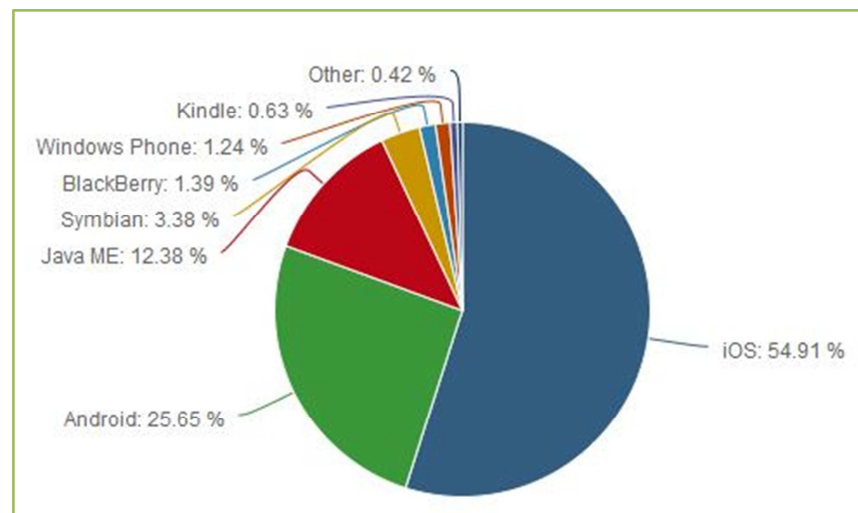


Figura 8: Ranking de Sistemas Operativos más usados en el 2013

Año 2014

En la fig. 6 se puede observar la gráfica comparativa de los Sistemas Operativos. Se puede observar que windows es un sistema muy minoritario, con un 0.55% del mercado, siendo el claro dominador Apple, con su iOS, 54,27%. En segundo lugar se tiene a Android 35,41% y después, muy de lejos, Java ME y Symbian con un 3,18% y Blackberry y Kindle con un 1,56% y 0.95% respectivamente [6].

La fig. 9 ilustra de forma gráfica el Ranking de los Sistemas Operativos más usados en el 2014.

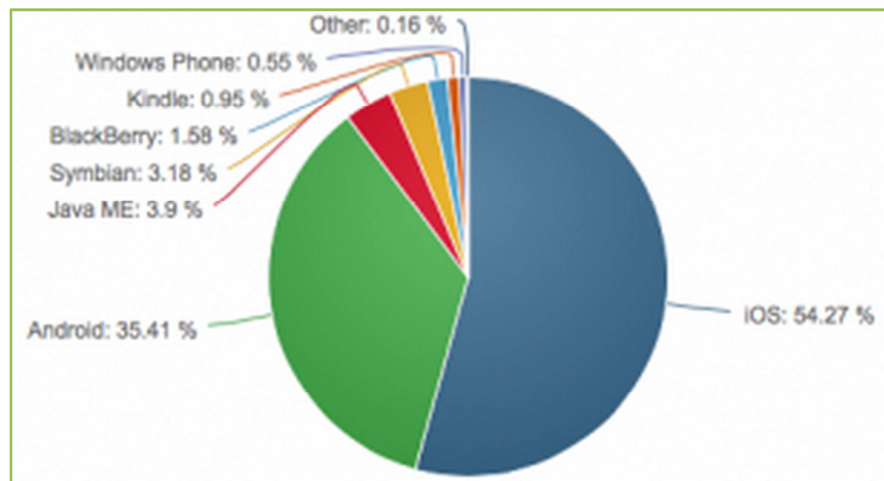


Figura 9: Ranking de Sistemas Operativos más usados en el 2014

1.7. Sistema Operativo Android

Android es un Sistema Operativo de código abierto basado en Linux (con desarrollo independiente), diseñado principalmente para móviles con pantalla táctil como teléfonos inteligentes o tabletas. Google liberó la mayoría del código de Android bajo la licencia Apache, una licencia libre y de código abierto.

La estructura del sistema operativo Android se compone de aplicaciones que se ejecutan en un framework Java de aplicaciones orientadas a objetos sobre el núcleo de las bibliotecas de Java en una máquina virtual con compilación en tiempo de ejecución.

Todas las aplicaciones están comprimidas en formato APK, que se pueden instalar sin dificultad desde cualquier explorador de archivos en la mayoría de dispositivos.

Las bibliotecas escritas en lenguaje C incluyen un administrador de interfaz gráfica (surface manager), un framework OpenCore, una base de datos relacional SQLite, una Interfaz de programación de API gráfica OpenGL ES 2.0 3D, un motor de renderizado Web Kit, un motor gráfico SGL, SSL y una biblioteca estándar de C Bionic. [7].

La fig. 10 representa el logotipo de Android.



Figura 10: Logotipo de Android

1.7.1. Características y especificaciones

- Diseño de dispositivo: la plataforma es adaptable a pantallas de mayor resolución.
- Conectividad Android soporta tecnologías de conectividad: Bluetooth, WiFi, etc.
- Contiene mensajería y navegador web.
- Android soporta formatos multimedia.
- Soporte para streaming, descarga progresiva de HTML.
- Soporte de Java y muchos formatos multimedia.
- Soporte para hardware adicional: Android soporta cámaras de fotos, de vídeo, pantallas táctiles, GPS.
- Entorno de desarrollo: Android incluye un emulador de dispositivos, herramientas para la depuración de memoria y análisis del rendimiento del software.
- El entorno de desarrollo se encuentra integrado en Eclipse.
- Google Play: Un catálogo de aplicaciones gratuitas o de pago.
- Multi-táctil, Bluetooth, Video llamada, Multitarea y Tethering, que permite al teléfono o Tablet ser usado como punto de acceso inalámbrico para la conexión a Internet.
- Utiliza SQLite para el almacenamiento de datos.
- Navegador web basado en Web Kit incluido.

1.7.2. Arquitectura del Sistema Operativo Android

El Sistema Operativo Android en su ámbito arquitectónico está estructurado por varias capas [8].

La fig. 11 ilustra de forma gráfica la arquitectura del sistema.



Figura 11: Arquitectura Android

A continuación se detallará cada uno de los componentes de la arquitectura del sistema operativo Android:

- **Kernel de Linux.** Este actúa como una capa de abstracción entre el hardware y el resto de las capas de la arquitectura. El kernel se encarga de gestionar los diferentes recursos del teléfono y del sistema operativo en sí: procesos, elementos de comunicación, etc.
- **Librerías:** La componen las librerías nativas de Android, están escritas en C o C++ y compiladas para la arquitectura hardware específica del teléfono. Su objetivo es proporcionar funcionalidad a las aplicaciones para tareas que se repiten con frecuencia, evitando tener que estar codificándolas y garantizando que se llevan a cabo de la forma más eficiente.
- **Entorno de Ejecución:** Su principal componente es la máquina virtual Dalvik, por ende las aplicaciones se compilan una única vez y se pueden distribuir con garantía de que podrán ser instaladas en cualquier dispositivo Android que disponga de la versión mínima del sistema operativo que requiera la aplicación.



- Framework de Aplicaciones: La siguiente capa está formada por todas las clases y servicios que utilizan las aplicaciones para realizar sus funciones, la mayoría de sus componentes son librerías de Java que acceden a los recursos de las capas anteriores a través de la máquina virtual Dalvik.

Entre sus principales librerías tenemos:

- **Activity Manager** se encarga de administrar la pila de actividades de la aplicación así como su ciclo de vida.
- **Windows Manager** se encarga de organizar lo que se mostrara en pantalla que posterior pasaran a ser ocupadas por las actividades,
- **Content Provider** crea una capa que encapsula los datos que se compartirán entre aplicaciones para tener control sobre cómo se accede a la información.
- **Views** las vistas ayudan a crear las interfaces de usuarios.
- **Notificacion Manager:** Engloba los servicios para notificar al usuario cuando requiera alguna alerta.

1.7.3. Máquina Virtual de Android

La máquina virtual utilizada por Android es Dalvik, esta trabaja en la capa de ejecución y ha sido diseñada para optimizar la memoria y los recursos de hardware en el entorno de los teléfonos móviles o Tablet. Dalvik es una máquina virtual que cumple la función de interprete en la ejecución de archivos en el formato (*.dex) [9].

Entre las características sobresalientes de Dalvik se encuentra:

- Máquina Virtual de aplicación o proceso, cada aplicación se ejecuta en un proceso independiente y con su propia instancia de la máquina virtual.
- La arquitectura está basada en registros.



Capítulo II: TECNOLOGÍAS UTILIZADAS

2. Aplicaciones

Para el desarrollo del presente proyecto de titulación se ha utilizado las tecnologías de Eclipse, Web Services y SDK de Android, las mismas que serán explicadas en el presente capítulo.

2.1. Web Service WSDL

WSDL (Web Service Description Language) es un protocolo basado en XML que describe los accesos al Web Service, se puede decir que es el manual de operación del mismo, porque indica cuáles son las interfaces que provee el Servicio Web y los tipos de datos necesarios para su utilización [10].

WSDL es el lenguaje para la descripción de Servicios Web y permite describir la interfaz de un servicio web en un formato XML. Una de sus ventajas es que permite separar la descripción abstracta de la funcionalidad ofrecida por un servicio, es decir, de los detalles concretos del mismo, como puede ser el enlace a un protocolo de red o un formato de mensaje concreto que puede ser SOAP, HTTP o MIME.

Un Web Service está compuesto por tres componentes principales:

- Búsqueda y localización de servicios UDDI (Universal Discovery, Description and Integration)
- Descripción del servicio WSDL (Web Service Description Language)
- Codificación de datos y mensajes SOAP (Simple Object Access Protocol)

El WSDL describe los servicios Web a través de los mensajes que se intercambian entre el proveedor del servicio y el cliente.

2.1.1. Utilización

En la fig. 12 se describe los pasos que se realizan al consumir el servicio que son los siguientes:

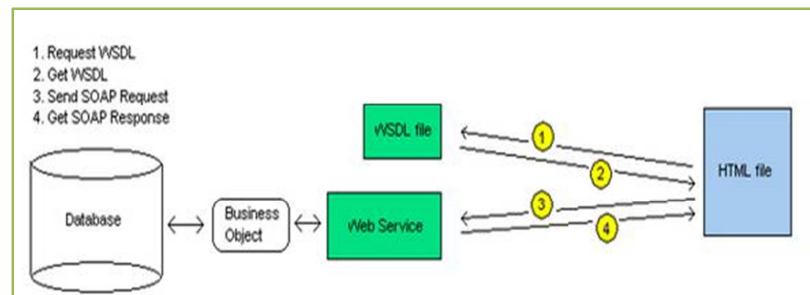


Figura 12: Pasos para consumir un servicio

1. Lo primero que realiza el cliente al hacer una solicitud al servicio es tomar la definición del archivo WSDL.
2. El servidor entrega el fichero WSDL. Este archivo indica a la petición los métodos y propiedades de ese servicio que están disponibles.
3. El cliente hace la petición en el formato que espera el servidor según las especificaciones del fichero WSDL en el que se dice qué parámetros acepta y de qué tipo.
4. El servidor entrega el resultado de la consulta.

2.1.2. Estructura

La fig. 13 presenta un ejemplo de la estructura de un documento WSDL es el siguiente:

```

1  <?xml version="1.0"?>
2  <definitions>
3      <types> ... </types>
4      <message> ... </message>
5      <portType> ... </portType>
6      <binding> ... </binding>
7  </definitions>

```

Figura 13: Ejemplo de estructura de un WSDL



Dónde:

TABLA I: DESCRIPCIÓN DE LA ESTRUCTURA DE UN WSDL

Elemento WSDL	Descripción
xml versión="1.0"	Un documento WSDL es como cualquier documento XML y se basa en los esquemas, por lo que debe comenzar con dicha etiqueta.
<definitions>	Comienzo del documento, este tag agrupa a todos los demás elementos
<types>	Se definen los tipos de datos utilizados en los mensajes. Se utilizan los tipos definidos en la especificación de esquemas XML.
<message>	Se definen los métodos y parámetros para realizar la operación. Cada message puede consistir en una o más partes (parámetros). Las partes pueden ser de cualquiera de los tipos definidos en la sección anterior.
<portType>	Esta sección es la más importante, ya que definen las operaciones que pueden ser realizadas, y los mensajes que involucran (por ejemplo el mensaje de petición y el de respuesta).
<binding>	Se definen el formato del mensaje y detalles del protocolo para cada portType.

2.1.3. Conexión del WSDL a la Base de Datos

La tabla muestra la conexión del Web Service con la base de datos, de esta manera se puede acceder a la información del cliente y permite almacenar y modificar los datos necesarios.

TABLA II: CÓDIGO CONEXIÓN A LA BASE DE DATOS

Código de conexión a la base de datos "fortunadb"
<pre><?xml version="1.0" encoding="UTF-8"?> <resources> <jdbc-resource enabled="true" jndi-name="jdbc/myFortunaDB" object-type="user" pool- name="connectionPool"> <description/> </jdbc-resource> <property name="URL" value="jdbc:oracle:thin:@192.168.28.103:1521:baselocal"/> </resources></pre>



```
<property name="User" value="system"/>  
<property name="Password" value="fortunadb"/>  
</jdbc-connection-pool>  
</resources>
```

2.2. Eclipse SDK Android

La principal herramienta que se utilizó para la construcción del proyecto Banca Móvil, se la describe a continuación.

2.2.1. Eclipse

Eclipse es una plataforma de desarrollo de código abierto basada en Java, es un marco de trabajo y un conjunto de servicios para la construcción del entorno de desarrollo de los componentes de entrada. Eclipse tiene un conjunto de complementos, incluidas las Herramientas de Desarrollo de Java (JDT) [11].

Ventajas:

- Emplea módulos para proporcionar toda su funcionalidad al frente de la Plataforma de Cliente.
- Este mecanismo de módulos es una plataforma ligera para componentes de software.
- La arquitectura plug-in permite escribir cualquier extensión deseada en el ambiente, como sería Gestión de la configuración.
- La definición que da el proyecto Eclipse acerca de su Software es: "una especie de herramienta universal - un IDE abierto y extensible para todo y nada en particular".
- Eclipse provee al programador con Frameworks para el desarrollo de aplicaciones gráficas, definición y manipulación de modelos de Software, Aplicaciones web, etc.
- El SDK de Eclipse incluye las herramientas de desarrollo de Java, ofreciendo un IDE con un compilador de Java interno y un modelo completo de los archivos fuente de Java. Esto permite técnicas avanzadas de refactorización y análisis de código.

- El IDE también hace uso de un espacio de trabajo, en este caso un grupo de metadata en un espacio para archivos plano, permitiendo modificaciones externas a los archivos en tanto se refresque el espacio de trabajo correspondiente.

La fig. 11 representa el entorno de trabajo de Eclipse:

- **Workspace:** directorio donde se almacenan todos los proyectos relacionados, mantienen sus propias propiedades.
- **View:** ventanas de utilidad, como gestor de proyectos, propiedades, consola, etc.
- **Perspective:** agrupación de vistas (views) que facilitan alguna tarea concreta.

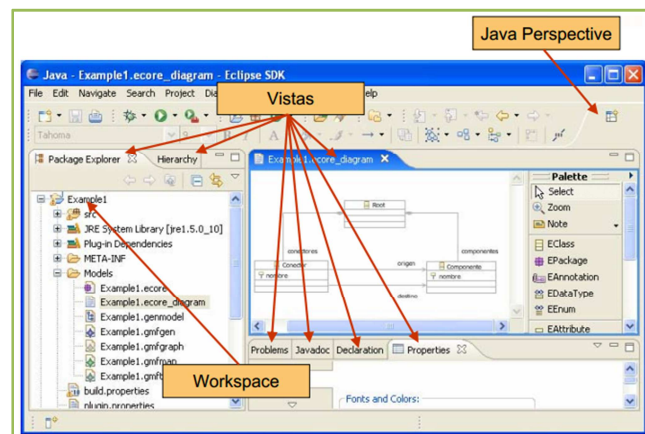


Figura 14: Entorno de Eclipse

2.2.2. SDK Android

Un kit de desarrollo de software o SDK (siglas en inglés de Software Development Kit) es un conjunto de herramientas de desarrollo de software, que permite al programador crear aplicaciones para un sistema concreto, por ejemplo paquetes de software, frameworks, plataformas de hardware, computadoras, videoconsolas, sistemas operativos, etc.

Es como una interfaz de programación de aplicaciones o API creada para permitir el uso de cierto lenguaje de programación, o puede también incluir hardware sofisticado para comunicarse con un determinado sistema embebido. Las herramientas más comunes incluyen soporte para la detección de errores de programación, un entorno de desarrollo integrado o IDE y otras utilidades.

En el caso de Android, las aplicaciones se desarrollan habitualmente en el lenguaje Java con Android Software Development Kit (Android SDK), esta herramienta facilita la creación de aplicaciones. Se encuentran integradas en Eclipse, el cual utiliza el plugin ADT (Android Developer Tools).

La fig. 15 representa el inicio de Android Developer Tools.



Figura 15: Plugin Android Developer Tools.

Android Developer Tools (ATD) es un plugin para el IDE Eclipse, integrado en el mismo para conseguir un mayor potencial en el desarrollo e integración de aplicaciones, confiriendo una mayor versatilidad para una mejor construcción de aplicaciones.

Android. ADT amplía las capacidades de Eclipse, permite configurar nuevos proyectos para Android, crear una interfaz de usuario de aplicación, agregar los paquetes basados en la API Framework Android, depurar sus aplicaciones utilizando las herramientas del SDK de Android e incluso exportar con firmas las APK generadas a fin de distribuirlas adecuadamente. En resumen, integra herramientas SDK que ejecutan estas acciones.

La fig. 16 representa la instalación del SDK de Android para Eclipse.

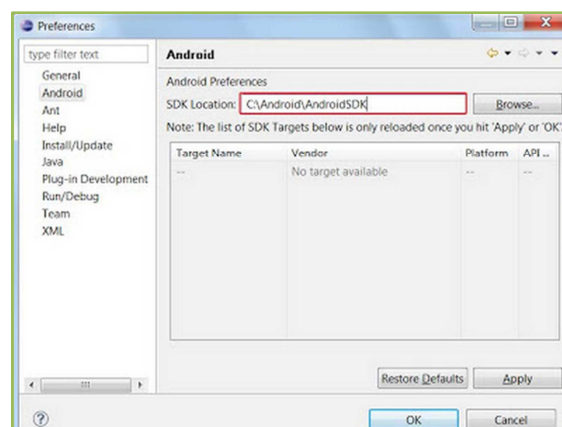


Figura 16: Instalación del SDK de Android para Eclipse.

2.2.2.1. Herramientas SDK

Entre las herramientas más importantes del SDK de Android se encuentra el Android Emulator y el Android Development Tools

Emulador de Android

Un dispositivo virtual de Android (AVD) es una configuración del emulador que permite simular un dispositivo real mediante la definición de las opciones de software que emula una versión Android y un hardware determinado, permite modelar diferentes configuraciones de dispositivos virtualizados con Android. El plugin ADT permite crear una AVD (Android Virtual Device) o lo que es lo mismo, un Dispositivo Virtual Android para lanzar y depurar la aplicación.

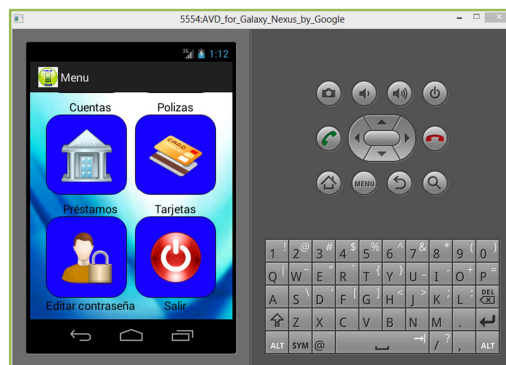


Figura 17: Emulador de BM Fortuna

Las herramientas SDK, fig. 18, se encuentran en el directorio "/tools", puedes lanzar el SDK Manager (en un fichero llamado "SDKManager.exe") ubicado en la carpeta "adt-bundle-windows" de la plataforma correspondiente (i386 o 64 bits).

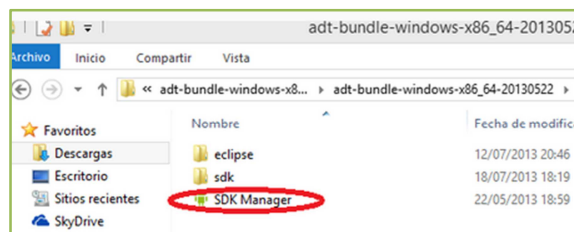


Figura 18: SDK Manager

Recomendaciones:

- Usar la documentación Android.
- Depurar aplicaciones con DDMS (Dalvik Debug Monitor Service) o ADM (Android Debug Monitor).
- Emulación Android. Usando AVD, la emulación persigue reproducir el comportamiento de las distintas plataformas Android, tratando de reproducir de la forma más precisa posible el funcionamiento de un dispositivo, como si estuviese siendo usado tal como lo haría el original.
- Conocimiento de la estructura del sistema Android (configuración, menús, pantallas).

Guía para desarrolladores

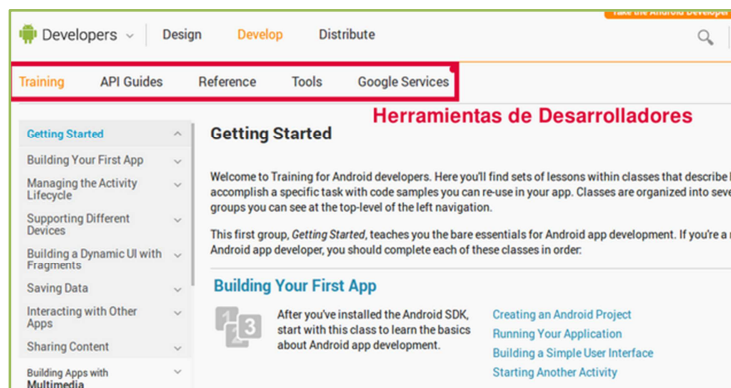


Figura 19: Guía para desarrolladores

- "Training". Entrenamiento, para aprender a realizar una aplicación desde cero.
- "API Guides". Guías de API. En esta sección se podrá encontrar distintos framework de aplicaciones de Android y explicaciones, que permitirán generar una nueva aplicación usando un conjunto de componentes ya creados que se podrá reutilizar.
- "Reference". Se dispone de un glosario explicativo que le permite conocer las API, Interfaces y Clases de Android, necesarios para la programación.
- "Tools". Las herramientas SDK. El Developer Tools plugin de Android (ADT) de Eclipse que proporciona un entorno de desarrollo de nivel profesional para la creación de aplicaciones Android.
- "Google Service". Servicios Google (Google Maps, Google + y otros).

En la tabla III se describe las principales definiciones del API.

TABLA III: API DE ANDROID

Palabra	Categoría	Definición
Adb	Herramientas	Es una herramienta que viene en el SDK de Android. Permite trastear con el móvil o un emulador mediante la línea de comandos.
APK	General	Es la extensión de las aplicaciones Android, Un .apk es cualquier aplicación que se puede instalar en el móvil.
AndroidManifest.xml	Programación	Es el archivo en el que se definen los rasgos principales de una aplicación.
Ddms	Herramientas	Permite visualizar los procesos que se están ejecutando en tu teléfono así como hacer capturas de pantalla.
Logcat	Programación	Un listado de los mensajes emitidos por el teléfono.
OTA	General	Es un método para distribuir actualizaciones a los usuarios de tal manera que se aparezca una notificación directamente en el teléfono y puedan descargarla directamente sin necesidad de conectarse a ningún ordenador.
SDK	Herramientas	Un kit de desarrollo de software • SDK es un conjunto de herramientas de desarrollo que le permite a un programador crear aplicaciones para un sistema concreto.
SPL	Root/Roms	Es un cargador secundario de programación <i>bootloader</i> que se encarga de cargar la <i>Radio/GSM</i> y el sistema operativo. De esta manera se podrá modificar tamaños de memoria para mejorar el rendimiento del teléfono.

2.3 Modelo Vista Controlador

Puesto que se hace uso de un lenguaje orientado a objetos, se emplea el patrón modelo-vista-controlador, dividiendo la aplicación en tres niveles, el nivel de datos correspondiente al modelo, el nivel de interfaz correspondiente a la vista y la lógica de la aplicación correspondiente al controlador.

La fig.20 representa el Modelo Vista Controlador de Android.

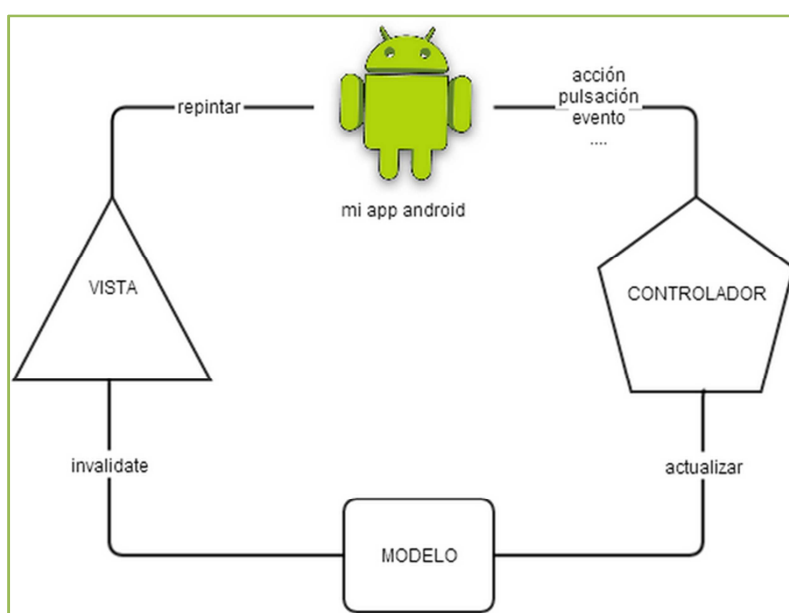


Figura 20: Modelo Vista Controlador de Android

1. Modelo

El modelo consiste en el comportamiento e información del dominio de la aplicación, así como del acceso a la base de datos y la propia base de datos. En cuanto al comportamiento e información del dominio destaca la colección de distintas clases empleadas para el desarrollo de la aplicación y vienen establecidas por el diagrama de clases. Esto permite encapsular la información obtenida de forma que pueda manejarse de forma cómoda y sencilla mediante el apartado controlador.



El modelo en el proyecto representa las estructuras de datos. Las clases del modelo contienen funciones que ayudarán a recuperar, insertar y actualizar la información en la base de datos.

2. Vista

La vista corresponde con la interfaz de usuario que se muestra. El punto principal de la interfaz es la simplicidad de ésta, puesto que la intención es que la aplicación sea rápida, intuitiva y fácil de usar. Por otra parte, gracias a que Android emplea el patrón anteriormente comentado, modelo-vista-controlador, en caso de querer realizar cambios en la interfaz, tan solo tendremos que modificar la propia interfaz, sin alterar la lógica o funcionamiento de la aplicación.

3. Controlador

En el controlador encontramos el código necesario para que la aplicación realice las funciones que se esperan de ella, es decir el código que se ejecuta cuando el usuario interacciona con la aplicación y ésta deben responder a la interacción. Se ha utilizado orientación a objetos para realizar la lógica de la aplicación, y como ya se ha comentado en el apartado correspondiente a la vista, debido a que Android hace uso del patrón modelo-vista-controlador, podemos modificar detalles del controlador sin que esto afecte al resultado mostrado.

2.4 Gestor de Base de Datos

Como gestor de Base de Datos se utilizó Oracle 11g, esta es la base que la COAC Fortuna tiene implementado para su Core Bancario Financiamiento 2.

2.4.1 Oracle

Oracle es básicamente una herramienta cliente/servidor para la gestión de Bases de Datos, es un sistema de gestión de base de datos relacional, la gran potencia que tiene y



su elevado precio hace que solo se vea en empresas muy grandes y multinacionales, por norma general.

2.4.2 Características

- Es una herramienta de administración gráfica, intuitiva y cómoda de utilizar.
- Ayuda a analizar datos y efectuar recomendaciones concernientes a mejorar el rendimiento y la eficiencia en el manejo de aquellos datos que se encuentran almacenados.
- Apoya en el diseño y optimización de modelos de datos.
- Asistir a los desarrolladores con sus conocimientos de SQL y de construcción de procedimientos almacenados y triggers, entre otros.
- Apoya en la definición de estándares de diseño y nomenclatura de objetos.
- Documentar y mantener un registro periódico de las mantenciones, actualizaciones de hardware y software, cambios en las aplicaciones y, en general, todos aquellos eventos relacionados con cambios en el entorno de utilización de una base de datos.

2.4.3 Estructuras de Oracle

Las estructuras de la Base de Datos Oracle pueden ser divididas en tres clases:

- Internas a la Base de Datos: Tablas, columnas, usuarios, esquemas, índices, clúster, vistas, secuencias, disparadores, triggers.
- Internas a las áreas de memoria (incluidas la memoria compartida y procesos): Área Global de Programa y Área Global del Sistema.
- Externas a la Base de Datos: Guardan datos almacenados en la base de datos como para gobernar la propia base de datos, Ficheros, log, ficheros de control, de traza.



e. Materiales y Métodos

El trabajo de titulación consta de tres fases, que permitieron familiarizarse con varios aspectos desconocidos relacionados al objetivo planteado, para determinar como el aplicativo móvil contribuirá a la banca local.

1. Métodos de Investigación

- ✓ **Método Inductivo:** Este método permitió conocer de manera particular la problemática de la COAC Fortuna, a través de pasos fundamentales como la observación y análisis de recursos, tanto económico como humanos para establecer la información requerida para el trabajo de titulación y así ejecutar una solución a los problemas encontrados.
- ✓ **Método Deductivo:** Con el uso de este método se logró ejecutar las soluciones definitivas basadas en conocimientos científicamente validados, relacionados con el funcionamiento de aplicativos de banca móvil, a lo que antes eran simples deducciones.
- ✓ **Método Científico:** Este método permitió realizar el estudio del origen del problema que se enfrenta la COAC Fortuna y su impacto con los clientes de su localidad, mediante la planificación, el uso de una metodología e instrumentos confiables, permitió realizar el análisis para la posterior implementación del sistema, generando así nuevo conocimiento.

2. Metodología

Fase de Análisis de Requisitos: La fase de inicio de la metodología XP, contempla el análisis del funcionamiento del Core Bancario de la COAC Fortuna, para determinar cuál es la información más relevante y de importancia para el cliente; para mayor validez se aplicó el estándar IEEE, el mismo que establece un profundo análisis de requisitos para definir los límites del sistema propuesto.



Además se aplicó métodos de investigación, así mismo el uso de técnicas de recolección de información como la entrevista, observación directa y la encuesta permitieron realizar la Ingeniería de Requisitos (Ver sección Resultados, apartado 1 Análisis de Requisitos), los mismos que facilitaron la aprobación del trabajo de titulación (Ver Anexo 4 Aprobación del Proyecto de Tesis)

Fase de Diseño e Implementación: Contempla la etapa dos y tres de la metodología antes mencionada, la primera es la **Elaboración** en la que se establece una solución preliminar en base al diseño y descripción de los Casos de Uso, el Modelo de Dominio y el Diagrama de Clases.

La segunda etapa es la **Construcción** que busca completar la funcionalidad del sistema, abarcando el diseño de la base de datos, diagrama de paquetes y la generación del código fuente (Ver sección Resultados apartado 5 Diseño y sección Resultados apartado 5.3, 5.4, y 5.5).

Fase de Validación y Verificación: Esta fase abarca la etapa de **Transición** de la metodología seleccionada, esta fase cuyo objetivo es asegurar la disponibilidad del sistema para los usuarios finales, ajustando los errores y defectos encontrados a través de las pruebas unitarias, de aceptación, carga, rendimiento y de estrés.

f. Resultados.

Como primer resultado se justifica la selección del desarrollo de la aplicación Banca Móvil como aplicación Nativa y Sistema Operativo Android.

FASE 1:

1. ¿Por qué escoger una aplicación Nativa?

Para el desarrollo del aplicativo de Banca Móvil de la COAC Fortuna se consideró lo expuesto en el artículo App Nativa o Web de la empresa Cuatroochenta, la fig. Comparativo de App Nativa y Web App se especifica las principales funcionalidades de cada uno.

Características	App Nativa	Web App
Acceso a las características del dispositivo	Total	Parcial
Velocidad de funcionamiento	Muy Rápido	Depende de conexión
Interfaz de usuario	Máxima usabilidad	Usabilidad básica
Publicación en App Store	Si	No Permitido
Uso sin conexión de datos (sin conexión a internet)	Si	Parcial
Coste de desarrollo	Alto	Bajo
Proceso de revisión en App Store	Si	No

Figura 21 Comparativo de App Nativa y Web App

Análisis

Para desarrollar una App nativa se debe tomar en cuenta todo aquello que afectará el desarrollo, desde los recursos y la complejidad, hasta el tipo de usuario al que se quiere orientar.

En comparación con la Web App, las aplicaciones nativas son las que ofrecen una mejor experiencia de uso ya que tienen una máxima usabilidad para la interfaz del usuario, una respuesta inmediata a las acciones y peticiones de los usuarios, velocidad de

funcionamiento y rendimiento, aunque su principal desventaja es que tienen un alto costo de desarrollo, en cambio Web App tiene un coste de desarrollo bajo pero la seguridad es menor, y la aplicación está expuesta a todo el código malicioso que se ejecuta desde la web, en cambio las aplicaciones híbridas usan el motor del navegador del dispositivo exponiendo de una u otra manera el aplicativo por ende al estar desarrollando una aplicación para una entidad financiera se debe considerar el máximo de seguridades a implementar ya que su código debe ser compilado y no accesible por terceros.

2. ¿Por qué Android?

Al haber seleccionado desarrollar una aplicación nativa, surgió otra gran pregunta ¿En qué Sistema Operativo se lo va a realizar?

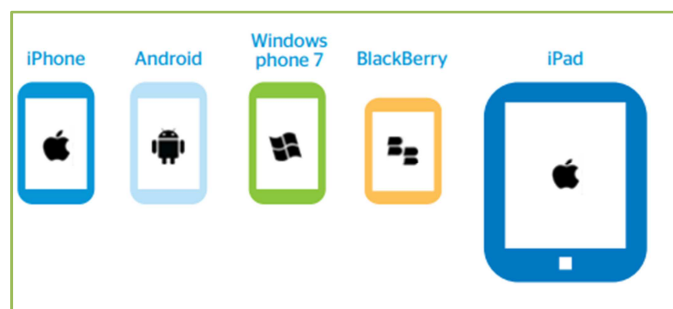


Figura 22: Principales sistemas operativos para dispositivos móviles

De acuerdo al artículo publicado por el Banco Bilbao Vizcaya Argentaria **BBVA** se ha podido recopilar que las principales entidades financieras del mundo tienen aplicaciones móviles nativas en diferentes Sistema Operativo que se detallan a continuación (ver Anexo 7):

En el mundo

- 📱 ActivoBank de Portugal y Millenium de Polonia: Reconocidos como mejores banco móvil europeo por la revista Global Finance, las soluciones bancarias móviles de dichos bancos están disponibles a través de Apps nativas para dispositivos Android e IOS.



- ScotiaBank de Canadá es el banco más internacional de Canadá con sucursales en el Caribe, América Central, América Latina y Asia tiene disponible su Apps en los Sistemas Operativos de IOS y Android.
- Citi Mobile y CitiBank son compatibles para iPod, iPad, y dispositivos Android y Kindle Fire (Tableta diseñada por Amazon).
- Chase Mobile Banking tiene aplicaciones compatibles con iOS, Android, Kindle Fire y BlackBerry.
- Bank of America ofrece aplicaciones nativas para iPhone, iPod Touch, iPad, BlackBerry, Android, Smartphones de Windows y Kindle Fire.
- ICBC de China es el ganador de dos premios AppStore por mejor aplicación financiera para iPhone.
- BBVA de España, Chile y Argentina ofrece sus aplicaciones para cuatro plataformas diferentes: iPhone, iPad, Android y BlackBerry. Fueron Galardonados por su aplicación de geo localización para iPhone y Android que ayuda al cliente a encontrar la sucursal o cajero más cercano.

Localmente

- Banco de Pichincha, Banco de Guayaquil, Banco Bolivariano, Banco Internacional, Banco del Austro, Banco Pro América, Banco del Pacifico y Banco de Loja le permite acceder desde Android, BlackBerry, Mac. *(Información tomada desde el sitio web de cada entidad bancaria).*

Análisis

De acuerdo a la información descrita en los párrafos anteriores, se resume que las entidades financieras han optado por desarrollar aplicaciones nativas en los sistemas operativos:

Utilizando la fórmula de proyección para Android 2015 = 43.48 y 2016 = 51.55 se puede observar que la demanda de Android es creciente, por ende este es el Sistema Operativo que se va a desarrollar el aplicativo Banca Móvil para la COAC Fortuna.



3. Análisis de Requisitos.

El análisis de requisitos mide la calidad de un sistema software determinando la concordancia entre sus requisitos y el mayor o menor grado de su consecución. En esta fase se procedió a realizar la ingeniería de Requisitos bajo el estándar IEEE Std. 830-1998, obteniendo así los requisitos, conceptos y objetos que están relacionados con las actividades que se realizan en la Cooperativa, teniendo como resultado la especificación de requisitos en formato IEEE.

3.1. Elicitación de Requisitos

Examina la información que se maneja en la entidad financiera, cuales son los requisitos de los usuarios a través de las técnicas de recolección de información:

Entrevista: Esta técnica se la aplico a la oficial de Servicios Cooperativos de la COAC Fortuna, quien tiene la responsabilidad de receptor las solicitudes o peticiones de los clientes (Ver Anexo 1 Entrevista).

Observación: Esta técnica se la realizo en las instalaciones de la Cooperativa Fortuna, en el área de Servicios Cooperativos, se permitió analizar y evaluar como es la recurrencia de los clientes y las peticiones que generalmente suelen realizar a las oficiales de atención al público (ver Anexo 2 Ficha de Observación).

Encuesta: Esta técnica se la dirigió al Jefe de Sistemas y Servicios Cooperativos e Inversiones de la COAC Fortuna (Ver Anexo 3 Modelo de Encuesta).

En base a las técnicas aplicadas, antes descritas se recolectaron los requisitos preliminares que se detallan en la tabla IV.

TABLA IV: REQUISITOS PRELIMINARES

CÓDIGO	REQUISITOS	FUENTE
RF01	Acceso a la información bancaria por medio de un teléfono celular.	Entrevista (Anexo1)
RF02	El cliente pueda realizar consultas de sus cuentas.	Encuesta(Anexo 3)
RF03	Revisar movimientos de cuentas.	Encuesta(Anexo 3)
RF04	Revisar depósitos realizados en las cuentas.	Encuesta(Anexo 3)
RF05	El cliente pueda revisar la fecha de vencimiento de	Observación



	su póliza.	(Anexo 2)
RF06	Se pueda revisar los préstamos que tiene el cliente.	Observación (Anexo 2)
RF07	Permita revisar las acreditaciones de intereses de las pólizas.	Observación (Anexo 2)
RF08	Permita revisar los movimientos de un crédito.	Observación (Anexo 2)
RF09	Permita saber con exactitud el valor de la cuota a cancelar.	Observación (Anexo 2)
RF10	Cumplir con normativa vigente de la SBS que permita el bloqueo de tarjetas de débito en cualquier lugar u hora.	Observación (Anexo 2)
RF11	Revisar la fecha de cancelación de las operaciones crediticias	Encuesta(Anexo 3)
RF12	Revisar las cuentas de ahorro que tiene como cliente.	Encuesta(Anexo 3)
RF13	Revisar las pólizas que tiene como cliente.	Encuesta(Anexo 3)
RF14	Revisar los créditos que se tiene actualmente.	Encuesta(Anexo 3)
RF15	Ingresar a sus movimientos financieros sin necesidad de ir a las oficinas de la cooperativa.	Encuesta(Anexo 3)
RF16	Ingresar a sus movimientos financieros sin necesidad de llamar o enviar correo a la cooperativa.	Encuesta(Anexo 3)
RF17	Consulta de saldos.	Encuesta(Anexo 3)
RF18	Verificar la información bancaria por medio de un dispositivo móvil.	Entrevista (Anexo1)
RF19	Se permita tener acceso a la información bancaria en cualquier lugar y en cualquier hora.	Entrevista (Anexo1)
RF20	Los clientes puedan saber su fecha de cancelación del crédito.	Encuesta(Anexo 3)
RF21	Los clientes puedan saber su valor a cancelar del crédito.	Encuesta(Anexo 3)



RF22	Los clientes puedan revisar los abonos o pagos realizados a los créditos.	Observación (Anexo 2)
RF23	El cliente pueda bloquear su tarjeta de débito.	Encuesta(Anexo 3)
RF24	Consulta de Valor a cancelar de cuota de crédito.	Encuesta(Anexo 3)
RF25	Consulta de Fecha de Vencimiento de crédito.	Encuesta(Anexo 3)
RF26	Consultar de fechas anteriores de los movimientos de las cuentas.	Observación (Anexo 2)
RF27	Bloqueo de tarjeta de débito por pedida o robo.	Encuesta(Anexo 3)
RF28	Ingresar a las cuentas bancarias a través de un celular.	Entrevista (Anexo1)
RF29	Monitorear su información en línea.	Entrevista (Anexo1)
RF30	Verificar estado de póliza.	Observación (Anexo 2)
RF31	Verificar fecha de cancelación de intereses.	Observación (Anexo 2)
RF32	Consultar tasa de interés de pólizas.	Observación (Anexo 2)
RF33	Consultar estado de crédito.	Observación (Anexo 2)
RF34	El usuario pueda revisar únicamente su información bancaria.	Entrevista (Anexo1)
RF35	Recuperar la contraseña cuando el usuario se olvide.	Encuesta(Anexo 3)
	Cambiar la contraseña cuando el usuario se registre por primera vez.	Encuesta(Anexo 3)
RF36	Ingresar al sistema con usuario y contraseña.	Encuesta(Anexo 3)
RF37	Considerar la cédula como usuario.	Encuesta(Anexo 3)

3.2. Análisis de Requisitos

Se analizaran los requisitos solicitados por los usuarios comprobando así, la existencia de solapamiento o complejidad lo cual se detalla en la Tabla V.



TABLA V: MATRIZ DE INTERACCIÓN.

RF	RF01	RF02	RF03	RF04	RF05	RF06	RF07	RF08	RF09	RF10	RF11	RF12	RF13	RF14	RF15	RF16	RF17	RF18	RF19	RF20	RF21	RF22	RF23	RF24	RF25	RF26	RF27	RF28	RF29	RF30	RF31	RF32	RF33	RF34	RF35	RF36	RF37		
RF01															S	S		S	S									S	S										
RF02												S																											
RF03			S																						S														
RF04																																							
RF05																																							
RF06																																							
RF07																																							
RF08																																							
RF09																					S			S															
RF10																						S						S											
RF11																				S					S														
RF12	S																																						
RF13																																							
RF14						S																																	
RF15																																							
RF16																																							
RF17																																							
RF18																																							
RF19																																							
RF20																																							
RF21																																							
RF22																																							
RF23																																							
RF24																																							
RF25																																							
RF26																																							
RF27																																							
RF28																																							
RF29																																							
RF30																																							
RF31																																							
RF32																																							
RF33																																							
RF34																																							
RF35																																							
RF36																																							
RF37																																							



3.3. Especificación de Requisitos

La especificación de requisitos se la realizó bajo el estándar IEEE Std 830-1998 la misma que se encuentra validada por las partes involucradas (Ver Anexo 8 Validación de la Especificación de Requisitos).

a) INTRODUCCIÓN

Este documento es una Especificación de Requisitos de Software (ERS) para el sistema de Banca Móvil de la COAC Fortuna de la ciudad de Loja, con la ejecución del sistema se pretende facilitar a los clientes de la cooperativa el monitorear su información financiera a través de dispositivos móviles.

1. Propósito

El presente documento tiene como propósito definir las especificaciones funcionales y no funcionales para el desarrollo de un sistema bancario que permita a los clientes de la COAC Fortuna monitorear sus transacciones financieras a través de dispositivos móviles.

2. Alcance

Esta especificación de requisitos está dirigida al usuario del sistema, quienes son clientes de la COAC Fortuna y quienes previamente han descargado e instalado el aplicativo móvil.

El trabajo de titulación a desarrollar denominado "Diseño e Implementación de Banca Móvil para la Cooperativa de Ahorro y Crédito de la Microempresa Fortuna de la Ciudad de Loja, con accesibilidad desde dispositivos Android" cumple con el principal objetivo de la gestión de información bancaria de los clientes de una institución financiera con accesibilidad desde dispositivos móviles con sistema operativo Android.

3. Personal involucrado

En la tabla VI, VII, VIII se encuentran detallado las personas involucradas en el desarrollo del sistema.



TABLA VI: PERSONAL INVOLUCRADO N°1

Nombre	Natali del Cisne Cruz González
Rol	Analista, Diseñadora y Programadora
Categoría Profesional	Egresada de Ingeniería en Sistemas
Responsabilidades	Análisis de Información, diseño y programación de la aplicación
Información de Contacto	natycruz1187@gmail.com ; ncruz@coopfortuna.fin.ec

TABLA VII: PERSONAL INVOLUCRADO N°2

Nombre	Roberth Paul Bravo Castro
Rol	Jefe de Sistemas (e) de COAC Fortuna.
Categoría Profesional	Ingeniero en Sistemas.
Responsabilidades	Responsable del departamento de Sistemas de la COAC Fortuna.
Información de Contacto	help_desk@coopfortuna.fin.ec

TABLA VIII: PERSONAL INVOLUCRADO N°3

Nombre	Rosa Romero Rojas
Rol	Servicios Cooperativos e Inversiones
Categoría Profesional	Economista.
Responsabilidades	Responsable del departamento de Atención al Cliente e Inversiones.
Información de Contacto	rromero@coopfortuna.fin.ec

4. Definiciones, acrónimos y abreviaturas

En la tabla IX se detalla las definiciones, acrónimos y abreviaturas que se utilizarán en el documento.

TABLA IX: DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS

Nombre	Descripción
Cliente	Usuario quien hará uso de la aplicación Banca Móvil.
Android	Es un sistema operativo basado en el kernel de Linux diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes o tabletas.
ADK	Es un conjunto de herramientas de desarrollo de software que le permite al programador crear aplicaciones para un sistema



	concreto.
RF	Requerimientos Funcionales.
RNF	Requerimientos no Funcionales.
BM-Fortuna	Es el nombre del aplicativo Banca Móvil Fortuna.
COAC	Cooperativa de Ahorro y Crédito.
SEPS	Superintendencia de Economía Popular y Solidaria.
WSDL	Son las siglas de Web Service Description Language, un formato XML que se utiliza para describir servicios Web.

5. Referencias

En la tabla X se presenta la resolución de la Junta Bancaria a la que debe someterse la COAC Fortuna.

TABLA X: REFERENCIA DE LA RESOLUCIÓN DE LA JUNTA BANCARIA SBS

Título del Documento	Ruta	Autor
Normas Generales para la aplicación de la ley General de Instituciones del Sistema Financiero.	Resolución JB-2012-2148	Junta Bancaria del Ecuador

6. Resumen

El presente documento consta de tres secciones. En la primera sección se realiza una introducción y se proporciona una visión global de la especificación de recursos del sistema.

En la segunda sección del documento se realiza una descripción general del aplicativo con la finalidad de conocer las principales funciones que realiza, los datos asociados, los factores, restricciones, supuestos y dependencias que afectan el desarrollo sin entrar en excesivos detalles.

Por último la tercera sección del documento es aquella en la que se definen detalladamente los requisitos que debe satisfacer el sistema y el apéndice donde se referencia los anexos que justifica el desarrollo del documento.

b) DESCRIPCIÓN GENERAL

1. Perspectiva del Producto

El sistema Banca Móvil Fortuna será un producto diseñado para trabajar bajo el entorno móvil específicamente sobre Android, lo que permitirá su utilización de forma rápida, segura y eficaz.

Para mejor comprensión en la fig. 23 se muestra la integración de todos los componentes que interactúan.

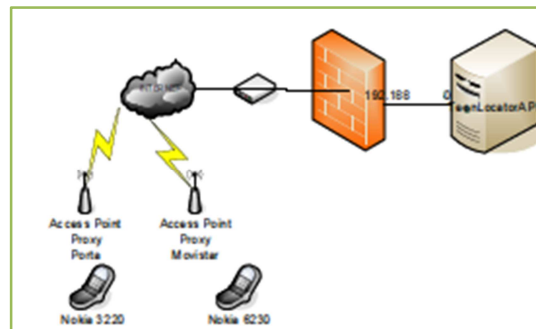


Figura 23: Estructura general del proyecto "Banca Móvil"

2. Funcionalidad del Producto

El aplicativo móvil únicamente permitirá el ingreso a los clientes de la COAC Fortuna, si algún usuario descarga el aplicativo, lo instala y se registra, el aplicativo validará la cédula contra la Base de Datos de la Institución Bancaria y emitirá los respectivos mensajes de ingreso.

El usuario podrá realizar consultas de sus movimientos financieros, registrarse en el sistema, cambiar clave, recibir correos, todas estas acciones las puede realizar en el aplicativo móvil.

3. Características de los usuarios

La tabla XI presenta las características que deben tener los usuarios para utilizar la aplicación móvil.



TABLA XI: CARACTERÍSTICAS DEL CLIENTE

Tipo de usuario	Cliente
Formación	Conocimiento en utilización en aplicativos móviles.
Habilidades	Consumidor Final.
Actividades	Administrar la aplicación móvil.

4. Restricciones

- Aplicativo de uso exclusivo para clientes de la COAC Fortuna.
- Actualizar el correo electrónico en la institución financiera previa instalación de la aplicación móvil.
- Los usuarios deben contar con dispositivos con Sistema Operativo Android con versión 2.1 o superiores.
- El aplicativo deberá tener un diseño e implementación sencilla, independientemente del lenguaje de programación.
- La aplicación debe responder de manera eficiente y rápida a cualquier operación o petición echo por el usuario.
- Para ingresar al aplicativo de banca móvil únicamente con internet sea Wifi o Plan de datos.
- La comunicación entre la aplicación móvil y el servidor de aplicaciones se lo hará mediante dirección IP.
- Las aplicaciones manejan el patrón MVC.

5. Suposiciones y dependencias

- Se asume que los requisitos aquí descritos son estables.
- Los equipos en los que se vaya a instalar el aplicativo deben estar en las versiones descritas en las restricciones.
- Para que el aplicativo Banca Móvil Fortuna funcione de manera se debe establecer comunicaciones con los módulos de captaciones Vista, Plazo y Cartera de manera que pueda ser monitoreada la información bancaria del cliente.
- El aplicativo Banca Móvil Fortuna es dependiente de los módulos antes descritos para presentar la información que hubiese sido ingresa a través del Core Bancario de la institución.
- El aplicativo Banca Móvil Fortuna cumple con las disposiciones de seguridad emitida por la Súper Intendencia de Bancos.



6. Evolución previsible del sistema

Para futuras versiones del aplicativo Banca Móvil se considera realizar transacciones interbancarias, de manera que pueda ofrecer más servicios con mayor rapidez, eficiencia, seguridad, tanto en el tratamiento de datos como en simples actividades.

Al estar relacionado con el Core Bancario de la institución financiera se debe desarrollar procesos que no afecten o alteren la comunicación de cualquier transacción.

Dado que las tecnologías a utilizar cambian con frecuencia (versiones), o por las necesidades que se presenten a la aplicación en un futuro, se sugiere dejar la posibilidad de que el sistema se pueda adaptar a los cambios de época, además se debe considerar que el aplicativo debe someterse a constantes cambios referente a normas o leyes determinadas por la SEPS que la institución debe cumplir para ofrecer el servicio de banca móvil.

c) REQUISITOS ESPECÍFICOS

Después de un análisis de los requisitos solicitados por los usuarios se presenta a continuación los requisitos específicos del aplicativo móvil.

Requisito Funcional 1

- **Autenticación de Usuarios:** El sistema permitirá el ingreso al usuario mediante su cédula y contraseña.

Requisito Funcional 2

- **Gestión de Cuentas:** El sistema permitirá al usuario ingresar a la opción de cuentas, de manera que el usuario pueda ver todas las transacciones financieras de sus cuentas bancarias.

Requisito Funcional 3

- **Gestión de Pólizas:** El sistema permitirá al usuario ingresar a la opción de pólizas, de manera que el usuario pueda ver todas las transacciones financieras de sus pólizas.



Requisito Funcional 4

- **Gestión de Créditos:** El sistema permitirá al usuario ingresar a la opción de créditos, de manera que el usuario pueda ver todos los movimientos del pago de créditos.

Requisito Funcional 5

- **Gestión de Tarjeta de Débito:** El sistema permitirá al usuario ingresar a la opción de tarjeta de débito, a través de esta opción el usuario también puede bloquear su tarjeta.

Requisito Funcional 6

- **Registro de Cliente:** El sistema permitirá al usuario ingresar a registrar usuario, con su número de cédula quien será el usuario.

Requisito Funcional 7

- **Editar Clave personal:** El sistema permitirá al usuario ingresar a la opción de Editar.

Requisito Funcional 8

- **Registrar pistas de Auditoria:** El sistema registrara las acciones del usuario.

Requisito Funcional 9

- **Enviar Correos:** El sistema enviará correos al registrarse y cambiar de clave respectivamente.

d) Requisitos comunes de las interfaces

Las diferentes pantallas que contiene el aplicativo Banca Móvil Fortuna podrán ser visualizadas desde un dispositivo Smartphone, deben estar orientadas a satisfacer las necesidades del usuario final, tener un alto grado de usabilidad sencilla y deben ser capaces de ser manipuladas por el usuario para mayor comprensión de su funcionamiento, por lo tanto incluirán diferentes vistas, botones e imágenes para su fácil comprensión y utilidad.



1. Interfaces de Usuario

La interfaz con el usuario consistirá en un conjunto de ventanas con botones, listas y campos de textos. Esta deberá ser construida específicamente para el sistema propuesto.

Vista Menú Principal: Esta pantalla contendrá un acceso a las vistas de Cuentas, Pólizas, Créditos, Tarjeta de Débito, Editar Clave y Botón de Salida, estas vistas son exclusivos para uso del cliente.

Vista Cuentas: esta vista contendrá la información de todas las cuentas que el cliente posee actualmente, esta es la vista principal, al seleccionar una cuenta específica se presentarán dos vistas correspondientes a Información de Cuenta, esta vista contendrá datos esenciales de la cuenta como fecha de apertura, valor disponible y en la vista de Movimientos de Cuenta se podrá determinar por un rango de fechas los movimientos que se han realizado sobre dicha cuenta.

Vista Pólizas: esta vista contendrá la información de todas las pólizas que el cliente posee actualmente, esta es la vista principal, al seleccionar una póliza específica se presentarán dos subvistas correspondientes a Información de Póliza, esta vista contendrá datos esenciales de la póliza como fecha de apertura, fecha de vencimiento, valor de apertura, tasa de interés y en la vista de Movimientos de Póliza se podrá determinar los movimientos que se han realizado sobre dicha póliza.

Vista Préstamos: esta vista contendrá la información de todos los créditos que el cliente posee actualmente, esta es la vista principal, al seleccionar un crédito específico se presentarán dos subvistas, la primera correspondiente a Información del Crédito, esta vista contendrá datos esenciales del crédito como fecha de desembolso, valor para estar al día en caso de vencimiento, valor a cancelar de toda la operación crediticia, número de días vencidos y estado actual del crédito y en la vista de Movimientos de Préstamos se podrá determinar los movimientos que se han efectuado sobre dicho crédito.

Vista Tarjeta Débito: Esta vista contendrá detalles e información esencial de la tarjeta de débito del cliente, además contiene una subvista Bloqueo de Tarjeta que permitirá ingresar un motivo y observaciones respecto al bloqueo de la tarjeta de débito.



Vista Editar Clave: Esta vista permitirá editar la clave de ingreso del aplicativo.

2. Interfaces de Hardware

Será necesario disponer de un dispositivo Smartphone y un servidor de aplicaciones en perfecto estado con las siguientes características:

- Adaptadores de Red
- Procesador: Processor Intel(R) Xeon(R) CPU E31220 @ 3.10GHz, 3100 MHz, 4 Core(s), 4 Logical Processor(s)
- Memoria: 6 GB
- Microsoft Windows Server 2008 R2 Standard

3. Interfaces de Software

Ya que esta aplicación será desarrollada con tecnología Android, será necesario que los usuarios que deseen ejecutar dicha aplicación, deberán tener algún dispositivo con sistema operativo Android con versiones desde 2.1 hasta la 4.0, además dicho dispositivo debe tener conexión a internet.

Para el servidor de aplicaciones Windows Server 2008 R2 con el afán de trabajar con las características de equipos que tiene la COAC Fortuna.

4. Interfaces de comunicación

Los usuarios que utilizarán la aplicación mediante internet, la comunicación entre aplicaciones y la base de datos son directos, es decir se utilizara el protocolo TCP para transmitir y administrar la información.



e) Requisitos Funcionales

Identificación del Requerimiento	RF01
Nombre del Requerimiento:	Autenticación de Usuarios
Características:	Los usuarios deberán identificarse para acceder a cualquier funcionalidad del aplicativo.
Descripción del Requerimiento:	El sistema podrá ser consultado por cualquier cliente de la COAC Fortuna, mediante su usuario y contraseña, debe tener asociado un correo electrónico con el gestor bancario, esta información se la otorga al ingresar los datos de cliente, si el usuario no tiene asociado un correo el sistema no le permitirá ingresar al aplicativo móvil, para ello el usuario debe actualizar su información con el gestor bancario.
Prioridad del Requerimiento:	Alta

Identificación del Requerimiento	RF02
Nombre del Requerimiento:	Gestión de Cuentas
Características:	Se vincula con el módulo de Captaciones Vista.
Descripción del Requerimiento:	La aplicación ofrecerá al usuario una opción que le permita al cliente monitorear todas las cuentas que tenga apertura independiente del estado que dicha cuenta se encuentre, el usuario podrá revisar información específica de cada cuenta, entre los datos a observar están fecha de apertura y movimientos de cada cuenta de acuerdo a un rango de fechas.
Prioridad del Requerimiento:	Alta

Identificación del Requerimiento	RF03
Nombre del Requerimiento:	Gestión de Pólizas
Características:	Se vincula con el módulo de Captaciones Plazo.



Descripción del Requerimiento:	La aplicación permitirá monitorear todas las pólizas, no se presentaran las pólizas que se encuentran canceladas, el aplicativo permitirá revisar la información de una póliza específica como la fecha de apertura, fecha de vencimiento, tasa de interés y puede revisar los movimientos de cada póliza como cancelación de interés.
Prioridad del Requerimiento:	Alta

Identificación del Requerimiento	RF04
Nombre del Requerimiento:	Gestión de Créditos
Características:	Se vincula con el módulo de Cartera.
Descripción del Requerimiento:	La aplicación debe permitir al cliente monitorear todos los créditos que tiene como activos, el aplicativo permite verificar la información de cada crédito como fecha de cancelación de cuota, en el caso de vencimiento días de vencimiento, valor a cancelar para estar al día, valor total del crédito a cancelar a la fecha y movimientos realizados en cada crédito.
Prioridad del Requerimiento:	Alta

Identificación del Requerimiento	RF05
Nombre del Requerimiento:	Gestión de Tarjeta de Débito
Características:	Se vincula con el módulo de Captaciones Vista.
Descripción del Requerimiento:	La aplicación debe contener una opción que le permita al cliente revisar su información respecto a Tarjeta de Débito y a su vez realizar bloqueo sobre dicha tarjeta, ingresando un motivo y observaciones.
Prioridad del Requerimiento:	Alta

Identificación del Requerimiento	RF06
Nombre del Requerimiento:	Registro de Cliente
Características:	Se vincula con el módulo de Personas.



Descripción del Requerimiento:	Todos los clientes para ingresar al aplicativo primero deben Registrarse, la contraseña le será enviada al correo electrónico que definió al gestor bancario, si necesita cambiar de correo es necesario que actualice sus datos en la institución financiera.
Prioridad del Requerimiento:	Alta

Identificación del Requerimiento	RF07
Nombre del Requerimiento:	Editar Clave.
Características:	Se relaciona con el requerimiento RF04
Descripción del Requerimiento:	El cliente luego de recibir la clave en su correo puede editar dicha clave, este registro quedara grabado en la base de históricos el cambio realizado.
Prioridad del Requerimiento:	Media

Identificación del Requerimiento	RF08
Nombre del Requerimiento:	Registrar pistas de Auditoria.
Características:	Se relaciona con todos los requerimientos antes descritos.
Descripción del Requerimiento:	Se registraran todos los movimientos realizados por el usuario.
Prioridad del Requerimiento:	Alta

Identificación del Requerimiento	RF09
Nombre del Requerimiento:	Enviar Correos.
Características:	Se relaciona con todos los requerimientos antes descritos.
Descripción del Requerimiento:	La aplicación permitirá al cliente conocer los procesos que se realiza recibiendo un correo electrónico.
Prioridad del Requerimiento:	Media



f) Requisitos no Funcionales

Identificación del Requerimiento	RNF01
Nombre del Requerimiento:	Usabilidad.
Características:	El aplicativo garantiza a los usuarios un aplicativo sencillo y fácil de dar uso, las interfaces son amigables para el usuario.
Descripción del Requerimiento:	El aplicativo presenta vistas sencillas e intuitivas para con los usuarios del sistema, de manera que presente imágenes que describan la funcionalidad de las vistas.
Prioridad del Requerimiento:	Alta

Identificación del Requerimiento	RNF02
Nombre del Requerimiento:	Seguridad de la Información.
Características:	El aplicativo garantiza a los usuarios las seguridades necesarias para el ingreso del aplicativo.
Descripción del Requerimiento:	Garantizar la confiabilidad, la seguridad y el desempeño del aplicativo a los diferentes usuarios. Garantizar la seguridad del sistema con respecto a la información y datos que se manejan como contraseñas e información en general. Seguridad de encriptación para la clave del usuario.
Prioridad del Requerimiento:	Alta

Identificación del Requerimiento	RNF04
Nombre del Requerimiento:	Confiabilidad.
Características:	El aplicativo garantiza a los usuarios un desempeño en cuanto a los datos almacenados en la base del Core Bancario, de esta manera garantiza la confiabilidad de la información.



Descripción del Requerimiento:	Garantizar el desempeño del aplicativo a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y monitoreados de forma permanente y simultáneamente, sin que afecte el tiempo de respuesta.
Prioridad del Requerimiento:	Alta

Identificación del Requerimiento	RNF05
Nombre del Requerimiento:	Disponibilidad
Características:	El aplicativo garantiza a los usuarios estar en funcionamiento las 24 horas los 7 días de la semana.
Descripción del Requerimiento:	La disponibilidad del aplicativo debe ser continua con un nivel de servicio para los usuarios de 7 días por 24 horas, garantizando un esquema adecuado que permita la posible falla en cualquiera de sus componentes.
Prioridad del Requerimiento:	Alta





Identificación del Requerimiento	RNF06
Nombre del Requerimiento:	Rendimiento
Características:	El aplicativo deberá funcionar sin inconvenientes en el Sistema Operativo Android.
Descripción del Requerimiento:	El aplicativo se basa en base a las restricciones definidas en el documento. Las vistas que contienen más de un registro deben presentarse de manera que puedan ser observadas por el cliente.
Prioridad del Requerimiento:	Alta



Identificación del Requerimiento	RNF07
Nombre del Requerimiento:	Escalabilidad
Características:	El aplicativo debe permitir ser actualizado y mejorado para integrar nuevas funcionalidades.
Descripción del Requerimiento:	La aplicación abarca sólo una parte del servicio que ofrece la cooperativa a sus clientes. Pero en un futuro podría desarrollarse una aplicación que integrase todos los servicios ofrecidos por la COAC Fortuna.
Prioridad del Requerimiento:	Alta

g) Apéndices

Para la justificación de la especificación de requisitos se detalla los siguientes anexos:

-  Anexo 1: Entrevista con el Jefe de Sistemas y Servicios Cooperativos e Inversiones de la COAC Fortuna.
-  Anexo 2: Ficha de Observación,
-  Anexo 3: Modelo de Encuesta.
-  Anexo 8: Validación de la Especificación de Requisitos.

FASE 2:

3.4. Modelo de Dominio

Luego de determinar los procesos, se obtienen los conceptos y las clases que se utilizaran en el desarrollo del sistema, las cuales se las represento a dichas clases en el modelo del dominio que se detallan en la fig. 24.

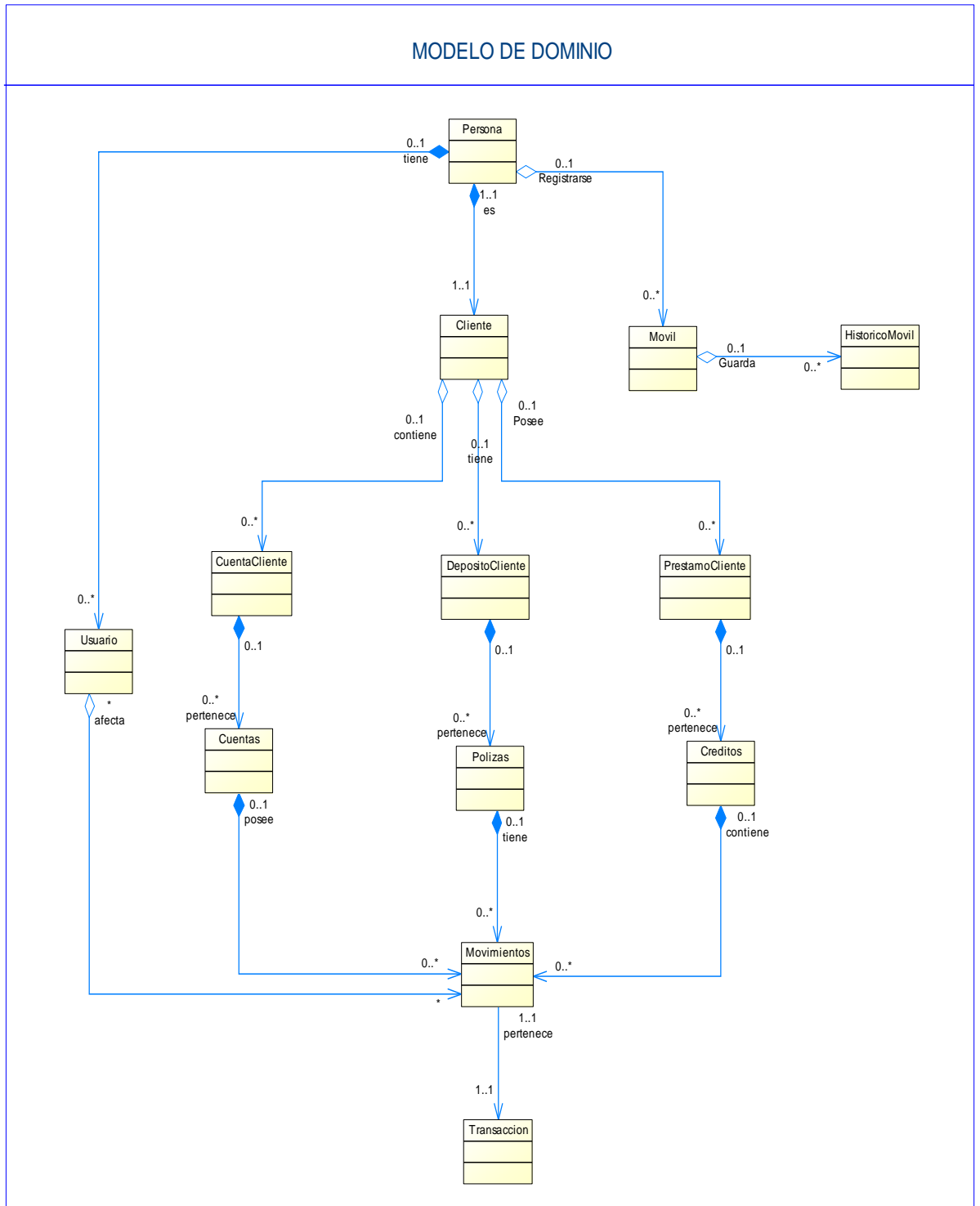


Figura 24: Modelo de Dominio de la aplicación BM Móvil

3.5. Modelo de Casos de uso

El modelo de Casos de Uso representa la forma en como el Cliente (Actor 1) y Banca Móvil (Actor 2), opera con el aplicativo móvil, además de la forma, tipo y orden en como los elementos interactúan (operaciones o casos de uso) tal como se muestra en las fig.25, 26, 27, 28 y 29.

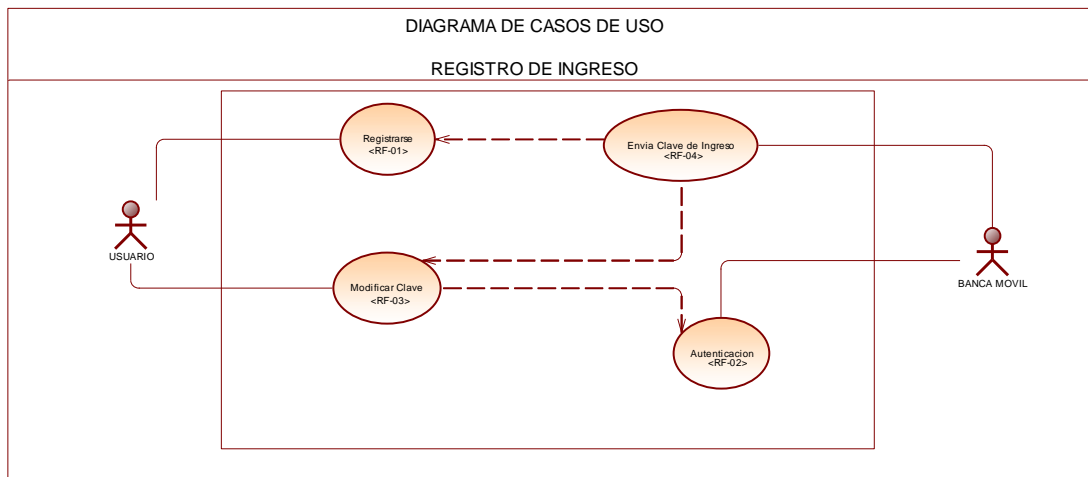


Figura 25: Diagrama de CU Registro de Ingreso

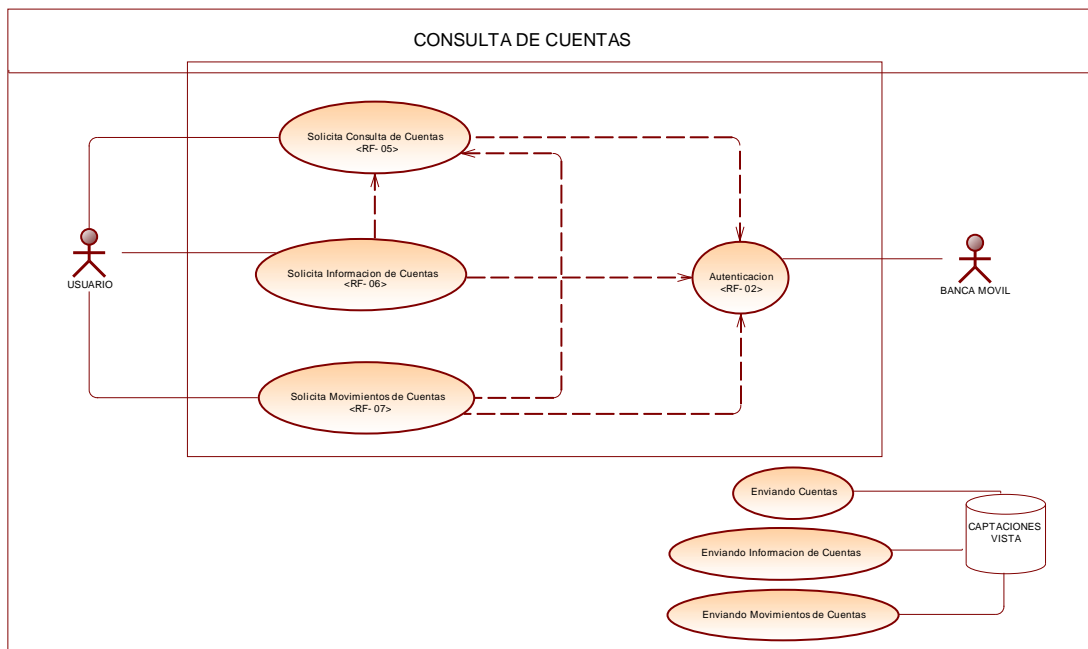


Figura 26: Diagrama de CU Consulta de Cuentas

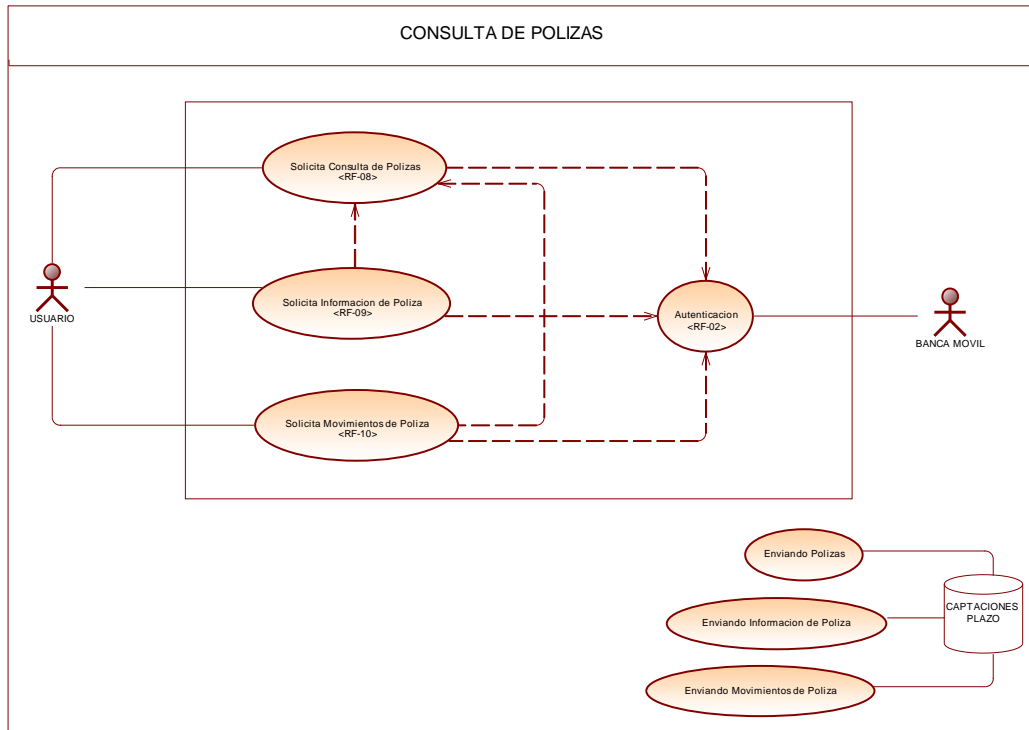


Figura 27: Diagrama de CU Consulta de Pólizas

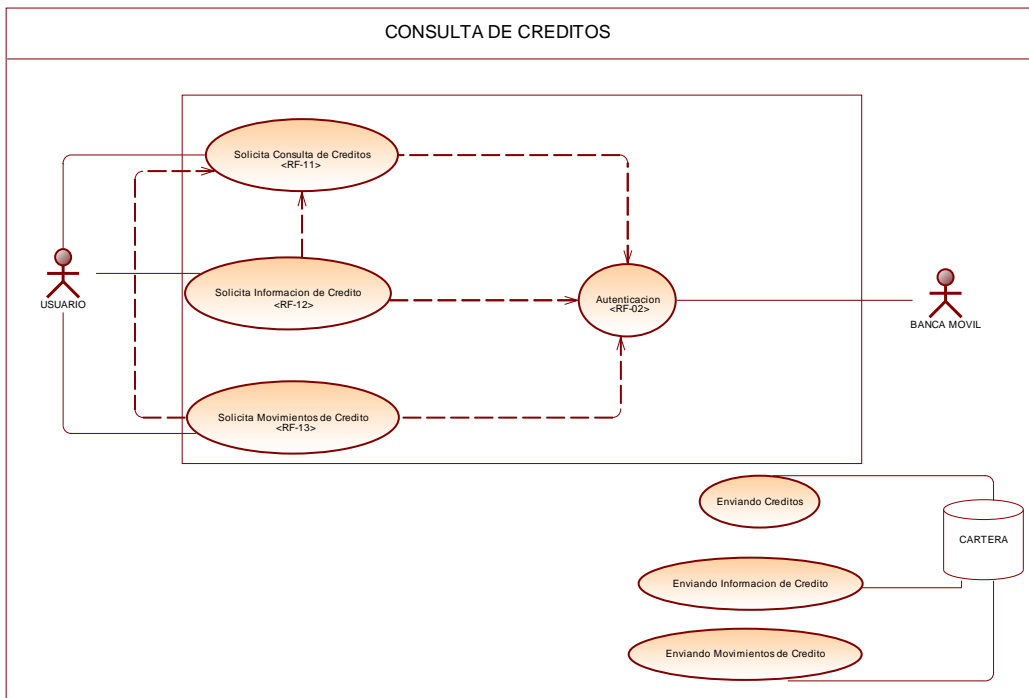


Figura 28: Diagrama de CU Consulta de Créditos

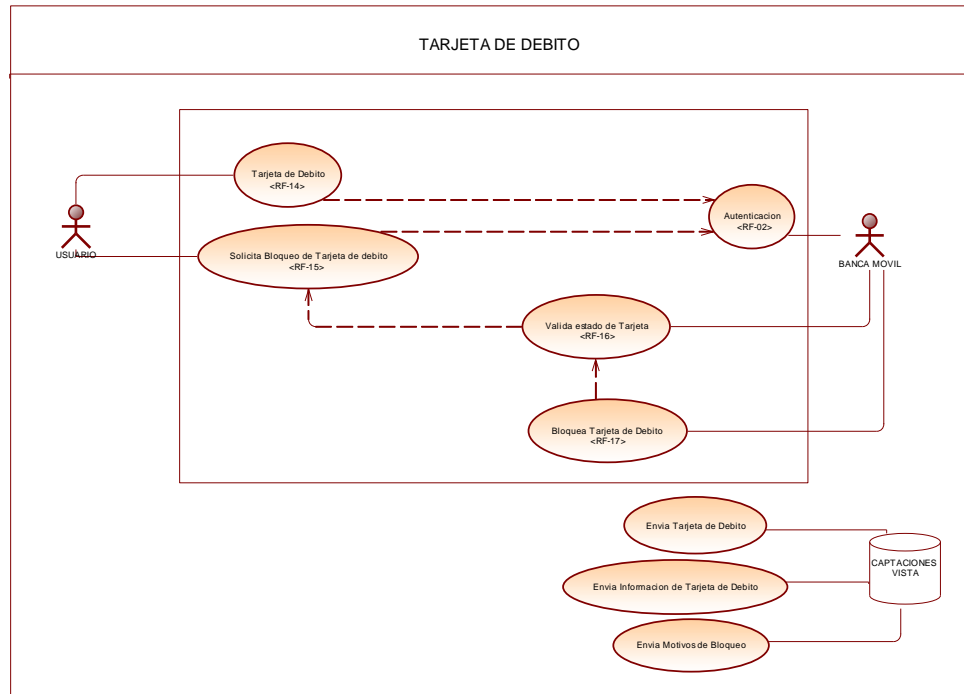


Figura 29: Diagrama de CU Tarjeta de Debito

4. Diseño e Implementación de los Requisitos

En esta fase consta el desarrollo del prototipado inicial del sistema móvil, descripción de los casos de uso, diagrama de clases, diseño de la base de datos, diagrama de paquetes e implementación del código fuente.

4.1. Prototipado Inicial

El prototipado es útil en diferentes fases del proyecto, durante la fase de análisis se usa para obtener los requerimientos del usuario. En la fase de diseño se usa para ayudar a evaluar muchos aspectos de la implementación seleccionada.

4.1.1. Prototipado Inicial del Aplicativo Banca Móvil

Para el prototipado inicial del aplicativo Banca Móvil, se utilizó el SDK de Android en Eclipse que permitió diseñar de manera rápida la interfaz de la aplicación.



Figura 30: Ingreso al sistema



Figura 31: Registro

En la fig. 30 se muestra la pantalla inicial que se presenta al usuario al abrir el aplicativo móvil, consta de las siguientes secciones:

- Una cabecera en la que está el logo de la COAC Fortuna.
- Los campos de cédula y contraseña para ser ingresados por el usuario de acuerdo a su información personal.
- La opción Registrar, cuando el usuario ingrese por primera vez.
- La opción Olvide Contraseña.

En la fig. 31 se muestra la pantalla de registro que se presenta al usuario consta de las siguientes secciones:

- Los campos de cédula y correo para ser ingresados por el usuario de acuerdo a su información personal.
- Solicitar código de ingreso.



Figura 32: Menú Principal



Figura 33: Cuentas

En la fig. 32 se muestra el menú principal en el que se presenta el nombre de cliente y el respectivo logo de la cooperativa, en su estructura consta de las siguientes secciones:

- Cuentas, Pólizas, Prestamos, Tarjeta de débito,
- Cerrar sesión y su respectiva confirmación.

En la fig. 33 se muestra las cuentas que el usuario tiene y consta de las siguientes secciones:

- Número de cuenta, Nombre de cuenta, Estado de la Cuenta,
- Saldo disponible,
- Información,
- Movimientos,
- Parte superior presenta el nombre del cliente.



Figura 34: Información Cuentas



Figura 35: Movimientos Cuentas

En la fig. 34 se muestra la información de la cuenta seleccionada y consta de las siguientes secciones:

- Nombre
- Oficina
- Fecha Apertura
- Saldo
- Disponible Ahorro y Bloqueo Ahorro

En la fig. 35 se muestran los movimientos de la cuenta seleccionada y consta de las siguientes secciones:

- Fecha desde
- Fecha hasta
- Fecha de transacción
- Nombre de Transacción
- Valor y Oficina



Figura 36: Pólizas



Figura 37: Información Póliza

En la fig.36 se muestra las pólizas que el cliente tiene y consta de las siguientes secciones:

- Código, Tipo de póliza
- Monto, Saldo
- Fecha de creación y vencimiento

En la fig. 37 se muestra la información de las pólizas que el cliente tiene y consta de las siguientes secciones:

- Número de secuencial de Cliente y nombre de Cliente
- Oficina
- Producto
- Fecha apertura, Monto
- Tasa, TEA
- Plazo



Figura 38: Movimientos Póliza



Figura 39: Créditos

En la fig. 38 se muestra los movimientos de las pólizas que el cliente selecciona y consta de las siguientes secciones:

- Fecha
- Transacción
- Tipo
- Valor
- Oficina

En la fig. 39 se muestra los créditos que el cliente tiene y consta de las siguientes secciones:

- Número de préstamo
- Deuda Inicial
- Saldo
- Fecha Adjudicación
- Estado



Figura 40: Información de Préstamo



Figura 41: Tarjeta de debito

En la fig. 40 se muestra la información del crédito seleccionado que el cliente tiene y consta de las siguientes secciones:

- Cliente, Oficina
- Número de Préstamo
- Deuda inicial y Saldo actual
- Fecha adjudicación, Fecha último pago y Fecha próximo vencimiento
- Valor a pagar
- Valor para estar al día



En la fig. 41 se muestra la tarjeta de débito que el cliente tiene y consta de las siguientes secciones:

- Código
- Nombre de Cliente
- Fecha de solicitud
- Estado y Oficina



Figura 42: Tarjeta de debito

En la fig. 42 se muestra la opción de bloqueo de la tarjeta de débito y consta de las siguientes secciones:

-  Tipo de bloqueo
-  Observaciones

4.2. Descripción de los Casos de Uso.

Luego de haber entregado el prototipado inicial del aplicativo móvil al personal de la COAC Fortuna y recibido las sugerencias se ha realizado el prototipado final con su respectiva descripción.

4.2.1. Prototipado Final de la aplicación Banca Móvil Fortuna

En la fig. 43 se presenta la pantalla principal de la aplicación y los servicios que presta los cuales son: Cuentas, Pólizas, Créditos, Tarjeta de Débito, Editar Contraseña y Salir.




Figura 43: Pantalla Ingreso

a) REGISTRARSE

En la Tabla XII se presenta la pantalla de registro de la aplicación.

TABLA XII: PANTALLA DE REGISTRO DEL APLICATIVO BANCA MÓVIL

Código: CU001	
Tipo de Interfaz: Android	
Caso de Uso: Registro	
Pantalla de Registro	Servicios de la aplicación
	
Pantalla de Clave en Correo	
	

En la Tabla XIII se describe el caso de uso Registro y la interacción entre el cliente y la aplicación móvil.



TABLA XIII: DESCRIPCIÓN DEL CASO DE USO REGISTRO

NOMBRE DEL CASO DE USO: Registro		CÓDIGO: CU001
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF001	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente registrarse en el aplicativo móvil y poder empezar a dar uso de las funcionalidades.	El usuario ingresa la cédula y el sistema envía la contraseña a su correo electrónico para acceder al servicio de la aplicación.	
PRE-CONDICIONES	POST CONDICIONES	
Registro del cliente en el Core Financiero de la COAC Fortuna y de su correo electrónico.	Verificar el correo electrónico donde se envía la clave de acceso.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Ingresa la cédula.		
2. Presiona el botón [Enviar Clave a Correo].		
	3. Válida la cédula del usuario.	
	4. Envía clave aleatoria al correo electrónico.	
5. Revisa correo electrónico.		
	6. Presenta la ventana de Autenticación del aplicativo de banca móvil. (CU continúa en el paso 1 del CU de Autenticación).	
	7. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
<p>Ítem 1.</p> <p>USUARIO NO ES CLIENTE DE COAC FORTUNA</p> <p>A1. Se válida la cédula contra la base de datos de la COAC Fortuna, al no tener ninguna coincidencia la aplicación presenta el mensaje: "Aún no es cliente de la Cooperativa FORTUNA".</p> <p>DATOS INCORRECTOS</p> <p>B1. El sistema muestra un mensaje: "Cédula/Pasaporte o contraseña incorrecta". El caso de uso continúa en el paso 1 del flujo normal de eventos.</p> <p>CAMPO CÉDULA VACÍA</p> <p>C1: El sistema muestra "Existen campos vacíos". El caso de uso continúa en el paso 1 del flujo normal de eventos.</p>		

b) AUTENTICACIÓN

En la Tabla XIV se presenta la pantalla de autenticación en la aplicación.

TABLA XIV: PANTALLA DE AUTENTICACIÓN DEL APLICATIVO BANCA MÓVIL

Código: CU002	
Tipo de Interfaz: Android	
Caso de Uso: Autenticación	
Pantalla Principal	Servicios de la aplicación
	



En la Tabla XV se describe el caso de uso Autenticación y la interacción entre el cliente y la aplicación móvil.

TABLA XV: DESCRIPCIÓN DEL CASO DE USO AUTENTICACIÓN

NOMBRE DEL CASO DE USO: Autenticación		CÓDIGO: CU002
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF002	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente ingresar al aplicativo de banca móvil y hacer uso del mismo.	El usuario ingresa la cédula y su contraseña para acceder al servicio de la aplicación.	
PRE-CONDICIONES	POST CONDICIONES	
Registro del cliente en el Core Financiero de la COAC Fortuna. Que el cliente tenga clave de ingreso. Que tenga conexión a internet. Que la COAC Fortuna mantenga activo el servidor de aplicaciones y la base de datos.	Se presenta la interfaz que contiene todos los servicios.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Ingresar la cédula y contraseña.		
2. Presiona el botón [Ingresar]		
	3. Valida que la información que ha ingresado sea la misma que esta almacenada en la base de datos.	
	4. Presenta la ventana principal del aplicativo de banca móvil.	
	5. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
<p>Ítem 2.</p> <p>USUARIO NO ES CLIENTE DE COAC FORTUNA</p> <p>A1. Se valida la cédula contra la base de datos de la COAC Fortuna, al no tener ninguna coincidencia la aplicación presenta el mensaje: "Aun no es cliente de la Cooperativa FORTUNA".</p> <p>DATOS INCORRECTOS</p> <p>B1. El sistema muestra un mensaje: "Cédula/Pasaporte o contraseña incorrecta". El caso de uso continúa en el paso 1 del flujo normal de eventos.</p>		

<p>CAMPO CÉDULA VACIA C1: El sistema muestra "Existen campos vacíos". El caso de uso continúa en el paso 1 del flujo normal de eventos.</p> <p>CAMPO CLAVE VACÍO D1: El sistema muestra "Existen campos vacíos". El caso de uso continúa en el paso 1 del flujo normal de eventos.</p>
--

c) MODIFICAR CLAVE

En la Tabla XVI se presenta la pantalla de modificar clave en la aplicación.

TABLA XVI: PANTALLA DE MODIFICAR CLAVE DEL APLICATIVO BANCA MÓVIL

Código: CU003	
Tipo de Interfaz: Android	
Caso de Uso: Modificar Clave	
Pantalla Principal	Servicios de la aplicación
	



En la Tabla XVII se describe el caso de uso Modificar Clave y la interacción entre el cliente y la aplicación móvil.

TABLA XVII: DESCRIPCIÓN DEL CASO DE USO MODIFICAR CLAVE




NOMBRE DEL CASO DE USO: Modificar Clave		CÓDIGO: CU003
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF003	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente modificar su clave de ingreso al aplicativo de banca móvil y hacer uso del mismo.	El usuario selecciona la interfaz Modificar Clave e ingresa su nueva clave de ingreso al aplicativo móvil.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente se encuentre registrado en el aplicativo banca móvil. Que el cliente tenga registrado el correo electrónico, información a ser ingresada en las oficinas de la Coop Fortuna.	Se modifica la clave de ingreso al aplicativo de banca móvil.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Ingresar nueva contraseña.		
2. Presiona el botón [Actualizar Clave]		
	3. Registra la nueva clave de ingreso de la aplicación banca móvil.	
	4. Envía mensaje de actualización y nueva clave de ingreso al correo electrónico.	
	5. Presenta la ventana de menú principal del aplicativo de banca móvil.	
	6. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
Ítem 3.		
CAMPOS VACÍOS		
A1. Si el cliente presiona directamente el botón [Consultar], se presentan el mensaje "No deben haber campos vacíos".		



d) CONSULTA DE CUENTAS

En la Tabla XVII se presenta la pantalla de consulta de cuentas en la aplicación.

TABLA XVIII: PANTALLA DE CONSULTA DE CUENTAS DEL APLICATIVO BANCA MÓVIL

Código: CU004	
Tipo de Interfaz: Android	
Caso de Uso: Consulta de Cuentas	
Icono de Cuentas	Cuentas
	
Icono de Cuentas	
	



En la Tabla XIX se describe el Caso de Uso Consulta de cuentas y la interacción entre el cliente y la aplicación móvil.

TABLA XIX: DESCRIPCIÓN DEL CASO DE USO CONSULTA DE CUENTAS

NOMBRE DEL CASO DE USO: Consulta de Cuentas		CÓDIGO: CU004
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF004	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente realizar consultas en el aplicativo de banca móvil y hacer uso del mismo.	El usuario selecciona el icono de cuentas del menú principal.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente se autentique en el aplicativo banca móvil.	Se presenta las cuentas que tiene asociadas a su cédula.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona el icono de Cuentas		
	2. Valida las cuentas que tiene asociadas al cliente.	
	3. Presenta las cuentas en la pantalla "Cuentas" del aplicativo de banca móvil.	
	4. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
Ítem 4.		
USUARIO NO TIENE CUENTAS		
A1. Si el cliente no tiene asociadas cuentas, el sistema presenta en vacío la pantalla "Cuentas".		
OPCIÓN INFORMACIÓN		
B1. El cliente selecciona Información presionando la opción [Información]: Se hace referencia al caso de uso "Información de Cuenta"		
OPCIÓN MOVIMIENTO		
C1. El cliente selecciona Movimientos presionando la opción [Movimientos]: Se hace referencia al caso de uso "Movimientos de Cuenta"		



e) INFORMACIÓN DE CUENTA

En la Tabla XXI se presenta la pantalla de Información de Cuenta en la aplicación.

TABLA XXI: PANTALLA DE INFORMACIÓN DE CUENTA DEL APLICATIVO BANCA MÓVIL

Código: CU005	
Tipo de Interfaz: Android	
Caso de Uso: Información de Cuenta	
Cuentas	Información de Cuenta

En la Tabla XXI se describe el caso de uso Información de Cuenta y la interacción entre el cliente y la aplicación móvil.



TABLA XXI: DESCRIPCIÓN DEL CASO DE USO INFORMACIÓN DE CUENTA

NOMBRE DEL CASO DE USO: Información de Cuenta		CÓDIGO: CU005
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF005	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente realizar consultas de la información de la cuenta en el aplicativo de banca móvil y hacer uso del mismo.	El usuario selecciona la opción de información en la cuenta.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente tenga al menos una cuenta creada.	Se presenta la interfaz que contiene la información de la cuenta específica.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona la cuenta.		
2. Clic derecho sobre la cuenta.		
3. Selecciona la opción Información.		
	4. Presenta la ventana Información Cuenta del aplicativo de banca móvil.	
	5. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
Ítem 5.		
CUENTA CERRADA		
A1. Se presentan datos en la pantalla de Información Cuenta independientemente del estado de la cuenta.		



f) MOVIMIENTO DE CUENTA

En la Tabla XXII se presenta la pantalla de movimiento de cuenta en la aplicación.

TABLA XXII: PANTALLA DE MOVIMIENTO DE CUENTA BANCA MÓVIL

Código: CU006	
Tipo de Interfaz: Android	
Caso de Uso: Movimiento de Cuenta	
Cuentas	Movimiento de Cuenta

En la Tabla XXIII se describe el caso de uso Movimiento de Cuenta y la interacción entre el cliente y la aplicación móvil.



TABLA XXIII: DESCRIPCIÓN DE CASO DE USO MOVIMIENTO DE CUENTA

NOMBRE DEL CASO DE USO: Movimiento de Cuenta		CÓDIGO: CU006
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF006	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente realizar consultas de los movimientos de la cuenta en el aplicativo de banca móvil.	El usuario selecciona la opción de movimiento en la cuenta.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente tenga al menos una cuenta creada.	Se presenta la interfaz que contiene los movimientos de la cuenta específica.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona la cuenta.		
2. Clic derecho sobre la cuenta.		
3. Selecciona la opción Movimiento.		
	4. Presenta la ventana Movimiento Cuenta del aplicativo de banca móvil.	
5. Selecciona rango de fechas.		
6. Presiona el botón [Consultar].		
	7. Valida las fechas seleccionadas y presenta la información.	
8. Verifica los movimientos realizados.		
	9. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
<p>Ítem 6.</p> <p>CAMPOS VACÍOS</p> <p>A1. Si el cliente presiona directamente el botón [Consultar], se presentan el mensaje "No deben haber campos vacíos".</p> <p>RANGO DE FECHAS</p> <p>B1. Si el cliente presiona fechas posteriores en las opciones Desde – Hasta, el sistema valida que no existen movimientos en dichas fechas y presenta vacío.</p>		



g) CONSULTA DE PÓLIZAS

En la Tabla XXIV se presenta la pantalla de consulta de pólizas en la aplicación.

TABLA XXIV: PANTALLA DE CONSULTA DE PÓLIZAS DEL APLICATIVO BANCA MÓVIL

Código: CU007	
Tipo de Interfaz: Android	
Caso de Uso: Consulta de Pólizas	
Icono de Póliza	Pólizas
Opciones de Póliza	



En la Tabla XXV se describe el caso de uso Consulta de Pólizas y la interacción entre el cliente y la aplicación móvil.

TABLA XXV: DESCRIPCIÓN DEL CASO DE USO CONSULTA DE PÓLIZAS

NOMBRE DEL CASO DE USO: Consulta de Pólizas		CÓDIGO: CU007
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF007	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente realizar consultas de sus pólizas en el aplicativo de banca móvil y hacer uso del mismo.	El usuario selecciona el icono de pólizas del menú principal.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente se autentique en el aplicativo banca móvil.	Se presenta las pólizas que tiene asociadas a su cédula.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona el icono de Pólizas		
	2. Valida las pólizas que tiene asociadas al cliente, se valida que sean diferente de cancelada.	
	3. Presenta las pólizas en la pantalla "Pólizas" del aplicativo de banca móvil.	
	4. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
Ítem 7.		
USUARIO NO TIENE PÓLIZAS		
A1. Si el cliente no tiene asociadas pólizas, el sistema presenta en vacío la pantalla "Cuentas".		
OPCIÓN INFORMACIÓN		
B1. El cliente selecciona Información presionando la opción [Información]: Se hace referencia al caso de uso "Información de Póliza"		
OPCIÓN MOVIMIENTO		
C1. El cliente selecciona Movimientos presionando la opción [Movimientos]: Se hace referencia al caso de uso "Movimientos de Póliza"		

h) INFORMACIÓN DE PÓLIZA

En la Tabla XXVI se presenta la pantalla de información de póliza en la aplicación.

TABLA XXVI: PANTALLA DE INFORMACIÓN DE PÓLIZA DEL APLICATIVO BANCA MÓVIL

Código: CU008	
Tipo de Interfaz: Android	
Caso de Uso: Información de Póliza	
Pólizas	Información de Póliza
	

En la Tabla XXVII se describe el caso de uso Información de Póliza y la interacción entre el cliente y la aplicación móvil.



TABLA XXVII: DESCRIPCIÓN DEL CASO DE USO INFORMACIÓN DE PÓLIZA

NOMBRE DEL CASO DE USO: Información de Póliza		CÓDIGO: CU008
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF008	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente realizar consultas de la información de la póliza en el aplicativo de banca móvil y hacer uso del mismo.	El usuario selecciona la opción de información en la póliza.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente tenga al menos una póliza aperturada.	Se presenta la interfaz que contiene la información de la póliza específica.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona la póliza.		
2. Clic derecho sobre la póliza.		
3. Selecciona la opción Información.		
	4. Presenta la ventana Información Póliza del aplicativo de banca móvil.	
	5. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
Ítem 8.		
PÓLIZA CERRADA		
A1. Se presentan las pólizas que tengan un estado diferente de cancelada.		

i) MOVIMIENTO DE PÓLIZA

En la Tabla XXVIII se presenta la pantalla de movimiento de póliza en la aplicación.

TABLA XXVIII: PANTALLA DE MOVIMIENTO DE PÓLIZA DEL APLICATIVO BANCA MÓVIL

Código: CU009	
Tipo de Interfaz: Android	
Caso de Uso: Movimiento de Póliza	
Pólizas	Movimiento de Póliza

En la Tabla XXIX se describe el caso de uso Movimiento de Póliza y la interacción entre el cliente y la aplicación móvil.



TABLA XXIX: DESCRIPCIÓN DEL CASO DE USO MOVIMIENTOS DE PÓLIZA

NOMBRE DEL CASO DE USO: Movimientos de Póliza		CÓDIGO: CU009
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF009	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente realizar consultas de los movimientos de la póliza en el aplicativo de banca móvil.	El usuario selecciona la opción de movimiento en la póliza.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente tenga al menos una póliza creada.	Se presenta la interfaz que contiene los movimientos de la póliza específica.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona la póliza.		
2. Clic derecho sobre la póliza.		
3. Selecciona la opción Movimiento.		
	4. Presenta la ventana Movimiento Póliza del aplicativo de banca móvil.	
	5. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
Ítem 9.		
USUARIO NO TIENE MOVIMIENTOS		
A1. Si el cliente no tiene asociados movimientos de la póliza, el sistema presenta en vacío la pantalla "Movimiento Póliza".		
FECHAS INCORRECTAS		
B1. Si el usuario selecciona fechas incorrectas el sistema presenta [Fechas Incorrectas].		



j) CONSULTA DE CRÉDITOS

En la Tabla XXX se presenta la pantalla de consulta de créditos en la aplicación.

TABLA XXX: PANTALLA DE CONSULTA DE CRÉDITO DEL APLICATIVO BANCA MÓVIL

Código: CU010	
Tipo de Interfaz: Android	
Caso de Uso: Consulta de Créditos	
Icono de crédito	Créditos
	
Opciones de crédito	
	



En la Tabla XXXI se describe el caso de uso Consulta de Créditos y la interacción entre el cliente y la aplicación móvil.

TABLA XXXI: DESCRIPCIÓN DE CASO DE USO CONSULTA DE CRÉDITOS

NOMBRE DEL CASO DE USO: Consulta de Créditos		CÓDIGO: CU010
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF010	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente realizar consultas de sus créditos en el aplicativo de banca móvil.	El usuario selecciona el icono de créditos del menú principal.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente se autentique en el aplicativo banca móvil.	Se presenta los créditos que tiene asociadas a su cédula.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona el icono de crédito.		
	2. Valida los créditos que tiene asociadas al cliente, se valida que sean diferentes de cancelada.	
	3. Presenta los créditos en la pantalla "Créditos" del aplicativo de banca móvil.	
	4. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
Ítem 10.		
USUARIO NO TIENE CRÉDITOS		
A1. Si el cliente no tiene asociados créditos, el sistema presenta en vacío la pantalla "Cuentas".		
OPCIÓN INFORMACIÓN		
B1. El cliente selecciona Información presionando la opción [Información]: Se hace referencia al caso de uso "Información de Crédito".		
OPCIÓN MOVIMIENTO		
C1. El cliente selecciona Movimientos presionando la opción [Movimientos]: Se hace referencia al caso de uso "Movimientos de Crédito"		



k) INFORMACIÓN DE CRÉDITO

En la Tabla XXXII se presenta la pantalla de información de crédito en la aplicación.

TABLA XXXII: PANTALLA DE INFORMACIÓN DE CRÉDITO DEL APLICATIVO BANCA MÓVIL

Código: CU011	
Tipo de Interfaz: Android	
Caso de Uso: Información de Crédito	
Créditos	Información de Crédito

En la Tabla XXXIII se describe el caso de uso Información de crédito y la interacción entre el cliente y la aplicación móvil.



TABLA XXXIII: DESCRIPCIÓN DEL CASO DE USO INFORMACIÓN DE CRÉDITO

NOMBRE DEL CASO DE USO: Información de crédito		CÓDIGO: CU011
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF011	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente realizar consultas de la información del crédito en el aplicativo de banca móvil y hacer uso del mismo.	El usuario selecciona la opción de información de crédito.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente tenga desembolsado al menos un crédito.	Se presenta la interfaz que contiene la información del crédito específico.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona un crédito.		
2. Clic derecho sobre el crédito.		
3. Selecciona la opción Información.		
	4. Presenta la ventana Información Prestamos del aplicativo de banca móvil.	
	5. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
Ítem 11.		
CRÉDITO CANCELADO		
A1. Se presentan los créditos que tengan un estado diferente de cancelado.		



I) MOVIMIENTO DE CRÉDITO

En la Tabla XXXIV se presenta la pantalla de movimiento de crédito en la aplicación.

TABLA XXXIV: PANTALLA DE MOVIMIENTO DE CRÉDITO DEL APLICATIVO BANCA MÓVIL

Código: CU012	
Tipo de Interfaz: Android	
Caso de Uso: Movimiento de Crédito	
Créditos	Movimiento de Créditos

En la Tabla XXXV se describe el caso de uso Movimiento de Crédito y la interacción entre el cliente y la aplicación móvil.



TABLA XXXV: DESCRIPCIÓN DEL CASO DE USO MOVIMIENTOS DE CRÉDITO

NOMBRE DEL CASO DE USO: Movimientos de Crédito		CÓDIGO: CU012
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF012	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente realizar consultas de los movimientos del crédito en el aplicativo de banca móvil.	El usuario selecciona la opción de movimiento en el crédito.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente tenga al menos un crédito desembolsado.	Se presenta la interfaz que contiene los movimientos del crédito específico.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona el crédito.		
2. Clic derecho sobre el crédito.		
3. Selecciona la opción Movimiento.		
	4. Presenta la ventana Movimiento Prestamos del aplicativo de banca móvil.	
	5. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
Ítem 12.		
USUARIO NO TIENE MOVIMIENTOS		
A1. Si el cliente no tiene asociados movimientos del, el sistema presenta en vacío la pantalla "Movimiento Crédito".		

m) ADMINISTRAR TARJETA DE DÉBITO

En la Tabla XXXVI se presenta la pantalla de Administrar Tarjeta de Débito en la aplicación.

TABLA XXXVI: PANTALLA DE ADMINISTRAR TARJETA DE DÉBITO DEL APLICATIVO BANCA MÓVIL

Código: CU013	
Tipo de Interfaz: Android	
Caso de Uso: Administrar Tarjeta de Débito	
Icono de Tarjetas	Tarjeta Débito
Icono de Cuentas	



En la Tabla XXXVII se describe el caso de uso Administrar Tarjeta de Débito y la interacción entre el cliente y la aplicación móvil.

TABLA XXXVII: DESCRIPCIÓN DEL CASO DE USO ADMINISTRAR TARJETA DE DÉBITO

NOMBRE DEL CASO DE USO: Administrar Tarjeta de Débito		CÓDIGO: CU013
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF013	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente realizar la administración de la tarjeta de débito en el aplicativo de banca móvil y hacer uso del mismo.	El usuario selecciona el icono de tarjetas del menú principal.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente se autentique en el aplicativo banca móvil.	Se presenta la tarjeta que tiene asociada a su cédula.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona el icono de Tarjetas		
	2. Valida la tarjeta que tiene asociada al cliente.	
	3. Presenta la tarjeta en la pantalla "Tarjetas" del aplicativo de banca móvil.	
	4. Finaliza el caso de uso.	
CURSO ALTERNO DE EVENTOS		
<p>Ítem 13.</p> <p>USUARIO NO TIENE TARJETA</p> <p>A1. Si el cliente no tiene asociadas una tarjeta, el sistema presenta en vacío la pantalla "Tarjetas".</p> <p>OPCIÓN BLOQUEAR</p> <p>B1. El cliente selecciona Bloquear presionando la opción [Bloquear]: Se hace referencia al caso de uso "Bloquear Tarjeta".</p>		

n) BLOQUEAR TARJETA DE DÉBITO

En la Tabla XXXVIII se presenta la pantalla de Bloquear Tarjeta de Débito en la aplicación.

TABLA XXXVIII: PANTALLA DE BLOQUEAR TARJETA DE DÉBITO DEL APLICATIVO BANCA MÓVIL

Código: CU014	
Tipo de Interfaz: Android	
Caso de Uso: Bloquear Tarjeta de Debito	
Opción Bloquear Tarjeta	Tarjeta Debito
	

En la Tabla XXXIX se describe el caso de uso Bloquear Tarjeta de Débito y la interacción entre el cliente y la aplicación móvil.



TABLA XXXIX: DESCRIPCIÓN DEL CASO DE USO BLOQUEAR TARJETA DE DÉBITO

NOMBRE DEL CASO DE USO: Bloquear Tarjeta de Débito		CÓDIGO: CU014
REFERENCIA DE REQUISITOS	ACTORES	TIPO DE C.U
RF014	Cliente	Primario y esencial
OBJETIVO	DESCRIPCIÓN	
Permitir al cliente bloquear la tarjeta de débito en el aplicativo de banca móvil y hacer uso del mismo.	El usuario selecciona la opción de bloquear de la pantalla Tarjetas.	
PRE-CONDICIONES	POST CONDICIONES	
Que el cliente tenga una tarjeta de débito activa.	Se bloquea la tarjeta de débito.	
CURSO NORMAL DE EVENTOS		
ACCIÓN DEL USUARIO	RESPUESTA DEL SISTEMA	
1. Selecciona la opción bloquear de la pantalla "Tarjetas".		
	2. Presenta la tarjeta en la pantalla "Bloquear Tarjeta" del aplicativo de banca móvil.	
3. Ingresa tipo de bloqueo y observaciones, presiona el botón [Bloquear]		
	4. Valida la tarjeta que tiene asociada al cliente este en un estado Activa.	
	5. Presenta el mensaje "Seguro que desea bloquear su tarjeta de débito"	
6. Digita el botón [Confirmar]		
	7. Presenta el mensaje "Bloqueando tarjeta de débito por : Tipo de bloqueo"	
	8. Finaliza el caso de uso	
CURSO ALTERNO DE EVENTOS		
Ítem.		
TIPO DE BLOQUEO		
A1. El cliente puede seleccionar el tipo de bloqueo sea por robo o pérdida.		
MENSAJE AL BLOQUEAR		
B1. El cliente selecciona Bloquear presionando la opción [Bloquear]: El sistema presenta el mensaje "Seguro que desea bloquear su tarjeta de débito"		



B2. Si el cliente digita el botón [Cancelar], el caso de uso continua en el paso 2 del presente caso de uso.

ESTADO TARJETA

C1. Si la tarjeta a bloquear ya se encuentra bloqueada, el sistema presenta el mensaje "Ésta tarjeta ya está bloqueada. Acérquese a la información más cercana para mayor información".

CONFIRMACIÓN DE BLOQUEO

D1. El cliente al digitar el botón [Bloquear], el sistema presenta el mensaje "¡IMPORTANTE: ¿Seguro que desea bloquear su tarjeta de débito 604869012450001590 (presenta el número de tarjeta del socio)?".

D2. Si el cliente digita el botón [Confirmar] el caso de uso continua en el paso 7 del presente caso de uso.

CAMPOS VACÍOS

E1. Si el cliente no selecciona el tipo de bloqueo e ingresa alguna observación, el sistema presentara "No deben haber campos vacíos".

4.3. Diagrama de Clases

En esta fase del diseño, se elaboró el diagrama de clases a partir del modelo del dominio, en este diagrama se encuentran establecidos sus atributos y sus respectivas relaciones entre clases, la misma que se encuentra en la fig. 44.

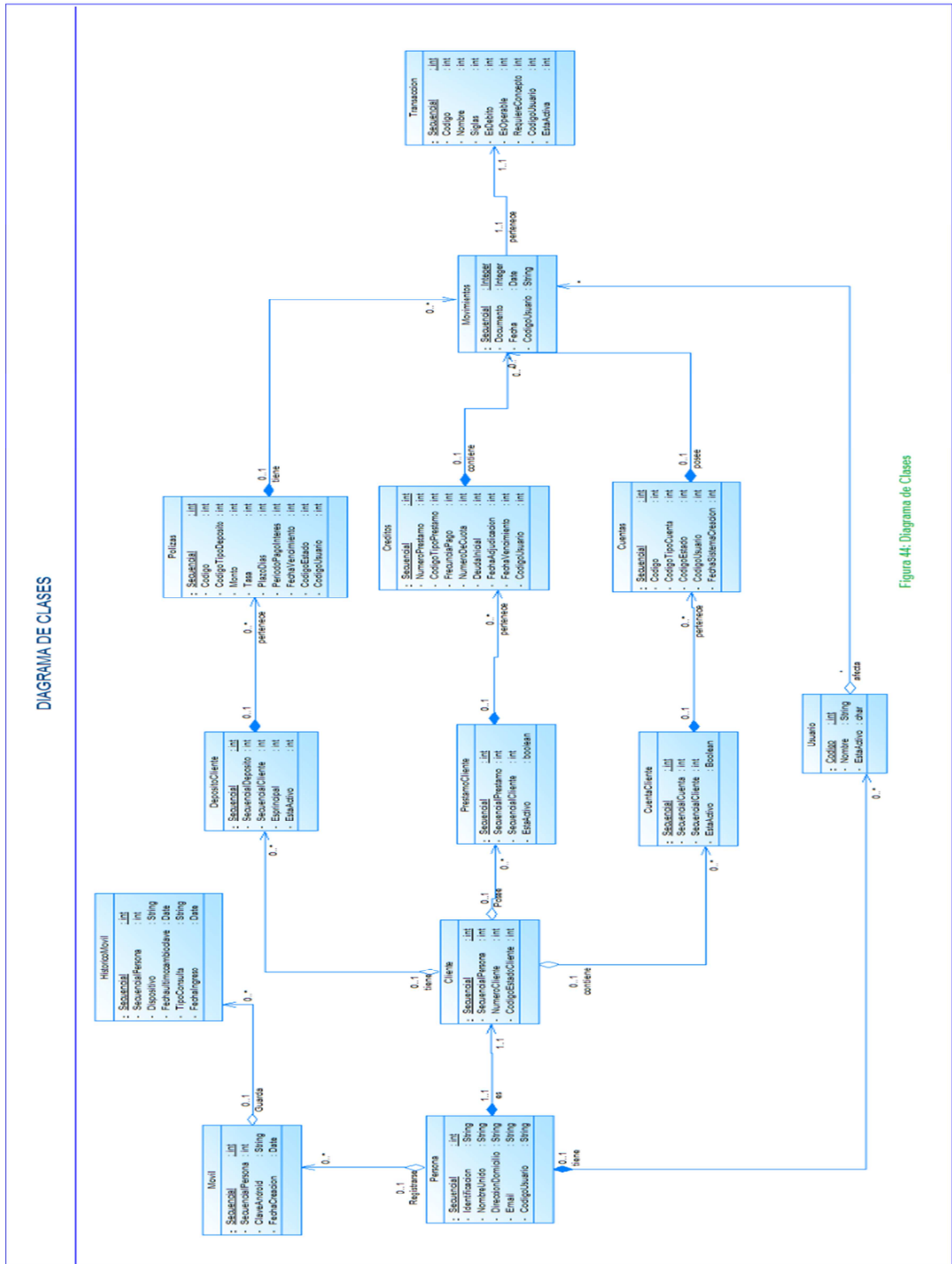


Figura 44: Diagrama de Clases

Figura 44 Diagrama de Clases



En la tabla XL se detalla las clases principales que se utiliza para el diseño del aplicativo móvil.

TABLA XL: DESCRIPCIÓN DEL DIAGRAMA DE CLASES

ENTIDADES	DESCRIPCIÓN
PERSONA	Clase que representa al cliente que va a utilizar la aplicación móvil.
CLIENTE	Clase interna para el acceso de la persona.
DEPÓSITO CLIENTE	Clase que contiene el secuencial del cliente relacionado con la póliza.
CUENTA CLIENTE	Clase que contiene el secuencial del cliente relacionado con la apertura de la cuenta.
PRÉSTAMO CLIENTE	Clase que contiene el secuencial del cliente relacionado con el crédito.
MOVIMIENTOS	Clase que representa los movimientos históricos y actuales de cuentas, pólizas y créditos.
CUENTAS	Clase que representa las cuentas del cliente.
PÓLIZAS	Clase que representa las pólizas del cliente.
CRÉDITOS	Clase que representa las pólizas e información que el cliente tiene desembolsados.
TRANSACCIÓN	Clase que representa el número de la transacción realizada.
TARJETA DE DÉBITO	Clase que representa la tarjeta de débito que el cliente tiene.
MÓVIL	Clase que representa el dispositivo móvil.
HISTORICO MÓVIL	Clase que representa el historial de la clase móvil.

4.4. Diseño de la base de datos

En la fig. 45 se presenta el diseño de la base de datos, denominada "fortunadb" que se encuentra alojada en el servidor, la misma que servirá para todo el desarrollo del aplicativo Banca Móvil.



En la tabla XLI se encuentra detallado las entidades principales que se encuentran en la base de datos "fortunadb" del diseño de la base de datos, las mismas que se han utilizado para la implementación del sistema móvil.

TABLA XLI: DESCRIPCIÓN DEL DISEÑO DE LA BASE DE DATOS

ENTIDADES	DESCRIPCIÓN
FBS_PERSONA	Representa en la BD al cliente que va a utilizar la aplicación móvil.
FBS_CAPTACIONESVISTA	Es la entidad que representa la consulta de cuentas, información y administración de tarjeta de débito.
FBS_CAPTACIONESPLAZO	Es la entidad que representa la consulta de pólizas y su información.
FBS_CARTERA	Es la entidad que representa la consulta de créditos y su información.
FBS_NEGOCIOSFINANCIEROS	Esta entidad representa los movimientos de cuentas, pólizas y créditos.
FBS_MOVIL	Esta entidad representa al dispositivo móvil.
FBS_HISTORICOMOVIL	Esta entidad representa los movimientos realizados en el dispositivo móvil. Contiene el registro para pistas de auditoria.
FBS_CLIENTE	Esta entidad representa un acceso rápido a persona.
FBS_DEPOSITOCLIENTE	Esta entidad representa los depósitos del pdf relacionados al cliente.
FBS_CUENTACLIENTE	Esta entidad representa las cuentas del cliente relacionados al cliente.
FBS_PRESTAMOCLIENTE	Esta entidad representa los préstamos del cliente relacionados al cliente.
FBS_SEGURIDADES	Esta entidad representa los usuarios que afectan los movimientos.
FBS_GENERALES	Esta entidad representa la oficina matriz de la cooperativa.

4.5. Diagrama de Paquetes

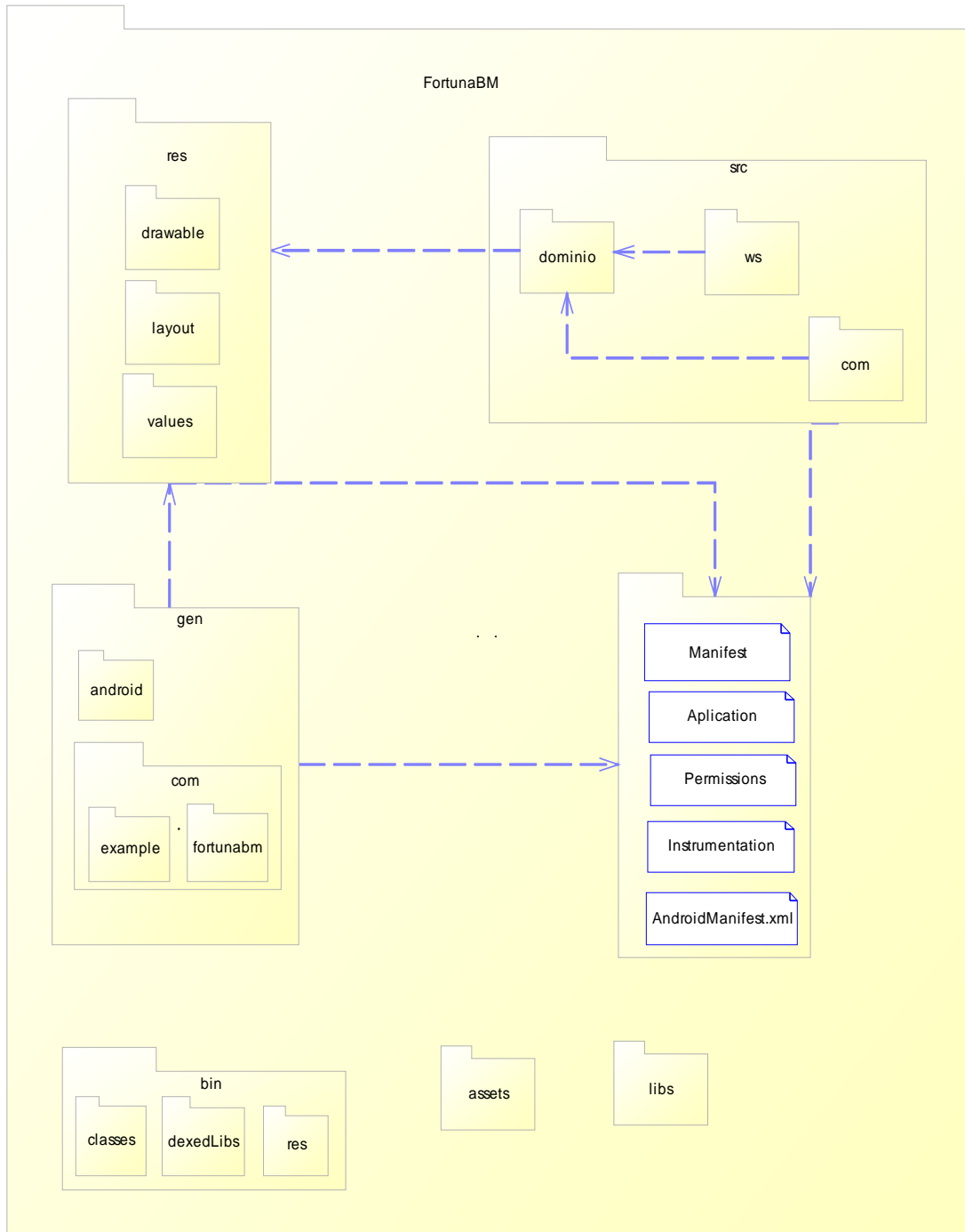


Figura 46 Diagrama de Paquetes



La fig. 46 representa lo siguiente:

La carpeta src

Esta es por defecto la carpeta donde se depositará el código fuente Java.

La carpeta res

Esa es una carpeta más compleja va a contener los recursos, como imágenes, textos, layouts, aquí como se van gestionar los recursos para dispositivos con características y configuraciones distintas (densidad de píxel, localización, ect...)

La carpeta gen

Esta es una carpeta de código fuente que va a contener archivos de Java como la carpeta src, pero no deberemos agregar o modificar los archivos contenidos ya que son generados automáticamente por el plugin de Android a partir del contenido de la carpeta res.

La carpeta bin

Esta carpeta como la carpeta gen se genera automáticamente, y la utiliza el compilador para preparar los archivos para el empaquetado final en forma de APK.

La carpeta assets

En esta carpeta se depositaran todos los archivos que acompañen a la aplicación, pueden estar organizados en carpetas y serán empaquetados en el apk final.

La carpeta libs

Está pensada para contener las librerías que se van a enlazar al proyecto, por lo general se van a depositar aquí los archivos JAR empaquetados de librerías portables.

El archivo AndroidManifest.xml

Es el archivo principal y que todas las aplicaciones o librerías deben contener en la raíz del proyecto. En este, se definen las características del proyecto como el nombre, paquete o los permisos que va a requerir la aplicación. También es en este archivo donde se describe los componentes de la aplicación.

FASE 3:

4.6. Implementación

En esta fase se pone en marcha la aplicación banca móvil, en la que se determinó la interactividad, accesibilidad y navegación en el software, comprobando la seguridad y comodidad del usuario en el momento de hacer uso de la aplicación móvil.

A continuación se detalla la arquitectura, tecnología usada y la configuración de las herramientas para su integración.

4.6.1. Arquitectura Implementada para la Aplicación Banca Móvil Fortuna

La arquitectura usada para el diseño de la aplicación banca móvil fortuna es la siguiente:

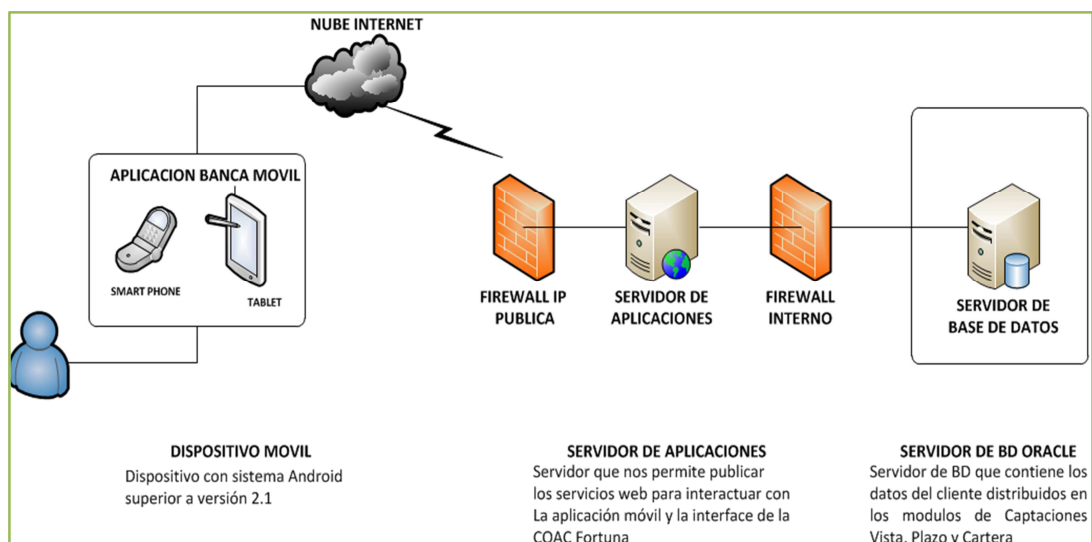


Figura 47: Arquitectura de la aplicación BM Fortuna.

En la fig. 47 se presenta la arquitectura de la aplicación "Banca Móvil Fortuna", la misma que se describe a continuación:

Ciente: Consiste el Smartphone con la aplicación nativa.

Red Pública: Consiste en la red inalámbrica o plan de datos por el cual se va a conectar los clientes con el servidor utilizando la IP pública.

Servidor de Aplicaciones: Contiene la Aplicación Android y el Web Service que permite la comunicación de la base de datos con la aplicación banca móvil.

Servidor de BD: Se encuentra la base de datos, proporciona al segundo los datos al servidor de aplicaciones para su correcto funcionamiento.

La aplicación móvil recogerá datos a petición del cliente en el aplicativo móvil con sistema operativo Android, luego se enviará al servidor de aplicaciones por medio de protocolo IP publica utilizando tecnología wifi o de datos, el cual por medio de la creación del Web Service se comunicará con el servidor de base de datos para obtener los datos requeridos por el usuario.

Para desarrollar la aplicación Banca Móvil, se utilizó Android SDK para desarrollar aplicaciones 100% nativas.

La aplicación cliente, es decir el aplicativo banca móvil se comunica con el servidor de aplicaciones para recibir/enviar datos, en el servidor de aplicaciones que contiene el web Service gestiona la lógica y se comunica con la base de datos de la COAC Fortuna.

4.6.2. Patrón Arquitectónico del Sistema Móvil

El sistema móvil desarrollado se basa en un patrón arquitectónico MVC, representado en la fig.48, los cuales se detallan a continuación.



Figura 48: Arquitectura de la aplicación BM Fortuna.



Los roles de modelo-vista-controlador son cubiertos de la siguiente manera:

MODELO: Gestiona la lógica del sistema y se comunica con la base de datos y se representa en la tabla XLII.

TABLA XLII: CÓDIGO DE MODELO DE CUENTAS

Código de Modelo de Cuentas.java	
<pre> package Dominio; public class Cuenta { String código, tipo, estado, saldo; public Cuenta(String codigo, String tipo, String estado, String saldo) { super(); this.codigo = codigo; this.tipo = tipo; this.estado = estado; this.saldo = saldo; } @Override public String toString() { return this.codigo + " " + this.tipo + " " + this.estado + " " + this.saldo; } public String getCodigo() { return codigo; } public String getEstado() { return estado; } </pre>	<pre> public String getSaldo() { return saldo; } public String getTipo() { return tipo; } public void setCodigo(String codigo) { this.codigo = codigo; } public void setEstado(String estado) { this.estado = estado; } public void setSaldo(String saldo) { this.saldo = saldo; } public void setTipo(String tipo) { this.tipo = tipo; } } </pre>

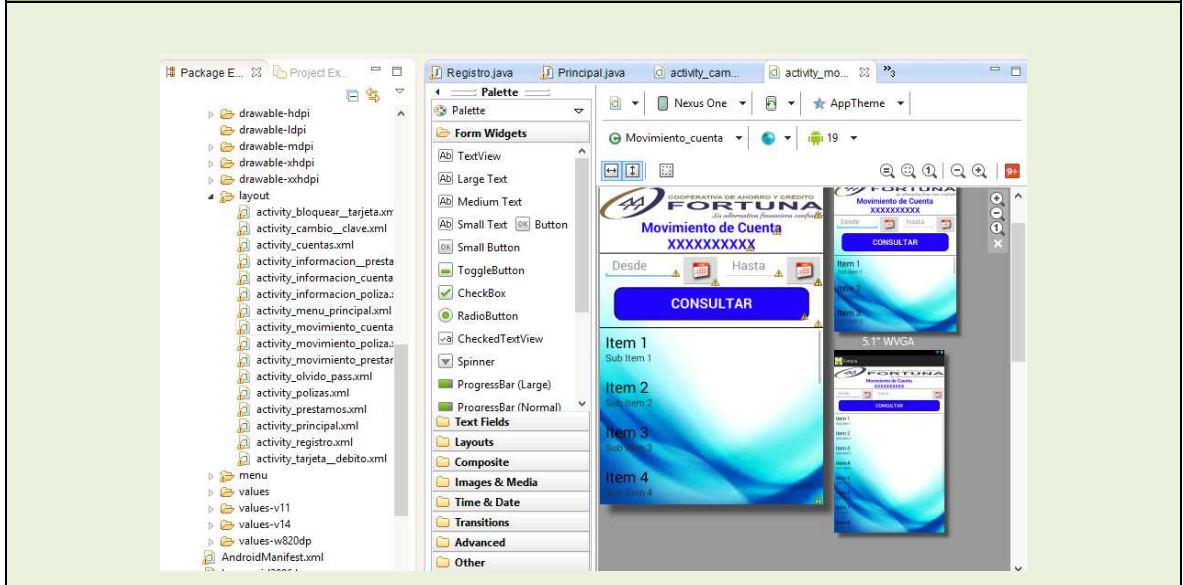
VISTA: Android, su función es presentar los datos al cliente y se representa en la tabla XLIII.

TABLA XLIII: CÓDIGO DE VISTA DE ACTIVITY_POLIZAS

Código de Vista de activity_polizas.xml	
<pre> <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:tools="http://schemas.android.com/tools" android:id="@+id/container" android:layout_width="match_parent" android:layout_height="match_parent" android:background="@drawable/images" tools:context="com.example.fortunabm.Informacion_cuenta" tools:ignore="MergeRootFrame" > <LinearLayout android:layout_width="match_parent" android:layout_height="match_parent" </pre>	<pre> <TextView android:id="@+id/clienteInformacionCuenta" android:layout_width="wrap_content" android:layout_height="match_parent" android:text="XXXXXX" android:textColor="@color/COBALTO" android:textSize="20sp" /> <TextView android:id="@+id/fechaemision" android:layout_width="wrap_content" </pre>

<pre> android:orientation="vertical" > <ImageView android:id="@+id/imageView1" android:layout_width="match_parent" android:layout_height="50dp" android:scaleType="fitXY" android:src="@drawable/logo" /> <LinearLayout android:layout_width="match_parent" android:layout_height="wrap_content" android:gravity="center_horizontal" android:orientation="horizontal" > <TextView android:id="@+id/clienteCuentas" android:layout_width="wrap_content" android:layout_height="wrap_content" android:text="XXXXXXXX" android:textColor="@color/COBALTO" android:textSize="20sp" /> </pre>	<pre> android:layout_height="match_parent" android:text="XX - XX - XXXX" android:textColor="@color/COBALTO" android:textSize="20sp" /> <TextView android:id="@+id/saldo" android:layout_width="wrap_content" android:layout_height="match_parent" android:text="0.00 USD" android:textColor="@color/ROJO" android:textSize="20sp" /> </pre>
---	---

Vista de Información de Cuenta



CONTROLADOR: Establece las comunicaciones y se representa en la tabla XLIV.

TABLA XLIV: CÓDIGO DE CONTROLADOR DE INFORMACION_CUENTA

Código de Controlador de Informacion_cuenta.java	
<pre> package com.example.fortunabm; import java.util.StringTokenizer; import org.ksoap2.serialization.PropertyInfo; import org.ksoap2.serialization.SoapObject; import ws.WebServicesFORTUNA; import android.annotation.SuppressLint; </pre>	<pre> PropertyInfo(); info.setName("numcuenta"); info.setValue(codigo); info.setType(String.class); infos[0] = info; </pre>



<pre> import android.os.Bundle; import android.os.StrictMode; import android.support.v7.app.ActionBarActivity; import android.widget.TextView; import android.widget.Toast; @SuppressLint("NewApi") public class Informacion_cuenta extends ActionBarActivity { String codigo; TextView textview, tipotxt, oficinatxt, aperturatxt, saldotxt, disponibletxt, bloqueotxt,clienteinformacionCuenta; String nombreParam,cédulaParam,emailParam; @Override protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity_informacion_cue nta); clienteinformacionCuenta= (TextView) findViewById(R.id.clienteInformacionCuenta); cédulaParam = getIntent().getStringExtra("cédulaX").toString(); nombreParam = getIntent().getStringExtra("nombreunido").toStrin g(); emailParam = getIntent().getStringExtra("email").toString(); clienteinformacionCuenta.setText(nombreParam); StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder() .permitAll().build(); StrictMode.setThreadPolicy(policy); textview = (TextView) findViewById(R.id.clienteCuentas); tipotxt = (TextView) findViewById(R.id.tipo); oficinatxt = (TextView) findViewById(R.id.oficina); aperturatxt = (TextView) findViewById(R.id.fechaemision); saldotxt = (TextView) findViewById(R.id.deudainicial); disponibletxt = (TextView) findViewById(R.id.saldoactual); bloqueotxt = (TextView) findViewById(R.id.adjudicado); codigo = getIntent().getStringExtra("codigo_cuenta").toStrin g(); PropertyInfo[] infos = new PropertyInfo[2]; PropertyInfo info = new </pre>	<pre> PropertyInfo info2 = new PropertyInfo(); info2.setName("cédula"); info2.setValue(cédulaParam); info2.setType(String.class); infos[1] = info2; SoapObject result = new WebServicesFORTUNA().ejecutarWebServices(WebServicesFORTUNA.METHOD_NAMEInformacionCu enta, infos, this); if (result != null) { int propertyCount = result.getPropertyCount(); String disponibleAhorro = "0", bloqueoAhorro = "0", oficina = "", apertura = "", numcuenta = "", tipo = ""; boolean bandera = true; for (int currentProperty = 0; currentProperty < propertyCount; currentProperty++) { Object input = result.getProperty(currentProperty); Object resultuax = (Object) (input.toString()); if (result != null) { StringTokenizer tokens = new StringTokenizer(resultuax.toString()); oficina = tokens.nextToken().replace("-", " "); apertura = tokens.nextToken(); numcuenta = tokens.nextToken(); tipo = tokens.nextToken().replace("-", " "); if (bandera) { disponibleAhorro = tokens.nextToken(); bandera = false; } else { bloqueoAhorro = tokens.nextToken(); } } double saldo = Double.parseDouble(disponibleAhorro) + Double.parseDouble(bloqueoAhorro); tipotxt.setText(tipo); oficinatxt.setText(oficina); aperturatxt.setText(apertura); saldotxt.setText(saldo + " USD"); disponibletxt.setText(disponibleAhorro + " USD"); bloqueotxt.setText(bloqueoAhorro + " USD"); textview.setText(codigo); } </pre>
--	---

- Todo parte cuando el usuario interactúa con la aplicación, el cliente actual es la VISTA. (El usuario quiere solicitar el ingreso a la aplicación banca móvil).



- El CONTROLADOR recibe la notificación de la acción solicitada. (Por medio de un handler verificará si el usuario está registrado).
- El MODELO es llamado para ser modificado, este a su vez se comunica con el WSDL para interactuar con la BD.
- De esta manera el usuario ya tiene la nueva interfaz para seguir interactuando con la aplicación y volver a iniciar el ciclo cuando solicite otra acción.

4.6.3. Detalle de la Implementación e Integración

A continuación se detalla la configuración de las herramientas que se utilizó.

a) Detalles de la Implementación

A la hora de desarrollar la aplicación se ha realizado especial hincapié en poder adaptar la aplicación a cualquier versión existente de la aplicación. Así pues, la aplicación se desarrolló con el objetivo de hacerla compatible con el mayor número de dispositivos, evitando disponer de una versión de Android actual, de forma que incluso dispositivos antiguos con versiones anteriores del sistema operativo y sin posibilidad de actualizarse puedan utilizar la aplicación. En definitiva, la aplicación funciona en cualquier dispositivo que disponga de Android 2.1 en adelante (Ver sección 3 de Validación y Verificación).

Por otro lado, uno de los objetivos es adaptar la interfaz a cualquier tamaño de pantalla o disposición. De esta forma, la aplicación muestra toda la información necesaria independientemente del tamaño de la pantalla o de la orientación del dispositivo, ya que funciona tanto en pantallas pequeñas, como grandes, o cuando el Smartphone se encuentra en posición vertical u horizontal. Este factor ha posibilitado también que la aplicación pueda utilizarse en Tablet, debiendo de adaptar mínimamente la interfaz para que se muestre la información correctamente en la pantalla de mayor tamaño que poseen las Tablet.

b) Construcción de Interfaces

Dentro del código reseñable cabe mencionar entre otras cosas la construcción de interfaces en Eclipse mediante el SDK de Android. Para analizarlo se toma de ejemplo la

Interfaz realizada para el menú cuando el dispositivo se encuentra en modo horizontal, también conocido como orientación horizontal o landscape. A continuación se muestra parte del código del menú principal en la fig. 49.

```
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="@color/BLANCO"
    android:fillViewport="true" >

    <RelativeLayout
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/images" >

        <!-- Header Starts -->

        <LinearLayout
            android:id="@+id/header"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content" >

            <!-- Logo Start -->

            <ImageView
                android:layout_width="match_parent"
                android:layout_height="120dp"
                android:scaleType="fitXY"
                android:src="@drawable/Logo" />

            <!-- Logo Ends -->
```

Figura 49: Fragmento de Código de Orientación Horizontal

Dentro de este código podemos destacar diversas características. En primer lugar Android emplea para las interfaces etiquetas para la distribución de los elementos, en este caso se utiliza una etiqueta <LinearLayout> para contener todos los elementos y a su vez se divide en otro <LinearLayout> que contiene a su vez un <RelativeLayout>. La diferencia principal entre <LinearLayout> y <RelativeLayout> es que el segundo permite alinear y situar componentes según la situación de otros componentes. Por ende, los botones se han situado según la posición del resto, como se puede observar en la fig. 50.

```
<Button
    android:id="@+id/btnLogin"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dip"
    android:background="@drawable/button"
    android:onClick="lanzar"
    android:text="Ingresar"
    android:textColor="@color/BLANCO"
    android:textSize="20sp"
    android:textStyle="bold" />
```

Figura 50: Alineación de Componentes

Todos los componentes disponen de un atributo `android_id` que sirve para identificarlos, como se puede ver en la fig. 51. Anteponiendo `@+id/` al id del botón se está indicando que el id debe añadirse al fichero `R.java`, un fichero generado de forma automática que permite acceder a los elementos de la interfaz en el código.

```
<TextView
    android:id="@+id/clienteParam"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:clickable="true"
    android:gravity="center"
    android:onClick="olvidopass"
    android:text="¿Olidaste tu contraseña?\nIngresa Aquí"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="@color/NEGRO"
    android:textSize="20sp" />
```

Figura 51: Atributo `Android_id`

La fig. 52 representa un atributo destacable es `android:layout_marginBottom` puesto que se puede observar que el valor no está en píxeles o medidas similares, sino en `dip`, que no es otra cosa que píxeles independientes de la densidad, de forma que se mantiene la proporción al emplear distintas pantallas.

```
<!-- Email Label -->
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Cédula / Pasaporte"
    android:textColor="@color/AZUL"
    android:textSize="20sp"
    android:textStyle="bold" />
<EditText
    android:id="@+id/passanterior"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20dip"
    android:layout_marginTop="5dip"
    android:hint="Ejm: 1100123456"
    android:singleLine="true" />
```

Figura 52: Atributo Android_Layout

A continuación podemos ver el elemento ImageButton, representado en la fig. 53, utilizado para crear un botón que use una imagen y no texto como un botón habitual:

```
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical" >
    <ImageButton
        android:id="@+id/Button03"
        android:layout_width="128dp"
        android:layout_height="128dp"
        android:layout_marginBottom="10dp"
        android:background="@drawable/button"
        android:drawableLeft="@drawable/abc_ic_cab_done_holo_light"
        android:onClick="LanzarPolizas"
        android:src="@drawable/polizas"
        android:text="@string/txtpolizas" />
    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="Polizas"
        android:textAppearance="?android:attr/textAppearanceMedium" />
</LinearLayout>
```

Figura 53: Atributo ImageButton

c) Desarrollo del código de la aplicación Android

Al hablar de detalles de implementación es necesario mencionar algunas particularidades de Android como las mostradas en el siguiente código:

```
listView1.setOnItemClickListener(new OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> parent, View view,
        int position, long id) {

        String item = ((TextView) view).getText().toString();
        final String cod_cuenta_num = item.substring(0,
            item.indexOf(" "));
        PopupMenu popup = new PopupMenu(Cuentas.this, view);
        // Inflating the Popup using xml file
        popup.getMenuInflater()
            .inflate(R.menu.cuentas, popup.getMenu());
        // registering popup with OnMenuItemClickListener
        popup.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {
            @Override
            public boolean onMenuItemClick(MenuItem arg0) {
                // TODO Auto-generated method stub
                progressDialog = ProgressDialog.show(Cuentas.this, "",
                    "Cargando...");

                if (arg0.getTitle().equals("Información")) { // Información
                    abrirCuenta(cod_cuenta_num,
                        Informacion_cuenta.class);
                } else { // movimiento
                    abrirCuenta(cod_cuenta_num, Movimiento_cuenta.class);
                }
                return true;
            }
        });
        popup.show(); // showing popup menu
    }
});
```

Figura 54: Evento OnItemClickListener

La fig. 54 representa el código mostrado corresponde al evento onItemClick disparado cuando se pulsa sobre una fila de una lista. El primer detalle es el tipo del parámetro parent, el cual es AdapterView<?> que corresponde a un manejador para las listas y tiene la posibilidad de ser personalizado.

```
</application>  
  
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />  
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />  
  
</manifest>
```

Figura 55: Permisos de Funcionalidades

La fig. 55 representa permisos tanto a nivel de funcionalidades, como por ejemplo permitir la conexión a Internet, como a nivel de los activities que se van a ejecutar. En caso contrario la aplicación dará error al intentar usar un fichero .class que no se ha declarado en el fichero AndroidManifest.xml representado en la fig. 56.

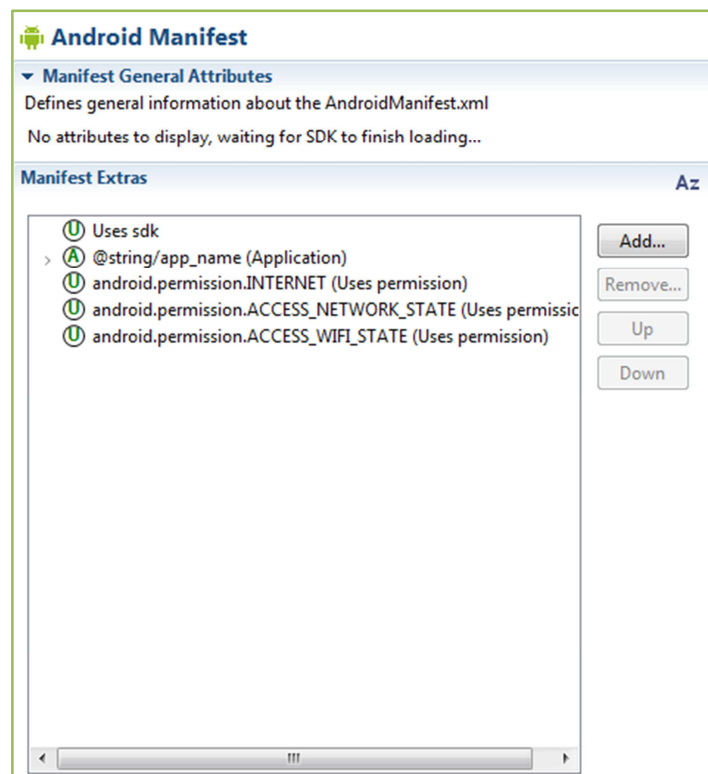


Figura 56: Fichero AndroidManifest

La fig. 57 Representa las líneas de código para recuperar el objeto entre activities.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_registro);
    cedulatxt = (EditText) findViewById(R.id.passanterior);
    String cedula = getIntent().getStringExtra("cedulaX").toString();
    cedulatxt.setText(cedula);
    StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder()
        .permitAll().build();
    StrictMode.setThreadPolicy(policy);
}
```

Figura 57: Objeto Com.Registro.

Por otro lado, mediante el siguiente código se puede ver el uso de dos interfaces distintas en una sola clase, es decir, se usan dos interfaces diferentes en la misma activity, de esta forma, se puede observar cómo es posible gestionar el uso de dos vistas sin necesitar para ello crear dos activities distintas y gestionar el paso de una a otra. Esto se realiza con el uso de dos métodos que intercambian la vista mediante el método setContentView, el cual establece el fichero XML que se usará para mostrar la interfaz, se lo representa en la fig. 58.

```
public class Registro extends ActionBarActivity {
    EditText cedulatxt;
    ProgressDialog progressDialog;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_registro);
        cedulatxt = (EditText) findViewById(R.id.passanterior);
        String cedula = getIntent().getStringExtra("cedulaX").toString();
        cedulatxt.setText(cedula);
        StrictMode.ThreadPolicy policy = new
        StrictMode.ThreadPolicy.Builder()
            .permitAll().build();
        StrictMode.setThreadPolicy(policy);
    }
    public void registrar(View v) {
        if (cedulatxt.getText().toString().equals("")) {
            Toast.makeText(Registro.this, "¡Existen campos vacíos!",
                Toast.LENGTH_LONG).show();
        } else {
            progressDialog = ProgressDialog.show(Registro.this, "",
                "Cargando...");
            registrarWS();
            progressDialog.dismiss();
        }
    }
}
```

Figura 58: Clase Registro y Método setContentView.

Por último presentar las ventanas emergentes. Esto se realiza mediante el siguiente código de la fig. 59.

```
public void bloquear(View view) {
    if (!observaciones.getText().toString().equals("")) {

        AlertDialog.Builder dialogo1 = new AlertDialog.Builder(this);
        dialogo1.setTitle("Importante");
        dialogo1.setMessage("¿ Seguro que desea bloquear su tarjeta de débito ?");
        dialogo1.setCancelable(false);

        dialogo1.setPositiveButton("Confirmar",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialogo1, int id) {
                    bloquearT();
                }
            });
        dialogo1.setNegativeButton("Cancelar",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialogo1, int id) {

                }
            });
        dialogo1.show();
    } else {
        Toast.makeText(Bloquear_Tarjeta.this,
            "No deben haber campos vacios", Toast.LENGTH_SHORT).show();
    }
}
```

Figura 59: Ventanas Emergentes.

Se puede ver como se crea un nuevo objeto, se indica el mensaje, si existirán botón para cancelar y por último, el botón para aceptar/confirmar. A continuación tan solo es necesario crear el mensaje emergente mediante el objeto creado, y mostrarlo con show.

d) Estructura del proyecto

Por último, con el fin de permitir el análisis del proyecto, su continuación o mejora, por parte de personas ajenas al desarrollo inicial se explicará la estructura que tiene, para así poder facilitar su entendimiento.

En primer lugar, se explicará cómo es la estructura Android los recursos necesarios para crear la interfaz de la aplicación. Para ello, al crear un proyecto se crea una carpeta llamada res que contiene varias carpetas dentro de ella.

Las tres carpetas principales dentro de res son drawable, layout y values como se puede ver en la fig. 60.

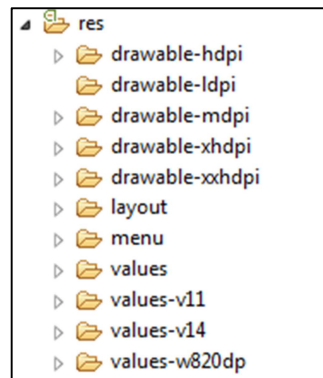


Figura 60: Estructura del Proyecto carpeta RES.

La fig., 61 representa la carpeta Drawable, la misma contiene las imágenes utilizadas por la aplicación, así como ficheros XML utilizados para crear el estilo de los botones y de las pestañas y poder mostrar una imagen cuando están pulsados. Esta carpeta contiene las imágenes del proyecto y los ficheros XML utilizados para crear los estilos de las pestañas del apartado de consulta de datos.

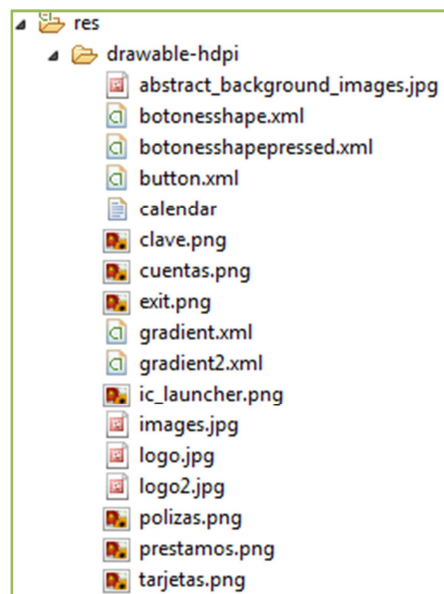


Figura 61: Estructura del Proyecto carpeta DRAWABLE.

Por su parte la carpeta Layout contiene las vistas encargadas de crear la interfaz, es decir, los ficheros XML a partir de los cuales se crea la interfaz.

Esta característica se utiliza para crear distintas interfaces según los distintos tipos de interfaz, tamaños u orientaciones. El proyecto hace uso de la carpeta layout como se puede observar en la fig. 62, para establecer la interfaz básica para dispositivos móviles con una resolución de tamaño normal, alrededor de los 480x800, correspondiente a la mayoría de móviles.

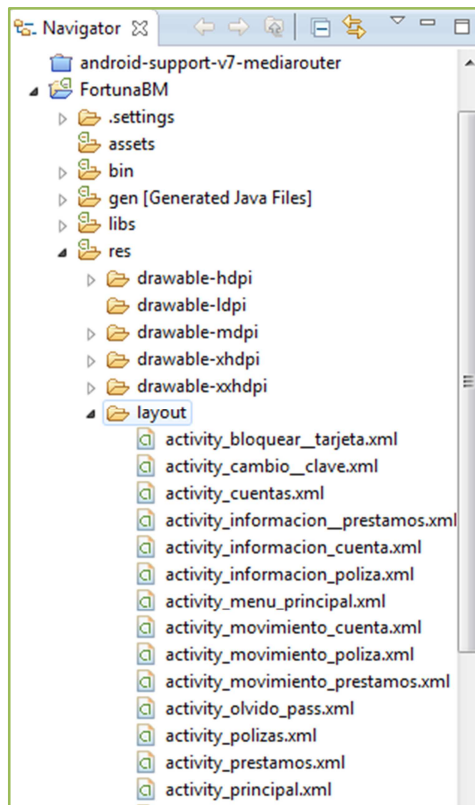


Figura 62: Estructura de la Carpeta Layout.

La última carpeta importante dentro de res es la carpeta values, representada en la fig. 63, utilizada para almacenar valores de la forma clave valor. Se utiliza principalmente para almacenar cadenas de caracteres y accederlas posteriormente haciendo uso del id. Es útil a la hora de realizar una aplicación puesto que permite modificar una cadena repetida varias veces a lo largo de la aplicación con solo cambiarla en un fichero.

Dentro de values encontramos los archivos string que se usa para almacenar cadenas.

La fig. 63 representa la estructura de la carpeta Valúes.

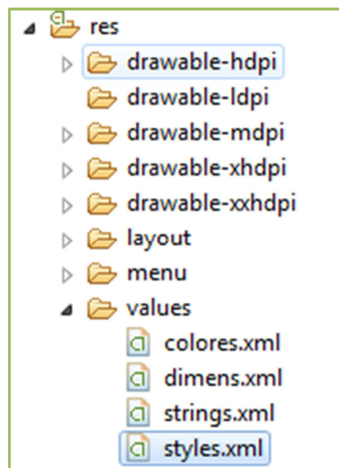








Figura 63: Estructura de Carpeta Valúes.

En cuanto al código que implementa las funcionalidades puede encontrarse en la carpeta src. Se hace una distinción entre archivos que actúan como controladores y archivos que actúan como objetos. Los controladores forman parte del patrón modelo-vista-controlador y se utilizan para comunicar las interacciones del usuario con el dispositivo con los cambios que se requieren hacer en el modelo.

Los objetos se utilizan para crear los objetos utilizados en la funcionalidad y se encuentran en la carpeta classes.

En cuanto a los controladores, se muestra a continuación una lista con el nombre y la funcionalidad de la que se encargan de forma que se pueda localizar fácilmente donde se encuentra una funcionalidad determinada para poder modificarla.

-  AutoResizeTextView.java
-  Bloquear_Tarjeta.java.
-  Cambio_Clave.java
-  Cuentas.java
-  Informacion_cuenta.java
-  Informacion_poliza.java



- Informacion_Prestmos.java
- Menú principal.java
- Movimiento_cuenta
- Movimiento póliza
- Movimientos_prestamos.java
- OlvidoPass.java
- Polizas.java
- Prestamos.java
- Principal.java
- Registro.java
- Tarjeta_Debito.java

e) Conexión a la Base de Datos

Para acceder a la información almacenada se emplea el WSDL que realiza la comunicación entre la base de datos y la aplicación. En estas funciones se realiza la petición correspondiente a la base de datos y devuelve los datos.

Para poder acceder a la BD primero se debe tener cargada la base de producción en el nuevo servidor para de ahí partir con las demás configuraciones.

La fig. 64 representa el comando para subir la base de producción al nuevo servidor de producción, fortunadb20140103_despues.dmp es el nombre de la base de producción con su fecha respectiva, se utiliza la sentencia **FULL=y** porque es la primera vez que se carga la base y es necesario cargar todos las instancias, esquemas, log, triggers, etc., el comando impdp es el **import** para cargar la base y la ruta donde se va a cargar dicha base.

```
C:\app>cd admin
C:\app\admin>cd baselocal
C:\app\admin\baselocal>cd dpdump
C:\app\admin\baselocal\dpdump>impdp system/fortunadb@baselocal file= fortunadb20
140103_despues.dmp FULL=y LOGFILE=impfull.og
```

Figura 64: Import de BD de Producción de COAC Fortuna

La fig. 65 representa la carga de la BD a nuestro nuevo Servidor de BD, cuyo nombre de la base de datos es BASELOCAL, en el que se observan todas las tablas y esquemas de la BD de producción.

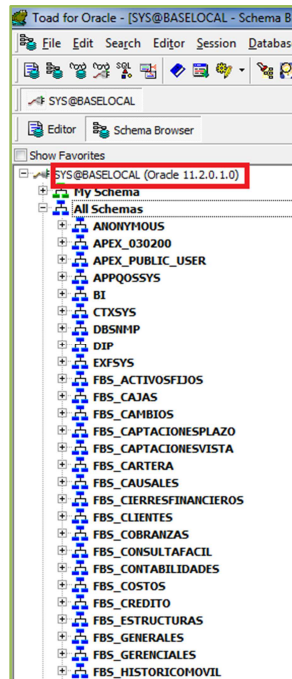


Figura 65: Subida de BD de COAC Fortuna

La fig. 66 presenta la conexión con la base de datos de la cooperativa, para esta conexión primeramente debemos tener carga la base de datos y luego realizar esta configuración.

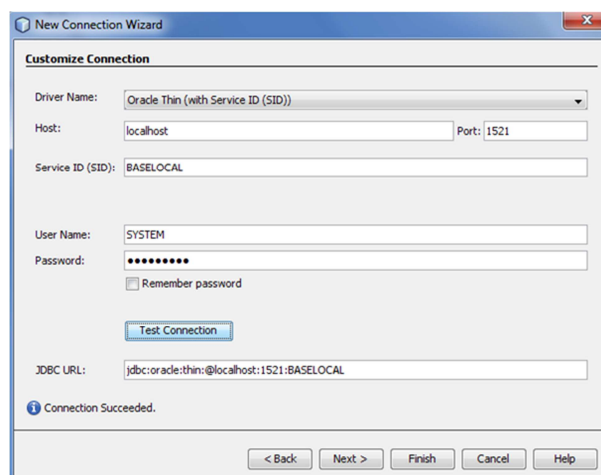


Figura 66: Conexión a la BD de la COAC Fortuna



La tabla XLV representa los métodos que se utilizan en el Web Service para la conexión con el servidor de base de datos.

TABLA XLV: MÉTODOS DE WEBSERVICE

Métodos de WebService WSDL
<pre>// *****CUENTA***** public static final String METHOD_NAMECuentas = "buscarCuentas"; public static final String METHOD_NAMEInformacionCuenta = "informacionCuenta"; public static final String METHOD_NAMEMovimientoCuenta = "movimientoCuenta"; // *****POLIZAS***** public static final String METHOD_NAMEPolizas = "buscarPolizas"; public static final String METHOD_NAMEInformacionPolizas = "informacionPoliza"; public static final String METHOD_NAMEMovimientoPoliza = "movimientoPoliza"; // *****PRESTAMOS***** public static final String METHOD_NAMEPrestamos = "buscarPrestamos"; public static final String METHOD_NAMEMovimientoPrestamos = "movimientoPrestamos"; public static final String METHOD_NAMEinformacionPrestamo = "informacionPrestamo"; // *****TARJETAS***** public static final String METHOD_NAMETarjetas = "buscarTarjetas"; public static final String METHOD_NAMEbuscarDatosTarjeta = "buscarDatosTarjeta"; public static final String METHOD_NAMEbloquearTarjeta = "bloquearTarjeta"; // *****ACCESO Y REGISTRO***** public static final String METHOD_NAMEValidarUser = "validarUser"; public static final String METHOD_NAMEEnviarClave = "enviarClave"; public static final String METHOD_NAMEOlvidoPass = "olvidoPass"; public static final String METHOD_NAMECambiarClave = "cambiarClave";</pre>

La fig. 67 representa la tabla FBS_MOVIL.APPMOVIL, aquí se guardan los registros de los móviles con su respectiva persona, clave encriptada, fecha de creación es decir la fecha de registro del dispositivo móvil y el ID del dispositivo, en esta celda se guarda el modelo y dirección Mac del dispositivo, de esta manera se controla que un único usuario pueda registrarse solo una vez en cada dispositivo, también en dicha imagen se puede corroborar que el mismo dispositivo puede registrar a varias personas. Esta tabla se relaciona con FBS_PERSONAS.PERSONA.



BANCA MÓVIL PARA LA COAC "FORTUNA" CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID

APPMOVIL: Created: 20/08/2014 13:38:43 Last DDL: 14/10/2014 22:39:46

SECUENCIAL	SECUENCIALPERSONA	CLAVEANDROID	FECHACREACION	IDDISPOSITIVO
17	240	f73f19bd7d5d42242fd04c83b9b1c3a7	07/10/2014 21:02:05	2C:44:01:BA:92:0C Linux/Android GT-S5570B
11	8	1593619c41f85d44665baeea2ef6919b	20/08/2014 17:42:54	2C:44:01:BA:92:0C Linux/Android GT-S5570B
20	102	164b6b70cfa3f5aab7c9ff5d2125100e	07/10/2014 21:06:52	2C:44:01:BA:92:0C Linux/Android GT-S5570B
10	101	202cb962ac59075b964b07152d234b70	20/08/2014 17:31:05	2C:44:01:BA:92:0C Linux/Android GT-S5570B
31	8518	ec40d16a2179b1b754718b487db99ee7	14/10/2014 22:56:29	2C:44:01:BA:92:0C Linux/Android GT-S5570B
34	8518	ec40d16a2179b1b754718b487db99ee7	14/10/2014 23:50:25	No tiene MAC - Linux/Android generic_x86
13	5245	e7f7cc6da46e656a148f68daa916e02c	18/09/2014 0:02:25	2C:44:01:BA:92:0C Linux/Android GT-S5570B
14	5244	0bbb2240276b3f84f4769afd63115a22	18/09/2014 11:22:23	2C:44:01:BA:92:0C Linux/Android GT-S5570B
9	105	5a0efb9e4e897453f1f158366989d81f	20/08/2014 17:00:52	2C:44:01:BA:92:0C Linux/Android GT-S5570B
21	241	c2baa03a175f40ae5bd778dbce06cbef	08/10/2014 17:02:52	2C:44:01:BA:92:0C Linux/Android GT-S5570B

Figura 67: Tabla de Registro de Móvil.

La fig. 68 representa la tabla FBS_HISTORICOMOVIL.HISTMOVIL, como su nombre lo indica guarda en el histórico todos los movimientos realizados por el usuario, esta tabla representa las pistas de auditoría, necesarias en cada sistema desarrollado, se guarda a la persona que ingreso con su usuario y contraseña, desde que dispositivo móvil realizó alguna transacción, fecha del último cambio de clave, tipo de consulta realizada y fecha de ingreso.

HISTMOVIL: Created: 27/08/2014 11:40:19 Last DDL: 28/08/2014 20:10:47

SECUENCIAL	SECUENCIALPERSONA	DISPOSITIVO	FECHAULTIMO CAMBIO CLAVE	TIPOCONSULTA	FECHAINGRESO
61	5245	2C:44:01:BA:92:0C Linux/Android GT-S5570B		BCue	18/09/2014 0:07:30
88	5245	No tiene MAC - Linux/Android generic_x86		BTar	18/09/2014 0:36:03
141	8518	2C:44:01:BA:92:0C Linux/Android GT-S5570B		BTar	27/09/2014 17:46:15
199	8518	No tiene MAC - Linux/Android generic_x86		BPol	27/09/2014 18:45:01
259	8518	2C:44:01:BA:92:0C Linux/Android GT-S5570B		BPol	27/09/2014 19:20:43
308	8518	04:18:0F:6C:D3:10 Linux/Android GT-S5570L		IPol	27/09/2014 19:47:05
372	240	2C:44:01:BA:92:0C Linux/Android GT-S5570B		RBMov	07/10/2014 21:02:05
405	8518	2C:44:01:BA:92:0C Linux/Android GT-S5570B	08/10/2014 17:26:41	OPas	08/10/2014 17:26:41
464	8518	2C:44:01:BA:92:0C Linux/Android GT-S5570B		BTar	15/10/2014 17:22:51
56	5245	9C:3A:AF:20:C1:31 Linux/Android espresso10rf		MPre	18/09/2014 0:05:00
89	5245	No tiene MAC - Linux/Android generic_x86		BDTar	18/09/2014 0:36:21
144	8518	2C:44:01:BA:92:0C Linux/Android GT-S5570B		BPre	27/09/2014 17:46:46
200	8518	No tiene MAC - Linux/Android generic_x86		IPol	27/09/2014 18:45:10
254	8518	2C:44:01:BA:92:0C Linux/Android GT-S5570B		BDTar	27/09/2014 19:10:19
309	8518	04:18:0F:6C:D3:10 Linux/Android GT-S5570L		BPol	27/09/2014 19:47:08
412	8518	2C:44:01:BA:92:0C Linux/Android GT-S5570B		MCue	08/10/2014 17:31:59
446	8518	No tiene MAC - Linux/Android generic_x86		BCue	15/10/2014 0:12:17

Figura 68: Tabla de Históricos.

La tabla XLVI describe los tipos de movimiento realizados por el cliente, el campo TipoConsulta es el que se almacena en la tabla HISTMOVIL.

TABLA XLVI: DESCRIPCIÓN DE TIPOS DE CONSULTA

Descripción de Tipos de Consulta	
Tipo Consulta	Descripción
BTar	Bloquear Tarjeta
IBMov	Información de Bloqueo por tipo de Movimiento
MCue	Movimientos de Cuentas
IPol	Información de Pólizas
CCla	Cambio de Clave
MPre	Movimientos Prestamos
IPre	Información de Prestamos
Opas	Olvido Password
BPre	Búsqueda de Prestamos
BDTar	Búsqueda de Tarjeta
BCue	Búsqueda de Cuentas
ICue	Información de Cuentas
BPol	Búsqueda de Pólizas
MPol	Movimientos de Póliza

4.6.4. Diagrama de Despliegue

En la fig.69 se presenta el diagrama de despliegue de la aplicación banca móvil.

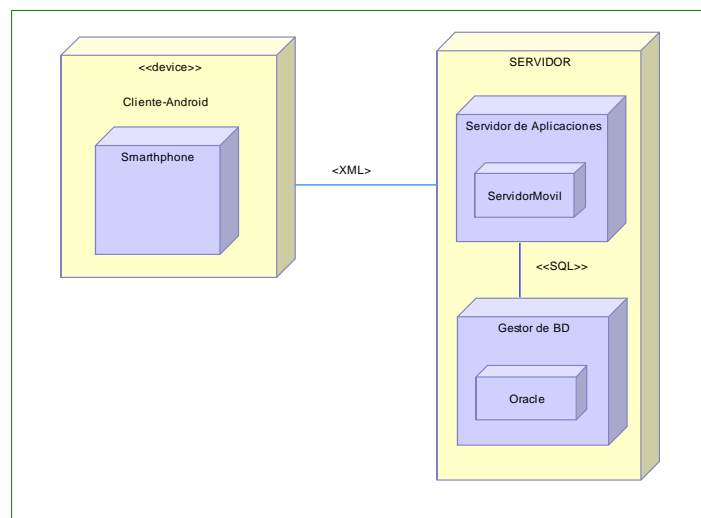


Figura 69: Diagrama de Despliegue.



FASE 4:

3 Validación y Verificación

Para comprobar el correcto funcionamiento de la aplicación se han llevado a cabo diversas evaluaciones y pruebas descritas a continuación.

3.3 Evaluación

Una vez terminada la implementación se han realizado distintas pruebas para garantizar el correcto funcionamiento, probando cada funcionalidad una a una y corrigiendo aquellas en las que se presentaba algún error, de forma que se ha conseguido depurar la aplicación lo máximo posible con el fin de evitar fallos a la hora de realizar uso del proyecto desarrollado.

3.4 Pruebas

En esta etapa se debe probar la aplicación en funcionamiento en distintos dispositivos, para ello se ha empleado el simulador incorporado en el propio SDK de Android. Mediante el simulador se pueden configurar diversos dispositivos virtuales, de forma que simulen las capacidades técnicas de un dispositivo real.

3.4.1 Pruebas Unitarias

En esta etapa se lleva a cabo la ejecución de actividades que permiten verificar que los componentes unitarios están codificados bajo condiciones de robustez, esto es:

- Las pruebas unitarias que se aplicaron fueron rutinas que validan el correcto funcionamiento de piezas del software, caso contrario es porque alguna parte del código hubo fallo de control y funcionamiento.
- Este tipo de pruebas ayuda a determinar los errores a tiempo para corregirlos sin mayor costo.
- Para realizar las pruebas unitarias se utilizó la librería JUnit, el mismo que permite probar métodos creados en el código.

Se preparó un conjunto de test para ser probados en lote, al pulsar el botón derecho en *AllTest* y seleccionar Run as JUnit Test, internamente eclipse lanzará todos los test que se tiene programado y presentara la siguiente figura.

La fig. 70 representa el Test de Pruebas Unitarias.

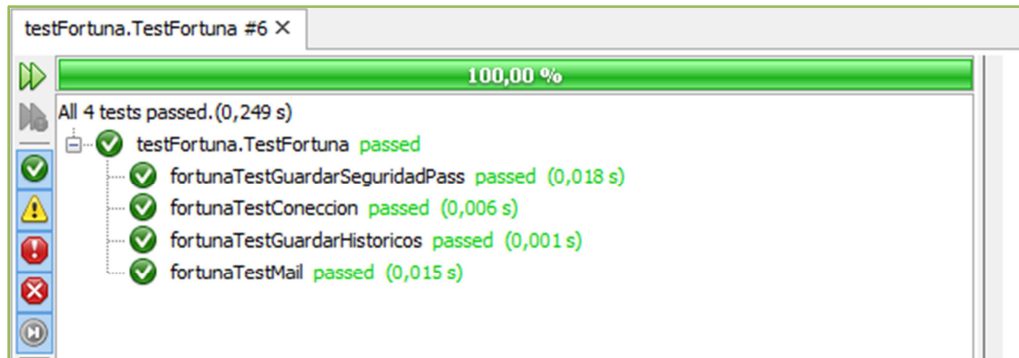






Figura 70: Test de Pruebas Unitarias.

De acuerdo a la fig. 70 todos los test salieron con resultados satisfactorios, siendo los métodos más importantes de la aplicación.

-  GuardarSeguridadPass
-  Conexión
-  Guardar Históricos
-  Test Mail

La fig. 71 representa la ejecución del proyecto test JUnit, pulsando sobre el proyecto "Run as" de las clases Tarjeta_Debito, Cuentas y ConeccionWS, siendo las clases más importantes del proyecto y funcionando de manera correcta.

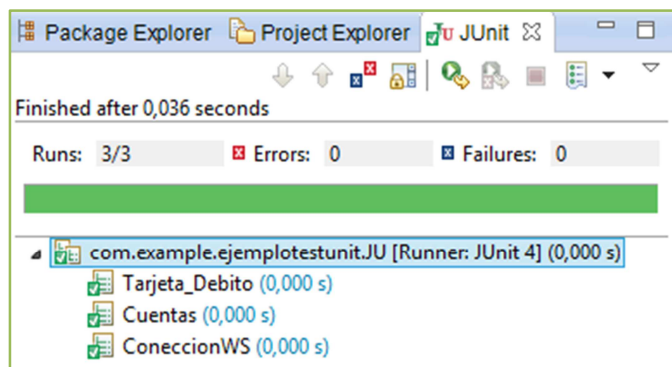


Figura 71 : Test JUnit.

3.4.2 Pruebas de Carga

Esta prueba se realizó con la finalidad de establecer si la aplicación rinde lo suficiente en caso que la carga real supere a la carga esperada.

Estas pruebas se las realizó con la herramienta Monkey, que es un programa que se ejecuta en el emulador y genera pruebas pseudo-aleatoria de los eventos de usuario, permite la verificación de una serie de eventos a nivel del sistema.

¿Cómo funciona?

Se envía una secuencia pseudo-aleatoria de los eventos de usuario en el sistema, que actúa como una prueba de esfuerzo en el software de la aplicación que está desarrollando. Opciones de configuración básicas, tales como la definición de la cantidad de eventos a intentar. Dificultades operacionales, tales como la restricción de la prueba a un solo paquete. Tipos y frecuencias de los acontecimientos.

Opciones de depuración. Cuando el monkey ejecuta, genera eventos y los envía al sistema. Se observa también el sistema sometido a prueba y busca tres condiciones, a las que trata especialmente:

Funcionalidad

Monkey permitió definir la compatibilidad de los equipos en los que la aplicación funciona de manera correcta en más de 260 dispositivos, representados en la fig. 72, 73, 74, 75, 76, 77 y 78.

Manufacturer	Android	Result	Failure Type	State	Name	Model
Alcatel	2.3	Success		Completed	One Touch	995
Fujitsu	2.3	Success		Completed	Arrows Kiss	F-03D
		Success		Completed	Lumix	F-05D
Gigabyte	2.3	Success		Completed	Gismart	G1355
HTC	2.2	Success		Completed	Droid Incredible	ADR6300
	2.3	Success		Completed	myTouch 4G Slide	PG59100
		Success		Completed	Status	PH06130
		Success		Completed	EVO 3D	PG86100
		Success		Completed	Explorer	PJ03100
		Success		Completed	Raider 4G	X710a
		Success		Completed	Sensation XL	X315E

Modelo de dispositivos móviles

Versiones de Android

Figura 72: Compatibilidad de Equipos 1.





		Success	Completed		Incredible 4G LTE	ADR6410
		Success	Completed		One V	T-320e
Huawei	2.3	Success	Completed		Yulan	U8850
		Success	Completed		Ideos	Y200
		Success	Completed		Activa 4G	M920
Kyocera	4.0	Success	Completed		Urbano Progreso	KYY04SSA
Lava	2.3	Success	Completed		Xolo	X900
Lenovo	4.0	Success	Completed		ideaphone	S686
		Success	Completed		ideaphone	A580
		Success	Completed		ideaphone	S680
LG	2.3	Success	Completed		Optimus Slider	LG-VM701
		Success	Completed		myTouch Q	C800

Figura 73 : Compatibilidad de Equipos 2.













		Success	Completed		Nitro HD	P930
		Success	Completed		DoublePlay	C729
		Success	Completed		Enlighten	LG-V5700
		Success	Completed		Viper 4G LTE	LS-840
		Success	Completed		Optimus	LGM5695
		Success	Completed		Optimus Sol	E730
Motorola	2.3	Success	Completed		Fire XT	XT530
		Success	Completed		Admiral	XT603
		Success	Completed		Fire	XT311
		Success	Completed		Moto	XT615
		Success	Completed		Droid Pro	XT610
	4.0	Success	Completed		Atrix HD	MOCD7

Figura 74: Compatibilidad de Equipos 3.

		Success	Completed		Motoluxe	XT685
		Success	Completed		Droid Razr M	XT907
		Success	Completed		Photon Q 4G LTE	XT897
Orange	2.3	Success	Completed		San Diego	AZ210
Panasonic	2.3	Success	Completed		P-07C	P-07C
Pantech	2.3	Success	Completed		Burst	P9070
		Success	Completed		Sky Vega 4G LTE Original	IM-A8105
Samsung	2.2	Success	Completed		Galaxy Prevail	SPH-M820
	2.3	Success	Completed		Galaxy R	GT-I9103
		Success	Completed		Galaxy S II	GT-I9100
		Success	Completed		Exhibit 4G	SGH-T759
		Success	Completed		Stratosphere	SCH-I405

Figura 75: Compatibilidad de Equipos 4.













	4.0	Success	Completed		Galaxy S II	SGH-T989
		Success	Completed		Admire	SCH-R720
		Success	Completed		Galaxy S II	SGH-I727
		Success	Completed		Galaxy 5.0 MP3 Player	YP-G70CWY/XAA
		Success	Completed		Galaxy Pocket	GT-S5300
		Success	Completed		Galaxy Exhilarate	SGH-I577
		Success	Completed		Galaxy Nexus	GT-19250
		Success	Completed		Galaxy Rush	SPH-M830
		Success	Completed		Galaxy Stellar	SCH-I200
		Sharp	2.3	Success	Completed	
Success	Completed				Chocolate	SH-04D
Success	Completed				Aquos	SH-13C

Figura 76: Compatibilidad de Equipos 5.











Sony	4.0	Success	Completed		Xperia Tipo	ST21a
		Success	Completed		Xperia GX	SO-04D
		Success	Completed		Xperia SL	LT26ii
Sony Ericsson	2.3	Success	Completed		XPERIA Arc	LT15I
		Success	Completed		XPERIA Neo	MT15I
		Success	Completed		XPERIA Pro	MK16I
		Success	Completed		XPERIA Arc S	LT18I
		Success	Completed		XPERIA Mini Pro	SK17I
		Success	Completed		XPERIA Active	ST17I
		Success	Completed		Xperia S	

Figura 77: Compatibilidad de Equipos 6.







Sprint	2.3	Success	Completed		Express	
ZTE	2.3	Success	Completed		Skate	T3
		Success	Completed		Warp	N860
		Success	Completed			V788D
		Success	Completed		Fury	N850
		Success	Completed		Grand X	U970

Figura 78: Compatibilidad de Equipos 7.

La fig. 79 permite ver el tiempo de ejecución de la aplicación (tiempo en instalar, ejecutar y desinstalar) en el dispositivo más lento. Dispondremos de información como el log de la aplicación, para saber si falló y poder observar en qué.

```
▼ Monkey Log

:Monkey: seed=0 count=1500
:AllowPackage: com.example.fortunabm
:IncludeCategory: android.intent.category.LAUNCHER
:IncludeCategory: android.intent.category.MONKEY
// Event percentages:
// 0: 15.0%
// 1: 10.0%
// 2: 2.0%
// 3: 15.0%
// 4: -0.0%
// 5: 25.0%
// 6: 15.0%
// 7: 2.0%
// 8: 2.0%
// 9: 1.0%
// 10: 13.0%
```

Figura 79: Log de Pruebas de Carga.

3.4.3 Pruebas de Stress

Para realizar las pruebas de Stress se utilizó la herramienta Monkey, el propósito es realizar una prueba de estrés de la aplicación Banca Móvil Fortuna para confirmar que, haga lo que haga el usuario con las pantallas, la aplicación no tendrá un comportamiento inesperado.

El input que realiza Monkey no tiene por qué tener sentido alguno. A continuación solicitamos 1000 eventos simulados cada 100 milisegundos obteniendo la lista de ellos y afectará a las aplicaciones del paquete com.example.fortunabm.

Monkey simulará eventos de teclado, tanto qwerty como teclas hardware especializadas, movimientos de trackball, apertura y cierre del teclado, rotaciones de la pantalla.



- La tabla XLVLL representa el inicio del proceso de testeo, librerías, llama a los métodos de entrada de Android, comienza con el ActivityManager es el que permite crear, borrar y administrar las actividades que son las pantallas en el dispositivo móvil y que el usuario ve una pantalla a la vez, realizar esta acción de ingresar a la aplicación toma un total de 120 milisegundos.

TABLA XLVII: CÓDIGO DE PRUEBA DE STRESS ACTIVITYMANAGER

```

Código de Prueba de Stress: ActivityManager
----- beginning of /dev/log/main
D/AndroidRuntime(23197):
D/AndroidRuntime(23197): >>>>>> AndroidRuntime START
com.android.internal.os.RuntimeInit <<<<<<<
D/AndroidRuntime(23197): CheckJNI is OFF
D/dalvikvm(23197): Trying to load lib libjavacore.so 0x0
D/dalvikvm(23197): Added shared lib libjavacore.so 0x0
D/dalvikvm(23197): Trying to load lib libnativehelper.so 0x0
D/dalvikvm(23197): Added shared lib libnativehelper.so 0x0
D/AndroidRuntime(23197): Calling main entry com.android.commands.pm.Pm

----- beginning of /dev/log/system
W/ActivityManager(20981): No content provider found for permission revoke:
file:///data/local/tmp/9R5sqY_10282
W/ActivityManager(20981): No content provider found for permission revoke:
file:///data/local/tmp/9R5sqY_10282
D/dalvikvm(20981): GC_CONCURRENT freed 1759K, 41% free 24598K/41543K, paused
3ms+15ms, total 120ms
D/dalvikvm(20981): WAIT_FOR_CONCURRENT_GC blocked 112ms
D/dalvikvm(20981): GC_CONCURRENT freed 2068K, 41% free 24572K/41543K, paused
14ms+13ms, total 120ms

```

- La tabla XLVIII representa las pruebas de instalación del apk fortunabm.apk, cambia los servicios, reconfigura el ingreso sobre los paquetes com.example.fortunabm con un total de 223milisegundos, al realizar una salida toma un tiempo de 3 ms.

TABLA XLVIII: CÓDIGO DE PRUEBA DE STRESS FORTUNABM.APK

```

Código de Prueba de Stress: fortunabm.apk
D/PackageManager(20981):           New           package           installed           in
/data/app/com.example.fortunabm-1.apk
I/RegisteredComponentCache(21151): ComponentInfo:           ResolveInfo{41924ba0

```



```

com.android.apps.tag.TagViewer p=0 o=0 m=0x108000}, techs: android.nfc.tech.Ndef,
D/dalvikvm(20981): WAIT_FOR_CONCURRENT_GC blocked 0ms
D/PackageManager(20981):
generateServicesMap(android.accounts.AccountAuthenticator): 3 services unchanged
I/InputReader(20981): Reconfiguring input devices. changes=0x00000010
I/Launcher(15515): setLoadOnResume
D/PackageManager(20981): generateServicesMap(android.content.SyncAdapter): 20
services unchanged
D/BackupManagerService(20981): Received broadcast Intent {
act=android.intent.action.PACKAGE_ADDED dat=package:com.example.fortunabm
flg=0x8000010 (has extras) }
D/dalvikvm(20981): GC_EXPLICIT freed 1991K, 41% free 24629K/41543K, paused
15ms+18ms, total 223ms
D/dalvikvm(20981): WAIT_FOR_CONCURRENT_GC blocked 98ms
D/dalvikvm(20981): WAIT_FOR_CONCURRENT_GC blocked 99ms
D/dalvikvm(20981): WAIT_FOR_CONCURRENT_GC blocked 106ms
D/AndroidRuntime(23197): Shutting down VM
D/dalvikvm(23197): GC_CONCURRENT freed 97K, 89% free 453K/4096K, paused
0ms+0ms, total 3ms
    
```

- La tabla XLIV representa las pruebas sobre los paquetes com.example.fortunabm para la clase principal con una demora de 13 milisegundos.

TABLA XLIX: CÓDIGO DE PRUEBA DE STRESS CLASE PRINCIPAL

Código de Prueba de Stress: Clase Principal
<pre> I/ActivityManager(20981): Start proc com.example.fortunabm for activity com.example.fortunabm/.Principal: pid=23231 uid=10062 gids={3003, 1028} D/dalvikvm(23231): GC_FOR_ALLOC freed 106K, 2% free 10880K/11079K, paused 12ms, total 13ms </pre>

- La tabla L representa las pruebas sobre la clase OlvidoPass actuando directamente sobre el ActivityManager con un total de 439 milisegundos.

TABLA L: CÓDIGO DE PRUEBA DE STRESS CLASE PRINCIPAL

Código de Prueba de Stress: Clase Principal
<pre> I/ActivityManager(20981): START {cmp=com.example.fortunabm/.OlvidoPass u=0} from pid 23231 I/ActivityManager(20981): Displayed com.example.fortunabm/.OlvidoPass: +439ms </pre>



3.4.4 Pruebas de Validación

a) Plan de la Prueba de Validación

Las pruebas de validación de una aplicación permiten:

- Verificar la existencia de posibles errores en el aplicativo móvil.
- Encontrar posibles soluciones para los problemas encontrados.

Para la realización de las pruebas de validación, no es necesario crear registros con anterioridad, el sistema toma los datos que ya se encuentran en la base de datos de la COAC Fortuna, se trabajara con una base de pruebas, la misma que será proporcionada por el Departamento de Sistemas de la cooperativa, en el siguiente cuadro se especifica los elementos que se utilizaron en la tabla LI.

TABLA LI: PRUEBA DE VALIDACIÓN RECURSOS

RECURSOS	
Humano	Rosa Romero, Servicios Cooperativos de COAC Fortuna.
	Paul Bravo, Jefe de Sistemas COAC Fortuna.
	Natali Cruz G, tesista y observadora.
Físicos	Oficina de Servicios Cooperativos en la COAC Fortuna.
	2 computadores con conexión a internet
	2 Smartphone y una Tablet con conexión a internet
	Router Huawei Claro con acceso a wifi
Plazo	Horario de Oficina 8:30 a 1:00 y de 2:00 a 5:30

- **Estructura de la Prueba:**

En la tabla LII se encuentra detallado el plan de pruebas de validación de los procesos de la aplicación móvil.

TABLA LII: ESTRUCTURA DEL PLAN DE PRUEBAS DE VALIDACIÓN

Procesos	Fecha	Participantes
<ul style="list-style-type: none"> ✓ Ingreso a la aplicación móvil. ✓ Registro en la aplicación móvil. ✓ Verificar envío de clave a correo. ✓ Autenticación en la aplicación móvil. ✓ Olvidar Clave. ✓ Ingresar cédula incorrecta. ✓ Ingresar cédula de cliente que no es socio de la COAC Fortuna. ✓ Ingreso al menú principal. 	23/10/2014	Cliente



<ul style="list-style-type: none"> ✓ Editar Clave. ✓ Verificar envío de clave a correo por actualización. ✓ Consulta de Cuentas. ✓ Solicitar información de una cuenta específica. ✓ Solicitar movimientos por un rango de fechas de una cuenta específica. ✓ No seleccionar fecha de rango de la cuenta. ✓ Consulta de pólizas. ✓ Solicitar información de una póliza específica. ✓ Solicitar movimientos de una póliza específica. ✓ Consulta de créditos. ✓ Solicitar información de un crédito. ✓ Solicitar movimientos de un crédito. 		
--	--	--

• **Escenarios para las Pruebas:**

En la tabla LIII se encuentra detallado los escenarios para la realización de las pruebas de validación.

TABLA LIII: ESCENARIOS PARA LAS PRUEBAS DE VALIDACIÓN

Escenario	Cumple		Incidencias
	SI	NO	
Preparación de equipos	x		Se utilizó una portátil que funciona como servidor de aplicaciones, otra portátil que funciona como servidor de base de datos,
Instalación del apk en los Smartphone.	x		Se instaló el aplicativo "Banca Móvil Fortuna" en cada uno de los teléfonos móviles.
Configuración de la red en los equipos a usar.	x		Se configuro una red para que estén en la misma red los dos servidores y los teléfonos móviles.
Comprobación del Servidor de Aplicaciones.	x		Comprobar que el servidor Glassfish este con el servicio levantado, el wsfortuna este corriendo.
Subir al servidor de base de datos una base actual.	x		Se utilizar una base de pruebas lo más actual posible, se procede a realizar la subida de un respaldo de la base de datos.
Arrancar los servicios para la aplicación móvil.	x		Verificar se encuentre levantado el servicio de Oracle y listener, para poder acceder a la base de datos.



Ejecución de la aplicación banca móvil Fortuna.	x		Comprobar el icono de acceso a la aplicación móvil se encuentre en un lugar visible.
---	---	--	--

- Diseño de la Prueba:

En la tabla LIV se detalla el diseño de las pruebas de validación que se realizaron a la aplicación.

TABLA LIV: DISEÑO DE LAS PRUEBAS DE VALIDACIÓN

Fecha:	08 de octubre de 2014
Alcance	Aplicación "Banca Móvil Fortuna"
Ítems a probar.	Autenticación. Registro. Cambiar contraseña. Consulta de los servicios que brinda la aplicación banca móvil.
Estrategia.	Manejo de las aplicaciones por parte de los usuarios. Obtener información a partir de las encuestas realizadas.
Recursos.	Computadores Smartphone Conexión a internet Aplicación "Banca Móvil Fortuna" Personal de Servicios Cooperativos.
Responsable.	Natali Cruz G.

b) Análisis de las Pruebas de Validación

A continuación se detallan las respuestas obtenidas a los encuestados de la COAC Fortuna.

Test de la Aplicación Móvil.

1. ACCESIBILIDAD

¿Tuvo inconvenientes al momento de registrar su usuario en la aplicación móvil?

SI () NO (X)

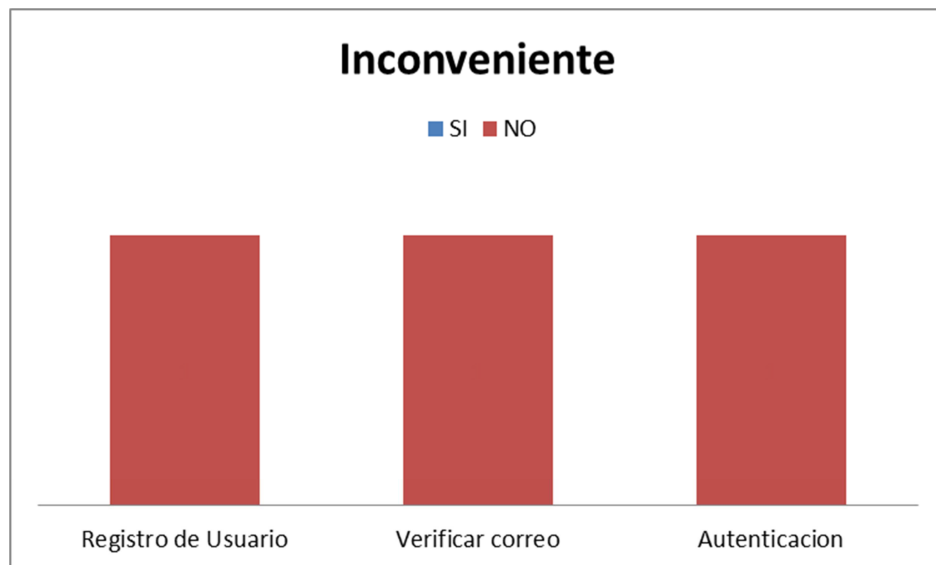
¿Tuvo inconvenientes al revisar en su correo electrónico la clave de seguridad que genera la aplicación?

SI () NO (X)

¿Tuvo inconvenientes al momento de ingresar el usuario y contraseña en la aplicación móvil?

SI ()

NO (X)



Análisis

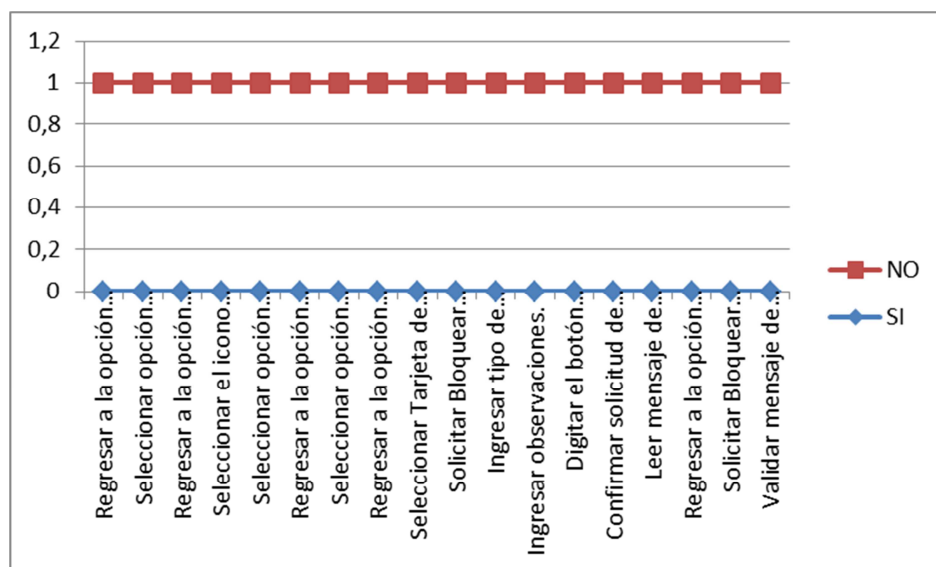
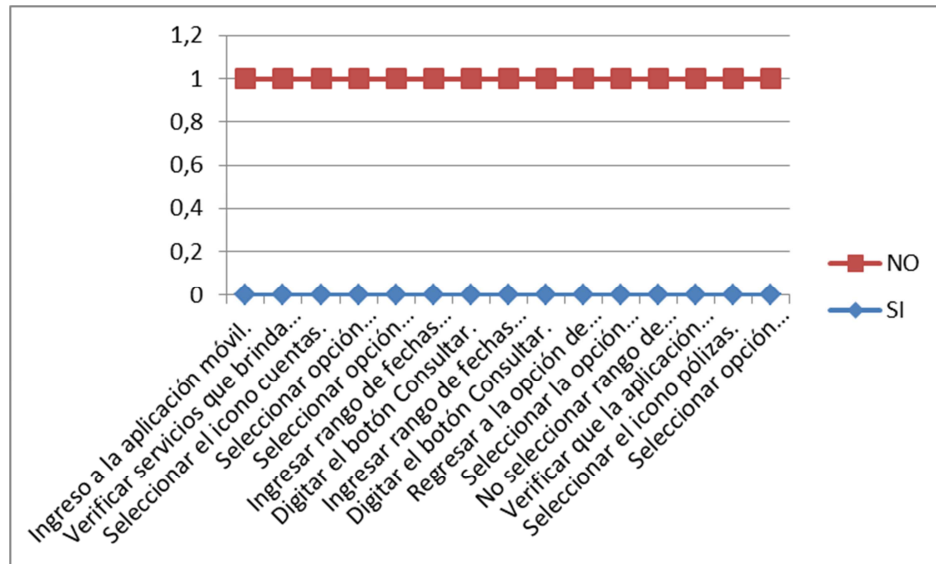
Se puede observar que la Eco. Rosa Romero Rojas no tuvo ningún tipo de problema al autenticarse, dejando claro que el software de la aplicación móvil es de fácil registro y autenticación, la verificación en el correo de la clave enviada por la aplicación móvil es un proceso externo y que sugiere que en el correo enviado por la aplicación móvil lleve el nombre del cliente, ya que hay diferentes clientes que utilizan la misma cuenta de correo entre uno o dos personas.



2. FUNCIONALIDAD

En el siguiente cuadro marque SI o NO según corresponda:

OPERACIÓN	DIFICULTAD	
	SI	NO
Ingreso a la aplicación móvil.		X
Verificar servicios que brinda la aplicación móvil.		X
Seleccionar el icono cuentas.		X
Seleccionar opción Información.		X
Seleccionar opción Movimiento.		X
Ingresar rango de fechas anteriores a la fecha actual.		X
Digitar el botón Consultar.		X
Ingresar rango de fechas posteriores a la fecha actual.		X
Digitar el botón Consultar.		X
Regresar a la opción de movimiento.		X
Seleccionar la opción Movimiento.		X
No seleccionar rango de fechas.		X
Verificar que la aplicación presente mensaje de campos vacíos.		X
Seleccionar el icono pólizas.		X
Seleccionar opción Información.		X
Regresar a la opción de información.		X
Seleccionar opción Movimiento.		X
Regresar a la opción de movimiento.		X
Seleccionar el icono créditos.		X
Seleccionar opción Información.		X
Regresar a la opción de información.		X
Seleccionar opción Movimiento.		X
Regresar a la opción de movimiento.		X
Seleccionar Tarjeta de Débito.		X
Solicitar Bloquear tarjeta.		X
Ingresar tipo de bloqueo.		X
Ingresar observaciones.		X
Digitar el botón Bloquear.		X
Confirmar solicitud de bloqueo.		X
Leer mensaje de bloqueo por el tipo de bloqueo ingresado que emite la aplicación.		X
Regresar a la opción de Tarjeta de Débito.		X
Solicitar Bloquear tarjeta.		X
Validar mensaje de bloque ya existente.		X



Análisis

La responsable del área de servicios cooperativos menciona que no tuvo ningún problema con los procesos que se evaluó, sin embargo supo manifestar que al momento de realizar el bloqueo de la tarjeta de débito se presenten dos solicitudes de confirmación de bloqueo al ingresar a la pantalla y luego al ya realizar el bloqueo, esto se solucionó en el web Service.

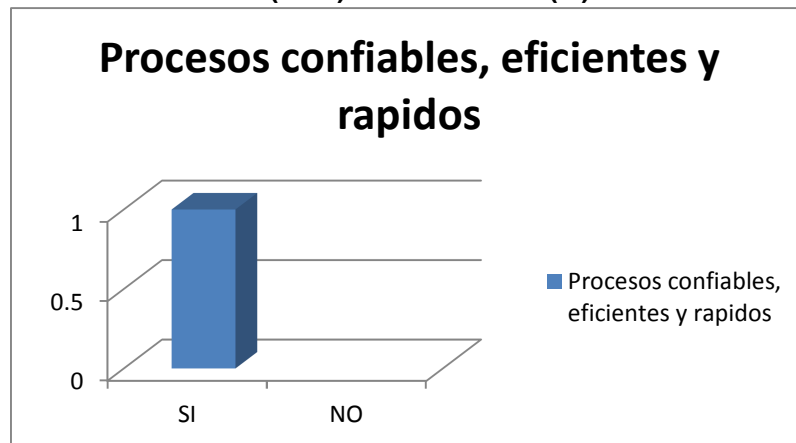
Considera Ud. Que la aplicación "Banca Móvil Fortuna" abarca las principales necesidades de los usuarios de la cooperativa.

SI (X) NO ()



Considera Ud. Que la aplicación "Banca Móvil Fortuna" realiza los procesos de manera confiable, eficiente y rápida.

SI (X) NO ()



Análisis

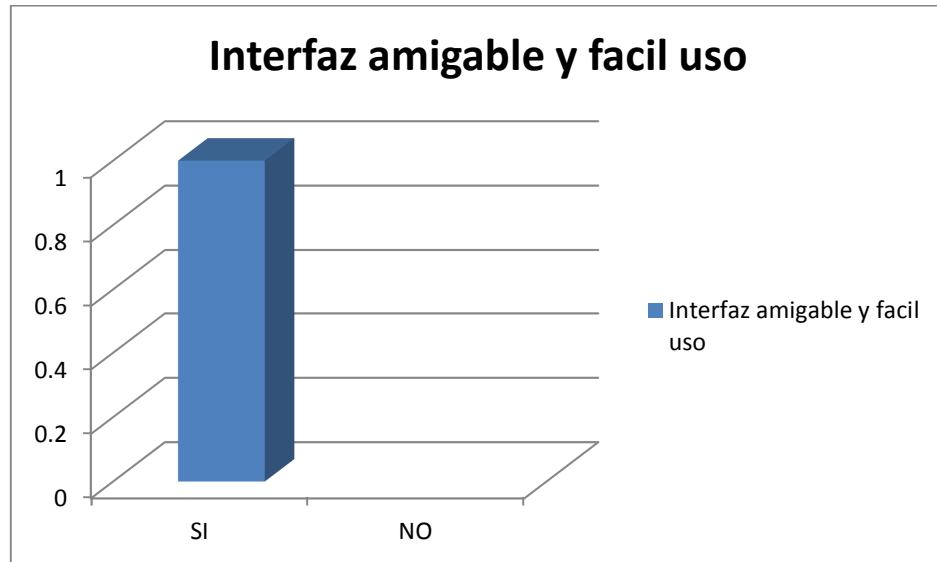
La responsable del área de servicios cooperativos menciona que el software le permite realizar los procesos de consulta y bloqueo de manera eficiente, rápida, segura, presenta la información acertada, esto se lo puede lograr gracias a la autenticación del cliente ya que únicamente se presenta la información que tiene asociada al cliente.

3. DISEÑO Y PRESENTACIÓN

¿Considera que la interfaz de la aplicación banca móvil es amigable y de fácil uso?

SI (X)

NO ()



Análisis

La responsable del área de servicios cooperativos menciona que la aplicación es de fácil uso y no tuvo ningún inconveniente al utilizarlo.



g. Discusión.

1. Desarrollo de la Propuesta Alternativa

El trabajo de titulación denominado "DISEÑO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO FORTUNA DE LA CIUDAD DE LOJA, CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID", dio como resultado la construcción de la aplicación "Banca Móvil Fortuna", desarrollada bajo el lenguaje de Eclipse con Android SDK, Oracle como gestor de base de datos y Glassfish como servidor de aplicaciones.

El objetivo general y los objetivos específicos fueron cumplidos en su totalidad, gracias al correcto manejo de métodos, técnicas y aplicación de las metodologías SWEBOK y XP, las mismas que fueron orientadas al desarrollo del proyecto, para la justificación que se detalla a continuación:

Objetivo Especifico 1: Analizar y determinar los requerimientos del sistema banca móvil para desarrollar los módulos lógicos y físicos de acuerdo a las necesidades que ha establecido el usuario.

Para cumplir con el objetivo se aplicó la ingeniería de requisitos métodos y técnicas de recolección de información, entre ellas:

Observación: Mediante esta técnica se logró examinar con atención el comportamiento más usual de los clientes, tipo de transacciones que se realizan con mayor regularidad en la COAC Fortuna. Cabe destacar que al ser parte de los procesos que se desarrollan en la cooperativa, se tiene una idea más clara de las principales necesidades del cliente, el estar relacionada al 100% con la herramienta tecnológica que utilizan para transaccionar diaria, permitió una fluidez de conocimientos entre el área involucrada con atención al cliente y mi persona.

Encuesta: Esta técnica se aplicó a un grupo específico de empleados de la COAC Fortuna, dando como resultado información cuantitativa para determinar los procesos actuales de atención al cliente y los canales electrónicos que tienen a disponibilidad de los clientes.



Entrevista: A través de esta técnica se recopiló información cualitativa, describiendo realmente las principales necesidades de los clientes, priorizando los procesos más urgentes desde el punto de vista de quienes están día a día con la atención al público.

Las técnicas anteriormente detalladas fueron seleccionadas porque son intencionadas, selectivas ilustradas e interpretadas dando como resultado información factible para establecer las necesidades o prioridades de los procesos que la COAC fortuna ofrece a sus clientes.

Además se realizó consultas en blog, libros y páginas electrónicas, como también se asistió de manera personal a las oficinas de servicios cooperativos principal oficina de atención al público, de la misma manera se dio importancia a los clientes que son el centro de economía de la COAC Fortuna como el área de crédito y por último el área de inversiones, todo esto se ha realizado con el afán de tener una visión general de la operatividad.

Siendo el análisis de requisitos uno de los puntos prioritarios en el ciclo de vida de desarrollo de software, para la identificación de las necesidades del usuario se documentó dichos requisitos bajo el formato IEEE Std 830-1998, para establecer las condiciones mediante el cual va a funcionar el sistema, evitando el malentendido de determinadas situaciones con el personal involucrado (Ver Sección Resultados apartado 3.3 Especificación de Requisitos).

Objetivo 2: Determinar el diseño de la arquitectura del software basándose en los requerimientos y en las tendencias actuales de sistemas de banca móvil de manera que se pueda obtener la interfaz gráfica y establecer el funcionamiento del sistema.

Para cumplirse este objetivo se procedió a la descripción de los casos de uso para establecer las actividades que deben seguir las personas involucradas con el sistema propuesto (Ver sección Resultados apartado 4.2 Descripción de los Casos de Uso), la creación del diagrama de clases (Ver sección de Resultados apartado 4.3 Diagrama de Clases), diseño de la base de datos con la herramienta de Oracle Toad Data Modeler, se utilizó dicha herramienta ya que la COAC Fortuna tiene su base de producción en Oracle, se crearon dos tablas para la aplicación móvil con el mismo esquema de la base actual



FBS_MOVIL, FBS_HISTORICOMOVIL y para la consulta se utilizan las principales tablas que suman un total de 33 tablas que servirán para la gestión de la información del proyecto de banca móvil (Ver sección de Resultados, apartado 2.4 Diseño de la Base de Datos).

Objetivo 3: Desarrollar e implementar la aplicación banca móvil haciendo uso de herramientas web para tener accesibilidad desde cualquier dispositivo Android.

Para el código fuente se utilizó el ID NetBeans ya que es uno de los ID más completos en soportar diferentes aplicaciones, entre ellos crear servicios web, además se utilizó el ID de Eclipse que se adapta perfectamente al SDK de Android, el mismo que permite crear aplicaciones nativas en el sistema operativo Android (Ver sección de Resultados, apartado 4.6 Implementación)

Objetivo 4: Planificar y organizar las pruebas y validaciones del aplicativo banca móvil para verificar que los componentes interactúen entre si de un modo adecuado y se permita certificar la implantación del sistema en un ambiente de producción.

Para verificar el funcionamiento de la aplicación móvil, se aplicó las pruebas pertinentes (Ver sección de Resultados, apartado 3.4.1 Pruebas unitarias), así mismo para validar las disponibilidad y correcto funcionamiento de la aplicación en diferentes dispositivos móvil y realizar pruebas de sobre carga hasta hacerlo inutilizable, para ello se realizaron pruebas de carga y stress (Ver sección Resultado, apartado 3.4.2 y 3.4.3 Pruebas Stress y Pruebas de Carga) y finalmente se aplicó las pruebas de validación a los usuarios de la COAC Fortuna (Ver sección Resultados, apartado 3.4.4 Pruebas de validación y anexo 6 Encuestas para la validación de la aplicación móvil).

Objetivo 5: Generar una base de conocimientos debidamente documentada sobre el desarrollo del proyecto banca móvil que sirva de guía para los usuarios y desarrolladores de manera que se permita mejorar, crear o actualizar el uso del sistema.

Para dar cumplimiento q este objetivo, se realizó el presente documento y los Manuales de Usuario y Programador para que la COAC Fortuna tenga la posibilidad de continuar con el presente proyecto y aumentar funcionalidades según vayan creciendo sus necesidades.



2. Valoración Técnica Económica y Ambiental

Referente a valoración técnica económica y ambiental del presente Trabajo de Titulación se detallan a continuación.

2.1. Valoración técnica-económica

El desarrollo del presente trabajo de titulación, es factible desde el punto de vista técnico, puesto que para la COAC Fortuna la aplicación permite ofrecer un nivel de servicio más personalizado a sus clientes, de la misma manera se adaptó al modelo de datos ya establecido en la institución con la finalidad de hacer uso del mismo y poder complementarlo con el aporte del proyecto.

Económicamente el proyecto de fin de carrera es factible ya que las herramientas utilizadas para su desarrollo son de licencia libre, además la COAC Fortuna cuenta con los requisitos necesarios para pagar la suscripción y mantenimiento para publicar la aplicación en Google Play, así mismo la implementación del servidor de aplicaciones donde se encuentra el desarrollo de la aplicación y a configurar las base de datos de producción, con la inserción de nuevas tablas y datos necesarios para el funcionamiento de la aplicación, de la misma manera la cooperativa cuenta con las seguridades necesarias establecidas por la SEPS, BCE y SBS.

Para llevar a cabo este proyecto, se utilizó los siguientes recursos:

Recursos Humanos: En la tabla LV se detalla la valoración económica de los recursos humanos que intervinieron en el desarrollo del presente trabajo como: analista, diseñador y programador.



TABLA LV: VALORACIÓN ECONÓMICA DE RECURSOS HUMANOS

DESCRIPCIÓN	Cantidad	Horas	Valor/Hora	Valor Total
Analista	1	120	\$ 10.00	\$1200
Diseñador	1	180	\$ 10.00	\$1800
Programador	1	320	\$ 10.00	\$3200
Recursos Humanos				\$ 6200

Recursos Técnicos:

Los recursos técnicos utilizados para el desarrollo se detallan en la tabla LVI con su respectiva valoración.

TABLA LVI: VALORACIÓN ECONÓMICA DE RECURSOS TÉCNICOS

DESCRIPCIÓN	Cantidad	Valor/Unitario	Horas	Valor/Hora	Valor Total
Hardware					
Computador Portátil Hp Core I5.	1		1600	\$0.50	\$ 800.00
Computador Portátil Sony Core I3.	1		1600	\$0.50	\$800.00
Impresora Lexmark serie 510.	1				\$75.00
Software					
Entorno de desarrollo (NetBeans, Eclipse)	2				\$0.00
SDK Android	1				\$0.00
Base de datos Oracle 11g.					\$0.00
Recursos Técnicos					\$1675.00

Recursos Materiales:

Los recursos materiales para las impresiones y consultas de información se detallan en la tabla LVII con su respectiva valoración económica.



TABLA LVII: VALORACIÓN ECONÓMICA DE RECURSOS MATERIALES

DESCRIPCIÓN	Cantidad	Valor/ Unitario	Valor Total
Resmas de Hojas	3	\$3.50	\$10.50
Internet (mensual)	10	\$12.00	\$120.00
Cartuchos	3	\$25.00	\$75.00
Cd's	2	\$0.80	\$1.60
Copias	80	\$0.02	\$1.60
Recursos Materiales			\$208.70

Recursos de Servicios: Los recursos de servicio como transporte se encuentran detallados en la tabla LVIII con su valoración económica.

TABLA LVIII: VALORACIÓN ECONÓMICA DE RECURSOS DE SERVICIOS

DESCRIPCIÓN	Cantidad	Valor/ Unitario	Valor Total
Transporte	35	\$0.25	\$8.75
Recursos de Servicios			\$8.75

El resumen del presupuesto para el desarrollo del presente trabajo se detalla en la tabla LIX.

TABLA LIX: TOTAL DE RECURSOS

DESCRIPCIÓN	VALOR
Recursos Humanos	\$ 6200
Recursos Técnicos	\$1675.00
Recursos Materiales	\$208.70
Recursos de Servicios	\$8.75
SUBTOTAL	\$8092.45
10 % (Imprevistos)	\$809.24
TOTAL	\$8901.69



h. Conclusiones.

- Se desarrolló e implementó el aplicativo Banca Móvil para la COAC Fortuna, el mismo que puede ser accedido desde dispositivos móviles que contengan un Sistema Operativo Android versión 2.1 o superior.
- El uso de tecnologías actuales como el SDK de Android permitió crear una aplicación móvil capaz de adaptarse a nuevos módulos según las necesidades del cliente y dar cumplimiento a los cambios o exigencias de los entes reguladores a los que la COAC Fortuna se rige.
- El uso del emulador de Android permitió desarrollar la aplicación móvil con una interfaz amigable para el usuario, de esta manera se puede realizar el proceso de consulta de manera confiable, eficiente y rápida.
- EL uso de Oracle como gestor de Base de Datos permitió incrementar el nivel de conocimientos en seguridad perimetral de datos, de esta manera se pudo resolver las peticiones en línea de acceso y respuesta a la información del cliente.
- El realizar pruebas al sistema móvil permitió evaluar el desempeño del mismo, verificar los métodos de control implementados como la encriptación de claves y su diseño responsivo para determinar el funcionamiento de la aplicación en los dispositivos sean estos Tablet o Smartphone independientemente del tamaño de la pantalla u orientación del dispositivo.



i. Recomendaciones.

- Analizar, comparar y justificar el uso de herramientas tanto en hardware como en software para evitar que la complejidad del proyecto se eleve al punto máximo, esto, al momento de ampliar los servicios de la aplicación móvil.
- Ampliar los servicios de la aplicación móvil de manera que el cliente pueda efectuar transacciones como pago de facturas, transferencias internas y externas e ingreso de solicitudes de crédito.
- Implementar la geo localización de cajeros automáticos de manera que facilite al cliente el retiro de dinero del punto más cercano a su posición.
- El uso de Métodos de Encriptación Síncronos para complementar los nuevos servicios de la aplicación Banca Móvil Fortuna para dar cumplimiento a las normativas de la Súper Intendencia de Bancos.
- Implementar seguridad dactilar para realizar transferencias desde la aplicación Banca Móvil Fortuna para incrementar la confiabilidad y seguridad del cliente al hacer uso del dispositivo móvil.






j. Bibliografía.

- [1] "Especial de Redes & Telecom", [Revista Online] Available: <http://www.redestelecom.es/revista> [Accessed: 22-Enero-2014].
- [2] Javier Cuello y José Vittone, *Diseñando App para móviles*, Edición Kindle, 2013 [Consultado: 04-Feb-2014].
- [3] "Tipos de Desarrollo para Aplicaciones Móviles", [Online]. Available: <http://geospatialtrainings.com/recursos-gratuitos/tipos-de-aplicaciones-moviles/>. [Accessed: 10-Mar-2014].
- [4] BBVA Chile Innovation Edge, *Banca Móvil*, Ediciones 2012.
- [5] "Ranking de Sistemas Operativos más usados en el 2013", [Online]. Available: <http://blog.uchceu.es/informatica/ranking-de-sistemas-operativos-mas-usados/>. [Accessed: 12-Mar-2014].
- [6] "Ranking de Sistemas Operativos más usados en el 2014", [Online]. Available: <http://blog.uchceu.es/informatica/ranking-de-sistemas-operativos-mas-usados-para-2014/>. [Accessed: 12-Mar-2014].
- [7] "Desarrollo de Aplicaciones en Android", [Online]. Available: <http://www.mibqyyo.com/articulos/2013/07/02/desarrollo-en-android-1-preparando-entorno-de-trabajo>. [Accessed: 17-Mar-2014].
- [8] "Android, Características" [Online]. Available: <http://androidos.readthedocs.org/en/latest/data/caracteristicas/>. [Accessed: 17-Mar-2014].
- [9] García García Carlos, Hernández Pulida David, Monterjo Javier, *Desarrollo de aplicaciones en Android*, Marzo 2010.
- [10] "WSDL", [Online]. Available: <http://programacion.jias.es/2012/01/web-service-definicion-utilizacion-estructura-del-wsdl/>. [Accessed: 18-Mar-2014].
- [11] Cristina Vicente Chicote y Diego Alonso Cáceres, *Herramientas Eclipse para desarrollo de software*, Vol. 1, N, Sistedes 2007.
- [11] http://www.ecured.cu/index.php/Eclipse,_entorno_de_desarrollo_integrado
<http://www.elandroidelibre.com/2010/09/que-significa-el-diccionario-android.html>
- [12] "La importancia de Android en MVC", [Online]. Available: <http://androideity.com/2012/05/10/la-importancia-del-mvc-en-android/>. [Accessed: 24-Mar-2014].



k. Anexo

Anexo 1 Entrevista

Resumen		
Fecha de entrevista:	13 de enero de 2014	
Participantes:	Rosa Romero, Servicios Cooperativos de la COAC Fortuna. Paul Bravo, Jefe de Sistemas (E) de la COAC Fortuna.	
Contenido:	<p>Servicios Cooperativos pronuncio lo siguiente:</p> <p>La mayor demanda que la cooperativa actualmente tiene es por la poca cobertura de oficinas y cajeros automáticos que tiene en la localidad, esto obliga a ciencia cierta al cliente a trasladarse hasta la oficina matriz ubicada en un punto no tan central para realizar una simple consulta o transacción, esto ha ocasionado el retiro de ciertos clientes importantes que tan solo para solicitar su saldo de cuenta necesitan recurrir al área de cajas con su libreta respectiva, o en su caso al área de servicios cooperativos teniendo que esperar el turno respectivo, de manera que es importante que la cooperativa pueda ofrecer un mecanismo de consulta de la información bancaria del cliente, de esta manera se evitaría el obligar al cliente a venir a las instalaciones y de cierta manera conservar el cliente por los servicios prestados.</p> <p>Jefatura de Sistemas pronuncio lo siguiente:</p> <p>Es importante que la cooperativa pueda estar tecnológicamente a la par con el resto de entidades bancarias, asumiendo el costo de implementar las seguridades exigidas por los entes de control externo. Sería de gran ventaja que se pueda utilizar dispositivos que están a la mano del cliente como lo son los dispositivos móviles, cualquier sistema que contemple las necesidades de la cooperativa deben adaptarse a nuestro Core Financiero Financial 2.0 y a la base de datos en Oracle, debe contemplar una aplicación de fácil uso para el cliente y que abarque las principales necesidades del usuario</p>	
Firman		
Tesista	Servicios Cooperativos	Jefe de Sistemas
Natali Cruz G 	Rosa Romero 	Paul Bravo COOPERATIVA DE AHORRO Y CREDITO "FORTUNA"  JEFE DE SISTEMAS



BANCA MÓVIL PARA LA COAC "FORTUNA" CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID

Anexo 2 Ficha de Observaciones (Core Bancario).

Consulta de Cuentas

Captaciones Vista.

Ficha Nemobibliográfica N°

Consulta de Cuentas: Saldo Cuenta

La imagen adjunta permite saber de que manera se pueden presentar los datos al cliente, en el cual constan NE de Cuenta, Tipo de cuenta, estado y Saldo Cuenta, oficina de apertura. Al seleccionar cada cuenta se puede observar Información Adicional y revisar movimientos por un rango de fechas.

Las imágenes adjuntas son tomadas del core financiero de la Cooperativa Fortuna, de esta manera el gestor bancario puede revisar la información de las cuentas de los clientes, no existe otro proceso alternativo de verificar dicha información.

Pol: Servicios Cooperativos

Consulta de Depósitos

Captaciones Vista.

Ficha Nemobibliográfica N°

Consulta de Depósitos: Políticas

Por medio de la imagen adjunta se puede observar el proceso normal de consulta de políticas.

Esta es la imagen principal en la que se detallan todas las políticas del cliente con su respectivo estado, monto, código, fecha creación, fecha vencimiento. Cada política permite revisar su información más detallada como tasa de interés, plazo en días. Además se puede verificar por cada política los movimientos que tiene relacionado como pago de intereses.

Una adición que se pudo distinguir es que el cliente que tiene políticas no está obligado a crear cuentas de ahorro, por ende no presenta cuentas atadas a dicho cliente.

Pol: Inversión Política



BANCA MÓVIL PARA LA COAC "FORTUNA" CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID

Administración de Tarjetas de Débito

Captaciones Vista Tarjetas de Débito Administración

Cliente: UN 5245 1100165540 BRAVO CASTRO LOBETH PAUL [Ingresos](#) [Seguridad](#)

Código	Tipo	Estado	Saldo Cuenta	Oficina	Producto
6048...	PAUL BRAVO	ACTIVA	0.000.00	MATIZ	Ahorros Débito

Administración de Tarjeta Débito

Detalle De Tarjeta Débito

Secuencia	Código	Nombre Tarjeta	Fecha Emisión	Tipo	Estado Tarjeta	Es Principal	Oficina de Rótulo
1	6048...	PAUL BRAVO	07-05-2013	Nueva	Activa	<input checked="" type="checkbox"/>	MATIZ

Ficha Nemobibliográfica N°

Captaciones Vista

Administración de Tarjetas de Débito

En captaciones vista se puede realizar la administración de tarjeta de débito, en la pantalla principal le presenta los datos de la cuenta a la que está asociada la tarjeta de débito como No de Código, Estado, Saldo Cuenta, Oficina, Producto.

En la administración de tarjeta de débito se puede realizar, el pedido de Nueva tarjeta, dar de baja una tarjeta, Bloquear, el cliente podrá tener acceso únicamente al bloqueo en el que se exige el tipo de bloqueo y las razones justificadas.

Rol: Responsable Tarjeta de débito

procler

Préstamos por Cliente

Préstamos por Cliente Préstamos por Cliente

Cliente: UN 5245 1100165540 BRAVO CASTRO LOBETH PAUL [Ingresos](#) [Seguridad](#)

Número	Tip	Deuda Total	Saldo	Morosos	Estado
01000001	CÓMERCIO BACK TO BACK	000.00	0.00	0.00	CANCELADO

Información Adicional

Cliente: 7801 CRUZ GONZALEZ NATALIE DE CONE

Número Préstamo: 01000001075

Oficina: MATIZ

Oficial: CRISTIANO USERRA RUIZ PAOLA

Tipo: COMERCIO BACK TO BACK

Estado: CANCELADO

Deuda Inicial: 0.00

Saldo Actual: 0.00

Ajustado: 08-12-2013

Último Pago: 08-12-2014

Préstamo vencimiento: 04-10-2014

Exigible Capital: 0

Exigible Interés: 0

Exigible Mora: 0

Exigible Otros: 0

Valor Para Estar al Día: 0

A Pagar Hasta Cuenta Actual: 0

Valor Para Cancelar: 0

Número Cuotas Vencidas: 0

Número Días Vencidos: 0

Movimientos Préstamo

Documento	Cuenta	Saldo	Valor	Fecha	Fecha Historia	Transacción	Usuario	Oficina	Estado
1100010001	Capital	1000.00	1000.00	08-12-2013	08-12-2013 20:02:01	Adjudicación de Préstamo	ACOSTILLO	MATIZ	A
1100010001	Seguro de Desgravamen	0.40	0.00	09-01-2014	09-01-2014 09:22:41	Alto Préstamo (Débito a Cuenta)	FRANCO	MATIZ	E
1100010001	Interés	0.00	0.00	09-01-2014	09-01-2014 09:22:41	Alto Préstamo (Débito a Cuenta)	FRANCO	MATIZ	E
1100010001	Capital	245.39	235.69	09-01-2014	09-01-2014 09:22:41	Alto Préstamo (Débito a Cuenta)	FRANCO	MATIZ	E
1100010001	Seguro de Desgravamen	0.00	0.00	09-02-2014	09-02-2014 21:02:11	Alto Préstamo (Débito a Cuenta)	FRANCO	MATIZ	E
1100010001	Interés	0.00	0.00	09-02-2014	09-02-2014 21:02:11	Alto Préstamo (Débito a Cuenta)	FRANCO	MATIZ	E
1100010001	Capital	245.37	235.62	09-02-2014	09-02-2014 21:02:11	Alto Préstamo (Débito a Cuenta)	FRANCO	MATIZ	E
1100010001	Seguro de Desgravamen	0.00	0.00	09-03-2014	09-03-2014 08:19:11	Alto Préstamo (Débito a Cuenta)	MOGOLICZ	MATIZ	E
1100010001	Interés	0.00	0.00	09-03-2014	09-03-2014 08:19:11	Alto Préstamo (Débito a Cuenta)	MOGOLICZ	MATIZ	E
1100010001	Capital	245.36	235.29	09-03-2014	09-03-2014 08:19:11	Alto Préstamo (Débito a Cuenta)	MOGOLICZ	MATIZ	E
1100010001	Seguro de Desgravamen	0.00	0.00	09-03-2014	09-03-2014 08:19:11	Alto Préstamo (Débito a Cuenta)	MOGOLICZ	MATIZ	E
1100010001	Capital	245.35	235.69	09-03-2014	09-03-2014 08:19:11	Alto Préstamo (Débito a Cuenta)	MOGOLICZ	MATIZ	E
1100010001	Interés	0.00	0.00	09-03-2014	09-03-2014 08:19:11	Alto Préstamo (Débito a Cuenta)	MOGOLICZ	MATIZ	E

Ficha Nemobibliográfica N°

Préstamos por Cliente: Créditos

Mediante esta pantalla se puede observar todos los créditos que un cliente tiene asociado. En esta pantalla se puede observar el No del crédito, nombre de crédito, deuda morosa, Saldo, Fecha de Adjudicación y Estado. Se presentan todos los créditos pendiente o están como cancelados. Cada crédito se puede revisar su información adicional, como el nombre del Oficial de crédito, fecha de vencimiento de la próxima cuota, número de cuotas vencidas, días de vencimiento, valor para estar al día, valor a cancelar de todo el crédito, así mismo se puede revisar los movimientos del crédito para verificar los pagos realizados.

Rol: Oficial de Crédito

procler



Anexo 3 Modelo de Encuesta.



UNIVERSIDAD NACIONAL DE LOJA

AREA DE LA ENERGIA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO RENOVABLES

Carrera de Ingeniería en Sistemas

TEMA: "DISEÑO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO FORTUNA DE LA CIUDAD DE LOJA, CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID"

FECHA: Octubre del 2014

1. ¿La cooperativa tiene algún mecanismo externo de atención al público?
.....
.....
.....
2. ¿La cooperativa presta algún servicio como tarjeta de débito, tarjeta de crédito? Y ¿Tienen algún mecanismo de bloqueo externo para dicho servicio?
.....
.....
.....
3. ¿Qué servicio debe prestar la aplicación móvil?
.....
.....
.....
4. ¿Qué tipo de consultas podrá realizar el cliente?
.....
.....
.....



5. ¿Qué tipo de cuentas puede revisar el cliente?

.....
.....
.....

6. ¿Qué información es la más relevante en la cuenta?

.....
.....
.....

7. ¿Qué información es la más relevante en pólizas?

.....
.....
.....

8. ¿Qué información es la más relevante en créditos?

.....
.....
.....

9. ¿Cuál es el servicio por el que más preguntan los clientes?

.....
.....
.....

10. ¿Qué medios de seguridad debe tener la aplicación móvil?

.....
.....
.....



Anexo 4 Aprobación del Proyecto de Tesis.

Sra. Lucía Toro de Mora

SECRETARIA GENERAL DE LA MICROEMPRESA "FORTUNA"

CERTIFICA:

La Cooperativa de Ahorro y Crédito de la Microempresa "FORTUNA", autoriza a la Srta. Nataly Cruz González para que realice la investigación Banca Móvil para la Cooperativa, con accesibilidad desde dispositivo Android.

Loja, noviembre de 2014

Sra. Lucía Toro de Mora
SECRETARIA GENERAL

Bolívar e Imbabura* Tel.:07- 2572688 - 2572954
www.coopfortuna.fin.ec
Loja-Ecuador



Anexo 5 Certificado de realización de pruebas.



Loja, 27 de Octubre de 2014

Ing.
Paul Bravo
JEFE DE SISTEMAS DE LA COAC FORTUNA (E)

CERTIFICO:

Que las pruebas correspondientes a la aplicación Banca Móvil Fortuna con accesibilidad desde dispositivos móvil con Sistema Operativo Android, fueron realizadas en las instalaciones del mismo, los días 24 y 25 de Octubre del presente año, dirigida por la Tesista Natali del Cisne Cruz González, quien aplico encuestas a los usuarios de la COAC Fortuna del sistema para conocer el grado de confiabilidad y aceptación.

Atentamente:



COOPERATIVA DE AHORRO Y
CRÉDITO "FORTUNA"



JEFE DE SISTEMAS

Ing. Paul Bravo

JEFE DE SISTEMAS DE LA COAC FORTUNA (E)



Anexo 6 Encuestas para la validación del sistema móvil.

Encuesta 1



UNIVERSIDAD NACIONAL DE LOJA

AREA DE LA ENERGIA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO RENOVABLES

Carrera de Ingeniería en Sistemas

TEMA: "DISEÑO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO FORTUNA DE LA CIUDAD DE LOJA, CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID"

NOMBRE: Eco. Rosa Romero

CARGO: Servicios Cooperativos / Inversiones

FECHA: Octubre 2014

1. ACCESIBILIDAD

¿Tuvo inconvenientes al momento de registrar su usuario en la aplicación móvil?

SI ()

NO (✓)

Porque: *La aplicación es de fácil manejo*

¿Tuvo inconvenientes al revisar en su correo electrónico la clave de seguridad que genera la aplicación?

SI ()

NO (✓)

Porque: *Se maneja correctamente el correo electrónico*

¿Tuvo inconvenientes al momento de ingresar el usuario y contraseña en la aplicación móvil?

SI ()

NO (✓)



Porque:.....
.....

2. Funcionalidad

En el siguiente cuadro marque SI o NO según corresponda:

OPERACION	DIFICULTAD	
	SI	NO
Ingreso a la aplicación móvil.		✓
Verificar servicios que brinda la aplicación móvil.		✓
Seleccionar el icono cuentas.		✓
Seleccionar opción Información.		✓
Seleccionar opción Movimiento.		✓
Ingresar rango de fechas anteriores a la fecha actual.		✓
Digitar el botón Consultar.		✓
Ingresar rango de fechas posteriores a la fecha actual.		✓
Digitar el botón Consultar.		✓
Regresar a la opción de movimiento.		✓
Seleccionar la opción Movimiento.		✓
No seleccionar rango de fechas.		✓
Verificar que la aplicación presente mensaje de campos vacíos.		✓
Seleccionar el icono pólizas.		✓
Seleccionar opción Información.		✓
Regresar a la opción de información.		✓
Seleccionar opción Movimiento.		✓
Regresar a la opción de movimiento.		✓
Seleccionar el icono créditos.		✓
Seleccionar opción Información.		✓
Regresar a la opción de información.		✓
Seleccionar opción Movimiento.		✓
Regresar a la opción de movimiento.		✓
Seleccionar Tarjeta de Débito.		✓
Solicitar Bloquear tarjeta.		✓
Ingresar tipo de bloqueo.		✓



Ingresar observaciones.		<input checked="" type="checkbox"/>
Digitar el botón Bloquear.		<input checked="" type="checkbox"/>
Confirmar solicitud de bloqueo.		<input checked="" type="checkbox"/>
Leer mensaje de bloqueo por el tipo de bloqueo ingresado que emite la aplicación.		<input checked="" type="checkbox"/>
Regresar a la opción de Tarjeta de Débito.		<input checked="" type="checkbox"/>
Solicitar Bloquear tarjeta.		<input checked="" type="checkbox"/>
Validar mensaje de bloque ya existente.		<input checked="" type="checkbox"/>

Considera Ud. Que la aplicación "Banca Móvil Fortuna" abarca las principales necesidades de los usuarios de la cooperativa.

SI () NO ()

Porque: Como la cooperativa no cuenta con sucursales, es una muy buena opción Banca móvil

Considera Ud. Que la aplicación "Banca Móvil Fortuna" realiza los procesos de manera confiable, eficiente y rápida.

SI () NO ()

3. DISEÑO Y PRESENTACION

¿Considera que la interfaz de la aplicación banca móvil es amigable y de fácil uso?

SI () NO ()

Porque: Es de fácil uso, una buena herramienta para verificar mis cuentas.

1103912661



Encuesta 2



UNIVERSIDAD NACIONAL DE LOJA

AREA DE LA ENERGIA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO RENOVABLES

Carrera de Ingeniería en Sistemas

TEMA: "DISEÑO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO FORTUNA DE LA CIUDAD DE LOJA, CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID"

NOMBRE: Ing. Paul Bravo.

CARGO: Jefe de Sistemas.

FECHA: Octubre 2014

1. ACCESIBILIDAD

¿Tuvo inconvenientes al momento de registrar su usuario en la aplicación móvil?

SI () NO (✓)

Porque: *El manejo de la aplicación es fácil*

¿Tuvo inconvenientes al revisar en su correo electrónico la clave de seguridad que genera la aplicación?

SI () NO (✓)

Porque: *el correo se lo usa constantemente, no es difícil el manejo*

¿Tuvo inconvenientes al momento de ingresar el usuario y contraseña en la aplicación móvil?

SI () NO (✓)



Porque:.....
.....

2. Funcionalidad

En el siguiente cuadro marque SI o NO según corresponda:

OPERACION	DIFICULTAD	
	SI	NO
Ingreso a la aplicación móvil.		/
Verificar servicios que brinda la aplicación móvil.		/
Seleccionar el icono cuentas.		/
Seleccionar opción Información.		/
Seleccionar opción Movimiento.		/
Ingresar rango de fechas anteriores a la fecha actual.		/
Digitar el botón Consultar.		/
Ingresar rango de fechas posteriores a la fecha actual.		/
Digitar el botón Consultar.		/
Regresar a la opción de movimiento.		/
Seleccionar la opción Movimiento.		/
No seleccionar rango de fechas.		/
Verificar que la aplicación presente mensaje de campos vacíos.		/
Seleccionar el icono pólizas.		/
Seleccionar opción Información.		/
Regresar a la opción de información.		/
Seleccionar opción Movimiento.		/
Regresar a la opción de movimiento.		/
Seleccionar el icono créditos.		/
Seleccionar opción Información.		/
Regresar a la opción de información.		/
Seleccionar opción Movimiento.		/
Regresar a la opción de movimiento.		/
Seleccionar Tarjeta de Débito.		/
Solicitar Bloquear tarjeta.		/
Ingresar tipo de bloqueo.		/



Ingresar observaciones.		/
Digitar el botón Bloquear.		/
Confirmar solicitud de bloqueo.		/
Leer mensaje de bloqueo por el tipo de bloqueo ingresado que emite la aplicación.		/
Regresar a la opción de Tarjeta de Débito.		/
Solicitar Bloquear tarjeta.		/
Validar mensaje de bloque ya existente.		/

Considera Ud. Que la aplicación "Banca Móvil Fortuna" abarca las principales necesidades de los usuarios de la cooperativa.

SI (/) NO ()

Porque: *Es que es una herramienta para verificar mis cuobts es buena opción ya que los cheques por tiempo no se acercan.*

Considera Ud. Que la aplicación "Banca Móvil Fortuna" realiza los procesos de manera confiable, eficiente y rápida.

SI (/) NO ()

3. DISEÑO Y PRESENTACION

¿Considera que la interfaz de la aplicación banca móvil es amigable y de fácil uso?

SI (/) NO ()

Porque: *Es amigable ya que es como un correo y son datos que el usuario maneja constantemente.*

COOPERATIVA DE AHORRO Y
CREDITO "FORTUNA"


JEFE DE SISTEMAS

Anexo 7 Estudio realizado por BBVA Mobile Banking



El primer servicio de móvil comercial se lanzó hace 32 años en Japón, cuando NTT DoCoMo comenzó a ofrecer un servicio de comunicaciones móviles el 1 de diciembre de 1979. Hoy en día estamos en la cúspide de una conectividad omnipresente y global, habiendo roto la barrera de los 6 mil millones de suscriptores en el 2011 (ver Visualización de la penetración móvil). Aunque mucha de la tecnología interna ha cambiado desde los primeros tiempos, en esencia, los dispositivos móviles han evolucionado de un simple teléfono a un mecanismo de computación y comunicación de tamaño portátil. Mientras el mundo sigue explorando y descubriendo las posibilidades adiciona-

les que pueden ofrecer las tecnologías móviles, estamos claramente en camino hacia una economía del "en todo momento y lugar". Con un modelo de negocio adecuado, las compañías pueden ofrecer servicio a los 6 mil millones de clientes usuarios de tecnologías móviles: creando y manteniendo con ellos una relación fiel, y asentando un canal principal para hacer llegar la propuesta de valor de la empresa, con un enfoque centrado en el cliente.

Fuente: Communities Dominate Brands! Celebrating 30 years of mobile phones. Thank you NTT of Japan, 2009.

Fuente: ITU (International Telecommunications Union) Noviembre del 2011.

Penetración de los móviles



APP

El término app (aplicación), nos puede recordar a esos botones tan vistosos de nuestros dispositivos portátiles. Técnicamente, esas aplicaciones son conocidas como apps nativas. Son aplicaciones hechas a medida que funcionan en plataformas OS específicas, a menudo programadas para un dispositivo o grupo de dispositivos concretos. Cuando las capacidades de estos dispositivos son incorporadas en la app, como las cámaras, GPS o NFC, ofrecen una rica experiencia al usuario. Sin embargo, hay que pagar un precio por esta riqueza: las apps nativas deben de ser desarrolladas individualmente para cada plataforma OS, lo cual puede incrementar considerablemente el gasto de producción.

Por otro lado, hay otro tipo de apps menos conocidas: las aplicaciones web. En esencia, cualquier sitio web preparado para acceso vía móvil puede ser considerado una app web. La ventaja de este enfoque reside en la capacidad del "escribe una vez, publica muchas veces", ya que las apps web son independientes del dispositivo. Mucha gente piensa que el nuevo estándar HTML5 está borrando las fronteras entre aplicaciones web y nativas. Esto puede que demuestre ser una forma eficaz y rentable para producir aplicaciones que proporcionen una rica experiencia de usuario.

Conectividad

Conexión a una línea de red

PROTAGONISTAS CLAVE:
Operadores de red móvil como Telefonica, AT&T, Verizon...

Dispositivos

Utilización de dispositivos móviles

PROTAGONISTAS CLAVE:
Apple, Samsung, HTC, Nokia, LG, Sony, Nintendo...

Plataformas OS

Sistemas operativos de las tabletas, teléfonos inteligentes y otros dispositivos móviles.

PROTAGONISTAS CLAVE:
Android, iOS, Blackberry, Windows phone...

App Store

Tienda virtual para comprar aplicaciones

PROTAGONISTAS CLAVE:
Apple App Store, Google Play, RIM App World, Nokia Store.

App

Nativa o web

PROTAGONISTAS CLAVE:
Desarrolladores de aplicaciones



Tecnologías de la banca móvil

USO. Generalizado en el mundo en desarrollo. Muchas aplicaciones de alerta y opciones de pago son utilizadas en el mundo desarrollado.

PROS. Se puede utilizar a través de operadores o plataformas, teléfonos convencionales o smartphones. Es fácil de usar y bastante seguro.

CONTRAS. Algunas cuestiones de seguridad no se permiten en ciertas regiones. Funcionalidad reducida.

USO. Generalizado en el mundo desarrollado, particularmente en los bancos de segundo nivel y regionales. Los navegadores móviles dan acceso al usuario a un entorno de banca internet especializado.

PROS. Conocido por los clientes de banca por internet, sencilla integración con las plataformas existentes, funciona en una amplia variedad de dispositivos y MNOs.

CONTRAS. Es un concepto bastante generalizado (no ofrece ninguna ventaja competitiva). El acceso conlleva una serie de pasos. No siempre se adapta bien las pantallas pequeñas.

USO. Está cobrando un fuerte impulso en el mundo desarrollado. Se utilizan en la banca móvil y servicios basados en cupones y localización.

PROS. Ofrece una rica experiencia de usuario. Más seguras y estables. Mantiene la fidelidad y ofrece oportunidades de venta cruzada.

CONTRAS. Cada dispositivo requiere una integración concreta. Debe ser instalado por el cliente. Puede ser más caro de implementar.



Anexo 8 Validación de la Especificación de Requisitos.

"DISEÑO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO FORTUNA DE LA CIUDAD DE LOJA, CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID"

Pág. 3

Especificación de Requisitos de Software

Documento validado por las partes en fecha: 12/03/2014.

Por la COAC Fortuna	Por la Universidad Nacional de Loja
<p>COOPERATIVA DE AHORRO Y CRÉDITO "FORTUNA" JEFE DE SISTEMAS Ing. Paul Bravo</p>	<p>Ing. Alex Virilio Padilla Encalada, Mgs. DIRECTOR DE TESIS</p>



Anexo 9 Certificación de traducción.



Yo, FREDDY PAÚL CASTILLO, certifico:

Que tengo el conocimiento y dominio de los idiomas español e inglés y que las traducciones de los siguientes:

Documento:

"DISEÑO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO DE LA MICROEMPRESA "FORTUNA" DE LA CIUDAD DE LOJA, CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID".

Para: Natali Cruz González

Firmado en Loja a los 20 días del mes de noviembre del 2014



Washington English Institute certifica la traducción del presente documento.
Av. 24 de mayo 11-36 y azuay
2579934-2573489



UNIVERSIDAD NACIONAL DE LOJA

AREA DE LA ENERGIA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO RENOVABLES

CARRERA DE INGENIERIA EN SISTEMAS

TEMA:

“DESARROLLO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA
DE AHORRO Y CRÉDITO FORTUNA DE LA CIUDAD DE LOJA, CON
ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID”

Tesis previa a la obtención de
grado de ingeniería en
Sistemas

AUTORA: Natali del Cisne Cruz González

LOJA – ECUADOR

2013



Anexo 10 Anteproyecto del trabajo de titulación.

A. TITULO

“DESARROLLO E IMPLEMENTACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO FORTUNA DE LA CIUDAD DE LOJA, CON ACCESIBILIDAD DESDE DISPOSITIVOS ANDROID”



B. PROBLEMÁTICA

1. SITUACIÓN PROBLEMÁTICA

Dentro de nuestro país la tecnología juega un rol importante para el desarrollo constante y significativo de sistemas que permitan ofrecer servicios, controles, accesos y buen funcionamiento de los mismos, de esta manera podemos mencionar que la implementación de nuevas tecnologías, nos llevan a desarrollar nuevas propuestas que satisfagan sistemas basados en tecnologías actuales.

Como la tecnología actual necesitó tiempo, estudio e investigación para su constante evolución, podemos referirnos como tecnología al gran crecimiento de sistemas enfocados principalmente en brindar oportunidades a los usuarios de realizar transacciones sin necesidad de acudir a la institución, dejando como puerta abierta los nuevos servicios bancarios que se pueden ir construyendo, dando así la necesidad de construir sistemas orientados a los nuevos dispositivos móviles, por estas razones es que la Universidad Nacional de Loja haciendo uso del Sistema Académico modular por Objetos de Transformación (SAMOT) que tiene como finalidad el formar profesionales capaces de desenvolverse en el mundo actual y la Carrera de Ingeniería en sistemas basada en el uso continuo de la tecnología y del desarrollo académico nos ha permitido analizar y llevar a cabo la realización de un proyecto de tesis tomando como fundamento los conocimientos obtenidos en el transcurso de nuestra carrera y las carencias bancarias que hoy por hoy existen.

Como profesionales se debe proporcionar a la empresa la oportunidad de mejorar de manera significativa la atención al cliente y aumentar la eficiencia de la organización, así como también mejorar las relaciones con clientes nuevos y existentes.

2. PROBLEMAS ESPECÍFICOS

Al ser la cooperativa Fortuna una institución de la banca privada que maneja un considerable número de clientes está propensa a ciertos cambios dentro de la forma en cómo evoluciona la tecnología, es por ello que se han encontrado algunos inconvenientes dentro de los cuales tenemos:



La Cooperativa:

- Por falta de un servicio en línea para realizar consulta de saldos o acreditaciones de las cuentas personales de los clientes, la institución ha dejado de contar con ciertos ingresos por la desvinculación de los mismos.
- Por no disponer de un mecanismo en línea en el que permita saber al cliente el valor exacto de su cuota y la fecha de pago del crédito ocasiona una alta morosidad e incurrir en pérdida de la institución.
- El sistema actual no permite realizar bloqueo en línea de tarjetas de débito exponiendo que se realicen transacciones y el acceso a la información bancaria de los clientes.
- El gran número de transacciones que se realizan de manera interna no cubre la demanda de clientes, ocasionando molestias por no ofrecer alternativas para acceder a su información personal, tomando en cuenta la atención personalizada que es el núcleo del servicio al cliente definido en esta institución.
- El no ofrecer nuevos servicios y la falta de innovación en la tecnología está originando pérdida en los ingresos de la cooperativa por no dar acceso al cliente a su información financiera sin que tenga que recurrir a las instalaciones, dificultando la realización de otras actividades de los socios.

3. PROBLEMA DE INVESTIGACIÓN

Una vez analizados los inconvenientes que se presentan en esta institución se considera firmemente que se debe desarrollar un sistema que brinde la posibilidad de revisar su información bancaria, realizar tareas administrativas de una forma más rápida y segura que ayudará a fomentar la lealtad de los clientes, para lo cual se concluye que el problema global se constituye en: **“Dificultad para realizar transacciones de manera personalizada de los clientes de la Cooperativa de Ahorro y Crédito Fortuna, a través de tecnología que se adapten a la realidad social”**.



C. JUSTIFICACIÓN

1. ACADÉMICA

La Universidad Nacional de Loja siendo uno de los principales centros de educación superior existentes en la Región Sur del Ecuador, tiene como objetivo involucrar a los estudiantes junto con la sociedad a través de la investigación científica, para fomentar en ellos un espíritu de colaboración y sentido más humano de la realidad que nos rodea.

El área de Energía, Industrias y Recursos Naturales no Renovables, en su afán de brindar a la sociedad profesionales capaces de solventar las dificultades que el mundo actual nos pone, tiene como objetivo el aplicar una educación técnica e investigativa de acuerdo a los intereses y necesidades que propone la sociedad actual.

La carrera de Ingeniería en Sistemas de la Universidad Nacional de Loja, a la cual pertenezco tiene como objetivo principal formar profesionales que puedan optimizar el uso de software y hardware, así mismo tiene como misión la educación de personas capaces de manejar, crear y resolver problemas de las nuevas tecnologías para de esta manera cumplir con su objetivo.

Es por esto que como estudiante de la carrera de Ingeniería en Sistemas; y con los conocimientos adquiridos y proporcionados por los docentes que aquí laboran; he creído conveniente desarrollar un aplicativo para banca móvil para así reforzar y adquirir nuevos conocimientos y experiencias útiles para una futura práctica profesional, mejorando el desenvolvimiento eficiente en el ámbito laboral.

2. ECONÓMICA

Para el desarrollo del presente proyecto se dispone de los recursos económicos adecuados, tomando en cuenta que se llevará un presupuesto conveniente que permita efectuar únicamente los gastos necesarios, y que a la vez permita cumplir a cabalidad con los objetivos planteados. Además, cabe puntualizar que todos los materiales a utilizar en el proyecto estarán al alcance y no será necesaria la adquisición de nuevos implementos.



3. SOCIAL

En la actualidad, las nuevas tecnologías se combinan en una sola; los celulares inteligentes ya no se limitan a la función de comunicar a dos personas entre sí, sino que ahora han evolucionado hasta incluir modalidades como la transmisión de datos desde una PC hacia un teléfono móvil.

El mercado mundial nos ofrece la posibilidad de interactuar con nuevos dispositivos y esto trae consigo innegables ventajas, acelera el ritmo al cual obtenemos información, facilita las comunicaciones, reduce los tiempos de emisión y respuesta; es decir, transforma la vida diaria en todo un acontecimiento tecnológico, todo esto asociado al crecimiento económico de las instituciones financieras por la prestación de servicios que pueden ofrecer al cliente de una manera confiable y accesible.

4. OPERATIVA

Es factible la creación de un sistema de banca móvil para la Cooperativa Fortuna de la Ciudad de Loja con accesibilidad desde aplicativos móviles que contengan Android ya que esto permitirá al usuario el acceso permanente a la consulta de su información bancaria, de la misma manera tendrá la oportunidad de realizar bloqueos de sus tarjetas de débito en cualquier instante del día salvaguardando la credibilidad, seguridad y confianza depositada en dicha institución.

Las autoridades de la Cooperativa Fortuna brindan todo el apoyo necesario para la realización de este proyecto, ya que la propuesta presentada les parece muy interesante, innovadora y muy necesaria dentro de su institución.

Además la finalidad principal de esta investigación es la de obtener la pertinencia como tema de tesis previo a la obtención del título de Ingenieros en Sistemas.



D. OBJETIVOS

1. OBJETIVO GENERAL

- ✓ Desarrollar e implementar el sistema de banca móvil para la Cooperativa de Ahorro y Crédito Fortuna de la ciudad de Loja, con accesibilidad desde dispositivos Android.

2. OBJETIVOS ESPECÍFICOS

- ✓ Analizar y determinar los requerimientos del sistema banca móvil para desarrollar los módulos lógicos y físicos de acuerdo a las necesidades que ha establecido el usuario.
- ✓ Determinar el diseño de la arquitectura del software basándose en los requerimientos y en las tendencias actuales de sistemas de banca móvil de manera que se pueda obtener la interfaz gráfica y establecer el funcionamiento del sistema.
- ✓ Desarrollar e implementar la aplicación banca móvil haciendo uso de herramientas web para tener accesibilidad desde cualquier dispositivo Android.
- ✓ Planificar y organizar las pruebas y validaciones del aplicativo banca móvil para verificar que los componentes interactúen entre si de un modo adecuado y se permita certificar la implantación del sistema en un ambiente de producción.
- ✓ Generar una base de conocimientos debidamente documentada sobre el desarrollo del proyecto banca móvil que sirva de guía para los usuarios y desarrolladores de manera que se permita mejorar, crear o actualizar el uso del sistema.