

**UNIVERSIDAD NACIONAL DE SAN CRISTOBAL DE HUAMANGA**

**Facultad de Ingeniería de Minas, Geología y Civil**

**Escuela de Formación Profesional de Ingeniería de Sistemas**



**“APLICACIÓN WEB Y MÓVIL DE GESTIÓN DE COBRANZAS PARA LA  
RECUPERACIÓN DE CRÉDITOS MOROSOS EN LA CAC. “SANTA MARIA  
MAGDALENA”, 2016”**

Tesis presentada por : Darwin Martin Huamán Huamání

Para optar el título profesional de : Ingeniero de Sistemas

Tipo de investigación : Descriptiva

Área de investigación : Ingeniería de Software

Asesor : MSc. Ing. Efraín Elías Porras Flores

Ayacucho, Agosto del 2016

**“APLICACIÓN WEB Y MÓVIL DE GESTIÓN DE COBRANZAS PARA LA RECUPERACIÓN DE CRÉDITOS MOROSOS EN LA CAC. “SANTA MARÍA MAGDALENA, 2016”**

**RECOMENDADO : 01 DE SETIEMBRE DEL 2016**

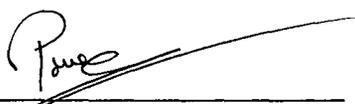
**APROBADO : 13 DE OCTUBRE DEL 2016**

  
MSc. CARLOS A. PRADO PRADO  
(Presidente)

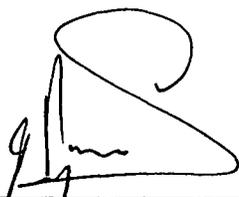
  
Ing. JUAN C. CARREÑO GAMARRA  
(Miembro)

  
Ing. MANUELA LAGOS BARZOLA  
(Miembro)

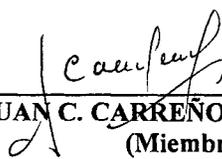
  
MSc. Ing. EFRAÍN E. PORRAS FLORES  
(Miembro)

  
Ing. JENNIFER R. PILLACA DE LA CRUZ  
(Secretaria Docente (e))

Según el acuerdo constatado en el Acta, levantada el 13 de octubre del 2016, en la Sustentación de Tesis Profesional presentado por el Bachiller en Ingeniería de Sistemas Sr. Darwin Martín HUAMÁN HUAMANÍ, con el Trabajo Titulado “APLICACIÓN WEB Y MÓVIL DE GESTIÓN DE COBRANZAS PARA LA RECUPERACIÓN DE CRÉDITOS MOROSOS EN LA CAC. “SANTA MARÍA MAGDALENA, 2016”, fue calificado con la nota de CATORCE (14) por lo que se da la respectiva APROBACIÓN.



MSc. CARLOS A. PRADO PRADO  
(Presidente)



Ing. JUAN C. CARREÑO GAMARRA  
(Miembro)



Ing. MANUEL A. LAGOS BARZOLA  
(Miembro)



MSc. Ing. EFRAÍN E. PORRAS FLORES  
(Miembro)



Ing. JENNIFER R. PILLACA DE LA CRUZ  
(Secretaria Docente (e))

## **DEDICATORIA**

La presente investigación  
está dedicada a mis padres y  
hermanos que en todo  
momento me apoyaron a  
pesar de las dificultades, y a  
dios por las bendiciones que  
me ofrece para seguir  
adelante.

## CONTENIDO

	Pág.
DEDICATORIA.....	i
CONTENIDO.....	ii
RESUMEN .....	iv
INTRODUCCIÓN .....	v
<b>CAPITULO I</b>	
<b>PLANTEAMIENTO DE LA INVESTIGACIÓN</b>	
1.1. DIAGNÓSTICO Y ENUNCIADO DEL PROBLEMA .....	1
1.2. DEFINICIÓN DEL PROBLEMA DE INVESTIGACIÓN .....	2
1.3. OBJETIVOS DE LA INVESTIGACIÓN.....	3
1.4. JUSTIFICACION Y DELIMITACION DE LA INVESTIGACIÓN.....	3
<b>CAPITULO II</b>	
<b>REVISIÓN DE LA LITERATURA</b>	
2.1. ANTECEDENTES DE LA INVESTIGACIÓN .....	5
2.2. MARCO TEÓRICO .....	6
2.2.1. GESTION DE COBRANZA .....	6
2.2.2. CREDITOS MOROSOS .....	11
2.2.3. MARCO LEGAL .....	14
2.2.4. PROGRAMACIÓN EXTREMA (XP) .....	15
2.2.5. PROGRAMACIÓN ORIENTADA A OBJETOS .....	23
2.2.6. SISTEMA GESTOR DE BASE DE DATOS RELACIONAL.....	25
2.2.7. APLICACION ANDROID .....	25
2.2.8. TECNOLOGIAS DE COMUNICACIÓN DE DISPOSITIVOS MOVILES .....	31
2.2.9. TECNOLOGIAS DE INTERNET .....	32
2.2.10. POBLACION Y MUESTRA.....	35
<b>CAPITULO III</b>	
<b>METODOLOGÍA DE LA INVESTIGACIÓN</b>	
3.1. TIPO DE INVESTIGACIÓN .....	37
3.2. DISEÑO DE LA INVESTIGACIÓN.....	37
3.3. POBLACION Y MUESTRA.....	37

3.4.	VARIABLES E INDICADORES.....	38
3.5.	OPERACIONALIZACIÓN DE LAS VARIABLES.....	40
3.6.	TECNICAS E INSTRUMENTOS.....	40

#### **CAPITULO IV**

##### **ANÁLISIS Y RESULTADOS DE LA INVESTIGACIÓN**

4.1.	ARTEFACTOS DEL SOFTWARE APLICANDO EL PROCESO XP.....	46
4.1.1.	FASE DE EXPLORACIÓN.....	46
4.1.2.	FASE DE PLANIFICACIÓN.....	50
4.1.3.	FASE DE ITERACIÓN.....	61

#### **CAPITULO V**

##### **CONCLUSIONES Y RECOMENDACIONES**

5.1.	CONCLUSIONES.....	111
5.2.	RECOMENDACIONES.....	112
	BIBLIOGRAFÍA.....	113
	ANEXO A.....	117
	ANEXO B.....	121
	ANEXO C.....	122

## **RESUMEN**

Las cooperativas de ahorro y créditos y en especial la CAC. “Santa María Magdalena” que se dedican a brindar créditos y que a través de estos créditos obtener algunas ganancias por medio de los intereses, se enfrentan al problema de los créditos morosos los cuales generan serios inconvenientes. Esto se debe en gran medida a una inadecuada gestión de cobranza, porque realizan el registro y control de las actividades de manera manual, desordenada e ineficiente, no existiendo ningún sistema portátil que ayude a los gestores de cobranzas a realizar mejor su trabajo, lo cual, provoca en muchos casos pérdida de información, desactualización del estado de las carteras, demora en la entrega de las notificaciones y una inadecuada gestión de los niveles de presión para el seguimiento y recuperación de los créditos morosos.

Esta investigación tiene como objetivo desarrollar una aplicación web y una aplicación móvil en Android, que automatizarán la gestión de cobranza, utilizando las herramientas de programación Netbeans para la aplicación web y Android Studio para la aplicación móvil y el lenguaje de programación java para ambas; además se implementaran servicios web que utilizan el estándar SOAP como mecanismo de comunicación con el lenguaje de programación C# para acceder a datos almacenados en un servidor de base de datos externo que en este caso es el SQL Server 2014, con la finalidad de recuperar los créditos morosos de la CAC. “Santa María Magdalena”.

Se utiliza la programación extrema para el desarrollo de las dos aplicaciones y las técnicas de entrevista al jefe de la oficina de recuperaciones, encuestas a los gestores de cobranzas y análisis documental de los reportes, normas y políticas de recuperación de créditos de CAC. “Santa María Magdalena” para el levantamiento de información.

Se obtuvo como resultado las aplicaciones web y móvil con sus respectivas funcionalidades e interfaces que automatizan la gestión de cobranza, proporcionando información necesaria de la cartera en cobranza, de las notificaciones de pago y de la clasificación del deudor.

### **Palabras claves**

Gestión de cobranza, Créditos Morosos, COOPAC, Aplicación Web, Aplicación Android, Proceso XP, Servicios Web, Java, C#, Netbeans, Android Studio, SQL Server 2008.

## INTRODUCCIÓN

La gestión de cobranza es el conjunto de acciones coordinadas y aplicadas a los socios para lograr la recuperación de los créditos morosos, de manera que los activos exigibles de la Cooperativa se conviertan en activos líquidos de la manera más rápida y eficiente, Brachfield (2004). El crédito moroso es aquel crédito que no ha sido cancelado oportunamente según el cronograma de pagos establecido al momento de ser otorgado, Susano (2014). La cartera en cobranza es la información de aquellos socios que mantienen deudas por pagar, Susano (2014). Las notificaciones de pago tienen por objeto comunicar a los deudores el interés de la cooperativa de recuperar los créditos vencidos y las acciones que estuviera por iniciar, Feban (s. f.). La clasificación de los deudores consiste en relacionar la voluntad de pago con la capacidad de pago de los deudores. Susano (2014).

La programación extrema es el más destacado de los procesos ágiles de desarrollo de software, se diferencia de las metodologías tradicionales en que pone más énfasis en la adaptabilidad que en la previsibilidad y que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable, Bautista (s. f.).

Actualmente no existe una aplicación que funcione en la plataforma Android o que pueda ser utilizado en dispositivos móviles el cual ayude a realizar mejor las actividades de la gestión de cobranza, que pueda incrementar el seguimiento de los clientes morosos para realizar sus pagos y ser usado como una herramienta por los gestores de cobranzas. La razón que motivó la realización de esta investigación es por la gran importancia que se les está dando cada día más a las aplicaciones para dispositivos móviles y por su gran potencial que estas tienen, además, por ser una tecnología nueva que está en constante desarrollo, también, para apoyar a las cooperativas y empresas del sistema financiero a ejecutar mejor las actividades de la gestión de cobranza.

Los objetivos específicos son: a) Explorar, planificar e iterar la información necesaria de la cartera en cobranza con la finalidad de recuperar los créditos morosos. b) Explorar, planificar e iterar las notificaciones de pago de los deudores a fin de recuperar los créditos morosos. c) Explorar, planificar e iterar la clasificación del deudor con la finalidad de recuperar los créditos morosos.

# **CAPITULO I**

## **PLANTEAMIENTO DE LA INVESTIGACIÓN**

### **1.1. DIAGNÓSTICO Y ENUNCIADO DEL PROBLEMA**

En la actualidad, las empresas del sistema financiero como las Cajas Rurales y las Cooperativas de ahorro y crédito que se dedican a brindar créditos y microcréditos a pequeñas y medianas empresas y que a través de estos créditos obtener algunas ganancias por medio del cobro de los intereses, saben que el tener portafolios de clientes con atrasos en el pago de sus cuotas es parte del negocio financiero pero el problema radica cuando este porcentaje sobrepasa los límites esperados.

El problema de los atrasos en los pagos de las cuotas por parte del socio o cliente, generalmente se debe a causas de apresuradas e incorrectas evaluaciones de créditos donde se podría verificar que el cliente tiene malas referencias o actitud negativa hacia el pago puntual, y también la aprobación del crédito que son influenciadas por la presión de alcanzar metas, y que también son basadas en la confianza o experiencia del analista sin el debido análisis de crédito, y sobre todo por una mala gestión de las actividades de cobranza que tiene por finalidad el cobro de un crédito cuando el deudor tiene pendiente una o varias cuotas del crédito otorgado y que es ejecutado por los gestores de cobranzas.

Los gestores de cobranza tienen estos problemas porque realizan el registro y control de las actividades de manera manual, desordenada e ineficiente, lo cual genera en muchos casos pérdida de información, desactualización del estado de las carteras, demora en la entrega de las notificaciones y una inadecuada gestión de los niveles de presión para el seguimiento y recuperación de los créditos moroso, además no se cuenta con un sistema de información portátil que ayude a los gestores de cobranzas a realizar su trabajo con mayor eficiencia, por lo que el resultado de los créditos brindados no es siempre exitoso ya que finalmente no termina recuperándose lo esperado como se acordó y se genera un desbalance en las cuentas que puede llevar a serios problemas a las empresas del sistema financiero.

Entre los diversos factores que explican la morosidad en los pagos de las obligaciones originadas por los créditos otorgados, se menciona el desconocimiento que tienen los acreedores de sus prestatarios. Esta falta de conocimiento del deudor, llevaría a los prestamistas a conceder créditos a quienes representan un alto potencial de riesgo, esto es enfrentar posibles fallas de recuperaciones e incluso la incobrabilidad de los préstamos, en otras palabras, no se trata sólo de averiguar sobre la capacidad de pago del negocio que recibe el crédito; es importante conocer la voluntad de pago de quienes son responsables ante los acreedores de la devolución del préstamo.

<b>AÑOS</b>	<b>INDICE DE MOROSIDAD</b>	<b>VARIACION</b>
2008	5.77%	-1.62%
2009	4.01%	-1.76%
2010	4.37%	0.36%
2011	6.14%	1.77%
2012	6.63%	0.49%
2013	6.29%	-0.34%
DIC. 2014	5.55%	-0.74%

Tabla N° 1.1: Índice de morosidad (CAC Santa María Magdalena, 2014)

Según la tabla N° 1.1 el índice de morosidad en la cooperativa Santa María Magdalena ha tenido un comportamiento variado a fluctuante teniendo como causa algunos factores externos e internos que han afectado este indicador. Es así que al cierre del mes de diciembre del 2014 la mora fue de 5.55% y la variación de -0.74%.

## **1.2. DEFINICIÓN DEL PROBLEMA DE INVESTIGACIÓN**

### **PROBLEMA GENERAL**

¿Cómo automatizar la gestión de cobranza para recuperar los créditos morosos en la “CAC. Santa María Magdalena”, 2016?

### **PROBLEMAS ESPECÍFICOS**

- a) ¿Qué información es necesaria de la cartera en cobranza para recuperar los créditos morosos?
- b) ¿De qué manera automatizar las notificaciones de pago de los deudores para recuperar los créditos morosos?
- c) ¿Cómo automatizar la clasificación del deudor para recuperar los créditos morosos?

### **1.3 OBJETIVOS DE LA INVESTIGACIÓN**

#### **OBJETIVO GENERAL**

Desarrollar una aplicación web y móvil para automatizar la gestión de cobranza, mediante técnicas e instrumentos, la programación extrema, un administrador de base de datos relacional, un lenguaje de programación orientado a objetos y tecnologías de internet, con la finalidad de recuperar los créditos morosos de la CAC. “Santa María Magdalena”, 2016.

#### **OBJETIVOS ESPECÍFICOS**

- a) Explorar, planificar e iterar la información necesaria de la cartera en cobranza con la finalidad de recuperar los créditos morosos.
- b) Explorar, planificar e iterar las notificaciones de pago de los deudores a fin de recuperar los créditos morosos.
- c) Explorar, planificar e iterar la clasificación del deudor con la finalidad de recuperar los créditos morosos.

### **1.4 JUSTIFICACIÓN Y DELIMITACIÓN DE LA INVESTIGACIÓN**

#### **1.4.1 IMPORTANCIA DEL TEMA**

##### **IMPORTANCIA TÉCNICA**

Se desarrollará una aplicación web y una móvil, que permitirá a los gestores de cobranzas consultar los datos de los deudores y la información de sus deudas en tiempo real desde cualquier lugar, y así evitar el uso de cuadernillos y agendas que puedan ser extraviadas, también se podrá tener un mejor control de las notificaciones de pago que se entregan a los deudores, y además se clasificará al deudor de acuerdo a su capacidad y voluntad de pago para poder elegir la mejor estrategia de cobro.

##### **IMPORTANCIA ECONÓMICA**

La aplicación móvil al poder mostrar la ubicación exacta de la dirección del deudor a través del mapa de la aplicación, el gestor de cobranza encontrara el domicilio del deudor con mayor rapidez, ahorrando tiempo y combustible si es que el gestor de cobranza utiliza algún vehículo motorizado, además se podrá realizar mayor cantidad de visitas domiciliarias y entrega de notificaciones por lo que la cooperativa prescindiría de personal de cobranza asignándoles otras tareas.

## **IMPORTANCIA SOCIAL**

Al saber las características emocionales de los deudores se puede aplicar estrategias de cobranza o asignar gestores de cobranzas de acuerdo a la personalidad de cada deudor, evitando crear rencores o intimidaciones, haciendo que el deudor se sienta en confianza y así mantener la buena disposición de los deudores para futuras negociaciones. Y además la cooperativa mantendría una buena imagen hacia la sociedad.

### **1.4.2 JUSTIFICACION**

La oficina de recuperaciones de la CAC. “Santa María Magdalena”, necesita herramientas tecnológicas que puedan brindar información confiable y oportuna en tiempo real y en cualquier lugar y momento de la cartera en cobranza, de la entrega de las notificaciones de pago y también poder clasificar a los deudores para así tener un mayor éxito en la recuperación de los créditos morosos.

### **1.4.3 DELIMITACIÓN**

La investigación se realizará solo para las actividades de la gestión de cobranza extrajudicial de la oficina de recuperaciones de la CAC. “Santa María Magdalena”, 2016.

## CAPITULO II

### REVISIÓN DE LA LITERATURA

#### 2.1. ANTECEDENTES DE LA INVESTIGACIÓN

Vera, (2011), en su investigación *Diseño de un sistema de gestión aplicado al área de crédito y cobranzas, para disminuir el riesgo de incobrabilidad de la cartera de una empresa comercial en la ciudad de Guayaquil*, Ecuador, se ha rediseñado la política y procedimientos de cobro a través del análisis de valor agregado de los procesos y antigüedad de la cartera; de modo que la propuesta de mejora de los procedimientos de cobranza garantizan la eficiencia y rapidez en el proceso de recuperación de cartera y disminuyen el periodo promedio de cobranza y las pérdidas por cuentas malas, manteniendo la buena disposición de los clientes para futuras negociaciones.

Galio, (2010), en su obra, *Sistema para la gestión y control de cobranzas a través de dispositivos móviles*, Guayaquil, Ecuador, manifiesta que la solución planteada es un instrumento de soporte a la toma de decisiones a la Gerencia de comercialización para personalizar la gestión de los clientes que estén al día, que tengan facturas por vencer, o que tengan n días de vencimiento para un control eficiente que permita reducir el índice de morosidad e incrementar su efectividad en la gestión de cobranza y por ende del retorno de la inversión y mejorar la razón circulante, además el producto se promocionara como una herramienta empresarial dedicada al control de la cobranza en todas las etapas e incluye el análisis de productividad y eficiencia de la gestión.

Vásquez (2013), en su investigación, *Análisis, Diseño e Implementación de un Sistema de recaudación de deudas*. Lima, Perú, se logró desarrollar un software capaz de efectuar el registro y control del seguimiento realizado durante el proceso de recaudación de deudas y que además brinda accesibilidad al usuario y permite la adición de futuras funciones gracias a la escalabilidad del mismo, Además de implementar un método que asigna los gestores de cobranza con más experiencia al seguimiento de las deudas más difíciles de cobrar gracias a la construcción de un algoritmo y la elaboración de una interface para realizar la carga masiva de la información de las carteras de deudas y desarrollar distintos tipos de reporte, los cuales reflejan información valiosa.

Calapiña, (2015), en su obra, *Sistema web para la gestión de procesos de crédito y recuperación de cartera*, Ambato, Ecuador, explica el desarrollo de un sistema para el seguimiento de créditos otorgados y créditos vencidos, que permite gestionar y manipular en la web la información de la Cooperativa de Ahorro y Crédito Santa Lucia Ltda., de la ciudad de Ambato, mediante tecnologías ASP.NET desarrolladas en Visual Studio Express 2012 y con herramientas Open Source, de esta manera mantener los procesos automatizados entre el personal administrativo y socios dentro de la institución financiera, permitiendo el acceso web desde cualquiera de las agencias.

## 2.2. MARCO TEÓRICO

### 2.2.1. GESTION DE COBRANZA

Brachfield (2004), manifiesta que la gestión de cobranza es el conjunto de acciones coordinadas y aplicadas a los clientes para lograr la recuperación de los créditos, de manera que los activos exigibles se conviertan en activos líquidos de la manera más rápida y eficiente posible, manteniendo en el proceso la buena disposición de los clientes para futuras negociaciones. La gestión de cobranza es un proceso bastante interactivo con los clientes, que parte del análisis de la situación del cliente, un oportuno y frecuente contacto, ofreciendo en el proceso de negociación alternativas de solución oportunas para cada caso y registrando las acciones ejecutadas para realizar un seguimiento continuo y el control del cumplimiento de los acuerdos negociados.

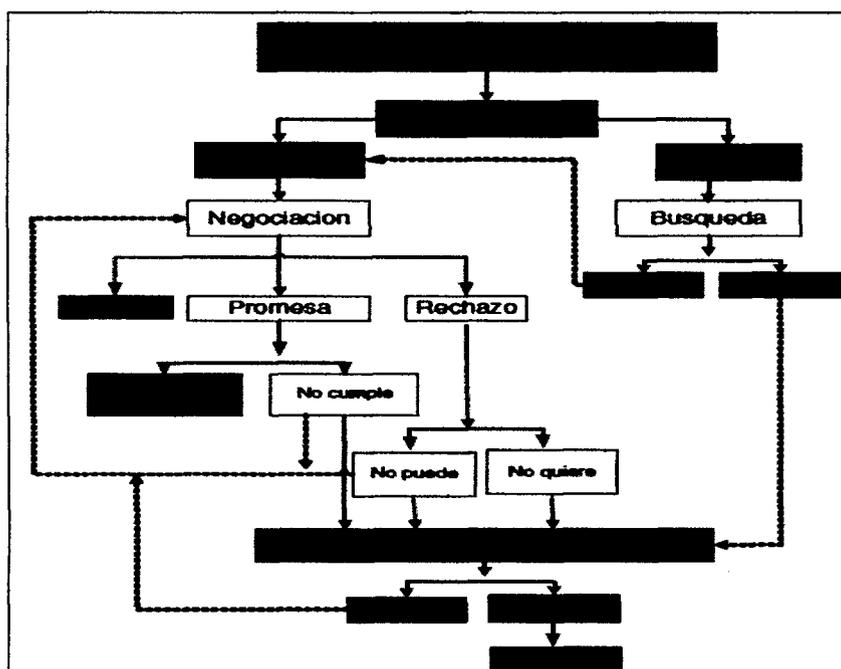


Figura 2.1: Acciones típicas en la gestión de cobranza. (Brachfield, 2004)

Mullo (2014) manifiesta que la cobranza son todos los tipos de gestión que involucran la recuperación de dinero producto de los créditos concedidos y que han sobrepasado la fecha de vencimiento de sus obligaciones. El seguimiento debe ser altamente dinámico y para ello debe contarse con medios eficientes de seguimiento no limitativos, sin restricciones y con la mayor libertad de funcionamiento posible. La finalidad del seguimiento es mantener un bajo nivel de morosidad que no afecte la rentabilidad de la cartera ni los niveles de liquidez de la institución. El proceso ideal de trabajo en recuperación es el siguiente:

- a) **Primer tramo: Cobranza Preventiva**, es la gestión que permite antes del vencimiento de una obligación recordar a los socios la fecha oportuna de pago de sus obligaciones, manejada ya sea por el call center o por el Oficial de Crédito, es importante en este tramo también utilizar servicios tecnológicos tales como la mensajería instantánea SMS.
- b) **Segundo tramo: Cobranza Administrativa**, generalmente hasta los 30 días dependiendo de la Cooperativa, y esta es de exclusiva responsabilidad del Oficial de Crédito.
- c) **Tercer tramo: Cobranza Extra Judicial**, generalmente manejada desde los 30 hasta los 90 días, esta cartera es manejada por el Oficial de Cobranzas en trabajo coordinado con el Oficial de Crédito. Es recomendable que en este tramo se realice ventas o encargos de cartera en mora.
- d) **Cuarto tramo: Cobranza Judicial**, manejada a partir de los 90 días, manejada por el Abogado de la Institución, al igual que la cartera extra judicial esta generalmente se vende o encarga su cobro a empresas especializadas en recuperación.

#### **A. CARTERA EN COBRANZA**

Según Susano (2014), se trata de la información de aquellos prestatarios que mantienen adeudos por pagar. De acuerdo con nuestro concepto de información de prestatarios, la cartera en cobranza es todavía un segmento de menor dimensión en número de clientes; ya que ésta se refiere exclusivamente a quienes tienen créditos por pagar, se encuentren al día o vencida. Una pregunta que a menudo surge cuando se analiza la información de la cartera en cobranza es ¿Cuál debería ser su contenido? La respuesta tiene un alcance mayor a la información proveniente del expediente de créditos. Este

último, sin duda, es la base para el acopio y análisis de la información para los gestores de recuperaciones; pero no es lo único que sirve a la recuperación del crédito. Los gestores de cobranza requieren de la información que muestre al crédito en funcionamiento. Esto es el vínculo que existe entre la capacidad y la voluntad de pagos del deudor. La información de la cartera en cobranza tiene que ser básicamente de "Inteligencia Comercial". Una concepción nueva en la gestión de información de prestatarios de los IF's. En este tipo de información, se presta atención a todo aquello que sirva para la recuperación de la cartera como: datos del negocio, de nuevas cargas financieras, del comportamiento de pagos del deudor, situación del mercado.

## B. NOTIFICACIONES DE PAGO

Para Susano (2014), una notificación, es aquella carta informativa en la que se induce al cliente o se le aconseja que efectúe un pago o que finalice el cobro que éste adeuda con la empresa o persona en cuestión. La notificación de pago es muy utilizado por aquellas empresas que tras reclamar un cobro vía telefónica, no obtienen ninguna respuesta y toman la iniciativa de hacerlo mediante un tipo de carta que en ocasiones es certificada. Dicha certificación o dicha carta notarial no tiene otro motivo que el de servir como prueba de cobro para el caso de que dicho cobro trascendiera y hubiera o hubiese que ir a un juicio.

	TIPO DE NOTIFICACION	PERSONA A QUIEN SE ENTREGA	DIAS	ENCARGADO DE LA COBRANZA
1	Aviso de vencimiento	Titular	0 – 8 días	Oficial de Negocios
2	Aviso de Cobranza	Titular	9 – 29 días	Oficial de Negocios
3	Requerimiento de Pago	Titular y Fiador	30 – 45 días	Oficial de Negocios
4	Notificación Urgente por falta de pago	Titular y Fiador	45 – 60 días	Gestor de Cobranza y Recuperaciones
5	Ultima Notificación	Titular y Fiador	61 – 80 días	Gestor de Cobranza y Recuperaciones
6	Carta notarial para crédito sin garantía mobiliaria	Titular y Fiador	81 a más días	Gestor de Cobranza y Recuperaciones
7	Carta notarial para crédito con hipoteca inmueble	Titular y Fiador	81 a más días	Gestor de Cobranza y Recuperaciones
8	Carta notarial para crédito con garantía mobiliaria vehicular	Titular y Fiador	81 a más días	Gestor de Cobranza y Recuperaciones

Tabla 2.1: Notificaciones de pago. Susano (2014)

Según Feban (s. f), el acto de la notificación de cartas o comunicaciones tiene por objeto poner en conocimiento de los deudores el interés de la cooperativa en el pago de sus créditos vencidos y las acciones que estuviera por iniciar a fin de recuperar sus acreencias. Otras notificaciones, son meramente cartas informativas a modo de factura en el que el cliente es informado de que se le ha sido cobrada cierta cantidad de dinero a través, por ejemplo, de una transferencia bancaria, en cuyo caso, antes de que la persona vea el cobro, se informa a la vez de que ha sido efectuado. Un ejemplo claro lo vemos en las cartas de cobro de las empresas de electricidad, en las que nos aparece la factura y adjunta una carta de cobro que nos informa del cobro efectuado a través del banco.

### **C. CLASIFICACION DEL DEUDOR**

Según Susano (2014), los recuperadores han logrado clasificar a los deudores relacionando: la voluntad de pago con la capacidad de pago de los prestatarios. Aplicándoles una escala diferenciada de alto a bajo. Alto (A) significa una situación buena o mejor, tanto para la voluntad como para la capacidad de pago. En cambio, bajo (B) expresa una situación contraria, que en el extremo es malo o peor. Esta escala ordinal sirve para ubicar al deudor en un plano de dos dimensiones relacionadas. La combinación de las escalas de voluntad y capacidad de pagos permite identificar hasta cuatro tipos de comportamiento del deudor frente a sus obligaciones:

Para Susano (2014), sin duda que, una vez identificado por esta vía el perfil del deudor, el gestor de cobranza estará en condiciones de ser efectivo en la recuperación de los préstamos. Esto es que el gestor diseñará la mejor estrategia y técnica para abordar al deudor, que le lleve a lograr la recuperación de los préstamos. Desde esta perspectiva, el recuperador basará su éxito en manejar su presión de cobro, incidiendo en la particular situación de la voluntad de pago frente a la capacidad de atender la deuda en su fecha.

Según Sofom (2010), la voluntad de pago se define como la capacidad moral del solicitante, medida por las condiciones de honestidad y responsabilidad personal. Por lo tanto, no es cuantificable en forma objetiva tan directamente como la capacidad de pago. Para tomar la decisión de otorgar el crédito se necesita evaluar la voluntad de pago del solicitante, para lo cual hay que considerar las condiciones tanto morales como técnicas del potencial deudor cuando se relacionan con el cumplimiento de las obligaciones contraídas, es decir su reputación personal y solvencia moral.

Para Sofom (2010), la capacidad de pago significa conocer concretamente si el solicitante, cuenta con ingresos suficientes para cubrir los pagos de las cuotas de un eventual crédito a conceder. La información necesaria para analizar la capacidad de pago de los solicitantes se recopila con los formularios de relevamiento de información de los cuales se obtiene una visión completa sobre la situación patrimonial, los flujos de ingresos y egresos del potencial acreditado y su capacidad de pago en función a su excedente y su endeudamiento patrimonial.

Para Mullo (2004), en la figura 4.2, se presentan las cuatro combinaciones. El punto aquí es que la voluntad de pago es el pivote de la clasificación. Allí se identificaría al deudor sano en la celda señala por la letra "A", de aquellos otros quienes representan dificultades para el cobro oportuno, denominados por las letras "B, C y D". En otras palabras, de una situación inmejorable establecida por la ubicación del deudor en la letra "A", se va deteriorando esta clasificación en grados menores de "B " a la "C", llegando a la peor clasificación la "D"; en la cual el deudor está en la categoría de incobrable.

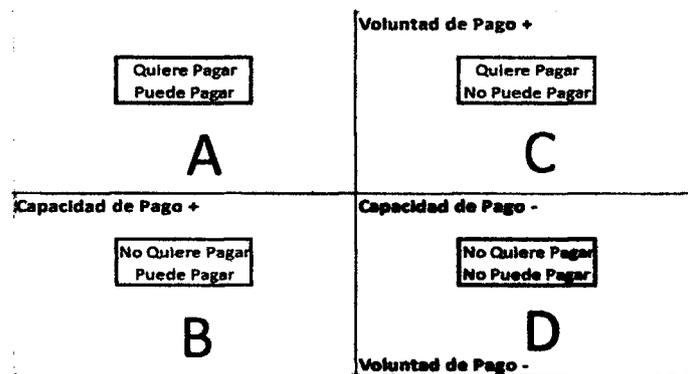


Figura 2.2: Ubicación y clasificación del deudor. (Mullo, 2004)

Según Susano (2014), a continuación se presentan los rasgos más resaltantes por cada uno de los cuatro tipos de prestatarios:

- a) El “deudor sano”, clasificación "A". Muestra una definida disciplina en sus pagos, lo cual influye en la adecuada asignación de fondos para cumplir con los acreedores. Tiene un orden en su gestión de obligaciones, motivado por su respeto al crédito. Su documentación y registros de la deuda se encuentran accesibles y puede hacer un seguimiento de la misma. En el plano financiero, calzan apropiadamente la rotación de cobros con el de los pagos, de tal manera que los vencimientos no le sorprenden. (Susano, 2014).

- b) El deudor "duro de cobrar", clasificación "B". Tiene una marcada indisciplina en sus pagos. Hay un comportamiento a exagerar el trabajar con fondos ajenos. Su actitud es hostil con sus acreedores. Siempre está dispuesto a dilatar los pagos, recurriendo a todo tipo de artimañas. Pretende rentabilizar al máximo todos los fondos disponibles, sin tomar en cuenta a los acreedores. En el lado financiero está en capacidad de atender las obligaciones. La documentación y registros de la deuda están al día, para tomar todas las ventajas disponibles para la dilación de los pagos. (Susano, 2014).
  
- c) El "deudor negligente", clasificación "C". A la vez que es indisciplinado en los pagos, se desenvuelve en gran desorden. Si bien es amigable y tiene actitudes por salvar el crédito, confronta serios problemas financieros para el pago oportuno. Sus registros son deficientes, los costos financieros se elevan por moras, castigos y penalidades. El descalce entre cobros y pagos es de tal magnitud que siempre confronta dificultades de liquidez. Los acreedores ejercen presión continua sobre este deudor y cobra quien es más audaz para hacerlo. (Susano, 2014).
  
- d) El "deudor problema", clasificación "D". Es indisciplinado en los pagos, actúa en desorden y muestra actitudes de irresponsabilidad frente a los acreedores. No tiene ningún interés por hacer frente a sus compromisos financieros. Está dispuesto a que le requieran los pagos judicialmente y pretende por esta vía, incluso eludir la cancelación del préstamo. Financieramente enfrenta alto riesgo de insolvencia y peor aún de quebranto. Muchas veces ha llegado a extremos de disponer de las garantías, contraviniendo cláusulas contractuales. No hay presión moral o social que lo haga cambiar. (Susano, 2014).

### **2.2.2. CREDITOS MOROSOS**

Según Charles (2003), un crédito moroso es aquel cuyo deudor no han cumplido con el pago de sus obligaciones a tiempo, por lo cual es considerado como moroso y requieren el inicio de actividades para contactarlo y lograr recuperar ese pago. Por consiguiente, un portafolio de créditos morosos para una empresa es un activo improductivo puesto que no se puede convertir en líquido fácilmente e implica un proceso engorroso y especializado, el cual no suele dar muchos resultados positivos debido a la elección de estrategias inapropiadas o falta de instrumentos pertinentes.

Para Susano (2014), son los créditos que no han sido cancelados oportunamente según el cronograma de pagos establecido al momento de otorgar el crédito. El sistema informático deberá estar preparado para la impresión y consulta en pantalla de listados de morosidad considerando diversos parámetros y niveles de detalle de forma que se pueda realizar un adecuado seguimiento y control de los créditos morosos. La problemática de los créditos morosos es común en pequeñas, medianas y grandes empresas, y su grado de afectación, a la tesorería y en consecuencia al normal funcionamiento económico de la misma, lo cual es tanto más grave en aquellas empresas de reducido tamaño y poca implantación.

#### **A. DEUDA**

Según Medina y Cubides (1999), para una persona natural o jurídica la deuda es la obligación de satisfacer una cuenta por pagar contraída a una parte acreedora, ello cuando el origen de la deuda es de carácter voluntario, es decir, que el deudor decidió libremente comprometerse al pago de dicha obligación. Dada esta situación, en la que el deudor no efectúe los pagos a su acreedor, éste podrá ejercer acciones legales contra el deudor para intentar recuperar la deuda correspondiente. Debemos hacer constar, aunque sea obvio, que si no hay ningún documento que sea capaz de demostrar la citada deuda, legalmente la prestación no existirá y no se podrá efectuar ningún tipo de acción legal contra el moroso.

Según Feban (2010), el socio que en virtud a dicha condición obtuvo el otorgamiento de un crédito a su favor por parte de la cooperativa, el cual se encuentra obligado a honrar su deuda en los términos en que asumió el compromiso.

Según Mullo (2014) existen cuatro tipos de deudores:

- a) Los que quieren pagar y pueden pagar, con ellos las estrategias de cobranza son más suaves, se necesita simplemente dialogar para llegar a un acuerdo de pago.
- b) Los que quieren pagar y no pueden pagar, con ellos necesitamos delinear una estrategia de renegociación de la deuda, es necesario re estructurar la operación crediticia.
- c) Los que no quieren pagar y pueden pagar, con ellos necesitamos directamente delinear estrategias judiciales o extra judiciales dependiendo el caso, estos son socios problemáticos que necesitan acciones fuertes y efectivas.

- d) Los que no quieren pagar y no pueden pagar, a este tipo de deudores no sirve realizar gestiones judiciales, ya que simplemente es perder el tiempo, con este tipo de socios lo que prima es el diálogo y crear acuerdos de pago parciales, el seguimiento a este tipo de socios es diario.

## **B. INTERES MORATORIO**

Según el BCRP (2003), La tasa de interés moratorio se determina por la libre competencia en el mercado financiero y es expresada en términos efectivos anuales por todo concepto. El interés moratorio se cobra sólo cuando se haya pactado y únicamente sobre el monto de la deuda correspondiente al capital impagado cuyo plazo esté vencido. El interés moratorio se computa y cobra a partir de la fecha en que el deudor incurre en mora, sin perjuicio del cobro del interés convencional compensatorio o legal pactado. Se cobra sobre el saldo del capital impagado. Por tanto, en los casos en que la devolución del préstamo se efectúe por cuotas, el cobro del interés moratorio procede únicamente sobre la parte correspondiente al capital de las cuotas vencidas e impagadas.

Para Bustamante (s. f.). Los intereses moratorios, se presumen de pleno derecho sufridos por el acreedor, por el solo hecho del retardo del deudor en la entrega de su capital; la mora es el retardo en la ejecución de la obligación, ya que no se cumple con la obligación en el tiempo prefijado. Y aunque la obligación se cumpla, resulta claro que el retardo ya se ha producido. Por lo que, el interés moratorio se genera por el cumplimiento tardío de las obligaciones, por lo que, tiene un carácter indemnizatorio. El interés “es moratorio cuando tiene por finalidad indemnizar la mora en el pago”, de lo que podemos inferir que el fin del interés surge ante el cumplimiento tardío de la obligación de devolver el capital.

## **C. DIAS DE ATRASO**

Según López (2002), son los días en los que socio deja de pagar su deuda y se cuenta a partir del día siguiente del vencimiento de su cuota. Los deudores están clasificados de acuerdo a los días de atraso. Para la S.B.S (2008), estos deudores deberán clasificarse conforme a los siguientes criterios de acuerdo a los días de atraso:

### **a) CATEGORÍA NORMAL (0)**

Son aquellos deudores que vienen cumpliendo con el pago de sus créditos de acuerdo a lo convenido o con un atraso de hasta ocho (8) días calendario.

- b) CATEGORÍA CON PROBLEMAS POTENCIALES (1)  
Son aquellos deudores que registran atraso en el pago de sus créditos de nueve (9) a treinta (30) días calendario.
- c) CATEGORÍA DEFICIENTE (2)  
Son aquellos deudores que registran atraso en el pago de sus créditos de treinta y uno (31) a sesenta (60) días calendario.
- d) CATEGORÍA DUDOSO (3)  
Son aquellos deudores que registran atraso en el pago de sus créditos de sesenta y uno (61) a ciento veinte (120) días calendario.
- e) CATEGORÍA PÉRDIDA (4)  
Son aquellos deudores que muestran atraso en el pago de sus créditos de más de ciento veinte (120) días calendario

### **2.2.3. MARCO LEGAL**

Según S.B.S (2008), la ley General del Sistema Financiero y del Sistema de Seguros y Orgánica de la Superintendencia de Banca y Seguros, Ley N° 26702 y sus modificatorias, en adelante Ley General, establece en su artículo 222° que, en la evaluación de las operaciones que integran la cartera crediticia deberá tenerse presente los flujos de caja del deudor, sus ingresos y capacidad de servicio de la deuda, situación financiera, patrimonio neto, proyectos futuros y otros factores relevantes para determinar la capacidad del servicio y pago de la deuda; señalando expresamente que el criterio básico es la capacidad de pago del deudor y que las garantías son subsidiarias.

La CAC “Santa María Magdalena”, Ltda. N° 219, en adelante “La Cooperativa”, se constituyó el día 09 de enero de 1963, quedando inscrita en los Registros Públicos de Ayacucho, con el nombre de COOPERATIVA DE AHORRO Y CREDITO “SANTA MARIA MAGDALENA” Ltda., 219 y en los Registros de las Cooperativas de la Superintendencia de Banca y Seguros con el N° 20, conforme a la Resolución SBS N° 720-93, de fecha 22 de diciembre de 1993. “La Cooperativa” se rige por la legislación cooperativa vigente en el Perú, las disposiciones de la Superintendencia de Banca y Seguros, de la Federación Nacional de Cooperativas de Ahorro y Crédito del Perú, el presente Estatuto y sus Reglamentos. Los casos no previstos por las indicadas normas se reirán por los Principios Generales de Cooperativismo. (CAC. S. M. M., 2006).

#### **2.2.4. PROGRAMACIÓN EXTREMA (XP)**

Según Bautista (s. f.), la programación extrema es un enfoque de la ingeniería de software. Es el más destacado de los procesos ágiles de desarrollo de software. Al igual que éstos, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos.

Letelier y Penandés (2004), manifiestan que XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

##### **A. CICLO DE VIDA DEL SOFTWARE EN XP**

Según Fernández (2002), la metodología XP define cuatro variables para cualquier proyecto de software: costo, tiempo, calidad y alcance, y sólo tres de ellas podrán ser fijadas arbitrariamente por actores externos al grupo de desarrolladores (clientes y jefes de proyecto). El valor de la variable restante podrá ser establecido por el equipo de desarrollo, en función de los valores de las otras tres. Este mecanismo indica que, por ejemplo, si el cliente establece el alcance y la calidad, y el jefe de proyecto el precio, el grupo de desarrollo tendrá libertad para determinar el tiempo que durará el proyecto. Como se detalló en los apartados anteriores, los ciclos de vida “tradicionales” proponen una clara distinción entre las etapas del proyecto de software, y tienen un plan bien preestablecido acerca del proceso de desarrollo. Asimismo, en todos ellos se parte de especificaciones claras, si no del total del proyecto, por lo menos de una buena parte inicial. El ciclo de vida de un proyecto XP incluye, al igual que las otras metodologías, entender lo que el cliente necesita, estimar el esfuerzo, crear la solución y entregar el

producto final al cliente. Sin embargo, XP propone un ciclo de vida dinámico, donde se admite expresamente que, en muchos casos, los clientes no son capaces de especificar sus requerimientos al comienzo de un proyecto. Por esto, se trata de realizar ciclos de desarrollo cortos (llamados iteraciones), con entregables funcionales al finalizar cada ciclo. En cada iteración se realiza un ciclo completo de análisis, diseño, desarrollo y pruebas, pero utilizando un conjunto de reglas y prácticas que caracterizan a XP. Típicamente un proyecto con XP lleva 10 a 15 ciclos o iteraciones. La siguiente figura esquematiza los ciclos de desarrollo en cascada e iterativos tradicionales (por ejemplo, incremental o espiral), comparados con el de XP.

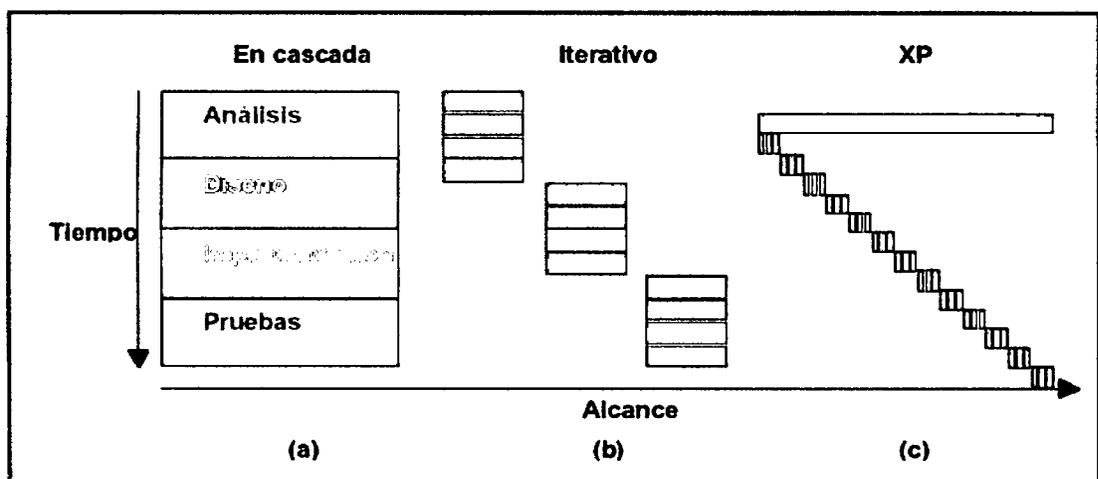


Figura N° 2.3: Evolución de los largos ciclos de desarrollo en cascada (a) a ciclos más cortos (b) y a la mezcla que hace XP (c) (Fernández, 2002)

## B. FASES DEL MODELO XP

Según Baird (2002), si bien el ciclo de vida de un proyecto XP es muy dinámico, se puede separar en fases. Varios de los detalles acerca de las tareas de estas fases se detallan más adelante, en la sección “Reglas y Practicas”:

- a) **Fase de exploración.-** Es la fase en la que se define el alcance general del proyecto. En esta fase, el cliente define lo que necesita mediante la redacción de sencillas “historias de usuarios”. Los programadores estiman los tiempos de desarrollo en base a esta información. Debe quedar claro que las estimaciones realizadas en esta fase son primarias (ya que estarán basadas en datos de muy alto nivel), y podrían variar cuando se analicen más en detalle en cada iteración. Esta fase dura típicamente un par de semanas, y el resultado es una visión general del sistema, y un plazo total estimado. (Baird, 2002).

- b) **Fase de planificación.-** La planificación es una fase corta, en la que el cliente, los gerentes y el grupo de desarrolladores acuerdan el orden en que deberán implementarse las historias de usuario, y, asociadas a éstas, las entregas. Típicamente esta fase consiste en una o varias reuniones grupales de planificación. El resultado de esta fase es un Plan de Entregas, como se detallará en la sección “Reglas y Practicas”. (Baird, 2002).
- c) **Fase de iteraciones.-** Esta es la fase principal en el ciclo de desarrollo de XP. Las funcionalidades son desarrolladas en esta fase, generando al final de cada una un entregable funcional que implementa las historias de usuario asignadas a la iteración. Como las historias de usuario no tienen suficiente detalle como para permitir su análisis y desarrollo, al principio de cada iteración se realizan las tareas necesarias de análisis, recabando con el cliente todos los datos que sean necesarios donde el cliente participa activamente. (Baird, 2002).
- d) **Fase de puesta en producción.-** Si bien al final de cada iteración se entregan módulos funcionales y sin errores, puede ser deseable por parte del cliente no poner el sistema en producción hasta tanto no se tenga la funcionalidad completa. En esta fase no se realizan más desarrollos funcionales, pero pueden ser necesarias tareas de ajuste. (Baird, 2002).

## C. REGLAS Y PRACTICAS

Para Wells (2006), la metodología XP tiene un conjunto importante de reglas y prácticas. En forma genérica, se pueden agrupar en: Reglas y prácticas para la Planificación, para el diseño, para el desarrollo y para las pruebas.

### C.1. PLANIFICACIÓN

Fernández (2002), XP plantea la planificación como un permanente diálogo entre la parte empresarial y técnica del proyecto, en la que los primeros decidirán el alcance, la prioridad, la composición de las versiones y la fecha de las mismas. En cuanto a los técnicos, son los responsables de estimar la duración requerida para implementar las funcionalidades deseadas por el cliente, de informar sobre las consecuencias de determinadas decisiones, de organizar la cultura de trabajo y, finalmente, de realizar la planificación detallada dentro de cada versión. XP no es sólo un método centrado en el

código, sino que es un método de gestión de proyectos software. Los conceptos básicos de esta planificación son los siguientes:

**Historias de usuarios.-** Las “Historias de usuarios” sustituyen a los documentos de especificación funcional, y a los “casos de uso”. Estas “historias” son escritas por el cliente, en su propio lenguaje, como descripciones cortas de lo que el sistema debe realizar. La diferencia más importante entre estas historias y los tradicionales documentos de especificación funcional se encuentra en el nivel de detalle requerido. Las historias de usuario deben tener el detalle mínimo como para que los programadores puedan realizar una estimación poco riesgosa del tiempo que llevará su desarrollo. Cuando llegue el momento de la implementación, los desarrolladores dialogarán directamente con el cliente para obtener todos los detalles necesarios. (Wesley, 2001).

Para Fernández (2002), las historias de usuarios tienen el mismo propósito que los casos de uso, pero no son lo mismo. Las escriben los propios clientes, tal y como ven ellos las necesidades del sistema. Por tanto serán descripciones cortas y escritas en el lenguaje del usuario, sin terminología técnica.

**Plan de entregas.-** El cronograma de entregas establece qué historias de usuario serán agrupadas para conformar una entrega, y el orden de las mismas. Este cronograma será el resultado de una reunión entre todos los actores del proyecto (cliente, desarrolladores, gerentes, etc.). XP denomina a esta reunión “Juego de planeamiento”, pero puede denominarse de la manera que sea más apropiada al tipo de empresa y cliente (por ejemplo, Reunión de planeamiento, típicamente el cliente ordenará y agrupará según sus prioridades las historias de usuario. El cronograma de entregas se realiza en base a las estimaciones de tiempos de desarrollo realizadas por los desarrolladores. Luego de algunas iteraciones es recomendable realizar nuevamente una reunión con los actores del proyecto, para evaluar nuevamente el plan de entregas y ajustarlo si es necesario. (Wesley, 2001).

**Plan de iteraciones.-** Las historias de usuarios seleccionadas para cada entrega son desarrolladas y probadas en un ciclo de iteración, de acuerdo al orden preestablecido. Al comienzo de cada ciclo, se realiza una reunión de planificación de la iteración. Cada historia de usuario se traduce en tareas específicas de programación. Asimismo, para cada

historia de usuario se establecen las pruebas de aceptación. Estas pruebas se realizan al final del ciclo en el que se desarrollan, pero también al final de cada uno de los ciclos siguientes, para verificar que subsiguientes iteraciones no han afectado a las anteriores. Las pruebas de aceptación que hayan fallado en el ciclo anterior son analizadas para evaluar su corrección, así como para prever que no vuelvan a ocurrir. (Wesley, 2001).

**Reuniones diarias de seguimiento.**- El objetivo de tener reuniones diarias es mantener la comunicación entre el equipo, y compartir problemas y soluciones. En la mayoría de estas reuniones, gran parte de los participantes simplemente escuchan, sin tener mucho que aportar. Para no quitar tiempo innecesario del equipo, se sugiere realizar estas reuniones en círculo y de pie. (Wesley, 2001).

## C.2. DISEÑO

La metodología XP hace especial énfasis en los diseños simples y claros. Los conceptos más importantes de diseño son los siguientes (Fernández 2002):

**Simplicidad.**- Siempre costará menos tiempo de implementar un diseño sencillo que uno complejo. Por lo que, trataremos siempre de realizar las cosas de la manera más sencilla posible. Si alguna parte de la implementación resulta especialmente compleja, deberías replantearla (divide y vencerás). Así, cualquier cambio y modificación será mucho más sencillo. En ocasiones, realizar un diseño sencillo puede resultar una tarea especialmente difícil. (Fernández 2002).

**Metáfora.**- Una metáfora para el sistema es una historia que todo el mundo puede contar acerca de cómo el sistema funciona. La tarea de elegir una metáfora para el sistema nos permitirá mantener la coherencia de nombres de todo aquello que se va a implementar. El nombre de los objetos o partes de nuestro sistema es muy importante. La tarea de “poner nombre”, sencilla a simple vista, no lo es tanto. Debemos elegir un sistema de nombres que permita que cualquiera que lo vea adivine la relación entre el objeto y aquello que representa. (Beck, 2000).

**Tarjetas CRC (Cargo o clase, Responsabilidad y Colaboración).**- Para poder diseñar el sistema como un equipo deberemos cumplir con tres principios: Cargo o Clase, Responsabilidad y Colaboración (CRC). Las tarjetas CRC permitirán desprendernos del

método de trabajo basado en procedimientos y trabajar con una metodología basada en objetos. Las tarjetas CRC permiten que el equipo completo contribuya en la tarea del diseño. Una tarjeta CRC representa un objeto. El nombre de la clase se coloca a modo de título en la tarjeta, las responsabilidades se colocan a la izquierda, y las clases que se implican en cada responsabilidad a la derecha. (Fernández 2002).

**Soluciones “spike”.**- Cuando aparecen problemas técnicos, o cuando es difícil de estimar el tiempo para implementar una historia de usuario, pueden utilizarse pequeños programas de prueba (llamados “spike”<sup>1</sup>), para explorar diferentes soluciones. Estos programas son únicamente para probar o evaluar una solución, y suelen ser desechados luego de su evaluación. (Joskowicz, 2008).

Un programa Spike, es un programa muy simple que explora una posible solución al problema. A partir de estos pequeños programas iremos construyendo la solución a nuestro problema. El sistema se pone por primera vez en producción en, a lo sumo, unos pocos meses, antes de estar completamente terminado. Las sucesivas versiones serán más frecuentes –entre un día y un mes–. (Fernández 2002).

**Recodificación.**- La recodificación consiste en escribir nuevamente parte del código de un programa, sin cambiar su funcionalidad, a los efectos de hacerlo más simple, conciso y/o entendible. Muchas veces, al terminar de escribir un código de programa, pensamos que, si lo comenzáramos de nuevo, lo hubiéramos hecho en forma diferente, más clara y eficiente. Sin embargo, como ya está pronto y “funciona”, rara vez es reescrito. XP sugieren recodificar cada vez que sea necesario. Si bien, puede parecer una pérdida de tiempo innecesaria en el plazo inmediato, los resultados de ésta práctica tienen sus frutos en las siguientes iteraciones, cuando sea necesario ampliar o cambiar la funcionalidad. La filosofía que se persigue es, como ya se mencionó, tratar de mantener el código más simple posible. (Joskowicz, 2008).

### **C.3. DESARROLLO**

Esta etapa debe reunir las siguientes características o cualidades:

**Disponibilidad del cliente.**- Según Fernández (2002), una de las pocas condiciones que impone la metodología XP es tener al usuario siempre disponible. No sólo para ayudar al

equipo de desarrollo, sino formando parte de él. Todas las fases que se realizan en un proyecto XP requieren de comunicación con el usuario, preferiblemente cara a cara, en persona, sin intermediarios. Durante la reunión del plan de entregas, el usuario propondrá qué historia de usuario se incluye en cada plan. También se negociarán los plazos de entrega. El usuario o cliente tomará las decisiones que le afecten para alcanzar los objetivos de su negocio. También es necesario que el cliente colabore en la realización de los test. Estos test comprobarán que el sistema está listo para pasar a la fase de producción.

**Uso de estándares.-** Para Fernández (2002), si bien esto no es una idea nueva, XP promueve la programación basada en estándares, de manera que sea fácilmente entendible por todo el equipo, y que facilite la recodificación. Para Joskowicz (2008), el código ha de ser desarrollado siguiendo los estándares de desarrollo para facilitar su lectura y modificación por cualquier miembro del equipo de desarrollo. Es decisiva, para poder plantear con éxito la propiedad colectiva del código. Ésta sería impensable sin una codificación basada en estándares que haga que todo el mundo se sienta cómodo con el código escrito por cualquier otro miembro del equipo.

**Programación dirigida por pruebas.-** Según Joskowicz (2008), en las metodologías tradicionales, la fase de pruebas, incluyendo la definición de los test, es usualmente realizada sobre el final del proyecto, o sobre el final del desarrollo de cada módulo. La metodología XP propone un modelo inverso, en el que, lo primero que se escribe son los test que el sistema debe pasar. Luego, el desarrollo debe ser el mínimo necesario para pasar las pruebas previamente definidas. Las pruebas a los que se refieren esta práctica, son las pruebas unitarias, realizados por los desarrolladores. La definición de estos test al comienzo, condiciona o “dirige” el desarrollo.

**Programación en pares.-** Fernández (2002), manifiesta que todo el código que formará parte del plan, será desarrollado por dos personas que trabajarán de forma conjunta en un ordenador. De esta manera, se incrementará la calidad del software desarrollado sin afectar al tiempo de entrega. Partimos de la base de que este equipo de dos personas posee unos conocimientos similares en cuanto a la tarea que van a realizar, es decir, están aproximadamente al mismo nivel. Mientras uno de ellos se encarga de pensar la táctica con la que se va a abordar el problema, el otro se encargará de pensar las estrategias que permiten llevar dichas tácticas a su máximo exponente. Ambos roles son intercambiables.

**Integraciones permanentes.-** Según Joskowicz (2008), todos los desarrolladores necesitan trabajar siempre con la “última versión”. Realizar cambios o mejoras sobre versiones antiguas causan graves problemas, y retrasan al proyecto. Es por eso que XP promueve publicar lo antes posible las nuevas versiones, aunque no sean las últimas, siempre que estén libres de errores. Idealmente, todos los días deben existir nuevas versiones publicadas. Para evitar errores, solo una pareja de desarrolladores puede integrar su código a la vez.

**Propiedad colectiva del código.-** Según Fernández (2002), esta filosofía nos permite que cualquiera contribuya al desarrollo de cualquier parte del proyecto. Cualquier programador podrá cambiar una línea de código para añadir funcionalidad o eliminar algún fallo. Las personas dejan de ser un cuello de botella, en cuanto a la programación. Cualquiera puede realizar un cambio en el mismo siempre y cuando beneficie a la arquitectura del mismo. La responsabilidad del funcionamiento recaerá sobre el equipo al completo.

**Ritmo sostenido.-** Según Joskowicz (2008), la metodología XP indica que debe llevarse un ritmo sostenido de trabajo. Anteriormente, ésta práctica se denominaba “Semana de 40 horas”. Sin embargo, lo importante no es si se trabajan, 35, 40 o 42 horas por semana. El concepto que se desea establecer con esta práctica es el de planificar el trabajo de manera de mantener un ritmo constante y razonable, sin sobrecargar al equipo. Cuando un proyecto se retrasa, trabajar tiempo extra puede ser más perjudicial que beneficioso. El trabajo extra desmotiva inmediatamente al grupo e impacta en la calidad del producto. En la medida de lo posible, se debería renegociar el plan de entregas.

#### **C.4 PRUEBAS**

**Pruebas unitarias.-** Joskowicz (2008), manifiesta que las pruebas unitarias son una de las piedras angulares de XP. Todos los módulos deben de pasar las pruebas unitarias antes de ser liberados o publicados. Por otra parte, como se mencionó anteriormente, las pruebas deben ser definidas antes de realizar el código. Que todo código liberado pase correctamente las pruebas unitarias es lo que habilita que funcione la propiedad colectiva del código. En este sentido, el sistema y el conjunto de pruebas deben ser guardados junto con el código, para que pueda ser utilizado por otros desarrolladores, en caso de tener que corregir, cambiar o recodificar parte del mismo.

**Pruebas de aceptación.-** Joskowicz (2008), manifiesta que las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada. Las pruebas de aceptación son consideradas como “pruebas de caja negra”. Los clientes son responsables de verificar que los resultados de estas pruebas sean correctos. Asimismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución. Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. Dado que la responsabilidad es grupal, es recomendable publicar los resultados de las pruebas de aceptación, de manera que todo el equipo esté al tanto de esta información.

**Detección y corrección de errores.-** Para Joskowicz (2008), cuando se encuentra un error (“bug”), éste debe ser corregido inmediatamente, y se deben tener precauciones para que errores similares no vuelvan a ocurrir. Asimismo, se generan nuevas pruebas para verificar que el error haya sido resuelto.

### 2.2.5. PROGRAMACIÓN ORIENTADA A OBJETOS

Para Joyanes (1998), la programación orientada a objetos, abreviada POO, es un paradigma de programación que usa objetos para representar los datos (atributos en POO) y procedimientos (métodos en POO) de un programa. Se trata de un paradigma que tiene como objetivo permitir el diseño de sistemas de software modulares y reutilizables. El paradigma orientado a objetos está basado en conceptos como la programación estructurada y los tipos abstractos de datos. Un objeto es un tipo abstracto de dato que tiene como agregados el polimorfismo y la herencia. En la programación estructura se hace una separación entre código y datos, cuando en la programación orientada a objetos ambos conceptos se unen para formar un objeto, el cual tiene un estado (atributos) y un comportamiento (métodos).

Algunos conceptos relacionados de programación orientada a objetos son (Joyanes, 1998):

- a) **Clase.-** Definición de los atributos y el comportamiento de un tipo de objeto.
- b) **Instancia.-** Se trata de la creación de un objeto a partir de una clase.

- c) **Atributos.-** Son las características del objeto (datos) que se definen en la clase.
- d) **Métodos.-** Son subrutinas asociadas a un objeto y que se definen en la clase para describir el comportamiento del objeto.
- e) **Abstracción.-** Es la definición de un objeto en base a sus características esenciales (atributos y comportamiento). La abstracción permite modelar un problema en base a un conjunto de clases.
- f) **Modularidad.-** Al crear clases, el software se separa naturalmente en módulos, que pueden ser utilizados como parte de varios programas.
- g) **Mensajes.-** Es la comunicación hacia un objeto que le indica que ejecute un método con los parámetros indicados en el mensaje.

Según Ceballos (1993), las características básicas de un lenguaje de programación orientado a objetos son: encapsulamiento, herencia y polimorfismo.

**Encapsulamiento.-** Para Ceballos (1993), el encapsulamiento se refiere a agregar dentro de un objeto todos los recursos necesarios para que el objeto funcione, esto es, agregar los métodos y los atributos que requiere. La idea del encapsulamiento es que el código no inflencie en otro código, ni sea influenciado por otro código. Las clases permiten que los datos y los métodos estén estrechamente ligados, formando una “cápsula” congruente con el objeto que se está definiendo.

**Herencia.-** Según Ceballos (1993), la herencia es una de las características más poderosas de la programación orientada a objetos, pues fomenta la reutilización del código. La herencia se refiere a que se pueden crear clases que son hijas de otra clase. A la clase padre se le llama superclase, la cual contiene atributos y métodos comunes para todas las clases hijas. Las clases hijas se llaman subclasses y son las que heredan estos elementos comunes de las superclases, pero contienen también atributos y métodos propios.

**Polimorfismo.-** Ceballos (1993), manifiesta que el polimorfismo está muy ligado a la herencia. El polimorfismo se refiere a que dentro del código se puede tratar a todas las subclasses como si se tratara de su superclase. Esto simplifica mucho la programación y la hace más clara.

- c) **Atributos.-** Son las características del objeto (datos) que se definen en la clase.
- d) **Métodos.-** Son subrutinas asociadas a un objeto y que se definen en la clase para describir el comportamiento del objeto.
- e) **Abstracción.-** Es la definición de un objeto en base a sus características esenciales (atributos y comportamiento). La abstracción permite modelar un problema en base a un conjunto de clases.
- f) **Modularidad.-** Al crear clases, el software se separa naturalmente en módulos, que pueden ser utilizados como parte de varios programas.
- g) **Mensajes.-** Es la comunicación hacia un objeto que le indica que ejecute un método con los parámetros indicados en el mensaje.

Según Ceballos (1993), las características básicas de un lenguaje de programación orientado a objetos son: encapsulamiento, herencia y polimorfismo.

**Encapsulamiento.-** Para Ceballos (1993), el encapsulamiento se refiere a agregar dentro de un objeto todos los recursos necesarios para que el objeto funcione, esto es, agregar los métodos y los atributos que requiere. La idea del encapsulamiento es que el código no influya en otro código, ni sea influenciado por otro código. Las clases permiten que los datos y los métodos estén estrechamente ligados, formando una “cápsula” congruente con el objeto que se está definiendo.

**Herencia.-** Según Ceballos (1993), la herencia es una de las características más poderosas de la programación orientada a objetos, pues fomenta la reutilización del código. La herencia se refiere a que se pueden crear clases que son hijas de otra clase. A la clase padre se le llama superclase, la cual contiene atributos y métodos comunes para todas las clases hijas. Las clases hijas se llaman subclasses y son las que heredan estos elementos comunes de las superclases, pero contienen también atributos y métodos propios.

**Polimorfismo.-** Ceballos (1993), manifiesta que el polimorfismo está muy ligado a la herencia. El polimorfismo se refiere a que dentro del código se puede tratar a todas las subclasses como si se tratara de su superclase. Esto simplifica mucho la programación y la hace más clara.

## **2.2.6. SISTEMA GESTOR DE BASE DE DATOS RELACIONAL**

Una base de datos es realmente un conjunto de objetos, de entidades. Estos objetos controlan la forma en la que se almacenan y gestionan los datos. En una base de datos relacional la unidad básica de almacenamiento es la tabla. Cada entidad dispone de unos atributos (o propiedades) que definen el tipo de datos que se almacenan en esa entidad. Un SGBD (Sistema de Gestión de Base de Datos) es un conjunto de programas que va a permitir insertar, modificar, borrar y buscar eficazmente datos específicos entre un volumen masivo de información compartida por todos los usuarios de la Base; pero también es una herramienta que va a permitir ordenar, buscar, reordenar y convertir datos (Gómez, Romera, Fraile, Argente y Alfaro, 1993).

Según Ruiz, Gómez, López y Cerruela (2002), un SGBD es una colección de programas de aplicación que proporciona al usuario de la base de datos los medios necesarios para realizar las siguientes tareas; Definición de los datos a los distintos niveles de abstracción, manipulación de los datos en la base de datos, es decir, la inserción, modificación, borrado y acceso o consulta a los mismo, Mantenimiento de la integridad de la base de datos, integridad en cuanto a los datos en sí, sus valores y las relaciones entre ellos, Control de la privacidad y seguridad de los datos en la base de datos.

## **2.2.7. APLICACION ANDROID**

Android constituye una pila de software pensada especialmente para dispositivos móviles y que incluye tanto un sistema operativo, como middleware y diversas aplicaciones de usuario. Representa la primera incursión seria de Google en el mercado móvil y nace con la pretensión de extender su filosofía a dicho sector. Todas las aplicaciones para Android se programan en lenguaje Java y son ejecutadas en una máquina virtual especialmente diseñada para esta plataforma, que ha sido bautizada con el nombre de Dalvik. El núcleo de Android está basado en Linux 2.6. (Android, 2008).

Cuando en noviembre de 2007 Google anunció su irrupción en el mundo de la telefonía móvil a través de Android, muchos medios especializados catalogaron este novedoso producto como un nuevo sistema operativo, libre y específico para teléfonos móviles. Sin embargo, los responsables del proyecto se han esforzado desde entonces en destacar que la motivación de Android es convertirse en algo más que un simple sistema operativo. Con Android se busca reunir en una misma plataforma todos los elementos necesarios que

permitan al desarrollador controlar y aprovechar al máximo cualquier funcionalidad ofrecida por un dispositivo móvil (llamadas, mensajes de texto, cámara, agenda de contactos, conexión Wi-Fi, Bluetooth, aplicaciones ofimáticas, videojuegos, etc.), así como poder crear aplicaciones que sean verdaderamente portables, reutilizables y de rápido desarrollo. (El Navegante, 2007).

## **A. ARQUITECTURA**

Como ya se ha mencionado, Android es una plataforma para dispositivos móviles que contiene una pila de software donde se incluye un sistema operativo, middleware y aplicaciones básicas para el usuario. Su diseño cuenta, entre otras, con las siguientes características. (Google, 2008):

- a) Busca el desarrollo rápido de aplicaciones, que sean reutilizables y verdaderamente portables entre diferentes dispositivos.
- b) Cuenta con su propia máquina virtual, Dalvik, que interpreta y ejecuta código escrito en Java.
- c) Permite la representación de gráficos 2D y 3D.
- d) Posibilita el uso de bases de datos.
- e) Soporta un elevado número de formatos multimedia.
- f) Servicio de localización GSM.
- g) Controla los diferentes elementos hardware: Bluetooth, Wi-Fi, cámara fotográfica o de vídeo, GPS, acelerómetro, infrarrojos, etc.,
- h) Cuenta con un entorno de desarrollo muy cuidado mediante un SDK disponible de forma gratuita.
- i) Ofrece un plug-in para uno de los entornos de desarrollo más popular, Eclipse, y un emulador integrado para ejecutar las aplicaciones.

En las siguientes líneas se dará una visión global por capas de cuál es la arquitectura empleada en Android. Cada capa utiliza servicios ofrecidos por las anteriores, y ofrece a su vez los suyos propios a las capas de niveles superiores. (Google, 2008).

La capa más inmediata es la que corresponde al núcleo de Android. Android utiliza el núcleo de Linux 2.6 como una capa de abstracción para el hardware disponible en los dispositivos móviles. Esta capa contiene los drivers necesarios para que cualquier componente hardware

pueda ser utilizado mediante las llamadas correspondientes. La elección de Linux 2.6 se ha debido principalmente a dos razones: la primera, su naturaleza de código abierto y libre se ajusta al tipo de distribución que se buscaba para Android, la segunda es que este kernel de Linux incluye de por sí numerosos drivers, además de contemplar la gestión de memoria, gestión de procesos, módulos de seguridad, comunicación en red y otras muchas responsabilidades propias de un sistemas operativo.

La siguiente capa se corresponde con las librerías utilizadas por Android. Éstas han sido escritas utilizando C/C++ y proporcionan a Android la mayor parte de sus capacidades más características. Junto al núcleo basado en Linux, estas librerías constituyen el corazón de Android. Al mismo nivel que las librerías de Android se sitúa el entorno de ejecución. Éste lo constituyen las Core Libraries, que son librerías con multitud de clases de Java, y la máquina virtual Dalvik. Los dos últimos niveles de la arquitectura de Android están escritos enteramente en Java. El framework de aplicaciones representa fundamentalmente el conjunto de herramientas de desarrollo de cualquier aplicación. Toda aplicación que se desarrolle para Android, ya sean las propias del dispositivo, las desarrolladas por Google o terceras compañías, o incluso las que el propio usuario cree, utilizan el mismo conjunto de API y el mismo framework, representado por este nivel. (Google, 2008).

Entre las API más importantes ubicadas aquí, se pueden encontrar las siguientes (Google, 2008):

- a) Activity Manager, importante conjunto de API que gestiona el ciclo de vida de las aplicaciones en Android (del que se hablará más adelante).
- b) Window Manager, gestiona las ventanas de las aplicaciones y utiliza la librería ya vista Surface Manager.
- c) Telephone Manager, incluye todas las API vinculadas a las funcionalidades propias del teléfono (llamadas, mensajes, etc.)
- d) Content Providers, permite a cualquier aplicación compartir sus datos con las demás aplicaciones de Android. Por ejemplo, gracias a esta API la información de contactos, agenda, mensajes, etc. será accesible para otras aplicaciones.
- e) View System, proporciona un gran número de elementos para poder construir interfaces de usuario (GUI), como listas, mosaicos, botones, check-boxes, tamaño de ventanas, control de las interfaces mediante tacto o teclado, etc.

- f) Location Manager, posibilita a las aplicaciones la obtención de información de localización y posicionamiento, y funcionar según ésta.
- g) Notification Manager, mediante el cual las aplicaciones, usando un mismo formato, comunican al usuario eventos que ocurran durante su ejecución: una llamada entrante, un mensaje recibido, conexión Wi-Fi disponible, ubicación en un punto determinado, etc.
- h) XMPP Service, colección de API para utilizar este protocolo de intercambio de mensajes basado en XML.

## **B. MÁQUINA VIRTUAL DALVIK**

En Android, todas las aplicaciones se programan en el lenguaje Java y se ejecutan mediante una máquina virtual de nombre Dalvik, específicamente diseñada para Android. Esta máquina virtual ha sido optimizada y adaptada a las peculiaridades propias de los dispositivos móviles (menor capacidad de proceso, baja memoria, alimentación por batería, etc.) y trabaja con ficheros de extensión .dex (Dalvik Executables). Dalvik no trabaja directamente con el bytecode de Java, sino que lo transforma en un código más eficiente que el original, pensado para procesadores pequeños. (Dalvik VM, s. f.).

## **C. COMPONENTES DE UNA APLICACIÓN**

Según Android (2008), todas las aplicaciones en Android pueden descomponerse en cuatro tipos de bloques o componentes principales. Cada aplicación será una combinación de uno o más de estos componentes, que deberán ser declarados de forma explícita en un fichero con formato XML denominado “AndroidManifest.xml”, junto a otros datos asociados como valores globales, clases que implementa, datos que puede manejar, permisos, etc. Este fichero es básico en cualquier aplicación en Android y permite al sistema desplegar y ejecutar correctamente la aplicación. A continuación se exponen los cuatro tipos de componentes en los que puede dividirse una aplicación para Android.

**Activity.-** Es el componente más habitual de las aplicaciones Android. Un componente Activity refleja una determinada actividad llevada a cabo por una aplicación, y que lleva asociada una ventana o interfaz de usuario; es importante señalar que no contempla únicamente el aspecto gráfico, sino que éste forma parte del componente Activity a través de vistas representadas por clases como View y sus derivadas. Este componente se implementa mediante la clase de mismo nombre Activity. Muy vinculado a este componente

se encuentran los Intents, una interesante novedad introducida por Android. Un Intent consiste básicamente en la voluntad de realizar alguna acción, generalmente asociada a unos datos. Lanzando un Intent, una aplicación puede delegar el trabajo en otra, de forma que el sistema se encarga de buscar qué aplicación entre las instaladas es la que puede llevar a cabo la acción solicitada. Por ejemplo, abrir una URL en algún navegador web, o escribir un correo electrónico desde algún cliente de correo. (Android, 2008).

**Broadcast Intent Receiver.-** Se utiliza para lanzar alguna ejecución dentro de la aplicación actual cuando un determinado evento se produzca (generalmente, abrir un componente Activity). Por ejemplo, una llamada entrante o un SMS recibido. No tiene interfaz de usuario asociada, pero puede utilizar el API Notification Manager, mencionada anteriormente, para avisar al usuario del evento producido a través de la barra de estado del dispositivo móvil. Se implementa a través de una clase de nombre BroadcastReceiver. Para que Broadcast Intent Receiver funcione, no es necesario que la aplicación en cuestión sea la aplicación activa en el momento de producirse el evento. El sistema lanzará la aplicación si es necesario cuando el evento monitorizado tenga lugar. (Android, 2008).

**Service.-** Un componente Service representa una aplicación ejecutada sin interfaz de usuario, y que generalmente tiene lugar en segundo plano mientras otras aplicaciones (éstas con interfaz) son las que están activas en la pantalla del dispositivo. Un ejemplo típico de este componente es un reproductor de música. La interfaz del reproductor muestra al usuario las distintas canciones disponibles, así como los típicos botones de reproducción, pausa, volumen, etc. En el momento en el que el usuario reproduce una canción, ésta se escucha mientras se siguen visionando todas las acciones anteriores, e incluso puede ejecutar una aplicación distinta sin que la música deje de sonar. La interfaz de usuario del reproductor sería un componente Activity, pero la música en reproducción sería un componente Service, porque se ejecuta en background. Este elemento está implementado por la clase de mismo nombre Service. (Android, 2008).

**Content Provider.-** Con este componente, cualquier aplicación en Android puede almacenar datos en un fichero, en una base de datos SQLite o en cualquier otro formato. Además, estos datos pueden ser compartidos entre distintas aplicaciones. Una clase que implemente el componente Content Provider contendrá una serie de métodos que permite almacenar, recuperar, actualizar y compartir los datos de una aplicación. Existe una

colección de clases para distintos tipos de gestión de datos en el paquete android.provider. Además, cualquier formato adicional que se quiera implementar deberá pertenecer a este paquete y seguir sus estándares de funcionamiento. (Android, 2008).

#### D. CICLO DE VIDA DE LAS APLICACIONES ANDROID

Según Google (s. f.), en Android, cada aplicación se ejecuta en su propio proceso. Esto aporta beneficios en cuestiones básicas como seguridad, gestión de memoria, o la ocupación de la CPU del dispositivo móvil. Android se ocupa de lanzar y parar todos estos procesos, gestionar su ejecución y decidir qué hacer en función de los recursos disponibles y de las órdenes dadas por el usuario. Android lanza tantos procesos como permitan los recursos del dispositivo. Cada proceso, correspondiente a una aplicación, estará formado por una o varias actividades independientes (componentes Activity) de esa aplicación. Cuando el usuario navega de una actividad a otra, o abre una nueva aplicación, el sistema duerme dicho proceso y realiza una copia de su estado para poder recuperarlo más tarde. El proceso y la actividad siguen existiendo en el sistema, pero están dormidos y su estado ha sido guardado. Es entonces cuando crea, o despierta si ya existe, el proceso para la aplicación que debe ser lanzada, asumiendo que existan recursos para ello. El componente Activity, probablemente el más importante, tiene un ciclo de vida como el mostrado en la Figura 2.4. (Google. s. f.)

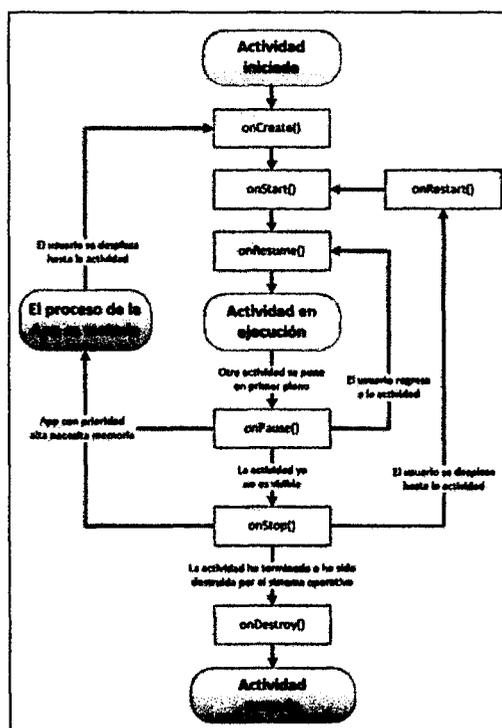


Figura 2.4: Ciclo de vida de un objeto Activity (Google. s. f.)

De la Figura 2.4 pueden sacarse las siguientes conclusiones. (Google. s. f.):

- a) **onCreate(), onDestroy():** abarcan todo el ciclo de vida. Cada uno de estos métodos representan el principio y el fin de la actividad.
- b) **onStart(), onStop():** representan la parte visible del ciclo de vida. Desde onStart() hasta onStop(), la actividad será visible para el usuario, aunque es posible que no tenga el foco de acción por existir otras actividades superpuestas.
- c) **onResume(), onPause():** delimitan la parte útil del ciclo de vida. Desde onResume() hasta onPause(), la actividad no sólo es visible, sino que además tiene el foco de la acción y el usuario puede interactuar con ella.

### 2.2.8. TECNOLOGIAS DE COMUNICACIÓN DE DISPOSITIVOS MOVILES

Para Marcia (2011), los teléfonos celulares se han convertido en una herramienta primordial para la gente común y de negocios; las hace sentir más seguras y las hace más productivas. A pesar de que la telefonía celular fue concebida estrictamente para la voz, la tecnología celular de hoy es capaz de brindar otro tipo de servicios, como datos, audio y video con algunas limitaciones. Sin embargo, la telefonía inalámbrica del mañana hará posible aplicaciones que requieran un mayor consumo de ancho de banda. Existen varios tipos de arquitecturas y protocolos de red celular, teniendo los siguientes: (Arquitectura y Protocolos de Red Celular s. f.).

**Acceso Múltiple por División de Tiempo (TDMA).**- La tecnología TDMA, permite a los operadores de red celular multiplexar transmisiones múltiples sobre una frecuencia de radio. Esto proporciona un mayor soporte de suscriptor utilizando el espectro de frecuencias disponibles. En Estados Unidos la tecnología AMPS y la TDMA deben ser compatibles para que los teléfonos celulares puedan operar automáticamente en “modo dual” (modo analógico y modo digital). (Arquitectura y Protocolos de Red Celular s. f.).

**Acceso Múltiple por División de Código (CDMA).**-En CDMA varias frecuencias se envían sobre la misma frecuencia sin multiplexado, en su lugar se agrega un código único para cada transmisión. De este modo los aparatos reciben todas las transmisiones, decodifican cada una y hallan el código correcto. A esta tecnología de dispersar frecuencias por todo el espectro se le conoce como Dispersión espectral. (Arquitectura y Protocolos de Red Celular s. f.).

**Sistema Global para Comunicaciones Móviles GSM.-** La primera red GSM fue implementada en Alemania en 1992. Debido al éxito que tuvo en esa época el desarrollo del ISDN y su posible implementación en redes inalámbricas, es que GSM se tuvo que implementar como una red digital. Así, el GSM se implementó en sus inicios en base a la señalización en base a la tecnología SS7. El GSM presenta varias características importantes como las llamadas en espera, identificación de la llamada, retención de llamada, llamadas en conferencia y velocidades de datos de 2400 hasta 9600 bps. (Arquitectura y Protocolos de Red Celular s. f.).

### **2.2.9. TECNOLOGIAS DE INTERNET**

Para Andrada (2010), internet es una red integrada por miles de redes y computadoras interconectadas en todo el mundo mediante cables y señales de telecomunicaciones, que utilizan una tecnología común para la transferencia de datos y el protocolo de comunicaciones que utiliza Internet se denomina TCP/IP (Transmission Control Protocol/Internet Protocol).

#### **A. APLICACION WEB**

Según Peñafiel (s.f.), en la ingeniería software se denomina aplicación web a aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web, y en la que se confía la ejecución de la aplicación al navegador.

Para Lujan (2002), una aplicación web es un tipo especial de aplicación cliente/servidor, donde tanto el cliente como el servidor y el protocolo mediante el que se comunican (HTTP) están estandarizados y no han de ser creados por el programador de aplicaciones. El protocolo HTTP forma parte de la familia de protocolos de comunicaciones Transmission Control Protocol/Internet Protocol (TCP/IP), que son los empleados en Internet. Estos protocolos permiten la conexión de sistemas heterogéneos, lo que facilita el intercambio de información entre distintos ordenadores.

#### **B. PROTOCOLO HTTP**

Para Mateu (2004), el protocolo HTTP (hypertext transfer protocol) es el protocolo base de la WWW. Se trata de un protocolo simple, orientado a conexión y sin estado. La razón de que esté orientado a conexión es que emplea para su funcionamiento un

protocolo de comunicaciones (TCP, transport control protocol) de modo conectado, un protocolo que establece un canal de comunicaciones de extremo a extremo (entre el cliente y el servidor) por el que pasa el flujo de bytes que constituyen los datos que hay que transferir, en contraposición a los protocolos de datagrama o no orientados a conexión que dividen los datos en pequeños paquetes (datagramas) y los envían, pudiendo llegar por vías diferentes del servidor al cliente. El protocolo no mantiene estado, es decir, cada transferencia de datos es una conexión independiente de la anterior, sin relación alguna entre ellas, hasta el punto de que para transferir una página web tenemos que enviar el código HTML del texto, así como las imágenes que la componen, pues en la especificación inicial de HTTP, la 1.0, se abrían y usaban tantas conexiones como componentes tenía la página, transfiriéndose por cada conexión un componente.

### **C. PROTOCOLO TCP/IP**

Según Estrada (2004), los protocolos IP (Protocolo de Internet) y TCP (Protocolo de Control de Transmisión) se originaron a principios de 1980 y fueron adoptados por la red ARPANET en 1983, que estaba integrada por cientos de computadoras de universidades, centros de investigación militar y algunas empresas. El e-mail (electronic mail) fue el servicio más comúnmente utilizado entonces, mientras que el sistema operativo más empleado era UNIX, en su versión BSD UNIX, desarrollada por la Universidad de California. Fue a mediados de los ochenta cuando fue creado el protocolo TCP/IP con la finalidad de contar con un lenguaje común a todas las computadoras conectadas a Internet, ya con la unión de las redes ARPANET, CSNET y MILNET. El protocolo TCP/IP representa, entonces, las reglas que hacen posible la conexión de computadoras de marcas y tecnología diferentes.

Para Gonzales (2000), TCP/IP es el protocolo común utilizado por todos los ordenadores conectados a Internet, de manera que éstos puedan comunicarse entre sí. Hay que tener en cuenta que en Internet se encuentran conectados ordenadores de clases muy diferentes y con hardware y software incompatibles en muchos casos, además de todos los medios y formas posibles de conexión. Aquí se encuentra una de las grandes ventajas del TCP/IP, pues este protocolo se encargará de que la comunicación entre todos sea posible. TCP/IP es compatible con cualquier sistema operativo y con cualquier tipo de hardware. TCP/IP no es un único protocolo, sino que es en realidad lo que se conoce con este nombre es un conjunto de los dos protocolos más importantes.

## D. SERVICIOS WEB

Curbera, William, Nagy y Weerawarana, (s. f.), definen a un Servicio Web como una aplicación de red habilitada para Internet mediante protocolos que sirven como estándares de comunicación entre aplicaciones, por medio de interfaces bien definidas y que son descritas, utilizando un lenguaje estándar. Otra definición nos dice que los Servicios Web son componentes de software que permiten a los usuarios usar aplicaciones de negocios que comparten datos con otros programas modulares, vía Internet. Son aplicaciones independientes de la plataforma que pueden ser fácilmente publicadas, localizadas e invocadas mediante protocolos Web estándar, como XML, SOAP, UDDI o WSDL. El objetivo principal de este tipo de estándares es crear un directorio en línea de Servicios Web, desde luego que pueda ser localizado por un gran número de personas o empresas y que a su vez tenga una alta fiabilidad.

El Simple Object Access Protocol (SOAP), deriva de un protocolo creado por David Winer, XML-RPC en 1998. Es un protocolo de intercambio de información estructurada basado en XML. Entre otras cosas, SOAP puede representar llamadas a procedimientos remotos (RPC), sus respuestas proporcionan un mecanismo para el intercambio de información y servicios entre aplicaciones. Un mensaje SOAP se compone de un sobre que contiene el cuerpo del mensaje y cualquier información de cabecera que se utiliza para describir le mensaje. A continuación se muestra un ejemplo de mensaje SOAP. (Curbera, et al, s. f.).

Mensaje SOAP
<pre>&lt;?xml version="1.0"?&gt; &lt;soap:Envelope xmlns:soap=<a href="http://schemas.xmlsoap.org/soap/envelope/">http://schemas.xmlsoap.org/soap/envelope/</a>&gt;   &lt;soap:Header&gt;     &lt;!--Aquí va información opcional de cabecera.--&gt;     &lt;To&gt; Scout &lt;/To&gt;     &lt;From&gt;Suzanne&lt;/From&gt;   &lt;/soap:Header&gt;    &lt;soap:Body&gt;     &lt;!--Aquí va el mensaje.--&gt;     Por favor , compra leche cuando vuelvas a casa del trabajo.   &lt;/soap:Body&gt;</pre>

Figura 4.5: Mensaje SOAP. (Curbera, et al, s. f.)

### **2.2.10. POBLACION Y MUESTRA**

Para García (2005), una vez definido el problema a investigar, formulados los objetivos y delimitadas las variables se hace necesario determinar los elementos o individuos con quienes se va a llevar a cabo el estudio o investigación. Esta consideración nos conduce a delimitar el ámbito de la investigación definiendo una población y seleccionando la muestra. Cualquier característica medible de la población se denomina parámetro, los valores de los parámetros calculados sobre muestras se conocen como estadísticos o estadígrafos, utilizan letras latinas ( $\mu$ ,  $s^2$ ,  $r$ ,..., media, varianza, coeficiente de correlación) y describen a las citadas muestras.

#### **A. POBLACION**

Población es un conjunto definido, limitado y accesible del universo que forma el referente para la elección de la muestra. Es el grupo al que se intenta generalizar los resultados. (Buendía, Colás y Hernández, 1998).

Según Tamayo (1997), la población es un conjunto de individuos de la misma clase, limitada por el estudio. La población se define como la totalidad del fenómeno a estudiar donde las unidades de población poseen una característica común la cual se estudia y da origen a los datos de la investigación. Entonces, una población es el conjunto de todas las cosas que concuerdan con una serie determinada de especificaciones. Un censo, por ejemplo, es el recuento de todos los elementos de una población.

#### **B MUESTRA**

Es una parte o subconjunto de una población normalmente seleccionada de tal modo que ponga de manifiesto las propiedades de la población. Su característica más importante es la representatividad, es decir, que sea una parte típica de la población en las características que son relevantes para la investigación. Una parte representativa de una población debidamente elegida, que se somete a observación científica en representación del conjunto, con el propósito de obtener resultados válidos. (Sierra, 1988).

Según Tamayo (1997), afirma que la muestra es el grupo de individuos que se toma de la población, para estudiar un fenómeno estadístico. La muestra es la que puede determinar la problemática ya que les capaz de generar los datos con los cuales se identifican las fallas dentro del proceso.

## **C MUESTRA POR JUICIO DE EXPERTOS**

La evaluación mediante el juicio de expertos, método de validación cada vez más utilizado en la investigación, “consiste, básicamente, en solicitar a una serie de personas la demanda de un juicio hacia un objeto, un instrumento, un material de enseñanza, o su opinión respecto a un aspecto concreto” (Cabero y Llorente, 2013). Se trata de una técnica cuya realización adecuada desde un punto de vista metodológico constituye a veces el único indicador de validez de contenido del instrumento de recogida de datos o de información (Escobar, 2008).

## **D MUESTRA POR CONVENIENCIA**

Es un procedimiento de muestreo cuantitativo en el que el investigador selecciona a los participantes, ya que están dispuestos y disponibles para ser estudiados. Se le conoce como selección intencionada, muestreo accidental o muestreo por oportunidad. Consiste en la elección por métodos no aleatorios de una muestra cuyas características sean similares a las de la población objetivo. En este tipo de muestreo la “representatividad” la determina el investigador de modo subjetivo, siendo este el mayor inconveniente del método ya que no podemos cuantificar la representatividad de la muestra. Presenta casi siempre sesgos y por tanto debe aplicarse únicamente cuando no existe alternativa. (Casal y Mateu, 2003).

## **CAPÍTULO III**

### **METODOLOGÍA DE LA INVESTIGACIÓN**

#### **3.1 TIPO DE INVESTIGACIÓN**

Se desarrollará dos productos software uno web y otro móvil, de Gestión de Cobranzas para la recuperación de Créditos Morosos en la CAC. “Santa María Magdalena”, 2016, usando el proceso XP. Según Bernal (2006), afirma que la investigación descriptiva es uno de los tipos o procedimientos investigativos más populares y utilizados por los principiantes en la actividad investigativa. En tales estudios se muestran, narran, reseñan o identifican hechos, situaciones, rasgos, características de un objeto de estudio, o se diseñan productos, modelos, prototipos, guías, etcétera, pero no se dan explicaciones o razones de las situaciones, los hechos, los fenómenos, etcétera. La investigación descriptiva se soporta principalmente en técnicas como la encuesta, la entrevista, la observación y la revisión documental. Por las consideraciones expuestas, el tipo de investigación es descriptiva.

#### **3.2 DISEÑO DE LA INVESTIGACIÓN**

Según Hernández et. al., (2010), indican que las investigaciones no experimentales son “aquellas que se realizan sin manipular deliberadamente las variables, es decir, no se varía intencionalmente la variable independiente, simplemente se observan las funciones tal y como se dan en su contexto natural para después analizarlo”.

Para Hernández et. al., (2010), la investigación no experimental está conformada por dos tipos de diseño: transversal y longitudinal; y manifiesta que en el diseño de investigación transversal se recolectan datos en un solo momento, en un tiempo único. Su propósito es describir variables, y analizar su incidencia e interrelación en un momento dado. Por las consideraciones mencionadas, el diseño de la investigación es no experimental y de tipo transversal.

#### **3.3 POBLACIÓN Y MUESTRA**

##### **POBLACIÓN**

La población estuvo compuesta por quince procesos finales de la gestión de cobranza de la C.A.C “Santa María Magdalena” en el año 2016.

## **MUESTRA**

Se tomó una muestra por juicio de expertos o conveniencia de cinco procesos finales de la gestión de cobranza de la C.A.C “Santa María Magdalena”, 2016

### **3.4 VARIABLES E INDICADORES**

#### **3.4.1 DEFINICIÓN CONCEPTUAL DE LAS VARIABLES**

##### **PRIMERA VARIABLE**

**Gestión de cobranza.-** La gestión de cobranza es el conjunto de acciones coordinadas y aplicadas adecuada y oportunamente a los socios para lograr la recuperación de los créditos, de manera que los activos exigibles (deudas) de la institución se conviertan en activos líquidos de la manera más rápida y eficiente posible, manteniendo en el proceso la buena disposición de los socios para futuras negociaciones.

##### **INDICADORES DE LA PRIMERA VARIABLE**

**Cartera en cobranza.-** Se trata de la información de aquellos socios que mantienen deudas por pagar. La cartera en cobranza es todavía un segmento de menor dimensión en número de socios; ya que ésta se refiere exclusivamente a quienes tienen créditos por pagar o vencidas y la cartera es gestionada por los gestores de cobranza.

**Notificaciones de pago.-** La notificación de pago es aquella carta informativa en la que se induce al cliente o se le aconseja que efectúe un pago o que finalice el cobro que éste adeuda con la empresa o persona en cuestión. Dicha certificación o dicha carta notarial no tiene otro motivo que el de servir como prueba de cobro para el caso de que dicho cobro trascendiera o hubiese que ir a un juicio.

**Clasificación del deudor.-** Los recuperadores han logrado clasificar a los socios relacionando: la voluntad de pago con la capacidad de pago. Aplicándoles una escala diferenciada de alto a bajo. Alto (A) significa una situación buena o mejor, tanto para la voluntad como para la capacidad de pago. En cambio, bajo (B) expresa una situación contraria, que en el extremo es malo o peor. Esta escala ordinal sirve para ubicar al socio en un plano de dos dimensiones relacionadas. La combinación de las escalas de voluntad y capacidad de pagos permite identificar hasta cuatro tipos de comportamiento del socio frente a sus obligaciones.

## **SEGUNDA VARIABLE**

**Créditos morosos.-** Son los créditos que no han sido cancelados oportunamente según el cronograma de pagos establecido al momento de otorgar el crédito, también pueden ser los créditos sin recibir el pago de una cuota en el día de vencimiento.

## **INDICADORES DE LA SEGUNDA VARIABLE**

**Deuda.-** La deuda es la obligación de satisfacer una cuenta por pagar contraída a una parte acreedora, ello cuando el origen de la deuda es de carácter voluntario, es decir, que el deudor decidió libremente comprometerse al pago de dicha obligación.

**Interés moratorio.-** El interés moratorio tiene por finalidad indemnizar la mora en el pago de una obligación, de lo que podemos inferir que el fin del interés surge ante el cumplimiento tardío de la obligación de devolver el capital.

**Días de atraso.-** Son los días que el socio deja de pagar su deuda y se cuenta a partir del día siguiente del vencimiento de su cuota. Los deudores están clasificados de acuerdo a los días de atraso.

### **3.4.2 DEFINICIÓN OPERACIONAL DE LAS VARIABLES**

#### **PRIMERA VARIABLE**

X: Gestión de cobranza

#### **INDICADORES**

X1: Cartera en cobranza

X2: Notificaciones de pago

X3: Clasificación del deudor

#### **SEGUNDA VARIABLE**

Y: Créditos morosos

#### **INDICADORES**

Y1: Deuda

Y2: Interés moratorio

Y3: Días de atraso

### **3.5 OPERACIONALIZACION DE LAS VARIABLES**

Se ha operado las variables de la investigación, a fin de obtener los reactivos, las técnicas e instrumentos de recolección de datos para levantar la información, el cual se muestra en el Anexo C.

### **3.6 TÉCNICAS E INSTRUMENTOS**

#### **3.6.1 TÉCNICAS PARA RECOLECTAR INFORMACIÓN**

Se utilizó las técnicas de; entrevista al Jefe de la oficina de recuperaciones, encuestas a los gestores de cobranzas, análisis documental de los reportes de cobranzas y de las normas y políticas de recuperación de créditos de la C.A.C “Santa María Magdalena”, 2016.

#### **3.6.2 INSTRUMENTOS PARA RECOLECTAR INFORMACIÓN**

Se diseñó el instrumento “Guía de entrevista al jefe de la oficina de recuperaciones de la C.A.C “Santa María Magdalena”, para la técnica de entrevista, presentado en el anexo A, tabla N° A.1, donde se muestra la información sobre: la cartera en cobranza, notificaciones de pago, clasificación de los deudores y las actividades realizadas por los gestores de cobranzas.

Se diseñó el instrumento: “Cuestionario para encuesta sobre como determinar la voluntad de pago y la capacidad de pago”, para la técnica de encuesta, presentado en el anexo A, tabla N° A.2, donde se muestra información sobre: las preguntas que son más importantes para determinar la capacidad de pago y la voluntad de pago.

Se diseñó el instrumento: “Registro de análisis documental a los reportes de cobranza” presentado en el anexo A, la tabla N° A.3, donde se muestra información de los tipos de reportes que puedan existir y de los datos que contienen.

Se diseñó el instrumento “Registro para análisis documental de las normas y políticas de recuperación de créditos”, presentado en el anexo A, tabla N° A.4, donde se muestra los lineamientos específicos de la recuperación de créditos y los procesos y actividades que realizan los gestores de cobranzas.

### 3.6.3 HERRAMIENTAS PARA EL TRATAMIENTO DE DATOS E INFORMACION

Las herramientas tecnológicas que utilizará son seleccionadas en función de lo que se quiere realizar, es decir, las herramientas que se necesitan para hacer una aplicación web y una aplicación Android con su respectiva base de datos.

SOFTWARE	FABRICANTE	SERVICIO
WINDOWS 8.1	Producida por Microsoft Corporation	Es la versión más estable de Microsoft Windows, línea de sistemas operativos producida por Microsoft Corporation.
JAVA	Desarrollado por Sun Microsystems	Java es un lenguaje de programación orientado a objetos, desarrollado por Sun Microsystems, que posee un sistema que interpreta y ejecuta los archivos para ser compilados.
NETBEANS	Netbeans fue desarrollado Sun Microsystems	NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE2 es un producto libre y gratuito sin restricciones de uso.
ANDROID STUDIO	Desarrollado por Google	Android Studio es un entorno de desarrollo integrado para la plataforma Android. Fue anunciado el 16 de mayo de 2013 en la conferencia Google I/O, y reemplazó a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android. La primera versión estable fue publicada en diciembre de 2014.
VISUAL STUDIO EXPRESS 2013	Producida por Microsoft Corporation	Microsoft Visual Studio Express Edition es un programa de desarrollo en entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta

		varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET.
MICROSOFT SQL SERVER 2008	Producida por Microsoft Corporation	Microsoft SQL Server es un sistema de manejo de bases de datos del modelo relacional. El lenguaje utilizado es Transact-SQL, una implementación del estándar ANSI del lenguaje SQL, utilizado para manipular y recuperar datos (DML), crear tablas y definir relaciones entre ellas (DDL).

Tabla N° 3.1: Herramientas tecnológicas para el tratamiento de datos

### 3.6.4 TECNICAS PARA APLICAR XP

Al revisar el marco teórico desarrollado en el capítulo II, se formula el proceso con apoyo de la referencia, que considera las fases para desarrollar la aplicación web y la aplicación móvil utilizando XP, como se indica en las tablas 3.2 al 3.4.

TAREA	ARTEFACTO	TÉCNICA	RESPONSABLES
Escribir historias de usuario	Historia de usuario	Describir brevemente la historia de usuario con la regla del negocio (lo que el sistema debe hacer) Dividir historias de usuario grandes	Cliente
Probar las tecnologías a utilizar	Arquitectura técnica inicial	Explorar posibilidades de uso de tecnologías Probar el rendimiento de las tecnologías Definir las tecnologías a usar	Cliente Programador Entrenador
Estimar esfuerzo para historia de usuario	Plan de alto nivel	Conocer previamente la historia de usuario Hacer una implementación rápida de historia de usuario Estimar esfuerzo (semana) para desarrollar la historia de usuario	Programador

Tabla N° 3.2: Fase de Exploración (Porrás, 2010)

<b>TAREA</b>	<b>ARTEFACTO</b>	<b>TÉCNICA</b>	<b>RESPONSABLES</b>
Rescribir las historias de usuario	Historia de usuario	Describir detalladamente la historia de usuario con la regla de negocio	Cliente
Formular el plan de versiones	Plan de versión (una iteración)	Introducir nuevos requisitos del software	Cliente
		Definir prioridad para cada historia de usuario por necesidad del negocio	
		Utilizar técnicas de elaboración del plan de alto nivel	Programador
		Estimar y asignar esfuerzo (semana) para cada historia de usuario en función a tiempo para planear, diseñar, implementar y probar.	
		Estimar y asignar riesgo a cada historia de usuario en función a situación que afecta la estimación del esfuerzo.	
		Actualizar tarjeta de historia de usuario	

Tabla Nº 3.3: Fase de planificación (Porras, 2010)

<b>TAREA</b>	<b>ARTEFACTO</b>	<b>TÉCNICA</b>	<b>RESPONSABLES</b>
Definir la arquitectura técnica	Arquitectura técnica	Actualizar la arquitectura técnica inicial. Usar características del negocio Utilizar arquitectura por capas Integrar frameworks.	Cliente Programador Entrenador
Escribir tareas de ingeniería	Tarea de ingeniería	Dividir cada historia de usuario en tareas, describir usando reglas del negocio cada tarea de ingeniería	Cliente programador

Formular el plan de iteraciones	Plan de iteración	Estimar y asignar esfuerzo para desarrollar una tarea de ingeniería	Programador
		Asignar una tarea de ingeniería al programador	Entrenador Programador
		Utilizar el plan de versión Actualizar el plan con tareas de ingeniería de la siguiente iteración. Actualizar el cuándo fallo prueba de aceptación. Actualizar el plan con tareas no concluidas. Actualizar las tarjetas de tarea de ingeniería	
Crear pruebas de aceptación	Caso de prueba de aceptación	Escribir pruebas de aceptación para cada historia de usuario por iteración	Cliente Encargado de pruebas
Implementar las interfaces	GUI	Diseñar con precisión la GUI relacionada a cada historia de usuario. Generar código para la interface usando herramienta	Cliente Programador
Escribir tarjetas CRC para cada tarea de ingeniería	Tarjeta CRC	Diseñar para una tarea de ingeniería de forma simple Rediseñar por falla de prueba de aceptación una tarea Identificar responsabilidades Identificar colaboración. Identificar Atributos	Cliente Programador
Implementar la base de datos física	Base de datos física	Escribir script usando tarjeta CRC. Ejecutar script usando DBMS	Programador
Implementar código para clases entidad	Código fuente	Escribir código fuente o generar con una herramienta usando tarjetas CRC	Programador
Crear pruebas unitarias para	Prueba unitaria	Escribir código fuente para una prueba unitaria, usando una	Programador

las clases control		herramienta	
Implementar código fuente	Código fuente	Codificar una tarea de ingeniería Hacer refactoring Mover programadores	Programador Supervisor
Ejecutar pruebas unitarias	Reporte de prueba unitaria	Ejecutar el módulo de cada prueba unitaria Modificar código fuente si la prueba unitaria muestra resultado incorrecto	Programador
Realizar integración continua	Código fuente	Integrar las tareas para una historia de usuario Mantener sistema integrado todo el tiempo	Programador
Ejecutar pruebas de integración para una historia de usuario	Reporte pruebas de integración	Integrar continuamente al concluir las tareas de una historia de usuario Verificar que las pruebas de integración pasan al 100%	Programador
Ejecutar pruebas de aceptación	Reporte de pruebas de aceptación	Correr la última versión de una iteración Utilizar los casos de prueba de aceptación	Cliente Encargado de pruebas

Tabla N° 3.4: Fase de iteración (Porrás, 2010)

## CAPÍTULO IV

### ANÁLISIS Y RESULTADOS DE LA INVESTIGACIÓN

#### 4.1 ARTEFACTOS DEL SOFTWARE APLICANDO EL PROCESO XP

##### 4.1.1 FASE DE EXPLORACION

Según el procedimiento desarrollado en las tablas 3.2 al 3.4 del capítulo III, para el proceso XP, descrito en el capítulo II, título 2.2.2, párrafo B, para la fase I; exploración, obtenemos, las historias de usuario, arquitectura inicial y el plan de alto nivel.

Nº	HISTORIA DE USUARIO	DESCRIPCIÓN
01	Iniciar sesión en la aplicación web	El usuario (jefe de recuperaciones, gestor de cobranza), podrán autenticarse para iniciar sesión en la aplicación web.
02	Consultar lista de deudores	El usuario (jefe de recuperaciones, gestor de cobranza), podrán consultar la lista de todos los deudores de la Cooperativa.
03	Consultar lista de nuevos deudores	El usuario (jefe de recuperaciones, gestor de cobranza), podrán consultar la lista de los nuevos deudores que no tienen gestor asignado.
04	Buscar deudor	El usuario (jefe de recuperaciones, gestor de cobranza), podrán buscar por DNI, nombres y apellidos, por categoría y lugar de domicilio del deudor.
05	Consultar información del deudor	El usuario (jefe de recuperaciones, gestor de cobranza), podrán consultar toda la información del deudor.
06	Consultar lista de gestores de cobranzas	El usuario (jefe de recuperaciones), podrá consultar la lista de los gestores de cobranzas que trabajan en la Cooperativa.
07	Consultar información del gestor de cobranzas	El usuario (jefe de recuperaciones), podrá consultar la información de los gestores de cobranzas que trabajan en la cooperativa.
08	Administrar gestor de cobranza	El usuario (jefe de recuperaciones), podrá ingresar, actualizar y eliminar la información gestores de cobranzas.
09	Consultar cartera de cobranza por gestor de cobranza	El usuario (jefe de recuperaciones), podrá consultar la lista de los deudores que le corresponde a cada gestor de cobranza.
10	Asignar deudor a gestor de cobranza	El usuario (jefe de recuperaciones), podrá asignar deudores a los gestores de cobranzas de acuerdo a su capacidad y experiencia.

11	Consultar notificaciones de pago por entregar	El usuario (jefe de recuperaciones, gestor de cobranza), podrán consultar las notificaciones que aún no han sido entregadas a los deudores.
12	Consultar notificaciones de pago del deudor	El usuario (jefe de recuperaciones, gestor de cobranza), podrán consultar las notificaciones que se entregaron a los deudores.
13	Registrar notificación de pago del deudor	El usuario (jefe de recuperaciones, gestor de cobranza), podrán registrar las notificaciones de pago que son entregadas a los deudores.
14	Consultar compromisos de pago	El usuario (jefe de recuperaciones, gestor de cobranza), podrán consultar los compromisos de pago.
15	Consultar compromisos de pago del deudor	El usuario (jefe de recuperaciones, gestor de cobranza), podrán consultar los compromisos del deudor.
16	Registrar compromisos de pago del deudor	El usuario (jefe de recuperaciones, gestor de cobranza), podrán registrar los compromisos de pago de los deudores.
17	Clasificar deudor	El usuario (jefe de recuperaciones, gestor de cobranza), podrán clasificar a los deudores.
18	Emitir reporte de cobranza	El usuario (jefe de recuperaciones), podrá visualizar e imprimir el reporte de cobranza.

Tabla N° 4.1: Historias de Usuario para la Aplicación web

N°	HISTORIA DE USUARIO	DESCRIPCIÓN
01	Iniciar sesión en la aplicación móvil	El usuario (gestor de cobranza), podrá autenticarse para iniciar sesión en la aplicación móvil.
02	Consultar lista de deudores	El usuario (gestor de cobranza), podrá consultar la lista de todos los deudores que están en su cartera de cobranzas.
03	Consultar información del deudor	El usuario (gestor de cobranza), podrá consultar la información necesaria del deudor.
04	Consultar notificaciones de pago del deudor	El usuario (gestor de cobranza), podrá consultar las notificaciones que ya fueron entregadas a los deudores.
05	Registrar notificaciones de pago del deudor	El usuario (gestor de cobranza), podrá registrar las notificaciones de pago que son entregadas a los deudores.
06	Consultar compromisos de pago del deudor	El usuario (gestor de cobranza), podrá consultar los compromisos del deudor.
07	Registrar compromisos de pago del deudor	El usuario (gestor de cobranza), podrá registrar los compromisos de pago de los deudores.

08	Consultar notificaciones de pago por entregar	El usuario (gestor de cobranza), podrá consultar las notificaciones que aún no han sido entregadas a los deudores
09	Consultar compromisos de pago del día	El usuario (gestor de cobranza), podrá consultar los compromisos de pago acordados para el día de hoy.
10	Clasificar deudor	El usuario (gestor de cobranza), podrá registrar la clasificación del deudor.
11	Consultar información del garante	El usuario (gestor de cobranza), podrá consultar la información del garante.
12	Consultar ubicación del domicilio del deudor en el mapa	El usuario (gestor de cobranza), podrá ubicar la dirección del deudor en el mapa de la aplicación.

Tabla N° 4.2: Historias de Usuario para la aplicación móvil.

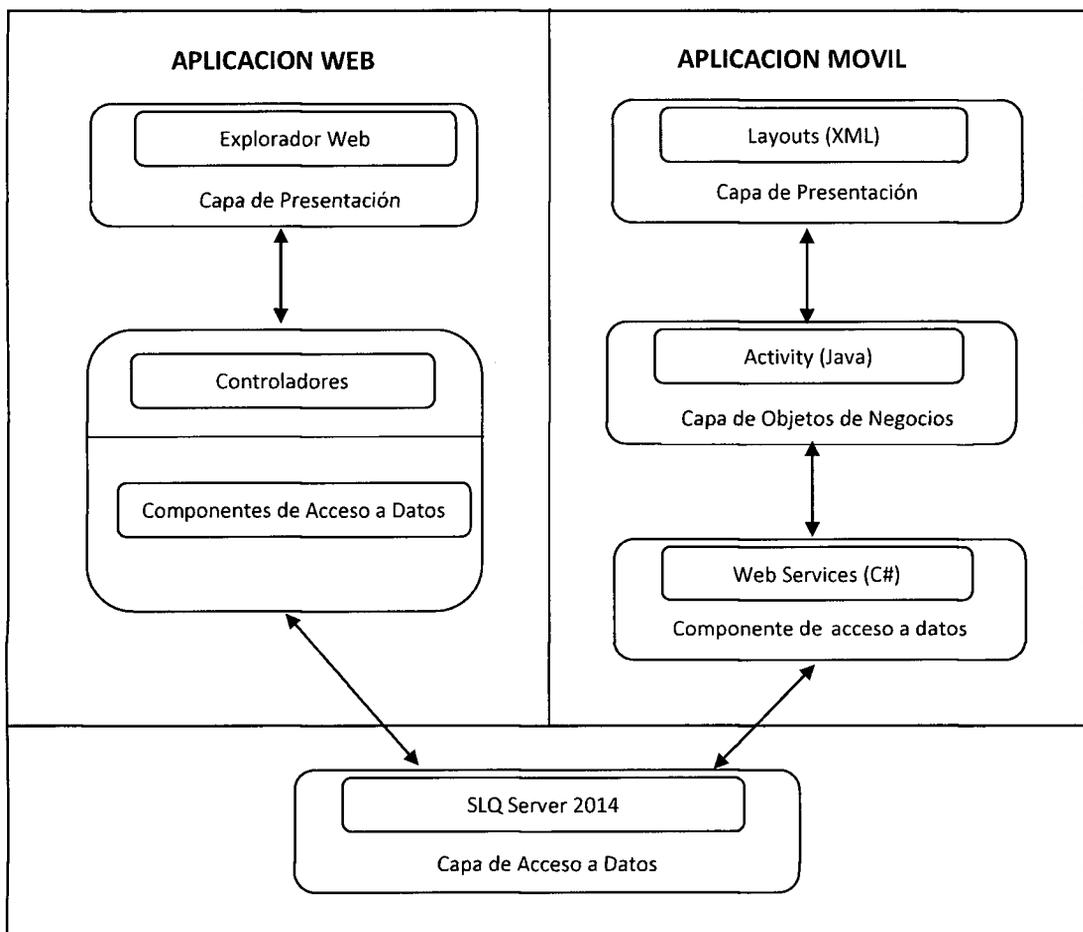


Figura N° 4.1: Arquitectura técnica inicial

<b>N°</b>	<b>HISTORIA DE USUARIO</b>	<b>ESFUERZO (Días)</b>
01	Iniciar sesión en la aplicación web	2
02	Consultar lista de deudores	2
03	Consultar lista de nuevos deudores	2
04	Buscar deudor	3
05	Consultar información del deudor	3
06	Consultar lista de gestores de cobranzas	2
07	Consultar información del gestor de cobranzas	2
08	Administrar gestor de cobranza	3
09	Consultar cartera de cobranza por gestor de cobranza	3
10	Asignar deudor a gestor de cobranza	2
11	Consultar notificaciones de pago por entregar	2
12	Consultar notificaciones de pago del deudor	3
13	Registrar notificación de pago del deudor	1
14	Consultar compromisos de pago	1
15	Consultar compromisos de pago del deudor	2
16	Registrar compromisos de pago del deudor	1
17	Clasificar deudor	1
18	Emitir reporte de cobranza	3

Tabla N° 4.3: Plan de alto nivel de la Aplicación web

<b>N°</b>	<b>HISTORIA DE USUARIO</b>	<b>ESFUERZO (Días)</b>
01	Iniciar sesión en la aplicación móvil	2
02	Consultar lista de deudores	2
03	Consultar información del deudor	2
04	Consultar notificaciones de pago del deudor	2
05	Registrar notificaciones de pago del deudor	2
06	Consultar compromisos de pago del deudor	2
07	Registrar compromisos de pago del deudor	2
08	Consultar notificaciones de pago por entregar	2
09	Consultar compromisos de pago del día	2
10	Clasificar deudor	2
11	Consultar información del garante	2
12	Consultar ubicación del domicilio del deudor en el mapa	3

Tabla N° 4.4: Plan de alto nivel de la Aplicación móvil.

#### 4.1.2 FASE DE PLANIFICACIÓN

Aplicamos la técnica para la fase de planificación, presentada en la tabla 3.3 del capítulo III, según la teoría del capítulo II, título 2.2.4, sección B, fase de planificación, obteniendo los artefactos, historias de usuario en detalle y el plan de versión.

#### HISTORIAS DE USUARIO

Las historias de usuario indican las funciones que debe realizar el sistema, es el mecanismo base de captura de requerimientos del proceso XP. Consisten en una breve descripción, no debe incluir sintaxis técnica, de modo que se centren en las necesidades y no en la especificación del aspecto de las interfaces de usuario.

#### A. HISTORIAS DE USUARIO DE LA APLICACIÓN WEB

Historia de Usuario	
<b>Número: 1</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Iniciar sesión en la aplicación web	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), ejecuta la aplicación y se muestra la página de Bienvenida y hace clic en Ingresar, luego la aplicación muestra la página “Login”. El usuario ingresa el nombre de usuario y contraseña; hace clic en “Enviar”, si los datos son correctos la aplicación muestra la página de “Menú principal”, en caso contrario la aplicación mostrara el mensaje de error “Ingrese nombre de usuario y contraseña correcta” y volverá a pedir los datos.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.5: Historia usuario. Iniciar sesión en la aplicación web.

Historia de Usuario	
<b>Número: 2</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Consultar lista de deudores	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), selecciona la opción “Lista de deudores” de la página de “Deudores”, y la aplicación muestra la página “Lista de Deudores” con la relación de todos los deudores de la Cooperativa.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.6: Historia usuario. Consultar lista de deudores.

Historia de Usuario	
<b>Número: 3</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Consultar lista de nuevos deudores	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), selecciona la opción “Nuevos deudores” de la página de “Deudores”, y la aplicación muestra la página “Lista de nuevos deudores” con la relación de todos los socios que recién han caído en deuda.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.7: Historia usuario. Consultar lista de nuevos deudores.

Historia de Usuario	
<b>Número: 4</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Buscar deudor	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “Deudores” selecciona la opción “Buscar deudores” y la aplicación muestra la página “Buscar deudores” en donde el usuario selecciona el tipo de búsqueda los cuales pueden ser por DNI, por nombres y apellidos, por categoría y por domicilio del deudor y luego hace clic en el botón buscar.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.8: Historia usuario. Buscar deudor.

Historia de Usuario	
<b>Número: 5</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Consultar información del deudor	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), hace clic en el botón “Ver detalle”, que está al lado de los datos del deudor de la Página “Lista de Deudores”, y la aplicación muestra la página “Datos del deudor” con toda la información del deudor como los datos personales, domicilio, datos de la deuda, etc.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.9: Historia usuario. Consultar información del deudor.

Historia de Usuario	
<b>Número: 6</b>	<b>Usuario:</b> Jefe de recuperaciones
<b>Nombre historia:</b> Consultar lista de gestores de cobranzas	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	

<b>Descripción:</b> El usuario (Jefe de recuperaciones), selecciona la opción “ <b>Gestor de cobranzas</b> ” de la página “ <b>Gestores de Cobranza</b> ”, y la aplicación muestra la página “ <b>Lista de gestores de cobranzas</b> ” con la información de los gestores de cobranzas que trabajan en la Cooperativa.
<b>Observaciones:</b> Ninguno

Tabla N° 4.10: Historia usuario. Consultar lista de gestores de cobranzas.

<b>Historia de Usuario</b>	
<b>Número: 7</b>	<b>Usuario:</b> Jefe de recuperaciones
<b>Nombre historia:</b> Consultar información del gestor de cobranzas	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), hace clic en el botón “ <b>Ver detalle</b> ”, que está al lado de los datos del gestor de cobranza en la Página “ <b>Lista de gestores de cobranzas</b> ”, y la aplicación muestra la página “ <b>Datos del gestor de cobranza</b> ” con toda la información del gestor de cobranzas.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.11: Historia usuario. Consultar información del gestor de cobranzas.

<b>Historia de Usuario</b>	
<b>Número: 8</b>	<b>Usuario:</b> Jefe de recuperaciones
<b>Nombre historia:</b> Administrar gestor de cobranza	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “ <b>Gestores de cobranza</b> ” selecciona la opción “ <b>Nuevo gestor de cobranza</b> ” y la aplicación muestra la página “ <b>Nuevo gestor de cobranza</b> ”, en donde el usuario podrá registrar a un nuevo gestor de cobranza. Y en la página “ <b>Datos del gestor de cobranza</b> ” el usuario podrá actualizar datos de los gestores de cobranza y eliminar gestores de cobranza.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.12: Historia usuario. Administrar gestor de cobranza

<b>Historia de Usuario</b>	
<b>Número: 9</b>	<b>Usuario:</b> Jefe de recuperaciones
<b>Nombre historia:</b> Consultar cartera de cobranza por gestor de cobranza	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “ <b>Lista de Gestores de cobranzas</b> ”, elige la opción “ <b>ver cartera</b> ” que está al lado de los datos de cada gestor de cobranza,	

y la aplicación muestra la página “Lista de Deudores” con la relación de todos los deudores que le corresponde a cada gestor de cobranzas.

**Observaciones:** Ninguno

Tabla N° 4.13: Historia usuario. Consultar cartera de cobranza por gestor de cobranza.

<b>Historia de Usuario</b>	
<b>Número: 10</b>	<b>Usuario:</b> Jefe de recuperaciones
<b>Nombre historia:</b> Asignar deudor a gestor de cobranza	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “Lista de nuevos deudores” selecciona la opción “Ver detalle” y la aplicación muestra la página “Asignación de deudores” con la información del deudor y una lista desplegable con los nombres de los gestores, luego el usuario selecciona a un gestor de la lista y hace clic en el botón “Asignar”, y la aplicación guarda la información en la base de datos y muestra la página de mensaje “La asignación se realizó con éxito”	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.14: Historia usuario. Asignar deudor a gestor de cobranza.

<b>Historia de Usuario</b>	
<b>Número: 11</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Consultar notificaciones de pago por entregar	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “Notificaciones y compromisos” hace clic en la opción “Notificaciones por entregar” y la aplicación muestra la página “Notificaciones por entregar” con la información de las notificaciones que ya deberían ser entregadas.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.15: Historia usuario. Consultar notificaciones de pago por entregar.

<b>Historia de Usuario</b>	
<b>Número: 12</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Consultar notificaciones de pago del deudor	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “Datos del deudor” hace clic en el botón “Notificaciones” y la aplicación muestra la página “Detalle de notificaciones” con la relación de todas las notificaciones de pago que ya fueron entregadas en su tiempo a los deudor.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.16: Historia usuario. Consultar notificaciones de pago del deudor.

Historia de Usuario	
<b>Número: 13</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Registrar notificación de pago del deudor	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “Detalle de notificaciones” hace clic en el botón “Registrar notificación” y la aplicación muestra la página “Registrar notificación” donde el usuario ingresa los datos de la notificación de pago y luego hace clic en el botón “Registrar” y la aplicación valida y almacena la información en la base de datos.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.17: Historia usuario. : Registrar notificación de pago del deudor.

Historia de Usuario	
<b>Número: 14</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Consultar compromisos de pago	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “Notificaciones y compromisos” hace clic en la opción “Compromisos de pago” y la aplicación muestra la página “Compromisos de pago” con la relación de todos los compromisos de pago acordado entre los deudores y los gestores de cobranza.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.18: Historia usuario. : Consultar compromisos de pago.

Historia de Usuario	
<b>Número: 15</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Consultar compromisos de pago del deudor	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “Datos del deudor” hace clic en el botón “Compromisos” y la aplicación muestra la página “Compromisos de pago del deudor” con la relación de todos los compromisos de pago acordados entre el gestor de cobranza y el deudor.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.19: Historia usuario. : Consultar compromisos de pago del deudor.

Historia de Usuario	
<b>Número: 16</b>	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Registrar compromisos de pago del deudor	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio

<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “ <b>Compromisos de pago del deudor</b> ” hace clic en el botón “ <b>Registrar compromiso</b> ” y la aplicación muestra la página “ <b>Registrar compromiso de pago del deudor</b> ” donde el deudor ingresa los datos del compromiso y luego hace clic en el botón Registrar y la aplicación almacena la información en la base de datos.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.20: Historia usuario. : Registrar compromisos de pago del deudor.

<b>Historia de Usuario</b>	
<b>Número:</b> 17	<b>Usuario:</b> Jefe de recuperaciones, gestor de cobranza
<b>Nombre historia:</b> Clasificar deudor	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), hace clic en el botón “ <b>Clasificar deudor</b> ” de la página “ <b>Datos del deudor</b> ”, y la aplicación muestra la página “ <b>Clasificación del deudor</b> ”, donde el usuario responde las preguntas de la capacidad y voluntad de pago del deudor en cada checkbox y luego hace clic en el botón procesar, y la aplicación procesa la información y lo almacena.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.21: Historia usuario. Clasificar deudor.

<b>Historia de Usuario</b>	
<b>Número:</b> 18	<b>Usuario:</b> Jefe de recuperaciones
<b>Nombre historia:</b> Emitir reporte de cobranza	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “ <b>Menú principal</b> ” selecciona la opción “ <b>Reportes</b> ” y la aplicación muestra la página “ <b>Reportes</b> ”, donde el usuario selecciona el tipo de reporte y luego hace clic en Ejecutar y la aplicación muestra el reporte generado.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.22: Historia usuario. : Emitir reporte de cobranza.

## B HISTORIAS DE USUARIO DE LA APLICACIÓN MOVIL

Historia de Usuario	
Número: 1	Usuario: Gestor de Cobranza
Nombre historia: Iniciar sesión en la aplicación móvil.	
Prioridad en negocio: Alto	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), selecciona el icono de la aplicación Android en el dispositivo móvil y se muestra ventana de “Login”. El usuario ingresa el nombre de usuario y contraseña; pulsa el botón “Iniciar sesión”, si los datos ingresados fueron correctos la aplicación muestra la ventana principal, en caso contrario la aplicación mostrara el mensaje de error “Usuario o contraseña incorrecta”.	
Observaciones: Ninguno	

Tabla N° 4.23: Historia usuario. Iniciar sesión en la aplicación móvil.

Historia de Usuario	
Número: 2	Usuario: Gestor de Cobranza
Nombre historia: Consultar lista de deudores	
Prioridad en negocio: Alto	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa la opción “Lista de deudores” de la ventana “Principal”, y la aplicación muestra la ventana “Lista de deudores” con los nombres, apellidos de los deudores que están en su cartera en cobranza.	
Observaciones: Ninguno	

Tabla N° 4.24: Historia usuario. Consultar lista de deudores.

Historia de Usuario	
Número: 3	Usuario: Gestor de Cobranza
Nombre historia: Consultar información del deudor	
Prioridad en negocio: Alto	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el nombre de un deudor de la ventana “Lista de deudores”, y la aplicación muestra la ventana “Datos del deudor” con la información del deudor, como nombres y apellidos, dirección y datos de la deuda.	
Observaciones: Ninguno	

Tabla N° 4.25: Historia usuario. Consultar información del deudor.

Historia de Usuario	
Número: 4	Usuario: Gestor de Cobranza
Nombre historia: Consultar notificaciones de pago del deudor	
Prioridad en negocio: Alto	Riesgo en desarrollo: Alto

<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “Notificaciones de pago” de la ventana “Datos del deudor”, y la aplicación muestra la ventana “Notificaciones de pago” con la información de las notificaciones que ya han sido entregadas al deudor.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.25: Historia usuario. Consultar notificaciones de pago del deudor.

<b>Historia de Usuario</b>	
<b>Número:</b> 5	<b>Usuario:</b> Gestor de Cobranza
<b>Nombre historia:</b> Registrar notificaciones de pago del deudor	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “Registrar notificación” de la ventana “Notificaciones de pago”, y la aplicación muestra la ventana “Registrar notificación”, donde el usuario selecciona el tipo de notificación e ingresa la hora y la fecha en que es entregada la notificación, luego pulsa el botón guardar y la aplicación almacena la información.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.27: Historia usuario. Registrar notificaciones de pago del deudor.

<b>Historia de Usuario</b>	
<b>Número:</b> 6	<b>Usuario:</b> Gestor de Cobranza
<b>Nombre historia:</b> Consultar compromisos de pago del deudor	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “Compromisos de pago” de la ventana “Datos del deudor”, y la aplicación muestra la ventana “Compromisos de pago” con la información de los compromisos de pago realizados por el deudor.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.28: Historia usuario. Consultar compromisos de pago del deudor.

<b>Historia de Usuario</b>	
<b>Número:</b> 7	<b>Usuario:</b> Gestor de Cobranza
<b>Nombre historia:</b> Registrar compromisos de pago del deudor	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “Registrar compromiso” en la ventana “Compromisos de pago”, y la aplicación muestra la ventana “Registrar	

<b>compromisos</b> ", donde el usuario ingresa los datos del compromiso y luego pulsa el botón <b>"Guardar"</b> y la aplicación almacena la información en la base de datos.
<b>Observaciones:</b> Ninguno

Tabla N° 4.29: Historia usuario. Registrar compromisos de pago del deudor.

<b>Historia de Usuario</b>	
<b>Número:</b> 8	<b>Usuario:</b> Gestor de Cobranza
<b>Nombre historia:</b> Consultar notificaciones de pago por entregar	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa la opción <b>"Notificaciones y compromisos"</b> de la ventana <b>"Principal"</b> , y la aplicación muestra la ventana <b>"Notificaciones y compromisos"</b> luego pulsa el botón <b>"Notificaciones"</b> y la aplicación muestra la ventana <b>"Notificaciones por entregar"</b> , con la lista de las notificaciones que deberían de ser entregadas.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.30: Historia usuario. Consultar notificaciones de pago por entregar.

<b>Historia de Usuario</b>	
<b>Número:</b> 9	<b>Usuario:</b> Gestor de Cobranza
<b>Nombre historia:</b> Consultar compromisos de pago del día	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa la opción <b>"Notificaciones y compromisos"</b> de la ventana <b>"Principal"</b> , y la aplicación muestra la ventana <b>"Notificaciones y compromisos"</b> luego pulsa el botón <b>"Compromisos"</b> y la aplicación muestra la ventana <b>"Compromisos del día"</b> , con la lista de los compromisos de pago que han sido acordados para el día de hoy.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.31: Historia usuario. Consultar compromisos de pago del día.

<b>Historia de Usuario</b>	
<b>Número:</b> 10	<b>Usuario:</b> Gestor de Cobranza
<b>Nombre historia:</b> Clasificar deudor	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón <b>"Clasificar deudor"</b> en la ventana <b>"Datos del deudor"</b> , y la aplicación muestra la ventana <b>"Clasificación del deudor"</b> , donde el usuario responde las preguntas de la capacidad y voluntad de pago del deudor	

en cada checkbox y luego pulsa el botón procesar, y la aplicación procesa la información y lo almacena en la base de datos.
<b>Observaciones:</b> Ninguno

Tabla N° 4.32: Historia usuario. Clasificar deudor.

<b>Historia de Usuario</b>	
<b>Número:</b> 11	<b>Usuario:</b> Gestor de Cobranza
<b>Nombre historia:</b> Consultar información del garante	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “ <b>Datos del garante</b> ” de la ventana “ <b>Datos del deudor</b> ” y la aplicación muestra la ventana “ <b>Datos del garante</b> ” con la información del garante.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.33: Historia usuario. Consultar información del garante.

<b>Historia de Usuario</b>	
<b>Número:</b> 12	<b>Usuario:</b> Gestor de Cobranza
<b>Nombre historia:</b> Consultar ubicación del domicilio del deudor en el mapa	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “ <b>Mostrar dirección en mapa</b> ” en la ventana “ <b>Datos del deudor</b> ”, y la aplicación muestra la ventana “ <b>Dirección del deudor</b> ”, donde se puede ubicar la dirección del deudor en el mapa de la aplicación.	
<b>Observaciones:</b> Ninguno	

Tabla N° 4.34: Historia usuario. Consultar ubicación del domicilio del deudor en el mapa.

N°	HISTORIA DE USUARIO	PRIORIDAD	RIESGO	ESFUERZO (DÍAS)	ITERACIÓN
01	Iniciar sesión en la aplicación web	Alto	Alto	2	1
02	Consultar lista de deudores	Alto	Alto	2	1
03	Consultar lista de nuevos deudores	Alto	Alto	2	1
04	Buscar deudor	Alto	Alto	3	1
05	Consultar información del deudor	Alto	Alto	3	1
06	Consultar lista de gestores de cobranzas	Alto	Alto	2	1
07	Consultar información del gestor de cobranzas	Alto	Alto	2	1
08	Administrar gestor de cobranza	Alto	Alto	3	1

09	Consultar cartera de cobranza por gestor de cobranza	Alto	Alto	3	1
10	Asignar deudor a gestor de cobranza	Alto	Alto	2	2
11	Consultar notificaciones de pago por entregar	Alto	Alto	2	2
12	Consultar notificaciones de pago del deudor	Alto	Alto	2	2
13	Registrar notificación de pago del deudor	Medio	Medio	1	2
14	Consultar compromisos de pago	Medio	Medio	1	2
15	Consultar compromisos de pago del deudor	Medio	Medio	2	2
16	Registrar compromisos de pago del deudor	Medio	Medio	1	2
17	Clasificar deudor	Medio	Medio	1	2
18	Emitir reporte de cobranza	Medio	Medio	3	2

Tabla N° 4.35: Plan de versión de la aplicación web

N°	HISTORIA DE USUARIO	PRIORIDAD	RIESGO	ESFUERZO (DÍAS)	ITERACIÓN
01	Iniciar sesión en la aplicación Android	Alto	Alto	2	1
02	Consultar lista de deudores	Alto	Alto	2	1
03	Consultar información del deudor	Alto	Alto	2	1
04	Consultar notificaciones de pago del deudor	Alto	Alto	2	1
05	Registrar notificaciones de pago del deudor	Alto	Alto	2	1
06	Consultar compromisos de pago del deudor	Alto	Alto	2	1
07	Registrar compromisos de pago del deudor	Alto	Alto	2	1
08	Consultar notificaciones de pago por entregar	Alto	Alto	2	1
09	Consultar compromisos de pago del día	Alto	Alto	2	1
10	Clasificar deudor	Medio	Medio	2	2
11	Consultar información del garante	Medio	Medio	2	2
12	Consultar ubicación del domicilio del deudor en el mapa.	Medio	Medio	3	2

Tabla N° 4.36: Plan de versión de la aplicación móvil

### 4.1.3 FASE DE ITERACIÓN

La fase de iteración presentado en el título 2.2 del capítulo II, fase de Iteración, la técnica referenciada en la tabla 3.4, permite obtener los entregables; arquitectura técnica, tareas de ingeniería, plan de iteración, casos de prueba de aceptación, GUI, tarjetas CRC, base de datos física, código fuente para clases entidad, pruebas unitarias, código fuente para tarea de ingeniería, reporte de pruebas unitarias, reporte de pruebas de integración y de aceptación.

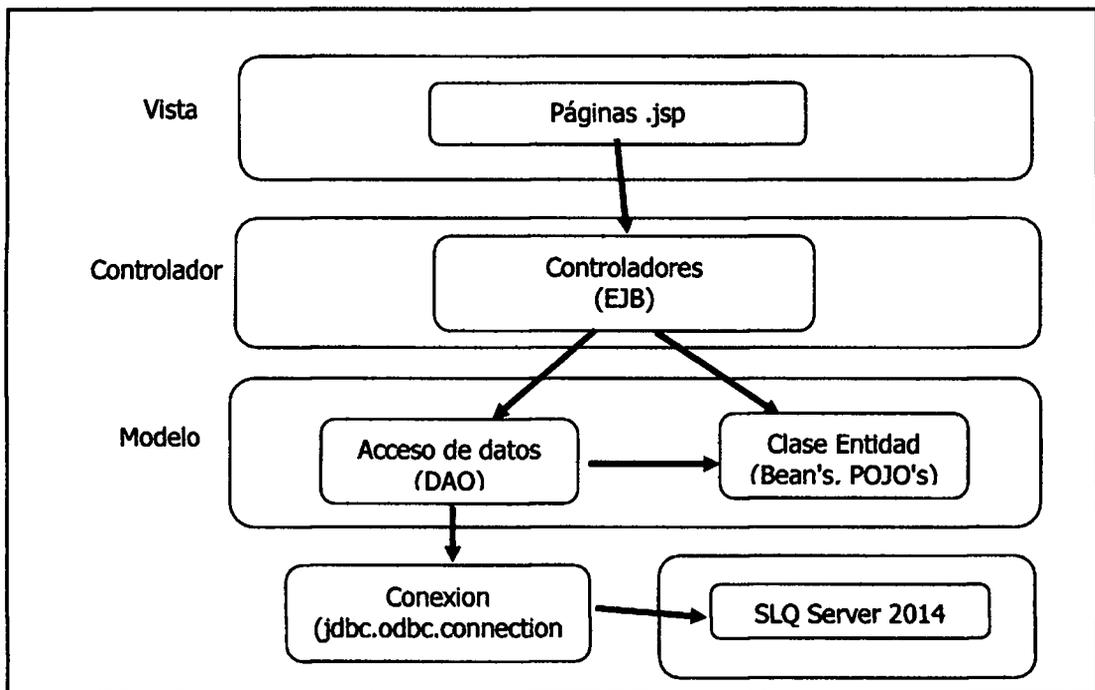


Figura N° 4.2: Arquitectura técnica final de la Aplicación web. Diagrama de componente

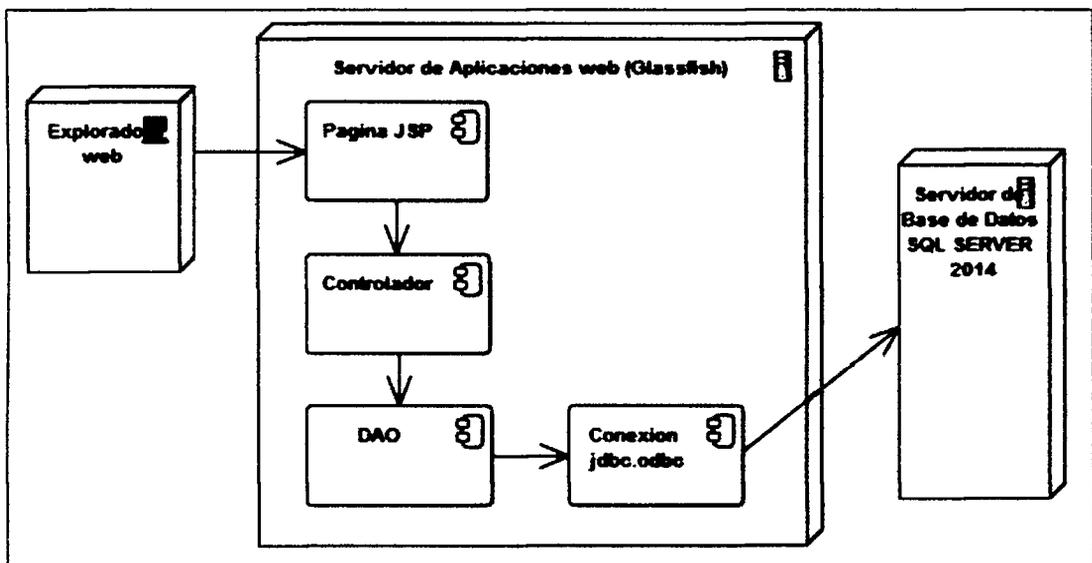


Figura N° 4.3: Arquitectura técnica final de la Aplicación web. Diagrama de despliegue

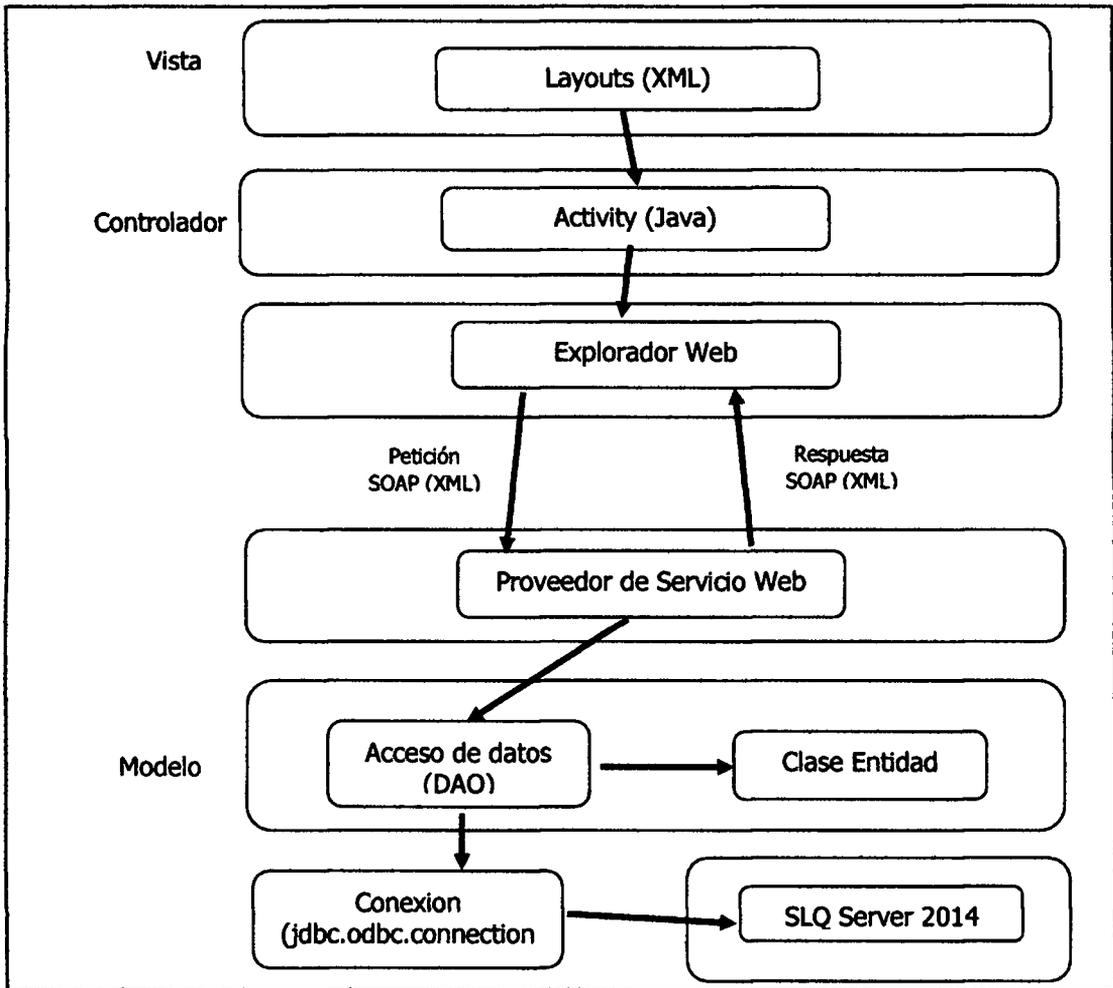


Figura N° 4.4: Arquitectura técnica final de la Aplicación móvil. Diagrama de componente

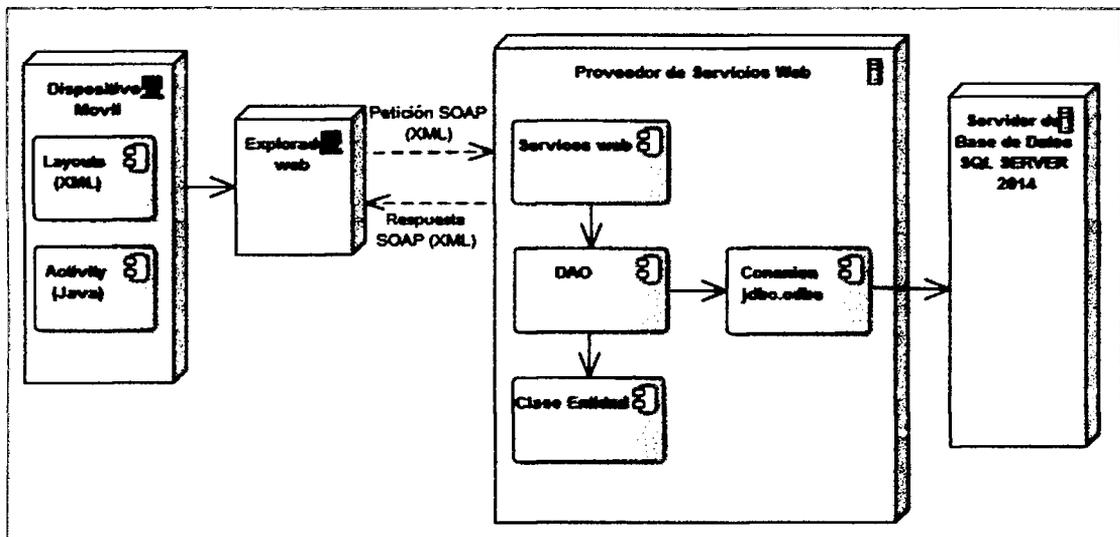


Figura N° 4.5: Arquitectura técnica final de la Aplicación móvil. Diagrama de despliegue

## A TAREAS DE INGENIERÍA DE LA APLICACIÓN WEB

Tarea de Ingeniería	
Número tarea de ingeniería: 1	Número historia de usuario: 1
Nombre tarea: Iniciar sesión en la aplicación web	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 01/06/2016	Fecha fin: 02/06/2016
Programador responsable: Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), ejecuta la aplicación y se muestra la página de Bienvenida donde hace clic en Ingresar, luego la aplicación muestra la página “Login”. El usuario ingresa el nombre de usuario y password; y hace clic en “Enviar”, si los datos son correctos la aplicación muestra la página de “Menú principal”, en caso contrario la aplicación mostrara el mensaje de error “Ingrese un usuario y password correcto”.	

Tabla N° 4.37: Tarea de ingeniería. Iniciar sesión en la aplicación web.

Tarea de Ingeniería	
Número tarea de ingeniería: 2	Número historia de usuario: 2
Nombre tarea: Consultar lista de deudores	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 03/06/2016	Fecha fin: 04/06/2016
Programador responsable: Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), selecciona la opción “Lista de deudores” de la página de “Deudores”, y la aplicación muestra la página “Lista de Deudores” con la relación de todos los deudores de la Cooperativa mostrando su dni, nombres, apellidos y categoría.	

Tabla N° 4.38: Tarea de ingeniería. Consultar lista de deudores.

Tarea de Ingeniería	
Número tarea de ingeniería: 3	Número historia de usuario: 3
Nombre tarea: Consultar lista de nuevos deudores	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 05/06/2016	Fecha fin: 06/06/2016
Programador responsable: Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), selecciona la opción “Nuevos deudores” de la página de “Deudores”, la aplicación muestra la página “Lista de nuevos deudores” con la relación de todos los socios que recién han caído en deuda, mostrando su dni, nombres y apellidos.	

Tabla N° 4.39: Tarea de ingeniería. Consultar lista de nuevos deudores.

Tarea de Ingeniería	
Número tarea de ingeniería: 4	Número historia de usuario: 4
Nombre tarea: Buscar deudor por dni	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 07/06/2016	Fecha fin: 07/06/2016

<b>Programador responsable:</b> Darwin Martin Huamán Huamani
<b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), en la página “Deudores” selecciona la opción “ <b>Buscar deudores</b> ” y la aplicación muestra la página “ <b>Buscar deudores</b> ” en donde el usuario ingresa el dni del deudor y luego hace clic en el botón buscar, la aplicación muestra el dni, nombres, apellidos y categoría del deudor

Tabla N° 4.40: Tarea de ingeniería. Buscar deudor por dni.

<b>Tarea de Ingeniería</b>	
<b>Número tarea de ingeniería:</b> 5	<b>Número historia de usuario:</b> 4
<b>Nombre tarea:</b> Buscar deudor por nombres y apellidos	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 08/06/2016	<b>Fecha fin:</b> 08/06/2016
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), en la página “Deudores” selecciona la opción “ <b>Buscar deudores</b> ” y la aplicación muestra la página “ <b>Buscar deudores</b> ” en donde el usuario ingresa los nombres o apellidos del deudor y luego hace clic en el botón buscar, la aplicación muestra el dni, nombres, apellidos y categoría de los deudores que tienen coincidencia de datos.	

Tabla N° 4.41: Tarea de ingeniería. Buscar deudor por nombres y apellidos.

<b>Tarea de Ingeniería</b>	
<b>Número tarea de ingeniería:</b> 6	<b>Número historia de usuario:</b> 4
<b>Nombre tarea:</b> Buscar deudor por categoría	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 09/06/2016	<b>Fecha fin:</b> 09/06/2016
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), en la página “Deudores” selecciona la opción “ <b>Buscar deudores</b> ” y la aplicación muestra la página “ <b>Buscar deudores</b> ” en donde el usuario selecciona la categoría a la cual pertenece el deudor y luego hace clic en el botón buscar, la aplicación muestra el dni, nombres, apellidos y categoría de los deudores que tienen coincidencia de datos.	

Tabla N° 4.42: Tarea de ingeniería. Buscar deudor por categoría.

<b>Tarea de Ingeniería</b>	
<b>Número tarea de ingeniería:</b> 7	<b>Número historia de usuario:</b> 4
<b>Nombre tarea:</b> Buscar deudor por domicilio	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 09/06/2016	<b>Fecha fin:</b> 09/06/2016
<b>Programador responsable:</b> Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “Deudores” selecciona la opción “ <b>Buscar deudores</b> ” y la aplicación muestra la página “ <b>Buscar deudores</b> ” en donde el usuario selecciona el departamento, luego la provincia y por último el distrito en donde radica el deudor, y luego hace clic en el botón buscar, la aplicación muestra el dni, nombres, apellidos y categoría de los deudores que tienen coincidencia de datos.	

Tabla N° 4.43: Tarea de ingeniería. Buscar deudor por domicilio.

Tarea de Ingeniería	
Número tarea de ingeniería: 8	Número historia de usuario: 5
Nombre tarea: Consultar información del deudor	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Fecha inicio: 10/06/2016	Fecha fin: 12/06/2016
Programador responsable: Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), hace clic en el botón “ <b>Ver detalle</b> ”, que está al lado de los datos del deudor de la Página “ <b>Lista de Deudores</b> ”, y la aplicación muestra la página “ <b>Datos del deudor</b> ” con toda la información del deudor como los datos personales, domicilio, datos de la deuda.	

Tabla N° 4.44: Tarea de ingeniería. Consultar información del deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 9	Número historia de usuario: 6
Nombre tarea: Consultar lista de gestores de cobranzas	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Fecha inicio: 13/06/2016	Fecha fin: 14/06/2016
Programador responsable: Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), selecciona la opción “ <b>Gestor de cobranzas</b> ” de la página “ <b>Gestores de Cobranza</b> ”, y la aplicación muestra la página “ <b>Lista de gestores de cobranzas</b> ” con los nombres, apellidos, fecha de ingreso y numero celular de los gestores de cobranzas que trabajan en la Cooperativa.	

Tabla N° 4.45: Tarea de ingeniería. Consultar lista de gestores de cobranzas.

Tarea de Ingeniería	
Número tarea de ingeniería: 10	Número historia de usuario: 7
Nombre tarea: Consultar información del gestor de cobranzas	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 15/06/2016	Fecha fin: 16/06/2016
Programador responsable: Darwin Martin Huamán Huamani	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), hace clic en el botón “ <b>Ver detalle</b> ”, que está al lado de los datos del gestor de cobranza en la Página “ <b>Lista de gestores de cobranzas</b> ”, y la aplicación muestra la página “ <b>Datos del gestor de cobranza</b> ” con el dni, nombres, apellidos, celular, dirección y fecha de ingreso de gestor de cobranzas.	

Tabla N° 4.46: Tarea de ingeniería. Consultar información del gestor de cobranzas.

Tarea de Ingeniería	
Número tarea de ingeniería: 11	Número historia de usuario: 8
Nombre tarea: Registrar nuevo gestor de cobranza	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 17/06/2016	Fecha fin: 17/06/2016
Programador responsable: Darwin Martin Huamán Huamani	
<b>Descripción:</b>	

El usuario (Jefe de recuperaciones), en la página “**Gestores de cobranza**” selecciona la opción “**Nuevo gestor de cobranza**” y la aplicación muestra la página “**Nuevo gestor de cobranza**”, en donde el usuario ingresa el dni, nombres, apellidos, teléfono, dirección, fecha de ingreso, usuario y password del nuevo gestor, y hace clic en el botón guardar, y la aplicación muestra el mensaje “**Datos registrados correctamente**”

Tabla N° 4.47: Tarea de ingeniería. Registrar nuevo gestor de cobranza.

Tarea de Ingeniería	
Número tarea de ingeniería: 12	Número historia de usuario: 8
Nombre tarea: Actualizar gestor de cobranza	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 18/06/2016	Fecha fin: 18/06/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “ <b>Datos del gestor de cobranza</b> ” hace clic en el botón “ <b>Actualizar datos</b> ”, la aplicación muestra la página “ <b>Actualizar gestor de cobranza</b> ”, en donde el usuario modifica los campos que cree conveniente y luego hace clic en el botón actualizar, y la aplicación muestra el mensaje “ <b>Datos actualizados correctamente</b> ”.	

Tabla N° 4.48: Tarea de ingeniería. Actualizar gestor de cobranza.

Tarea de Ingeniería	
Número tarea de ingeniería: 13	Número historia de usuario: 8
Nombre tarea: Eliminar gestor de cobranza	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 19/06/2016	Fecha fin: 19/06/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “ <b>Datos del gestor de cobranza</b> ” hace clic en el botón “ <b>Eliminar</b> ”, la aplicación muestra el mensaje “ <b>Esta seguro de eliminar al gestor de cobranza</b> ”, si el usuario selecciona SI la aplicación elimina al gestor de la base de datos y muestra el mensaje “ <b>Gestor eliminado correctamente</b> ” y si el usuario selecciona NO la aplicación regresa a la página anterior.	

Tabla N° 4.49: Tarea de ingeniería. Eliminar gestor de cobranza.

Tarea de Ingeniería	
Número tarea de ingeniería: 14	Número historia de usuario: 9
Nombre tarea: Consultar cartera de cobranza por gestor de cobranza	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 20/06/2016	Fecha fin: 22/06/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “ <b>Lista de Gestores de cobranzas</b> ”, elige la opción “ <b>ver cartera</b> ” que está al lado de los datos de cada gestor de cobranza, y la aplicación muestra la página “ <b>Lista de Deudores</b> ” con la relación de todos los deudores que le corresponde a cada gestor de cobranzas.	

Tabla N° 4.50: Tarea de ingeniería. Consultar cartera de cobranza por gestor de cobranza.

Tarea de Ingeniería	
Número tarea de ingeniería: 15	Número historia de usuario: 10
Nombre tarea: Asignar deudor a gestor de cobranza	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 23/06/2016	Fecha fin: 24/06/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “Lista de nuevos deudores” selecciona la opción “Ver detalle” y la aplicación muestra la página “Asignación de deudores” con la información del deudor y una lista desplegable con los nombres de los gestores, luego el usuario selecciona a un gestor de la lista y hace clic en el botón “Asignar”, y la aplicación guarda la información en la base de datos y muestra la página de mensaje “La asignación se realizó con éxito”	

Tabla N° 4.51: Tarea de ingeniería. Asignar deudor a gestor de cobranza.

Tarea de Ingeniería	
Número tarea de ingeniería: 16	Número historia de usuario: 11
Nombre tarea: Consultar notificaciones de pago por entregar	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 25/06/2016	Fecha fin: 26/06/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), en la página “Notificaciones y compromisos” hace clic en la opción “Notificaciones por entregar” y la aplicación muestra la página “Notificaciones por entregar” con los nombres y apellidos de los deudores y el tipo de notificación que deberían de ser entregadas.	

Tabla N° 4.52: Tarea de ingeniería. Consultar notificaciones de pago por entregar.

Tarea de Ingeniería	
Número tarea de ingeniería: 17	Número historia de usuario: 12
Nombre tarea: Consultar notificaciones de pago del deudor	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 27/06/2016	Fecha fin: 29/06/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), en la página “Datos del deudor” hace clic en el botón “Notificaciones” y la aplicación muestra la página “Detalle de notificaciones” con la relación de todas las notificaciones que ya fueron entregadas al deudor, mostrando el tipo de notificación, fecha de entrega, hora y estado.	

Tabla N° 4.53: Tarea de ingeniería. Consultar notificaciones de pago del deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 18	Número historia de usuario: 13
Nombre tarea: Registrar notificación de pago del deudor	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 30/06/2016	Fecha fin: 30/06/2016
Programador responsable: Darwin Martin Huamán Huamaní	

<p><b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), en la página “Detalle de notificaciones” hace clic en el botón “Registrar notificación” y la aplicación muestra la página “Registrar notificación” donde el usuario ingresa el tipo de notificación, A quien se le entrego, fecha, hora, cargo y observaciones, luego hace clic en el botón “Registrar” y la aplicación muestra el mensaje “La notificación se registró correctamente” y almacena la información en la base de datos.</p>
---

Tabla N° 4.54: Tarea de ingeniería. Registrar notificación de pago del deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 19	Número historia de usuario: 14
Nombre tarea: Consultar compromisos de pago	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 01/07/2016	Fecha fin: 01/07/2016
Programador responsable: Darwin Martin Huamán Huamani	
<p><b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “Notificaciones y compromisos” hace clic en la opción “Compromisos de pago” y la aplicación muestra la página “Compromisos de pago” con la relación de todos los compromisos de pago acordado entre los deudores y los gestores de cobranza, mostrando los nombres, apellidos, tipo de contacto, fecha de contacto, fecha de compromiso y observaciones.</p>	

Tabla N° 4.55: Tarea de ingeniería. Consultar compromisos de pago.

Tarea de Ingeniería	
Número tarea de ingeniería: 20	Número historia de usuario: 15
Nombre tarea: Consultar compromisos de pago del deudor	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 02/07/2016	Fecha fin: 03/07/2016
Programador responsable: Darwin Martin Huamán Huamani	
<p><b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), en la página “Datos del deudor” hace clic en el botón “Compromisos” y la aplicación muestra la página “Compromisos de pago del deudor” con la relación de todos los compromisos de pago efectuados por el deudor, mostrando los nombres, apellidos, tipo de contacto, fecha de contacto, fecha de compromiso y observaciones.</p>	

Tabla N° 4.56: Tarea de ingeniería. Consultar compromisos de pago del deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 21	Número historia de usuario: 16
Nombre tarea: Registrar compromisos de pago del deudor	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 04/07/2016	Fecha fin: 04/07/2016
Programador responsable: Darwin Martin Huamán Huamani	
<p><b>Descripción:</b> El usuario (Jefe de recuperaciones, gestor de cobranza), en la página “Compromisos de pago del deudor” hace clic en el botón “Registrar compromiso” y la aplicación muestra la página “Registrar compromiso de pago del deudor” donde el deudor</p>	

ingresa el tipo de contacto, fecha de contacto, fecha de compromiso y observaciones y luego hace clic en el botón Registrar y la aplicación muestra el mensaje “**Datos registrados correctamente**” y almacena la información en la base de datos.

Tabla N° 4.57: Tarea de ingeniería. Registrar compromisos de pago del deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 22	Número historia de usuario: 17
Nombre tarea: Clasificar deudor	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 26/07/2016	Fecha fin: 27/07/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza, gestor de cobranza), pulsa el botón “ <b>Clasificar deudor</b> ” en la página “ <b>Datos del deudor</b> ”, y la aplicación muestra la página “ <b>Clasificación del deudor</b> ”, donde el usuario responde las preguntas de la capacidad y voluntad de pago del deudor en cada checkbox y luego pulsa el botón procesar, y la aplicación procesa la información y lo almacena en la base de datos.	

Tabla N° 4.58: Tarea de ingeniería. Clasificar deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 23	Número historia de usuario: 18
Nombre tarea: Emitir reporte de cobranza	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 05/07/2016	Fecha fin: 07/07/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (Jefe de recuperaciones), en la página “ <b>Menú principal</b> ” selecciona la opción “ <b>Reportes</b> ” y la aplicación muestra la página “ <b>Reportes</b> ”, donde el usuario selecciona el tipo de reporte y luego hace clic en Ejecutar y la aplicación muestra el reporte generado.	

Tabla N° 4.59: Tarea de ingeniería. Emitir reporte de cobranza.

## B TAREAS DE INGENIERÍA DE LA APLICACIÓN MOVIL

Tarea de Ingeniería	
Número tarea de ingeniería: 1	Número historia de usuario: 1
Nombre tarea: Iniciar sesión en la aplicación móvil	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 08/07/2016	Fecha fin: 09/07/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), selecciona el icono de la aplicación Android en el dispositivo móvil y se muestra ventana de “ <b>Login</b> ”. El usuario ingresa el nombre de usuario y contraseña; pulsa el botón “ <b>Iniciar sesión</b> ”, si los datos ingresados fueron correctos la aplicación muestra la ventana principal y el mensaje “ <b>Sesión iniciada</b> ”, en caso contrario la aplicación mostrara el mensaje de error “ <b>Usuario o contraseña incorrecta</b> ”.	

Tabla N° 4.60: Tarea de ingeniería. Iniciar sesión en la aplicación móvil.

Tarea de Ingeniería	
Número tarea de ingeniería: 2	Número historia de usuario: 2
Nombre tarea: Consultar lista de deudores	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 10/07/2016	Fecha fin: 11/07/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa la opción “ <b>Lista de deudores</b> ” de la ventana “ <b>Principal</b> ”, y la aplicación muestra la ventana “ <b>Lista de deudores</b> ” con los nombres, apellidos de los deudores que están en su cartera en cobranza.	

Tabla N° 4.61: Tarea de ingeniería. Consultar lista de deudores.

Tarea de Ingeniería	
Número tarea de ingeniería: 3	Número historia de usuario: 3
Nombre tarea: Consultar información del deudor	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 12/07/2016	Fecha fin: 13/07/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el nombre de un deudor de la ventana “ <b>Lista de deudores</b> ”, y la aplicación muestra la ventana “ <b>Datos del deudor</b> ” con la información del deudor, como nombres y apellidos, dirección y datos de la deuda.	

Tabla N° 4.62: Tarea de ingeniería. Consultar información del deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 4	Número historia de usuario: 4
Nombre tarea: Consultar notificaciones de pago del deudor	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 14/07/2016	Fecha fin: 15/07/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “ <b>Notificaciones de pago</b> ” de la ventana “ <b>Datos del deudor</b> ”, y la aplicación muestra la ventana “ <b>Notificaciones de pago</b> ” con la información de las notificaciones que ya han sido entregadas al deudor.	

Tabla N° 4.63: Tarea de ingeniería. Consultar notificaciones de pago del deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 5	Número historia de usuario: 5
Nombre tarea: Registrar notificaciones de pago del deudor	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 16/07/2016	Fecha fin: 17/07/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “ <b>Registrar notificación</b> ” de la ventana “ <b>Notificaciones de pago</b> ”, y la aplicación muestra la ventana “ <b>Registrar notificación</b> ”, donde el usuario selecciona el tipo de notificación e ingresa la hora y la	

fecha en que es entregada la notificación, luego pulsa el botón guardar y la aplicación almacena la información.

Tabla N° 4.64: Tarea de ingeniería. Registrar notificaciones de pago del deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 6	Número historia de usuario: 6
Nombre tarea: Consultar compromisos de pago del deudor	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 18/07/2016	Fecha fin: 19/07/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “ <b>Compromisos de pago</b> ” de la ventana “ <b>Datos del deudor</b> ”, y la aplicación muestra la ventana “ <b>Compromisos de pago</b> ” con la información de los compromisos de pago realizados por el deudor.	

Tabla N° 4.65: Tarea de ingeniería. Consultar compromisos de pago del deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 7	Número historia de usuario: 7
Nombre tarea: Registrar compromisos de pago del deudor	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 20/07/2016	Fecha fin: 21/07/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “ <b>Registrar compromiso</b> ” en la ventana “ <b>Compromisos de pago</b> ”, y la aplicación muestra la ventana “ <b>Registrar compromisos</b> ”, donde el usuario ingresa el tipo de contacto, la fecha de contacto, fecha de compromiso y luego pulsa el botón “ <b>Guardar</b> ” y la aplicación almacena la información en la base de datos.	

Tabla N° 4.66: Tarea de ingeniería. Registrar compromisos de pago del deudor.

Tarea de Ingeniería	
Número tarea de ingeniería: 8	Número historia de usuario: 8
Nombre tarea: Consultar notificaciones de pago por entregar	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 22/07/2016	Fecha fin: 23/07/2016
Programador responsable: Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa la opción “ <b>Notificaciones y compromisos</b> ” de la ventana “ <b>Principal</b> ”, y la aplicación muestra la ventana “ <b>Notificaciones y compromisos</b> ” luego pulsa el botón “ <b>Notificaciones</b> ” y la aplicación muestra la ventana “ <b>Notificaciones por entregar</b> ”, con la lista de las notificaciones que deberían de ser entregadas.	

Tabla N° 4.67: Tarea de ingeniería. Consultar notificaciones de pago por entregar.

Tarea de Ingeniería	
Número tarea de ingeniería: 9	Número historia de usuario: 9
Nombre tarea: Consultar compromisos de pago del día	

<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 24/07/2016	<b>Fecha fin:</b> 25/07/2016
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa la opción “Notificaciones y compromisos” de la ventana “Principal”, y la aplicación muestra la ventana “Notificaciones y compromisos” luego pulsa el botón “Compromisos” y la aplicación muestra la ventana “Compromisos del día”, con la lista de los compromisos de pago que han sido acordados para el día de hoy.	

Tabla N° 4.68: Tarea de ingeniería. Consultar compromisos de pago del día.

<b>Tarea de Ingeniería</b>	
<b>Número tarea de ingeniería:</b> 10	<b>Número historia de usuario:</b> 10
<b>Nombre tarea:</b> Clasificar deudor	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 26/07/2016	<b>Fecha fin:</b> 27/07/2016
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “Clasificar deudor” en la ventana “Datos del deudor”, y la aplicación muestra la ventana “Clasificación del deudor”, donde el usuario responde las preguntas de la capacidad y voluntad de pago del deudor en cada checkbox y luego pulsa el botón procesar, y la aplicación procesa la información y lo almacena en la base de datos.	

Tabla N° 4.69: Tarea de ingeniería. Clasificar deudor.

<b>Tarea de Ingeniería</b>	
<b>Número tarea de ingeniería:</b> 11	<b>Número historia de usuario:</b> 11
<b>Nombre tarea:</b> Consultar información del garante	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 28/07/2016	<b>Fecha fin:</b> 29/07/2016
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “Datos del garante” de la ventana “Datos del deudor” y la aplicación muestra la ventana “Datos del garante” con la información del garante.	

Tabla N° 4.70: Tarea de ingeniería. Consultar información del garante.

<b>Tarea de Ingeniería</b>	
<b>Número tarea de ingeniería:</b> 12	<b>Número historia de usuario:</b> 12
<b>Nombre tarea:</b> Consultar ubicación del domicilio del deudor en el mapa	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 30/07/2016	<b>Fecha fin:</b> 31/07/2016
<b>Programador responsable:</b> Darwin Martin Huamán Huamaní	
<b>Descripción:</b> El usuario (gestor de cobranza), pulsa el botón “Mostrar dirección en mapa” en la ventana “Datos del deudor”, y la aplicación muestra la ventana “Dirección del deudor”, donde el usuario puede ubicar la dirección del deudor en el mapa de la aplicación.	

Tabla N° 4.71: Tarea de ingeniería. Consultar información del garante.

## A PLAN DE ITERACIÓN DE LA APLICACIÓN WEB

Nº	HISTORIA DE USUARIO	Nº	TAREAS DE INGENIERÍA
01	Iniciar sesión en la aplicación web	01	Iniciar sesión en la aplicación web
02	Consultar lista de deudores	02	Consultar lista de deudores
03	Consultar lista de nuevos deudores	03	Consultar lista de nuevos deudores
04	Buscar deudor	04	Buscar deudor por DNI
		05	Buscar deudor por nombres y apellidos
		06	Buscar deudor por categoría
		07	Buscar deudor por domicilio
05	Consultar información del deudor	08	Consultar información del deudor
06	Consultar lista de gestores de cobranzas	09	Consultar lista de gestores de cobranzas
07	Consultar información del gestor de cobranzas	10	Consultar información del gestor de cobranzas
08	Administrar gestor de cobranza	11	Registrar nuevo gestor de cobranza
		12	Actualizar gestor de cobranza
		13	Eliminar gestor de cobranza
09	Consultar cartera de cobranza por gestor de cobranza	14	Consultar cartera de cobranza por gestor de cobranza
10	Asignar deudor a gestor de cobranza	15	Asignar deudor a gestor de cobranza
11	Consultar notificaciones de pago por entregar	16	Consultar notificaciones de pago por entregar
12	Consultar notificaciones de pago del deudor	17	Consultar notificaciones de pago del deudor
13	Registrar notificación de pago del deudor	18	Registrar notificación de pago del deudor
14	Consultar compromisos de pago	19	Consultar compromisos de pago
15	Consultar compromisos de pago del deudor	20	Consultar compromisos de pago del deudor
16	Registrar compromisos de pago del deudor	21	Registrar compromisos de pago del deudor
17	Clasificar deudor	22	Clasificar deudor
18	Emitir reporte de cobranza	23	Emitir reporte de cobranza

Tabla N° 4.72: Plan de iteración de la aplicación web

## B PLAN DE ITERACIÓN DE LA APLICACIÓN MOVIL

Nº	HISTORIA DE USUARIO	Nº	TAREAS DE INGENIERÍA
01	Iniciar sesión en la aplicación móvil.	01	Iniciar sesión en la aplicación móvil.
02	Consultar lista de deudores	02	Consultar lista de deudores
03	Consultar información del deudor	03	Consultar información del deudor
04	Consultar notificaciones de pago del deudor	04	Consultar notificaciones de pago del deudor

05	Registrar notificación de pago del deudor	05	Registrar notificación de pago del deudor
06	Consultar compromisos de pago del deudor	06	Consultar compromisos de pago del deudor
07	Registrar compromisos de pago del deudor	07	Registrar compromisos de pago del deudor
08	Consultar notificaciones de pago por entregar	08	Consultar notificaciones de pago por entregar
09	Consultar compromisos de pago del día	09	Consultar compromisos de pago del día
10	Clasificar deudor	10	Clasificar deudor
11	Consultar información del garante	11	Consultar información del garante
12	Consultar ubicación del domicilio del deudor en el mapa de la aplicación	12	Consultar ubicación del domicilio del deudor en el mapa de la aplicación

Tabla N° 4.73: Plan de iteración de la aplicación móvil

N° T. I	N° H. U.	FECHA INICIO	FECHA FIN	PROGRAMADOR
1	1	01/06/2016	02/06/2016	Darwin Martin Huamán Huamaní
2	2	03/06/2016	04/06/2016	Darwin Martin Huamán Huamaní
3	3	05/06/2016	06/06/2016	Darwin Martin Huamán Huamaní
4	4	07/06/2016	07/06/2016	Darwin Martin Huamán Huamaní
5	4	08/06/2016	08/06/2016	Darwin Martin Huamán Huamaní
6	4	09/06/2016	09/06/2016	Darwin Martin Huamán Huamaní
7	4	09/06/2016	09/06/2016	Darwin Martin Huamán Huamaní
8	5	10/06/2016	12/06/2016	Darwin Martin Huamán Huamaní
9	6	13/06/2016	14/06/2016	Darwin Martin Huamán Huamaní
10	7	15/06/2016	16/06/2016	Darwin Martin Huamán Huamaní
11	8	17/06/2016	17/06/2016	Darwin Martin Huamán Huamaní
12	8	18/06/2016	18/06/2016	Darwin Martin Huamán Huamaní
13	8	19/06/2016	19/06/2016	Darwin Martin Huamán Huamaní
14	9	20/06/2016	22/06/2016	Darwin Martin Huamán Huamaní
15	10	23/06/2016	24/06/2016	Darwin Martin Huamán Huamaní
16	11	25/06/2016	26/06/2016	Darwin Martin Huamán Huamaní
17	12	27/06/2016	29/06/2016	Darwin Martin Huamán Huamaní
18	13	30/06/2016	30/06/2016	Darwin Martin Huamán Huamaní
19	14	01/07/2016	01/07/2016	Darwin Martin Huamán Huamaní
20	15	02/07/2016	03/07/2016	Darwin Martin Huamán Huamaní
21	16	04/07/2016	04/07/2016	Darwin Martin Huamán Huamaní
22	17	05/07/2016	06/07/2016	Darwin Martin Huamán Huamaní
23	18	06/07/2016	07/07/2016	Darwin Martin Huamán Huamaní

Tabla N° 4.74: Plan de iteración clasificados en fechas de desarrollo de la aplicación web.

Nº T. I	Nº H. U.	FECHA INICIO	FECHA FIN	PROGRAMADOR
1	1	08/07/2016	09/07/2016	Darwin Martin Huamán Huamani
2	2	10/07/2016	11/07/2016	Darwin Martin Huamán Huamani
3	3	12/07/2016	13/07/2016	Darwin Martin Huamán Huamani
4	4	14/07/2016	15/07/2016	Darwin Martin Huamán Huamani
5	5	16/07/2016	17/07/2016	Darwin Martin Huamán Huamani
6	6	18/07/2016	19/07/2016	Darwin Martin Huamán Huamani
7	7	20/07/2016	21/07/2016	Darwin Martin Huamán Huamani
8	8	22/07/2016	23/07/2016	Darwin Martin Huamán Huamani
9	9	24/07/2016	25/07/2016	Darwin Martin Huamán Huamani
10	10	26/07/2016	27/07/2016	Darwin Martin Huamán Huamani
11	11	28/07/2016	29/07/2016	Darwin Martin Huamán Huamani
12	12	30/07/2016	31/07/2016	Darwin Martin Huamán Huamani

Tabla Nº 4.75: Plan de iteración clasificados en fechas de desarrollo de la aplicación móvil.

Nº H.U.	REQUISITO	Nº C.P.	CASO DE PRUEBA (C.P.)
1	La aplicación debe ser capaz de iniciar una sesión.	1	Verificar que el nombre de usuario y password son correcto y permitir el acceso al sistema, y si son incorrectos mostrar mensaje de error.
2	La aplicación debe ser capaz de mostrar la lista de todos los deudores de la cooperativa	2	Comprobar que la aplicación muestra la lista de todos los deudores de la cooperativa
3	La aplicación debe ser capaz de mostrar la lista de los nuevos deudores de la cooperativa	3	Comprobar que la aplicación muestra la lista de los nuevos deudores de la cooperativa
4	La aplicación debe ser capaz de buscar deudores por el dni.	4	Verificar que la aplicación busca al deudor por el dni ingresado
	La aplicación debe ser capaz de buscar deudores ingresando los nombres o apellidos.	5	Verificar que la aplicación busca al deudor por los nombres o apellidos ingresados
	La aplicación debe ser capaz de buscar deudores por categoría.	6	Verificar que la aplicación busca al deudor por el tipo de categoría.
	La aplicación debe ser capaz de buscar deudores seleccionado el departamento, provincia y distrito en donde vive el deudor.	7	Verificar que la aplicación busca al deudor por el departamento, provincia y distrito en donde vive.
5	La aplicación debe ser capaz de mostrar la información del deudor	8	Comprobar que la aplicación muestra toda la información del deudor
6	La aplicación debe ser capaz de mostrar la lista de los gestores de cobranza que laboran en la cooperativa.	9	Comprobar que la aplicación muestra la lista de los gestores de cobranza que laboran en la cooperativa.
7	La aplicación debe ser capaz de mostrar la información del gestor de cobranza	10	Comprobar que la aplicación muestra toda la información del gestor de cobranza

8	La aplicación debe ser capaz de registrar a un nuevo gestor de cobranza.	11	Verificar que la aplicación valida los datos de ingreso, registra al nuevo gestor de cobranza y muestra el mensaje de confirmación.
	La aplicación debe ser capaz de actualizar la información del gestor de cobranza.	12	Verificar que la aplicación valida los datos de ingreso, actualiza la información del gestor de cobranza y muestra el mensaje de confirmación.
	La aplicación debe ser capaz de eliminar a un gestor de cobranza de la base de datos.	13	Verificar que la aplicación muestra el mensaje de advertencia, elimina al gestor de cobranza y muestra el mensaje de confirmación.
9	La aplicación debe ser capaz de mostrar la lista de deudores que le corresponde a cada gestor de cobranza	14	Comprobar que la aplicación muestra la lista de deudores que le corresponde a cada gestor de cobranza.
10	La aplicación debe ser capaz de mostrar los datos del nuevo deudor y cargar la lista de gestores de cobranza para la asignación.	15	Comprobar que la aplicación muestra los datos del nuevo deudor y carga la lista de los gestores de cobranza para la asignación.
	La aplicación debe ser capaz de asignar al nuevo deudor a un gestor de cobranza.	16	Verificar que la aplicación asigna al nuevo deudor a un gestor de cobranza seleccionado.
11	La aplicación debe ser capaz de mostrar la lista de notificaciones de pago por entregar	17	Comprobar que la aplicación muestra la lista de notificaciones de pago por entregar.
12	La aplicación debe ser capaz de mostrar las notificaciones de pago de cada deudor.	18	Comprobar que la aplicación muestra las notificaciones de pago de cada deudor.
13	La aplicación debe ser capaz de registrar las notificaciones de pago del deudor.	19	Verificar que la aplicación valida los datos de ingreso, registra la notificación de pago y muestra el mensaje de confirmación.
14	La aplicación debe ser capaz de mostrar la lista de compromisos de pago de todos los deudores.	20	Comprobar que la aplicación muestra la lista de compromisos de pago de todos los deudores.
15	La aplicación debe ser capaz de mostrar los compromisos de pago de cada deudor.	21	Comprobar que la aplicación muestra los compromisos de pago de cada deudor.
16	La aplicación debe ser capaz de registrar el compromiso de pago de cada deudor	22	Verificar que la aplicación valida los datos de ingreso, registra el compromiso de pago y muestra el mensaje de confirmación.
17	La aplicación debe ser capaz de clasificar al deudor a través de un cuestionario.	23	Verificar que la aplicación clasifica al deudor a través de un cuestionario.
18	La aplicación debe ser capaz de emitir reportes de cobranza	24	Comprobar que la aplicación emite reportes de cobranza.

Tabla N° 4.76: Casos de prueba de aceptación de la aplicación web.

Nº H.U.	REQUISITO	Nº C.P.	CASO DE PRUEBA (C.P.)
1	La aplicación móvil debe ser capaz de iniciar una sesión.	1	Verificar que el nombre de usuario y password son correcto y permitir el acceso al sistema, y si son incorrectos mostrar mensaje de error.
2	La aplicación móvil debe ser capaz de mostrar la lista de deudores que le corresponde a cada gestor de cobranza.	2	Comprobar que la aplicación móvil muestra la lista de deudores que le corresponde a cada gestor de cobranza.
3	La aplicación móvil debe ser capaz de mostrar la información del deudor	3	Comprobar que la aplicación móvil muestra la información necesaria del deudor.
4	La aplicación móvil debe ser capaz de mostrar las notificaciones de pago de cada deudor.	4	Comprobar que la aplicación móvil muestra las notificaciones de pago de cada deudor.
5	La aplicación móvil debe ser capaz de registrar las notificaciones de pago del deudor.	5	Verificar que la aplicación móvil valida los datos de ingreso, registra la notificación de pago y muestra el mensaje de confirmación.
6	La aplicación móvil debe ser capaz de mostrar los compromisos de pago de cada deudor.	6	Comprobar que la aplicación móvil muestra los compromisos de pago de cada deudor.
7	La aplicación móvil debe ser capaz de registrar los compromisos de pago del deudor.	7	Verificar que la aplicación móvil valida los datos de ingreso, registra los compromisos de pago y muestra el mensaje de confirmación.
8	La aplicación móvil debe ser capaz de mostrar la lista de notificaciones de pago por entregar	8	Comprobar que la aplicación móvil muestra la lista de notificaciones de pago por entregar.
9	La aplicación móvil debe ser capaz de mostrar la lista de los compromisos del pago del día.	9	Comprobar que la aplicación móvil muestra la lista de compromisos de pago del día.
11	La aplicación móvil debe ser capaz de clasificar al deudor a través de un cuestionario.	10	Verificar que la aplicación móvil clasifica al deudor a través de un cuestionario.
11	La aplicación móvil debe ser capaz de mostrar la información del garante del deudor.	11	Comprobar que la aplicación móvil muestra la información del garante del deudor.
12	La aplicación móvil debe ser capaz de mostrar a través de un mapa la ubicación del domicilio del deudor.	12	Comprobar que la aplicación móvil muestra a través de un mapa la ubicación del domicilio del deudor.

Tabla Nº 4.77: Casos de prueba de aceptación de la aplicación móvil.

## INTERFAZ DE USUARIO

En el diseño de los interfaces se consideró estándares de usabilidad, según los objetivos de la tesis no considera su evaluación; los factores para estándares utilizados son: botones, títulos de páginas y menús, fuentes y colores, tamaños de fuentes, etc., que permiten tener interfaces amigables y, fáciles de usar.

### A. INTERFACES DE LA APLICACIÓN WEB

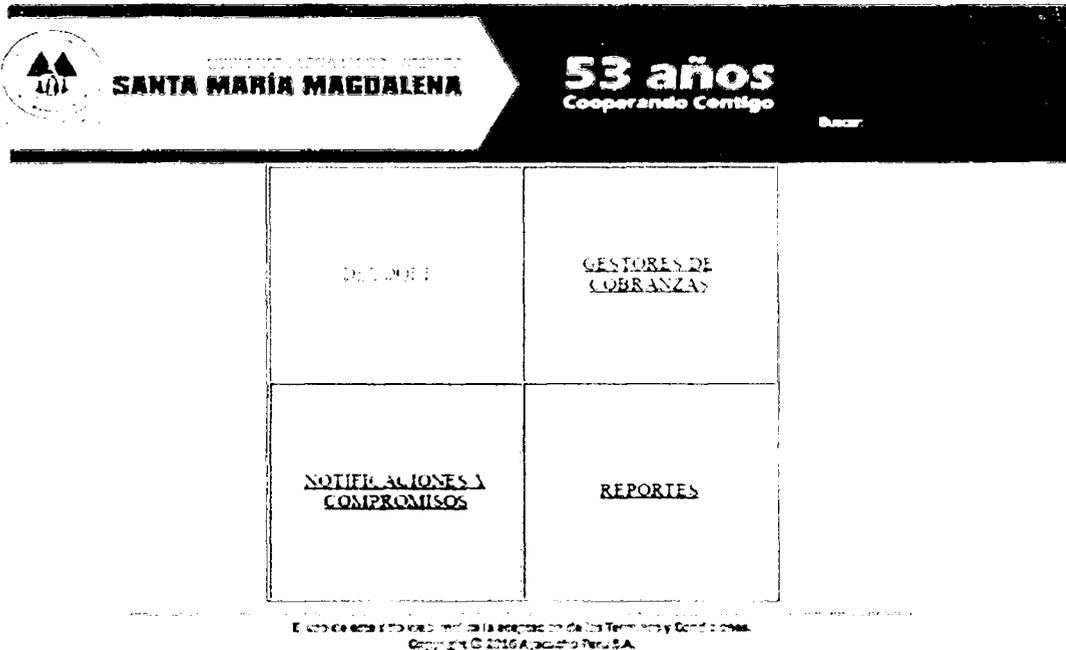


Figura N° 4.6: Interfaz menú principal

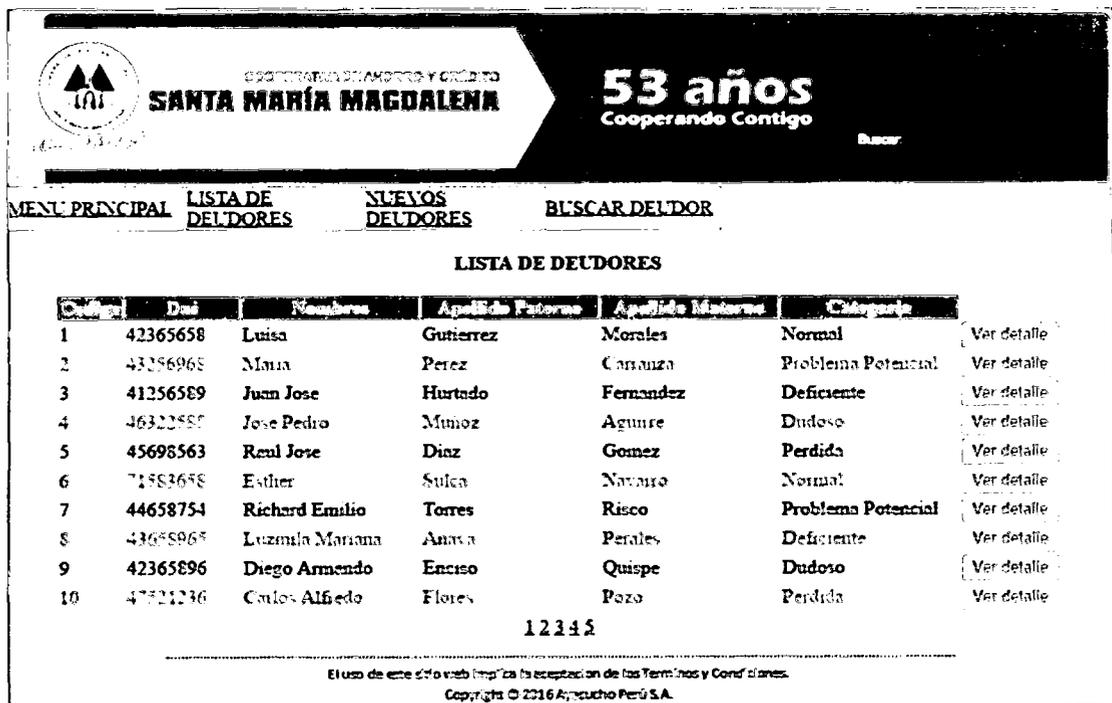


Figura N° 4.7: Interfaz lista de deudores



COOPERATIVA DE AHORRO Y CRÉDITO  
**SANTA MARÍA MAGDALENA**

**53 años**  
Cooperando Contigo

---

[MENU PRINCIPAL](#)
[LISTA DE DEUDORES](#)
[NUEVOS DEUDORES](#)
[BUSCAR DEUDOR](#)

**DATOS PERSONALES DEL DEUDOR**

Nombre y Apellidos: Luisa Gutierrez Morales		Cltar: A	
Estado Civil: Casada	Sexo: Femenino	Edad: 35	
Tel Cel: 966586969	E-mail: luise@hotmail.com		
Departamento: Ayacucho	Provincia: Huasmanga	Distrito: Ayacucho	
Direccion: Jr Lima N° 355			

**CONYUGE**

Nombre y Apellidos: Juan Gonzales Fernandez			
Estado Civil: Casado	Sexo: Masculino	Edad: 35	
Tel Cel: 958636556	E-mail: juan@hotmail.com		

**CREMITO**

Producto: Mipymes	Tipo: Ordinario	Destino: Capital de trabajo
Monto: 10000	Plazo: 12	Moneda: Soles

**DEUDA**

Numero de cuota: 2	Fecha de pago: 01-08-2016	Dias de atraso: 2
Capital: 737.25	Interes: 220.00	Deuda: 958.25

**GARANTE**

Nombre y Apellidos: Carlos Manuel Aguirre Casavilca			
Tel Cel: 956363654	E-mail: cmanuel@hotmail.com		
Departamento: Ayacucho	Provincia: Huasmanga	Distrito: San Juan Bautista	
Direccion: Jr Ica N° 258			

El uso de este sitio web implica la aceptación de los Términos y Condiciones.  
Copyright © 2016 Ayacucho Perú S.A.

Figura N° 4.8: Interfaz datos del deudor



COOPERATIVA DE AHORRO Y CRÉDITO  
**SANTA MARÍA MAGDALENA**

**53 años**  
Cooperando Contigo

---

[MENU PRINCIPAL](#)
[NOTIFICACIONES POR ENTREGAR](#)
[COMPROMISOS DE PAGO](#)

**NOTIFICACIONES**

Nombres y Apellidos: Luisa Gutierrez Morales

N°	Tipo de Notificación	Fecha de entrega	Hora	Estado
1	Aviso de vencimiento	04/06/2016	10:00 AM	Entregado
2	Aviso de cobranza	15 06 2016	11 00 AM	Entregado
3	Requerimiento de pago	08/06/2016	9:00 am	Entregado
4	Notificación urgente	01 02 2016	9:00 am	Entregado
5	Ultima notificación	08/06/2016	9:00 am	Entregado
6	Carta notarial			No entregado

El uso de este sitio web implica la aceptación de los Términos y Condiciones.  
Copyright © 2016 Ayacucho Perú S.A.

Figura N° 4.9: Interfaz notificaciones



**COOPERATIVA DE AHORRO Y CREDITO  
SANTA MARÍA MAGDALENA**

**53 años**  
Cooperando Contigo

Buscar

---

**MENU PRINCIPAL**   **LISTA DE DEUDORES**   **NUEVOS DEUDORES**   **BUSCAR DEUDOR**

**BUSCAR POR DNI**

Numero de DNI	
	Buscar

**BUSCAR POR NOMBRES Y APELLIDOS**

Nombres	
Apellido Paterno	
Apellido Materno	
	Buscar

**BUSCAR POR CATEGORIA**

Tipo de categoria	Normal ▾
	Buscar

**BUSCAR POR DIRECCION**

Departamento:	-Seleccione un departamento ▾	
Provincia:	-Seleccione una provincia ▾	
DISTRITO:	-Seleccione un distrito ▾	
		Buscar

El uso de este sitio web implica la aceptación de los Términos y Condiciones.  
Copyright © 2016 Ayacucho Perú S.A.

Figura N° 4.10: Interfaz buscar deudor



**COOPERATIVA DE AHORRO Y CREDITO  
SANTA MARÍA MAGDALENA**

**53 años**  
Cooperando Contigo

Buscar

---

**MENU PRINCIPAL**   **NOTIFICACIONES POR ENTREGAR**   **COMPROMISOS DE PAGO**

Tipo de contacto:

Codigo del deudor:

Fecha de contacto:

Fecha de compromiso:

Observacion:

**Agosto 2016**

Do	Lu	Ma	Mi	Ju	Vi	Sa
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

El uso de este sitio web implica la aceptación de los Términos y Condiciones.  
Copyright © 2016 Ayacucho Perú S.A.

Figura N° 4.11: Interfaz registrar compromiso



COOPERATIVA DE AHORRO Y CRÉDITO  
**SANTA MARÍA MAGDALENA**

**53 años**  
Cooperando Contigo

Inicio

MENU PRINCIPAL   GESTORES DE COBRANZAS   ASIGNAR DELDOR   NUEVO GESTOR DE COBRANZAS

**GESTORES DE COBRANZAS**

Nombre	Apellido Paterno	Apellido Materno	Fecha de Ingreso	Códnig	Ver detalle	Ver cartera
Jose Raul	Montes	Reyes	25/03/2014	987541236	Ver detalle	Ver cartera
Fredy Richard	Pidenn	Rodriguez	15/05/2014	966321456	Ver detalle	Ver cartera
Wilder	Dipaz	Gutierrez	20/10/2015	925635465	Ver detalle	Ver cartera
Jorge	Cardenas	Ramirez	05/01/2015	966754869	Ver detalle	Ver cartera
Laura	Quispe	Lujan	12/08/2015	966325847	Ver detalle	Ver cartera
Saida	Morales	Pelaez	24/09/2015	987459632	Ver detalle	Ver cartera
Diego Armando	Hurtado	Perales	24/09/2015	987459632	Ver detalle	Ver cartera
Pedro Jose	Morales	Garcia	24/09/2015	987459632	Ver detalle	Ver cartera
Mario	Torres	Sicha	24/09/2015	987459632	Ver detalle	Ver cartera
Rosa Maria	Lopez	Gamarra	15/05/2015	965236587	Ver detalle	Ver cartera
Diana	LLanos	Quispe	14/02/2015	987546585	Ver detalle	Ver cartera

El uso de este sitio web implica la aceptación de los Términos y Condiciones.  
Copyright © 2016 Ayscucho Perú S.A.

Figura N° 4.12: Interfaz gestores de cobranza



COOPERATIVA DE AHORRO Y CRÉDITO  
**SANTA MARÍA MAGDALENA**

**53 años**  
Cooperando Contigo

Inicio

MENU PRINCIPAL   GESTORES DE COBRANZAS   ASIGNAR DELDOR   NUEVO GESTOR DE COBRANZAS

Dni: \_\_\_\_\_

Nombres: \_\_\_\_\_

Apellido Paterno: \_\_\_\_\_

Apellido Materno: \_\_\_\_\_

Telefono: \_\_\_\_\_

Direccion: \_\_\_\_\_

Fecha de Ingreso: \_\_\_\_\_

Usuario: \_\_\_\_\_

Password: \_\_\_\_\_

El uso de este sitio web implica la aceptación de los Términos y Condiciones.  
Copyright © 2016 Ayscucho Perú S.A.

Figura N° 4.13: Interfaz registrar nuevo gestor de cobranza



COOPERATIVA DE AHORRO Y CRÉDITO  
**SANTA MARÍA MAGDALENA**

**53 años**  
Cooperando Contigo

Buscar

**MENU PRINCIPAL**   **NOTIFICACIONES POR ENTREGAR**   **COMPROMISOS DE PAGO**

Tipo de notificación: **Aviso de vencimiento**   A quien se entrego: **Titular**

Codigo del deudor:

Fecha:

Hora:

Cargo:

Observacion:

El uso de este sitio web implica la aceptación de los Términos y Condiciones.  
Copyright © 2016 Ayacucho Perú S.A.

Figura N° 4.14: Interfaz registrar notificación



COOPERATIVA DE AHORRO Y CRÉDITO  
**SANTA MARÍA MAGDALENA**

**53 años**  
Cooperando Contigo

Buscar

**CLASIFICACION DEL DEUDOR**

PREGUNTAS	SI	NO
¿Tiene actitudes por salvar el crédito?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
¿Es amable y cordial con sus acreedores?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
¿Está en capacidad de atender las obligaciones?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
¿Financieramente cuenta con solvencia económica?	<input type="checkbox"/>	<input checked="" type="checkbox"/>

El uso de este sitio web implica la aceptación de los Términos y Condiciones.  
Copyright © 2016 Ayacucho Perú S.A.

Figura N° 4.15: Interfaz clasificación del deudor

## B. INTERFACES DE LA APLICACIÓN MOVIL

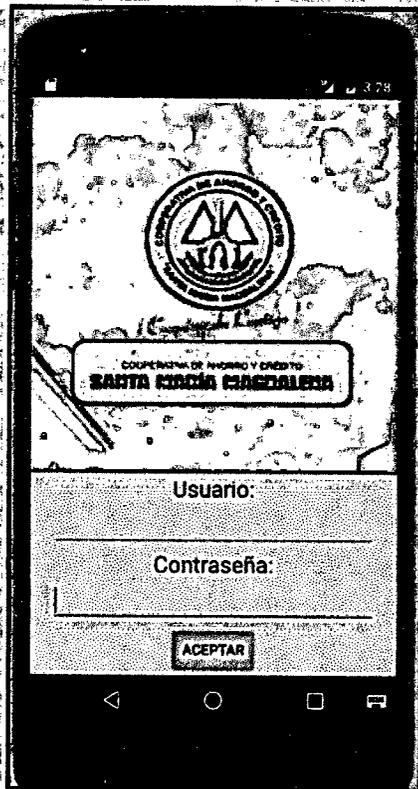


Figura N° 4.16: Interfaz iniciar sesión

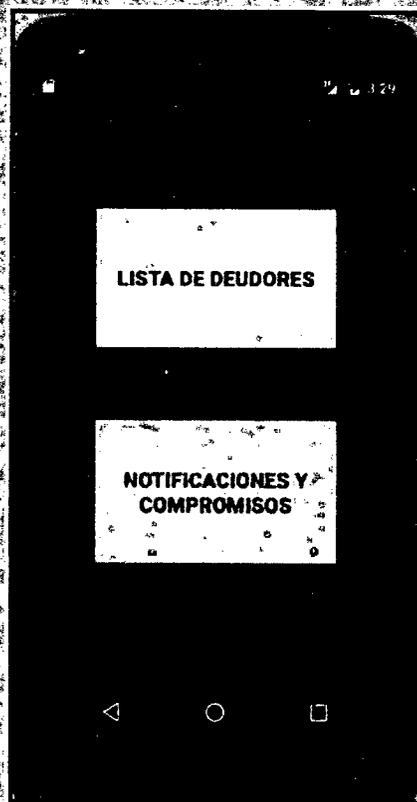


Figura N° 4.17: Interfaz menú principal

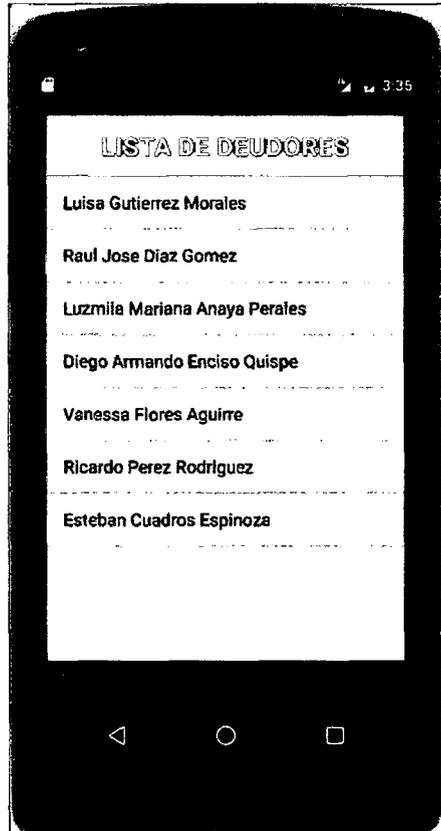


Figura N° 4.18: Interfaz lista de deudores

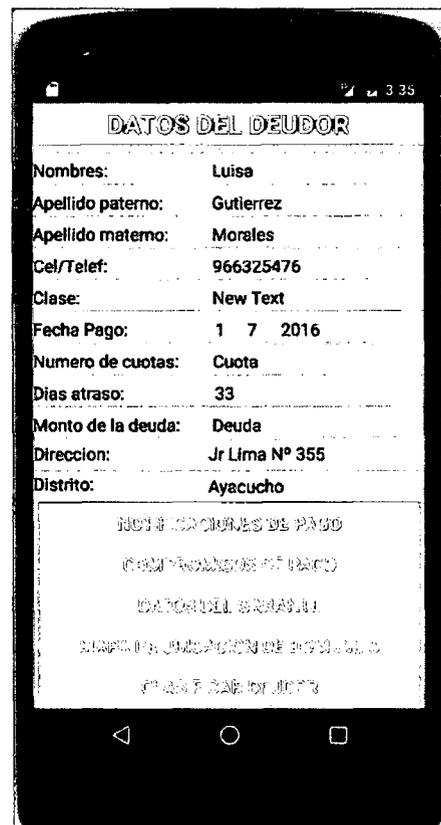


Figura N° 4.19: Interfaz datos del deudor

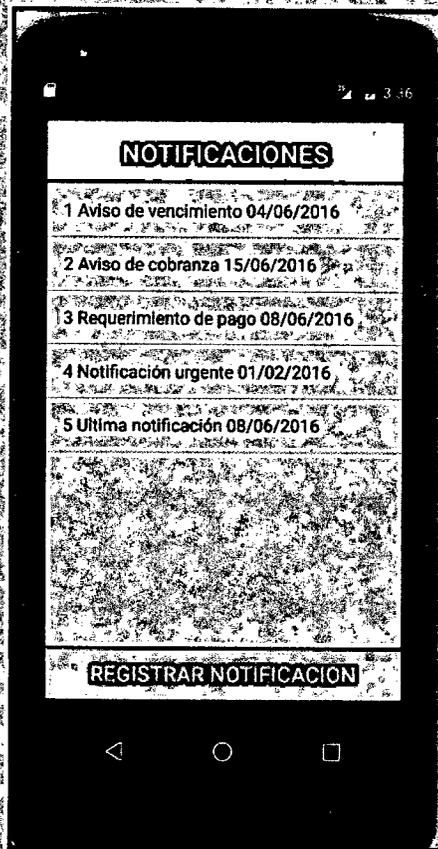


Figura N° 4.20: Interfaz notificaciones

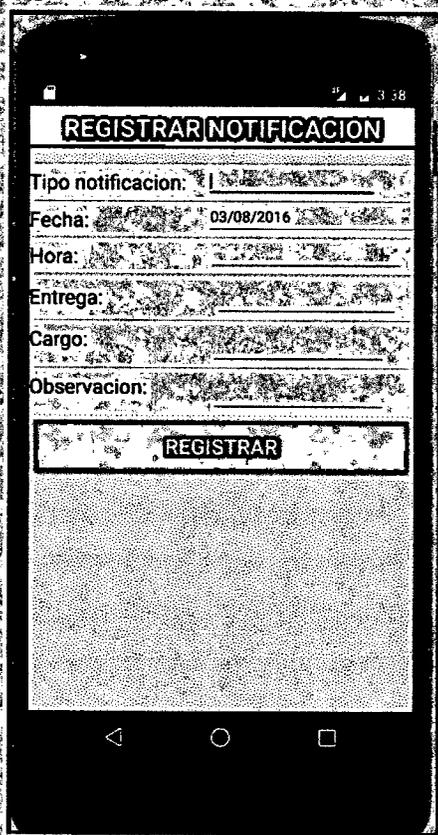


Figura N° 4.21: Interfaz registrar notificación

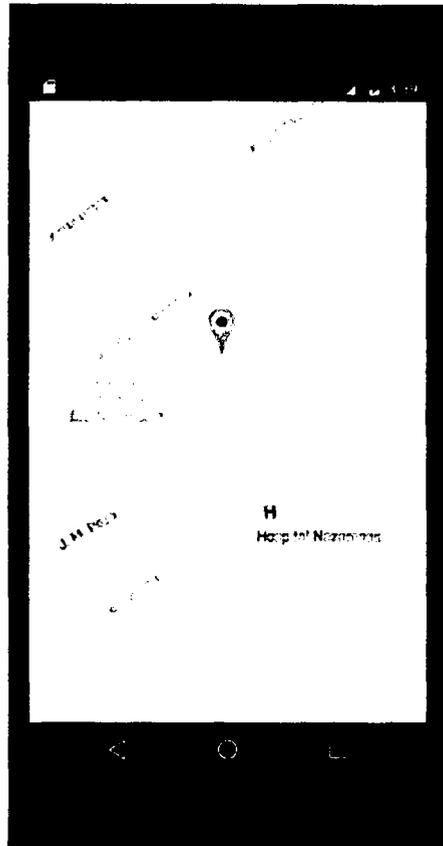


Figura N° 4.22: Interfaz mapa de ubicación del domicilio del deudor

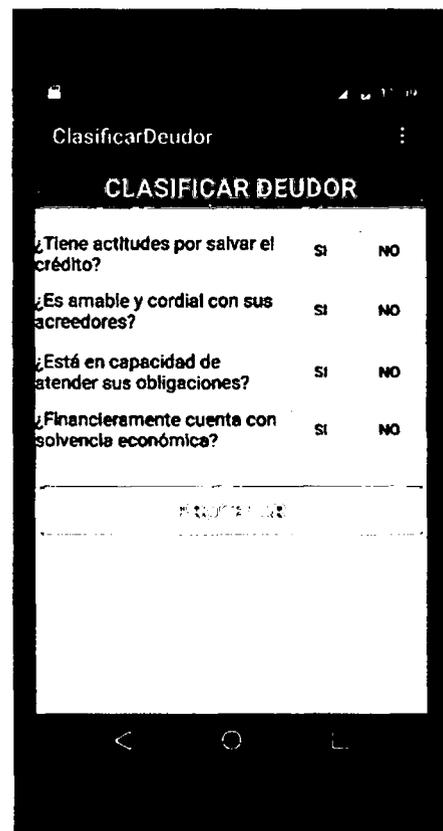


Figura N° 4.23: Interfaz clasificar deudor

## TARJETAS CLASE RESPONSABILIDAD Y COLABORACIÓN (CRC)

Para un diseño simple, se procede a definir tarjetas CRC, que permitan identificar las clases, sus responsabilidades y las colaboraciones que realiza para una historia de usuario y las tareas correspondientes, según se desarrolla en el capítulo II, sección 2.2.

TARJETA CRC		
Nombre Clase: Deudor		
Responsabilidades	Colaboradores	Atributos
<ul style="list-style-type: none"> <li>- Listar deudores</li> <li>- Listar nuevos deudores</li> <li>- Mostrar información del deudor</li> <li>- Buscar deudor por dni</li> <li>- Buscar deudor por nombres y apellidos</li> <li>- Buscar deudor por categoría</li> <li>- Buscar deudor por domicilio</li> </ul>	<ul style="list-style-type: none"> <li>- Dirección</li> <li>- Categoría</li> <li>- Crédito</li> <li>- Cónyuge</li> <li>- Garante</li> <li>- Compromiso</li> <li>- Notificación</li> <li>- GestorCobranza</li> </ul>	<ul style="list-style-type: none"> <li>- Dni</li> <li>- Nombres</li> <li>- ApellidoPaterno</li> <li>- ApellidoMaterno</li> <li>- Sexo</li> <li>- Edad</li> <li>- EstadoCivil</li> <li>- Telefono</li> <li>- Correo</li> </ul>

Tabla N° 4.78: Tarjeta CRC N° 1

TARJETA CRC		
Nombre Clase: Deuda		
Responsabilidades	Colaboradores	Atributos
<ul style="list-style-type: none"> <li>- Mostrar deuda</li> </ul>	<ul style="list-style-type: none"> <li>- Crédito</li> </ul>	<ul style="list-style-type: none"> <li>- Fecha</li> <li>- DiasAtraso</li> <li>- Capital</li> <li>- InteresMoratorio</li> <li>- Totaldeuda</li> </ul>

Tabla N° 4.79: Tarjeta CRC N° 2

TARJETA CRC		
Nombre Clase: Crédito		
Responsabilidades	Colaboradores	Atributos
<ul style="list-style-type: none"> <li>- Mostrar crédito</li> </ul>	<ul style="list-style-type: none"> <li>- Deudor</li> <li>- Deuda</li> <li>- PlanPagos</li> </ul>	<ul style="list-style-type: none"> <li>- Producto</li> <li>- TipoCredito</li> <li>- Destino</li> <li>- Moneda</li> <li>- Monto</li> <li>- Plazo</li> </ul>

Tabla N° 4.80: Tarjeta CRC N° 3

TARJETA CRC		
Nombre Clase: PlanPagos		
Responsabilidades	Colaboradores	Atributos
– Mostrar plan de pagos	– Crédito	– NumeroCuotas – FechaPagos – Capital – Interés – Cuota

Tabla N° 4.81: Tarjeta CRC N° 4

TARJETA CRC		
Nombre Clase: Cónyuge		
Responsabilidades	Colaboradores	Atributos
– Mostrar información del cónyuge	– Deudor	– Dni – Nombres – ApellidoPaterno – ApellidoMaterno – Edad – Telefono – EstadoCivil

Tabla N° 4.82: Tarjeta CRC N° 5

TARJETA CRC		
Nombre Clase: Garante		
Responsabilidades	Colaboradores	Atributos
– Mostrar información del garante	– Deudor – Dirección	– Dni – Nombres – ApellidoPaterno – ApellidoMaterno – Sexo – Edad – EstadoCivil – Telefono – Correo – Direccion

Tabla N° 4.83: Tarjeta CRC N° 6

TARJETA CRC		
Nombre Clase: GestorCobranzas		
Responsabilidades	Colaboradores	Atributos
<ul style="list-style-type: none"> <li>- Listar a los gestores de cobranzas</li> <li>- Mostrar información del gestor de cobranzas</li> <li>- Actualizar datos del gestor de cobranzas</li> <li>- Eliminar gestor de cobranzas</li> </ul>	<ul style="list-style-type: none"> <li>- Empleado</li> <li>- Deudor</li> </ul>	<ul style="list-style-type: none"> <li>- CodigoEmpleado</li> <li>- FechaIngreso</li> </ul>

Tabla N° 4.84: Tarjeta CRC N° 7

TARJETA CRC		
Nombre Clase: Empleado		
Responsabilidades	Colaboradores	Atributos
<ul style="list-style-type: none"> <li>- Mostrar información del empleado</li> </ul>	<ul style="list-style-type: none"> <li>- GestorCobranza</li> </ul>	<ul style="list-style-type: none"> <li>- Dni</li> <li>- Nombres</li> <li>- ApellidoPaterno</li> <li>- ApellidoMaterno</li> <li>- Sexo</li> <li>- Edad</li> <li>- EstadoCivil</li> <li>- Telefono</li> <li>- Correo</li> </ul>

Tabla N° 4.85: Tarjeta CRC N° 8

TARJETA CRC		
Nombre Clase: Dirección		
Responsabilidades	Colaboradores	Atributos
<ul style="list-style-type: none"> <li>- Mostrar dirección</li> </ul>	<ul style="list-style-type: none"> <li>- Deudor</li> <li>- Garante</li> </ul>	<ul style="list-style-type: none"> <li>- Dirección</li> <li>- Distrito</li> <li>- Provincia</li> <li>- Departamento</li> </ul>

Tabla N° 4.86: Tarjeta CRC N° 9

TARJETA CRC		
Nombre Clase: JefeRecuperacion		
Responsabilidades	Colaboradores	Atributos
<ul style="list-style-type: none"> <li>- Iniciar sesión</li> <li>- Asignar deudores a gestores de cobranzas</li> </ul>	<ul style="list-style-type: none"> <li>- Empleado</li> <li>- Usuario</li> </ul>	<ul style="list-style-type: none"> <li>- CodigoJefeRecuperaciones</li> <li>- CodigoUsuario</li> </ul>

Tabla N° 4.87: Tarjeta CRC N° 10

TARJETA CRC		
Nombre Clase: Notificación		
Responsabilidades	Colaboradores	Atributos
<ul style="list-style-type: none"> <li>- Listar notificaciones</li> <li>- Mostrar información de las notificaciones del deudor</li> <li>- Registrar notificación</li> </ul>	<ul style="list-style-type: none"> <li>- Deudor</li> </ul>	<ul style="list-style-type: none"> <li>- TipoNotificacion</li> <li>- Fecha</li> <li>- Hora</li> <li>- Entrega</li> <li>- Cargo</li> <li>- Observación</li> <li>- Estado</li> </ul>

Tabla N° 4.88: Tarjeta CRC N° 11

TARJETA CRC		
Nombre Clase: Compromiso		
Responsabilidades	Colaboradores	Atributos
<ul style="list-style-type: none"> <li>- Listar compromisos</li> <li>- Mostrar información de los compromisos del deudor</li> <li>- Registrar compromiso</li> </ul>	<ul style="list-style-type: none"> <li>- Deudor</li> </ul>	<ul style="list-style-type: none"> <li>- Tipo</li> <li>- Fecha</li> <li>- Observación</li> </ul>

Tabla N° 4.89: Tarjeta CRC N° 12

TARJETA CRC		
Nombre Clase: Usuario		
Responsabilidades	Colaboradores	Atributos
<ul style="list-style-type: none"> <li>- Ingresar al sistema</li> </ul>	<ul style="list-style-type: none"> <li>- JefeRecuperacion</li> <li>- GestorCobranza</li> </ul>	<ul style="list-style-type: none"> <li>- NombreUsuario</li> <li>- Contraseña</li> <li>- TipoUsuario</li> </ul>

Tabla N° 4.90: Tarjeta CRC N° 13



```

package pe.com.emp.ds;

import ...4 lines

public class DBRecuperacion {

    private String driver;
    private String url;
    private String username;
    private String pwname;

    public DBRecuperacion(ServletContext ctx) {
        this.driver = ctx.getInitParameter("driver");
        this.url = ctx.getInitParameter("url");
        this.username = ctx.getInitParameter("username");
        this.pwname = ctx.getInitParameter("pwname");
    }

    public Connection getConnection() {
        try {
            Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver").newInstance();
            Connection conn
                = DriverManager.getConnection("jdbc:sqlserver://PROFESIONAL:1433;databa
            return conn;
        } catch (SQLException sqle) {
            System.out.println("Error getConn=" + sqle.getMessage());
            return null;
        } catch (Exception e) {
            System.out.println("Error getConn=" + e.getMessage());
            return null;
        }
    }
}

```

Figura Nº 4.23: Código fuente de la conexión a la base de datos

```

import java.sql.Connection;
import javax.servlet.ServletContext;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import pe.com.emp.bean.BListaDeudor;
import pe.com.emp.dao.DAODEudor;
import pe.com.emp.ds.DBRecuperacion;

public class SlistarDeudor extends HttpServlet {

    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html; charset=UTF-8");

        String pag = request.getParameter("pg");
        int pg = Integer.parseInt(pag);

        ServletContext ctx = this.getServletConfig().getServletContext();
        DBRecuperacion ds = new DBRecuperacion(ctx);
        Connection conn = ds.getConnection();

        BListaDeudor blistaDeudor = new BListaDeudor();

        DAODEudor daodeudor = new DAODEudor(conn);
        daodeudor.listarDeudor(blistaDeudor, pg);

        request.setAttribute("blistaDeudor", blistaDeudor);
        request.getRequestDispatcher("listar_deudor.jsp").forward(request, response);
    }
}

```

Figura Nº 4.24: Código fuente del servlet listar deudor

```

public class RegistrarNotificacion extends Activity {

    private EditText txtTipoNotificacion,txtFecha,txtHora,txtEntrega,txtCargo,txtObservacion;
    private Button btnRegistrar;
    public String codigo;
    String res = "0";
    private int dia;
    private int mes;
    private int año;
    private static final int TIPO_DIALOGO = 0;
    private static DatePickerDialog.OnDateSetListener oventSelectorFecha;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_registrar_notificacion);

        txtTipoNotificacion = (EditText) findViewById(R.id.txtTipoNotificacion);
        txtFecha = (EditText) findViewById(R.id.txtFecha);
        txtHora = (EditText) findViewById(R.id.txtHora);
        txtEntrega = (EditText) findViewById(R.id.txtEntrega);
        txtCargo = (EditText) findViewById(R.id.txtCargo);
        txtObservacion = (EditText) findViewById(R.id.txtObservacion);
        btnRegistrar = (Button) findViewById(R.id.BtnRegistrar);

        Calendar calendario = Calendar.getInstance();
        dia = calendario.get(Calendar.DAY_OF_MONTH);
        mes = calendario.get(Calendar.MONTH);
        año = calendario.get(Calendar.YEAR);
        mostrarFecha();
    }
}

```

Figura N° 4.25: Código fuente Registrar notificación Android

```

private class IareaWSConsulta extends AsyncTask<String,Integer,Boolean> {

    private BDeudor[] listaDeudor;

    protected Boolean doInBackground(String... params) {

        boolean regul = true;
        res = "0";

        final String NAMESPACE = "http://martin.org/";
        final String URL="http://192.168.1.33//ServicioWebRecuperacion/ServicioClientes.asmx";
        final String METHOD_NAME = "RegistrarNotificacion";
        final String SOAP_ACTION = "http://martin.org/RegistrarNotificacion";

        SoapObject request = new SoapObject(NAMESPACE, METHOD_NAME);

        request.addProperty("hora", txtHora.getText().toString());
        request.addProperty("fecha", txtFecha.getText().toString());
        request.addProperty("entrega", txtEntrega.getText().toString());
        request.addProperty("cargo", txtCargo.getText().toString());
        request.addProperty("observacion", txtObservacion.getText().toString());
        request.addProperty("deudor", codigo);
        request.addProperty("tiponotificacion", txtTipoNotificacion.getText().toString());

        SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.VER11);
        envelope.dotNet = true;
        envelope.setOutputSoapObject(request);
        HttpTransportSE transport = new HttpTransportSE(URL);
    }
}

```

Figura N° 4.26: Código fuente Registrar notificación Android

```

public class ServicioClientes : System.Web.Services.WebService
{
    public static string nombreUsuario;

    [WebMethod]
    public Deudor[] ListadoClientes()
    {
        String usuario = nombreUsuario;
        SqlConnection con = new SqlConnection(@"Data
        Source=PROFESIONAL\SQLEXPRESS;Initial
        Catalog=DBRECUPERACIONES;Integrated Security=True");
        con.Open();
        string sql = "SELECT DeudorId, Nombre, ApellidoPaterno,
        ApellidoMaterno, Telefono FROM Deudor
        WHERE Usuario.NombreUsuario = @usu";
        SqlCommand cmd = new SqlCommand(sql, con);
        cmd.Parameters.Add("@usu", System.Data.SqlDbType.NVarChar).Value =
        usuario;
        SqlDataReader reader =
        cmd.ExecuteReader(System.Data.CommandBehavior.CloseConnection);
        List<Deudor> lista = new List<Deudor>();
        while (reader.Read())
        {
            lista.Add(new Deudor(reader.GetInt32(0), reader.GetString(1),
            reader.GetString(2), reader.GetString(3), reader.GetString(4)));
        }
        con.Close();
        return lista.ToArray();
    }

    [WebMethod]
    public Notificacion[] ListarNotificacionesEntregar()
    {
        String usuario = nombreUsuario;
        SqlConnection con = new SqlConnection(@"Data
        Source=PROFESIONAL\SQLEXPRESS;Initial
        Catalog=DBRECUPERACIONES;Integrated Security=True");
        con.Open();
        string sql = "SELECT Deudor.Nombre, Deudor.ApellidoPaterno, FROM
        Notificacion WHERE Notificacion.Estado = 'No entregado' AND
        Usuario.NombreUsuario = @usu";
        SqlCommand cmd = new SqlCommand(sql, con);
        cmd.Parameters.Add("@usu", System.Data.SqlDbType.NVarChar).Value =
        usuario;
        SqlDataReader reader =
        cmd.ExecuteReader(System.Data.CommandBehavior.CloseConnection);

        List<Notificacion> lista = new List<Notificacion>();
        while (reader.Read())
        {
            lista.Add(new Notificacion(reader.GetString(0),
            reader.GetString(1), reader.GetString(2),
            reader.GetInt32(3), reader.GetString(4)));
        }
        con.Close();
        return lista.ToArray();
    }
}

```

Figura N° 4.26: Código fuente de los servicios web

## PRUEBAS UNITARIAS DE LA APLICACIÓN WEB

Para las pruebas unitarias se utilizó como herramienta JUnit, el cual sirve para hacer las pruebas unitarias de las aplicaciones desarrolladas en Java.

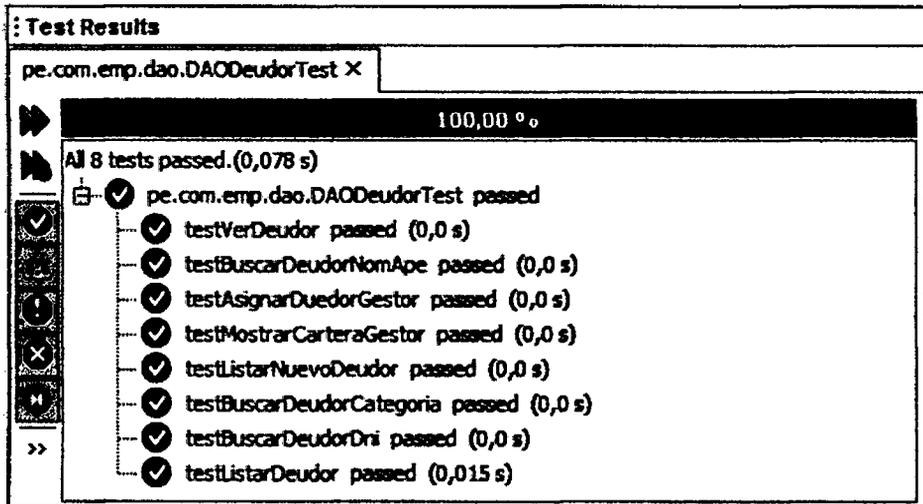


Figura N° 4.27: Resultado de las pruebas unitarias de Deudor

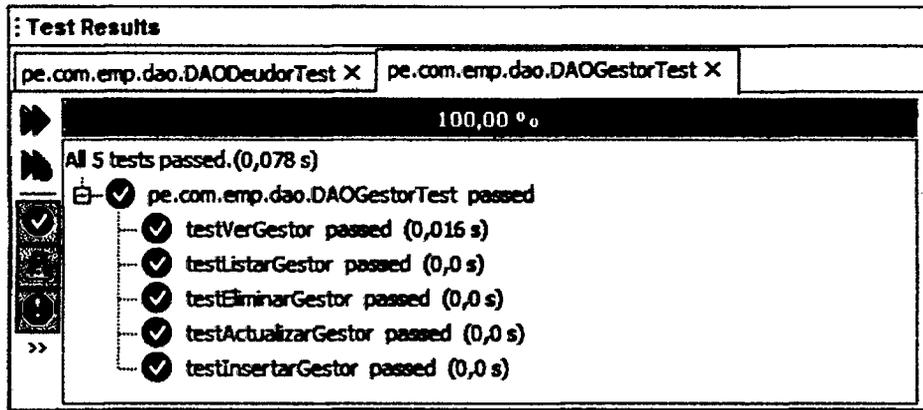


Figura N° 4.28: Resultado de las pruebas unitarias de Gestor

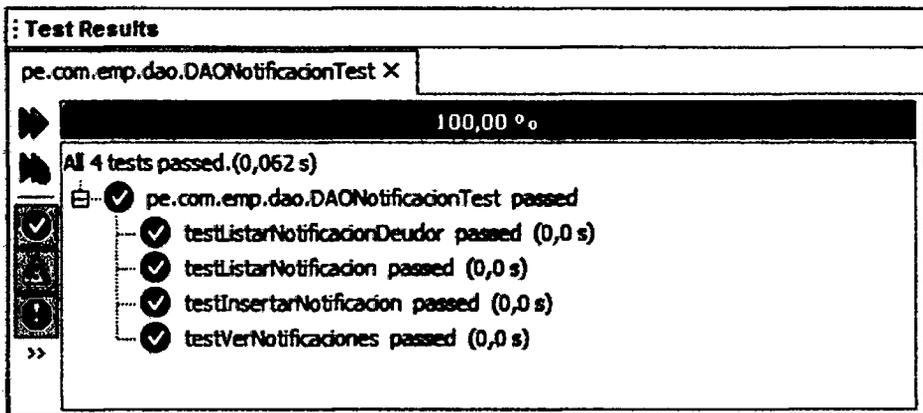


Figura N° 4.29: Resultado de las pruebas unitarias de Notificaciones

```

public void testBuscarDeudorDni () {
    System.out.println("buscarDeudorDni");
    String deudor = "1";
    DAODEudor instance = new DAODEudor();
    String expectedResult = "41166260";
    String result = instance.buscarDeudorDni(deudor);
    assertEquals(expResult, result);
}
/**
 * Test of buscarDeudorNomApe method, of class DAODEudor.
 */
@Test
public void testBuscarDeudorNomApe () {
    System.out.println("buscarDeudorNomApe");
    String deudor = "5";
    DAODEudor instance = new DAODEudor();
    String expectedResult = "María García Torres";
    String result = instance.buscarDeudorNomApe(deudor);
    assertEquals(expResult, result);
}
/**
 * Test of buscarDeudorCategoria method, of class DAODEudor.
 */
@Test
public void testBuscarDeudorCategoria () {
    System.out.println("buscarDeudorCategoria");
    String categoria = "3";
    DAODEudor instance = new DAODEudor();
    String expectedResult = "Juan Perez Gomez";
    String result = instance.buscarDeudorCategoria(categoria);
    assertEquals(expResult, result);
}
/**
 * Test of listarGestor method, of class DAOGestor.
 */
@Test
public void testListarGestor () {
    System.out.println("listarGestor");
    int pg = 0;
    DAOGestor instance = new DAOGestor();
    String expectedResult = "Lista de gestores";
    instance.listarGestor(pg);
    String result = instance.listarGestor(pg);
    assertEquals(expResult, result);
}
/* Test of verGestor method, of class DAOGestor.
 */
@Test
public void testVerGestor () {
    System.out.println("verGestor");
    int codigo = 8;
    DAOGestor instance = new DAOGestor();
    String result = instance.verGestor(codigo);
    String expectedResult = "Juaquin Hurtado Peres";
    assertEquals(expResult, result);
}

```

Figura Nº 4.30: Código fuente de las pruebas unitarias de la aplicación web

## PRUEBAS UNITARIAS DE LA APLICACIÓN MOVIL

Para las pruebas unitarias se utilizó el IDE Android Studio con la herramienta JUnit, el cual realiza las pruebas unitarias de las aplicaciones Android desarrolladas en Java.

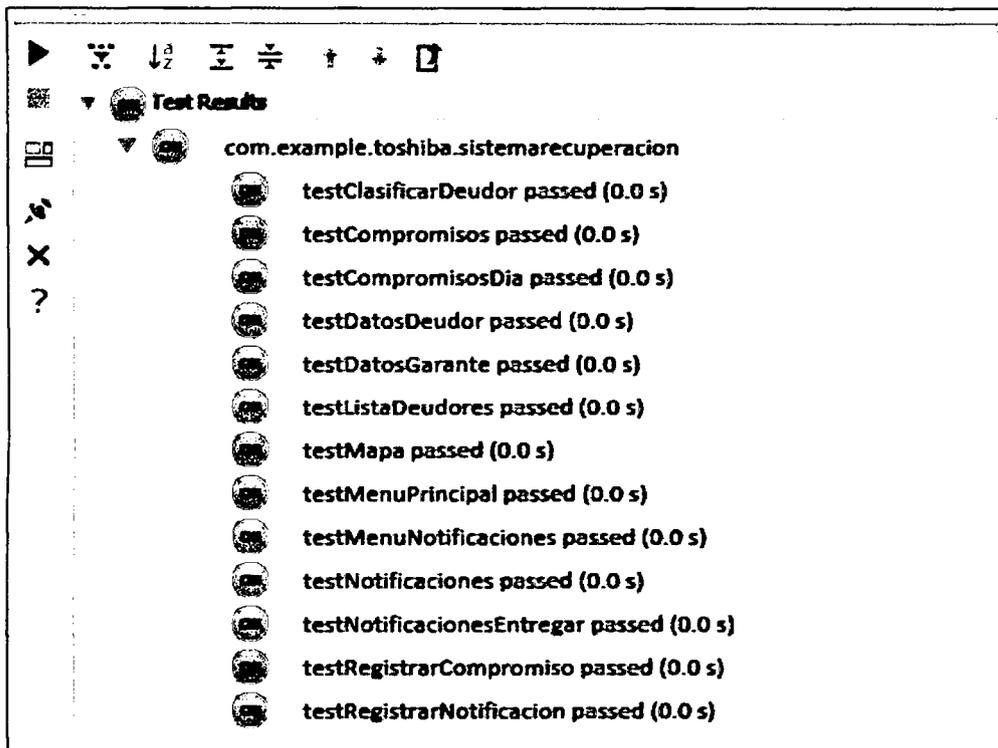


Figura N° 4.31: Resultado de las pruebas unitarias de la aplicación móvil.

```
public class ApplicationTest extends ApplicationTestCase<Application> {  
    public ApplicationTest() {  
        super(Application.class);  
    }  
  
    @Test  
    public void testClasificarDeudor(){  
        ClasificarDeudor instance = new ClasificarDeudor();  
        String expectedResult = "Deudor Clasificado";  
        String codigo = "1";  
        String result = instance.ClasificarDeudor(codigo);  
        assertEquals(expectedResult, result);  
    }  
  
    @Test  
    public void testCompromisos(){  
        Compromisos instance = new Compromisos();  
        String expectedResult = "Lista de compromisos";  
        String codigo = "3";  
        String result = instance.Compromisos(codigo);  
        assertEquals(expectedResult, result);  
    }  
}
```

Figura N° 4.32: Código fuente de las pruebas unitarias de la aplicación móvil.

## REPORTES DE PRUEBAS UNITARIAS DE LA APLICACIÓN WEB

Nº H. U.	Nº T. I.	TAREA DE INGENIERIA	RESULTADO
1	1	Iniciar sesión en la aplicación web	Satisfactorio
2	2	Consultar lista de deudores	Satisfactorio
3	3	Consultar lista de nuevos deudores	Satisfactorio
4	4	Buscar deudor por dni	Satisfactorio
	5	Buscar deudor por nombres y apellidos	Satisfactorio
	6	Buscar deudor por categoría	Satisfactorio
	7	Buscar deudor por domicilio	Satisfactorio
5	8	Consultar información del deudor	Satisfactorio
6	9	Consultar lista de gestores de cobranzas	Satisfactorio
7	10	Consultar información del gestor de cobranzas	Satisfactorio
8	11	Registrar nuevo gestor de cobranza	Satisfactorio
	12	Actualizar gestor de cobranza	Satisfactorio
	13	Eliminar gestor de cobranza	Satisfactorio
9	14	Consultar cartera de cobranza por gestor	Satisfactorio
10	15	Asignar deudor a gestor de cobranza	Satisfactorio
11	16	Consultar notificaciones de pago por entregar	Satisfactorio
12	17	Consultar notificaciones de pago del deudor	Satisfactorio
13	18	Registrar notificación de pago del deudor	Satisfactorio
14	19	Consultar compromisos de pago	Satisfactorio
15	20	Consultar compromisos de pago del deudor	Satisfactorio
16	21	Registrar compromisos de pago del deudor	Satisfactorio
17	22	Clasificar deudor	Satisfactorio
18	23	Emitir reporte de cobranza	Satisfactorio

Tabla Nº 4.91: Reporte de pruebas unitarias de la aplicación web

## REPORTES DE PRUEBAS UNITARIAS DE LA APLICACIÓN MOVIL

Nº H. U.	Nº T. I.	TAREA DE INGENIERIA	RESULTADO
1	1	Iniciar sesión en la aplicación móvil	Satisfactorio
2	2	Consultar lista de deudores	Satisfactorio
3	3	Consultar información del deudor	Satisfactorio
4	4	Consultar notificaciones de pago del deudor	Satisfactorio
5	5	Registrar notificación de pago del deudor	Satisfactorio
6	6	Consultar compromisos de pago del deudor	Satisfactorio
7	7	Registrar compromisos de pago del deudor	Satisfactorio
8	8	Consultar notificaciones de pago por entregar	Satisfactorio
9	9	Consultar compromisos de pago del día	Satisfactorio
10	10	Clasificar deudor	Satisfactorio
11	11	Consultar información del garante	Satisfactorio
12	12	Consultar ubicación del domicilio del deudor en el mapa	Satisfactorio

Tabla Nº 4.92: Reporte de pruebas unitarias de la aplicación móvil.

## REPORTE DE PRUEBA DE ACEPTACIÓN DE LA APLICACIÓN WEB

<b>Nº CASO PRUEBA</b>	1
<b>Propósito</b>	Verificar que el nombre de usuario y password son correcto y permitir el acceso al sistema, y si son incorrectos mostrar mensaje de error.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ejecutar la aplicación e ingresar a la página de login.
<b>Descripción de datos de entrada</b>	El usuario ingresa el nombre de usuario y password.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra página del menú principal
<b>Reales</b>	Muestra página del menú principal

Tabla N° 4.93: Reporte de prueba de aceptación. Iniciar sesión en la aplicación web.

<b>Nº CASO PRUEBA</b>	2
<b>Propósito</b>	Comprobar que la aplicación muestra la lista de todos los deudores de la cooperativa.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Selección la opción lista de deudores.
<b>Descripción de datos de entrada</b>	El usuario hace clic en el botón “Lista de deudores” de la página “Deudores”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra los nombres, apellidos y categoría de deudores de la cooperativa
<b>Reales</b>	Muestra página “Lista de deudores” con la relación de todos los deudores de la cooperativa.

Tabla N° 4.94: Reporte de prueba de aceptación. Consultar lista de deudores.

<b>Nº CASO PRUEBA</b>	3
<b>Propósito</b>	Comprobar que la aplicación muestra la lista de los nuevos deudores de la cooperativa.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Selección la opción lista de nuevos deudores.
<b>Descripción de datos de entrada</b>	El usuario hace clic en el botón “Lista de nuevos deudores” de la página “Deudores”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra la relación de los nuevos deudores de la cooperativa
<b>Reales</b>	Muestra página “Lista de nuevos deudores” con la relación de todos los socios que recién entraron en mora.

Tabla N° 4.95: Reporte de prueba de aceptación. Consultar lista de nuevos deudores.

<b>N° CASO PRUEBA</b>	4
<b>Propósito</b>	Verificar que la aplicación busca al deudor por el dni ingresado.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Buscar deudor”.
<b>Descripción de datos de entrada</b>	El usuario ingresa el dni del deudor en la caja de texto de dni y hace clic en el botón buscar.
<b>RESULTADOS</b>	
<b>Esperado</b>	Nombres, apellidos, categoría del deudor.
<b>Reales</b>	Muestra página “Lista de deudores” con los nombres, apellidos y categoría del deudor.

Tabla N° 4.96: Reporte de prueba de aceptación. Buscar deudor por dni.

<b>N° CASO PRUEBA</b>	5
<b>Propósito</b>	Verificar que la aplicación busca al deudor por los nombres o apellidos ingresados.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Buscar deudor”.
<b>Descripción de datos de entrada</b>	El usuario ingresa el nombre o el apellido paterno o el apellido materno del deudor en sus respectivas cajas de texto y hace clic en el botón buscar.
<b>RESULTADOS</b>	
<b>Esperado</b>	Nombres, apellidos, categoría de los deudores.
<b>Reales</b>	Muestra página “Lista de deudores” con los nombres, apellidos y categoría de los deudores.

Tabla N° 4.97: Reporte de prueba de aceptación. Buscar deudor por nombres y apellidos.

<b>N° CASO PRUEBA</b>	6
<b>Propósito</b>	Verificar que la aplicación busca al deudor por el tipo de categoría seleccionada.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Buscar deudor”.
<b>Descripción de datos de entrada</b>	El usuario selecciona el tipo de categoría y hace clic en el botón buscar.
<b>RESULTADOS</b>	
<b>Esperado</b>	Nombres, apellidos, categoría de los deudores.
<b>Reales</b>	Muestra página “Lista de deudores” con los nombres, apellidos y categoría de los deudores.

Tabla N° 4.98: Reporte de prueba de aceptación. Buscar deudor por categoría.

<b>N° CASO PRUEBA</b>	7
<b>Propósito</b>	Verificar que la aplicación busca al deudor por el departamento, provincia y distrito en donde vive.

<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Buscar deudor”.
<b>Descripción de datos de entrada</b>	El usuario selecciona el departamento, la provincia y el distrito del domicilio del deudor y hace clic en el botón buscar.
<b>RESULTADOS</b>	
<b>Esperado</b>	Nombres, apellidos, categoría de los deudores.
<b>Reales</b>	Muestra página “Lista de deudores” con los nombres, apellidos y categoría de los deudores.

Tabla N° 4.99: Reporte de prueba de aceptación. Buscar deudor por domicilio.

<b>N° CASO PRUEBA</b>	8
<b>Propósito</b>	Comprobar que la aplicación muestra toda la información del deudor
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Seleccionar la opción “ver detalle” en la página “Lista de deudores”.
<b>Descripción de datos de entrada</b>	El usuario hace clic en el botón “ver detalles” que se encuentra al lado de los datos del deudor.
<b>RESULTADOS</b>	
<b>Esperado</b>	Nombres, apellidos, categoría, dirección, datos del cónyuge, datos del crédito y de la deuda.
<b>Reales</b>	Muestra página “Datos del deudor” con los nombres, apellidos, categoría, dirección, datos del cónyuge, datos del crédito y datos de la deuda.

Tabla N° 4.100: Reporte de prueba de aceptación. Consultar información del deudor.

<b>N° CASO PRUEBA</b>	9
<b>Propósito</b>	Comprobar que la aplicación muestra la lista de los gestores de cobranza que laboran en la cooperativa
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Seleccionar la opción “Gestores de cobranza” en la página “Gestores”.
<b>Descripción de datos de entrada</b>	El usuario hace clic en el botón “Gestores de cobranza” que se encuentra en la página “Gestores”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra la relación con los nombres, apellidos y celular de los gestores de cobranza.
<b>Reales</b>	Muestra página “Gestores de cobranza” con los nombres, apellidos, y celular de los gestores de cobranza.

Tabla N° 4.101: Reporte de prueba de aceptación. Consultar lista de gestores de cobranzas.

<b>N° CASO PRUEBA</b>	10
<b>Propósito</b>	Comprobar que la aplicación muestra toda la información del gestor de cobranza.

<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Seleccionar la opción “ver detalle” en la página “Gestores de cobranza”.
<b>Descripción de datos de entrada</b>	El usuario hace clic en el botón “ver detalles” que se encuentra al lado de los datos del gestor de cobranza.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra los nombres, apellidos, dirección, fecha de ingreso y celular del gestor de cobranza.
<b>Reales</b>	Muestra página “Datos de gestor de cobranza” con los nombres, apellidos, dirección, fecha de ingreso y celular del gestor de cobranza.

Tabla N° 4.102: Reporte de prueba de aceptación. Consultar información del gestor de cobranzas.

<b>N° CASO PRUEBA</b>	11
<b>Propósito</b>	Verificar que la aplicación valida los datos de ingreso, registra al nuevo gestor de cobranza y muestra el mensaje de confirmación.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Nuevo gestores de cobranza”.
<b>Descripción de datos de entrada</b>	El usuario ingresa los nombres, apellidos, celular o teléfono, dirección, fecha de ingreso del nuevo gestor de cobranza y luego hace clic en el botón “Guardar”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Mensaje de confirmación de que los datos fueron ingresados correctamente.
<b>Reales</b>	Muestra el mensaje de confirmación de que los datos fueron ingresados correctamente y que los datos fueron almacenados en la base de datos.

Tabla N° 4.103: Reporte de prueba de aceptación. Registrar nuevo gestor de cobranza.

<b>N° CASO PRUEBA</b>	12
<b>Propósito</b>	Verificar que la aplicación valida los datos de ingreso, actualiza la información del gestor de cobranza y muestra el mensaje de confirmación.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Datos del gestores de cobranza” y hacer clic en el botón “actualizar”.
<b>Descripción de datos de entrada</b>	El usuario actualiza o corrige los nombres, apellidos, celular o teléfono, dirección, fecha de ingreso del gestor de cobranza y luego hace clic en el botón “Registrar”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Mensaje de confirmación de que los datos fueron actualizados correctamente.
<b>Reales</b>	Muestra el mensaje de confirmación de que los datos fueron actualizados correctamente.

Tabla N° 4.104: Reporte de prueba de aceptación. Actualizar gestor de cobranza.

<b>Nº CASO PRUEBA</b>	13
<b>Propósito</b>	Verificar que la aplicación muestra el mensaje de advertencia, elimina al gestor de cobranza de la base de datos y muestra el mensaje de confirmación.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Datos del gestores de cobranza” y hacer clic en el botón “Eliminar”.
<b>Descripción de datos de entrada</b>	El usuario hacer clic en el botón “Eliminar”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Mensaje de advertencia y luego mensaje que confirma la eliminación.
<b>Reales</b>	Muestra el mensaje de advertencia y luego mensaje que confirma la eliminación del gestor de cobranza de la base de datos.

Tabla N° 4.105: Reporte de prueba de aceptación. Eliminar gestor de cobranza.

<b>Nº CASO PRUEBA</b>	14
<b>Propósito</b>	Comprobar que la aplicación muestra la lista de deudores que le corresponde a cada gestor de cobranza.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Lista de gestores de cobranza”
<b>Descripción de datos de entrada</b>	El usuario hace clic en el botón “Ver cartera” que está al lado de los datos del gestor de cobranza de la página “Lista de gestores de cobranza”
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra la relación de deudores de que está en su cartera en cobranza
<b>Reales</b>	Muestra página “Lista de deudores” con la relación de todos los deudores que están en su cartera en cobranza de cada gestor de cobranza.

Tabla N° 4.106: Reporte de prueba de aceptación. Consultar cartera de cobranza por gestor de cobranza.

<b>Nº CASO PRUEBA</b>	15
<b>Propósito</b>	Comprobar que la aplicación muestra los datos del nuevo deudor y carga la lista de los gestores de cobranza para la asignación.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Selecciona la opción “Ver detalle” de la página “Lista de nuevos deudores”
<b>Descripción de datos de entrada</b>	El usuario hace clic en el botón “ver detalle” que se encuentra al lado de los datos del nuevo deudor.
<b>RESULTADOS</b>	
<b>Esperado</b>	Nombres, apellidos, categoría, dirección, datos del cónyuge, datos del crédito y de la deuda del deudor; y una lista desplegable con los nombres y apellidos del gestor de cobranza

<b>Reales</b>	Muestra página “Datos del deudor” con los nombres, apellidos, categoría, dirección, datos del cónyuge, datos del crédito, datos de la deuda y una lista desplegable con los nombres y apellidos del gestor de cobranza.
---------------	---

Tabla N° 4.107: Reporte de prueba de aceptación. Cargar lista de gestor de cobranza.

<b>N° CASO PRUEBA</b>	16
<b>Propósito</b>	Verificar que la aplicación asigna al nuevo deudor a un gestor de cobranza seleccionado.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Selecciona la opción “Ver detalle” de la página “Asignación de deudores”
<b>Descripción de datos de entrada</b>	El usuario selecciona a un gestor de cobranza de la lista desplegable y luego hace clic en el botón “Asignar”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra el mensaje “La asignación se realizó con éxito”
<b>Reales</b>	Se asigna el nuevo deudor al gestor de cobranza y muestra el mensaje “La asignación se realizó con éxito”.

Tabla N° 4.108: Reporte de prueba de aceptación. Asignar deudor a gestor de cobranza.

<b>N° CASO PRUEBA</b>	17
<b>Propósito</b>	Comprobar que la aplicación muestra la lista de notificaciones de pago por entregar.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Notificaciones y compromisos”
<b>Descripción de datos de entrada</b>	El usuario hace clic en la opción “Notificaciones por entregar” que está en la página “Notificaciones y compromisos”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra nombres, apellidos y tipo de notificación del deudor.
<b>Reales</b>	Muestra la página “Notificaciones por entregar”, con los nombres, apellidos y el tipo de notificación que aún no han sido entregados.

Tabla N° 4.109: Reporte de prueba de aceptación. Consultar notificaciones de pago por entregar

<b>N° CASO PRUEBA</b>	18
<b>Propósito</b>	Comprobar que la aplicación muestra las notificaciones de pago de cada deudor.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Datos del deudor”
<b>Descripción de datos de entrada</b>	El usuario hace clic en el botón “Notificaciones” que está en la página “Datos del deudor”.
<b>RESULTADOS</b>	

<b>Esperado</b>	Muestra nombres, apellidos y del deudor y una lista con el tipo de notificación, fecha de entrega, hora y estado.
<b>Reales</b>	Muestra la página “Notificaciones”, con los nombres, apellidos, una lista con el tipo de notificación, fecha de entrega, hora y estado.

Tabla N° 4.110: Reporte de prueba de aceptación. Consultar notificaciones de pago del deudor.

<b>N° CASO PRUEBA</b>	19
<b>Propósito</b>	Verificar que la aplicación valida los datos de ingreso, registra la notificación de pago y muestra el mensaje de confirmación.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Registrar notificación”
<b>Descripción de datos de entrada</b>	El usuario selecciona el tipo de notificación, a quien se entregó, ingresa la fecha, hora, cargo y observación y luego hace clic en el botón registrar.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra el mensaje de que la notificación fue registrada con éxito.
<b>Reales</b>	Muestra el mensaje de que la notificación fue registrada con éxito y almacena la información en la base de datos.

Tabla N° 4.111: Reporte de prueba de aceptación. Registrar notificación de pago del deudor.

<b>N° CASO PRUEBA</b>	20
<b>Propósito</b>	Comprobar que la aplicación muestra la lista de compromisos de pago de todos los deudores.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Notificaciones y compromisos”
<b>Descripción de datos de entrada</b>	El usuario hace clic en la opción “Compromisos de pago” que está en la página “Notificaciones y compromisos”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra los nombres, apellidos, tipo de contacto, fecha de contacto, fecha de compromiso y observación del compromiso de pago.
<b>Reales</b>	Muestra la página “Compromiso de pago” con los nombres, apellidos, tipo de contacto, fecha de contacto, fecha de compromiso y observación del compromiso de pago

Tabla N° 4.112: Reporte de prueba de aceptación. Consultar compromisos de pago.

<b>N° CASO PRUEBA</b>	21
<b>Propósito</b>	Comprobar que la aplicación muestra los compromisos de pago de cada deudor.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Datos del deudor”
<b>Descripción de datos de entrada</b>	El usuario hace clic en el botón “Compromisos” que está en la página “Datos del deudor”.
<b>RESULTADOS</b>	

<b>Esperado</b>	Muestra los nombres, apellidos, tipo de contacto, fecha de contacto, fecha de compromiso y observación del compromiso de pago.
<b>Reales</b>	Muestra la página “Compromiso de pago” los nombres, apellidos, tipo de contacto, fecha de contacto, fecha de compromiso y observación del compromiso de pago del deudor.

Tabla N° 4.113: Reporte de prueba de aceptación. Consultar compromisos de pago del deudor.

<b>N° CASO PRUEBA</b>	22
<b>Propósito</b>	Verificar que la aplicación valida los datos de ingreso, registra el compromiso de pago y muestra el mensaje de confirmación.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Registrar compromiso”
<b>Descripción de datos de entrada</b>	El usuario selecciona el tipo de contacto, ingresa la fecha de contacto, fecha de compromiso y observación, y luego hace clic en el botón registrar.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra el mensaje de que el compromiso fue registrado con éxito.
<b>Reales</b>	Muestra el mensaje de que el compromiso fue registrado con éxito y almacena la información en la base de datos.

Tabla N° 4.114: Reporte de prueba de aceptación. Registrar compromisos de pago del deudor.

<b>N° CASO PRUEBA</b>	23
<b>Propósito</b>	Verificar que la aplicación clasifica al deudor a través de un cuestionario.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Clasificación del deudor”
<b>Descripción de datos de entrada</b>	El usuario responde las preguntas de la página “Clasificación del deudor” y luego hace clic en el botón procesar.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra el mensaje “Deudor clasificado”
<b>Reales</b>	Muestra el mensaje “Deudor clasificado”

Tabla N° 4.115: Reporte de prueba de aceptación. Clasificar deudor.

<b>N° CASO PRUEBA</b>	24
<b>Propósito</b>	Comprobar que la aplicación emite reportes de cobranza.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la página “Reportes”
<b>Descripción de datos de entrada</b>	El usuario selecciona el tipo de reporte y hace clic en el botón ejecutar
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra el reporte seleccionado.
<b>Reales</b>	Muestra el reporte seleccionado.

Tabla N° 4.116: Reporte de prueba de aceptación. Emitir reporte de cobranza.

## REPORTE DE PRUEBA DE ACEPTACIÓN DE LA APLICACIÓN MOVIL

<b>Nº CASO PRUEBA</b>	1
<b>Propósito</b>	Verificar que el nombre de usuario y password son correcto y permitir el acceso al sistema, y si son incorrectos mostrar mensaje de error.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ejecutar la aplicación e ingresar a la página de login.
<b>Descripción de datos de entrada</b>	El usuario ingresa el nombre de usuario y password.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra la ventana del menú principal
<b>Reales</b>	Muestra la ventana del menú principal

Tabla Nº 4.117: Reporte de prueba de aceptación. Iniciar sesión.

<b>Nº CASO PRUEBA</b>	2
<b>Propósito</b>	Comprobar que la aplicación móvil muestra la lista de deudores que le corresponde a cada gestor de cobranza.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la ventana del Menú Principal.
<b>Descripción de datos de entrada</b>	El usuario hace clic en el botón “Lista de deudores” de la ventana del “Menú Principal.”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra la relación de deudores con sus nombres y apellidos que están en la cartera en cobranza
<b>Reales</b>	Muestra la ventana “Lista de deudores” con los nombres y apellidos de los deudores.

Tabla Nº 4.118: Reporte de prueba de aceptación. Iniciar sesión.

<b>Nº CASO PRUEBA</b>	3
<b>Propósito</b>	Comprobar que la aplicación móvil muestra la información necesaria del deudor.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Seleccionar un nombre de la ventana “Lista de deudores”.
<b>Descripción de datos de entrada</b>	El usuario pulsa un nombre que está en la ventana “Lista de deudores”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Nombres, apellidos, celular, fecha de pago, días de atraso, monto de la deuda, dirección, distrito.
<b>Reales</b>	Muestra ventana “Datos del deudor” con los nombres, apellidos, celular, fecha de pago, días de atraso, monto, dirección y distrito.

Tabla Nº 4.119: Reporte de prueba de aceptación. Iniciar sesión.

<b>Nº CASO PRUEBA</b>	4
<b>Propósito</b>	Comprobar que la aplicación móvil muestra las notificaciones de pago de cada deudor.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la ventana “Datos del deudor”
<b>Descripción de datos de entrada</b>	El usuario pulsa el botón “Notificaciones de pago” de la ventana “Datos del deudor”
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra el tipo de notificación y fecha de entrega.
<b>Reales</b>	Muestra ventana “Notificaciones de pago” con el tipo de notificación y fecha de entrega.

Tabla Nº 4.120: Reporte de prueba de aceptación. Iniciar sesión.

<b>Nº CASO PRUEBA</b>	5
<b>Propósito</b>	Verificar que la aplicación móvil valida los datos de ingreso, registra la notificación de pago y muestra el mensaje de confirmación.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la ventana “Registrar notificación”
<b>Descripción de datos de entrada</b>	El usuario ingresa el tipo de notificación, fecha, hora, entrega, cargo y observación y luego pulsa el botón “Registrar”.
<b>RESULTADOS</b>	
<b>Esperado</b>	Ingresar los datos y muestra el mensaje “Notificación registrada con éxito”
<b>Reales</b>	Almacena los datos en la base de datos y muestra el mensaje “Notificación registrada con éxito”

Tabla Nº 4.121: Reporte de prueba de aceptación. Iniciar sesión.

<b>Nº CASO PRUEBA</b>	6
<b>Propósito</b>	Comprobar que la aplicación móvil muestra los compromisos de pago de cada deudor.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la ventana “Datos del deudor”
<b>Descripción de datos de entrada</b>	El usuario pulsa el botón “Compromisos de pago” de la ventana “Datos del deudor”
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra la fecha y observación del compromiso de pago.
<b>Reales</b>	Muestra la ventana “Compromisos de pago” con la fecha y observación del compromiso de pago.

Tabla Nº 4.122: Reporte de prueba de aceptación. Iniciar sesión.

<b>N° CASO PRUEBA</b>	7
<b>Propósito</b>	Verificar que la aplicación móvil valida los datos de ingreso, registra los compromisos de pago y muestra el mensaje de confirmación.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la ventana “Registrar compromiso”
<b>Descripción de datos de entrada</b>	El usuario ingresa el tipo de contacto, la fecha de contacto, fecha de compromiso y observación, luego pulsa el botón “Registrar”
<b>RESULTADOS</b>	
<b>Esperado</b>	Ingresar los datos y muestra el mensaje “Compromiso registrado con éxito”
<b>Reales</b>	Almacena los datos en la base de datos y muestra el mensaje “Compromiso registrado con éxito”

Tabla N° 4.123: Reporte de prueba de aceptación. Iniciar sesión.

<b>N° CASO PRUEBA</b>	8
<b>Propósito</b>	Comprobar que la aplicación móvil muestra la lista de notificaciones de pago por entregar.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la ventana “Notificaciones y Compromisos”
<b>Descripción de datos de entrada</b>	El usuario pulsa el botón “Notificaciones por entregar” de la ventana “Notificaciones y Compromisos”
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra los nombres y apellidos del deudor y el tipo de notificación.
<b>Reales</b>	Muestra la ventana “Notificaciones por entregar” con los nombres y apellidos del deudor y el tipo de notificación

Tabla N° 4.124: Reporte de prueba de aceptación. Iniciar sesión.

<b>N° CASO PRUEBA</b>	9
<b>Propósito</b>	Comprobar que la aplicación móvil muestra la lista de compromisos de pago del día.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la ventana “Notificaciones y Compromisos”
<b>Descripción de datos de entrada</b>	El usuario pulsa el botón “Compromisos” de la ventana “Notificaciones y Compromisos”
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra los nombres y apellidos de los deudores.
<b>Reales</b>	Muestra la ventana “Compromisos del día” con los nombres y apellidos de los deudores.

Tabla N° 4.125: Reporte de prueba de aceptación. Iniciar sesión.

<b>Nº CASO PRUEBA</b>	10
<b>Propósito</b>	Verificar que la aplicación móvil clasifica al deudor a través de un cuestionario.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la ventana “Clasificación del deudor”
<b>Descripción de datos de entrada</b>	El usuario responde las preguntas de la ventana “Clasificación del deudor” y luego pulsa el botón procesar.
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra el mensaje “Deudor clasificado”
<b>Reales</b>	Muestra el mensaje “Deudor clasificado”

Tabla Nº 4.126: Reporte de prueba de aceptación. Clasificar deudor.

<b>Nº CASO PRUEBA</b>	11
<b>Propósito</b>	Comprobar que la aplicación móvil muestra la información del garante del deudor.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la ventana “Datos del deudor”
<b>Descripción de datos de entrada</b>	El usuario pulsa el botón “Garante” de la ventana “Datos del deudor”
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra nombres, apellidos, celular, dirección, y distrito del garante.
<b>Reales</b>	Muestra ventana “Datos del garante” con los nombres, apellidos, celular, dirección y distrito del garante.

Tabla Nº 4.127: Reporte de prueba de aceptación. Iniciar sesión.

<b>Nº CASO PRUEBA</b>	12
<b>Propósito</b>	Comprobar que la aplicación móvil muestra a través de un mapa la ubicación del domicilio del deudor.
<b>ACTIVIDAD</b>	
<b>Inicialización</b>	Ingresar a la ventana “Datos del deudor”
<b>Descripción de datos de entrada</b>	El usuario pulsa el botón “Mapa de domicilio” de la ventana “Datos del deudor”
<b>RESULTADOS</b>	
<b>Esperado</b>	Muestra el mapa de ubicación del domicilio del deudor.
<b>Reales</b>	Muestra ventana “Mapa” con el mapa de ubicación del domicilio del deudor.

Tabla Nº 4.128: Reporte de prueba de aceptación. Iniciar sesión.

## **CAPÍTULO V**

### **CONCLUSIONES Y RECOMENDACIONES**

#### **5.1 CONCLUSIONES**

- a. Según el marco teórico desarrollado en el capítulo II, sección 2.2, desarrollado en el capítulo III, sección 3.6, sustentado en el capítulo II, la sección 2.2.4. Los artefactos obtenidos en el capítulo IV; las historias de usuario de la aplicación web 2, 3, 4, 5 y 9 de la tabla N° 4.1, y las historias de usuario de la aplicación móvil 2, 3 de la tabla N° 4.2, las interfaces de usuario de las figuras N° 4.7, 4.8, 4.10, 4.18 y 4.19; las tarjetas CRC del N° 1 al 13; brindan información necesaria de la cartera en cobranza para recuperar los créditos morosos.
  
- b. Según el marco teórico desarrollado en el capítulo II, sección 2.2, desarrollado en el capítulo III, sección 3.6, sustentado en el capítulo II, la sección 2.2.4. Los artefactos obtenidos en el capítulo IV; las historias de usuario de la aplicación web 11, 12, 13, 14, 15 y 16 de la tabla N° 4.1, y las historias de usuario de la aplicación móvil 4, 5, 6, 7, 8 y 9 de la tabla N° 4.2, las interfaces de usuario de las figuras N° 4.9, 4.11, 4.14, 4.20, 4.21; las tarjetas CRC del N° 1 al 13; automatizan las notificaciones de pago de los deudores para recuperar los créditos morosos.
  
- c. Según el marco teórico desarrollado en el capítulo II, sección 2.2, desarrollado en el capítulo III, sección 3.6, sustentado en el capítulo II, la sección 2.2.4. Los artefactos obtenidos en el capítulo IV; las historias de usuario de la aplicación web 5 y 17 de la tabla N° 4.1, y las historias de usuario de la aplicación móvil 3 y 10, de la tabla N° 4.2, las interfaces de usuario de las figuras N° 4.15 y 4.23; las tarjetas CRC del N° 1 al 13 y el cuestionario para determinar la capacidad y voluntad de pago del anexo B; automatizan la clasificación del deudor para recuperar los créditos morosos.

## 5.2 RECOMENDACIONES

- a. Implementar un software georeferencial que ayude a los analistas de créditos a ubicar y registrar las coordenadas geográficas de los domicilios de los deudores a través de aplicaciones que funcionen en dispositivos móviles que cuenten con herramientas de GPS.
- b. Implementar un sistema experto que pueda evaluar las características emocionales, psicológicas y financieras de los socios que solicitan créditos el cual pueda ser utilizada por los analistas de créditos para apoyar la evaluación crediticia.
- c. Se recomienda la posibilidad de implementar un data warehouse para el área de créditos, recuperaciones, contabilidad y finanzas con el objetivo de obtener información confiable y ayudar a la toma de decisiones con el fin de disminuir el índice de morosidad.

## BIBLIOGRAFÍA

1. Andrada, A. (2010), *Nuevas tecnologías de la información y la comunicación NTICX*. Buenos Aires, Argentina. Editorial Maipue.
2. Brachfield, J. (2004). *El recobro de los impagados mediante la negociación efectiva con los morosos*. Barcelona, España: Ediciones Gestión.
3. Buendía I.; Colás B., y Hernández P., (1998): *Métodos de investigación en Psicopedagogía*. Madrid España. McGraw-Hill.
4. Bautista, J. (s. f.), *XP-Extreme-Programing [Online]*. Recuperado el 12 de diciembre de 2015, de [http://ingenieriadesoftware.mex.tl/52753\\_XP---Extreme-Programing.html](http://ingenieriadesoftware.mex.tl/52753_XP---Extreme-Programing.html)
5. Beck, K. (2001). *Manifiesto for Agile Software Development*. Recuperado el 13 de diciembre de 2015, de <http://agilemanifesto.org/>
6. Baird, S. (2002). *A Project Manager's Primer*, Prentice Hall, Professional Technical Reference. Recuperado el 20 de diciembre de 2015, de <http://www.phptr.com/articles/article.asp?p=26060&seqNum>.
7. Beck, k. (2000). *Extreme Programming Explained: Embrace Change*. Addison Wesley Longman.
8. Banco Central de Reserva del Perú (2003). *CIRCULAR N° 007-2003-EF/90*. Recuperado el 13 de febrero de 2016 de <http://www.bcrp.gob.pe/docs/Transparencia/Normas-Legales/Circulares/2003/Circular-007-2003-BCRP.pdf>
9. Bustamante, C. (s. f.). *Tratamiento contable de los ingresos por intereses compensatorios y moratorios* Recuperado el 13 de febrero de 2016 de [http://www.caballeroibustamante.com.pe/plantilla/2012/tratamiento\\_contable.pdf](http://www.caballeroibustamante.com.pe/plantilla/2012/tratamiento_contable.pdf)
10. Bernal, C. (2006). *Metodología de la Investigación*, Segunda Edición en Español, 2006.
11. Ceballos, F. (1993). *Programación orientada a objetos con C++*. Washington, Estados Unidos: Addison - Wesley Iberoamericana.
12. Campos, S. (2006). *Aprenda ya. Programación orientada a objetos*. Lima, Perú: Editorial Universitaria.
13. Curbera, F., William A. Nagy and Weerawarana, S., (s. f.). *Web Services: Why and How*, Recuperado el 3 de enero de 2016 de <http://www.research.ibm.com/people/b/bth/OOWS2001/nagy.pdf>.
14. Casal, J., y Mateu, E. (2003). *Tipos de muestreo*, Recuperado de <http://minnie.uab.es/~veteri/21216/TiposMuestreo1.pdf>

15. Calapiña, D. (2015), *Sistema web para la gestión de procesos de crédito y recuperación de cartera*, Ambato, Ecuador
16. Cabero, J., y Llorente, C. (2013), *La aplicación del juicio de experto como técnica de evaluación de las tecnologías de la información (TIC)*. Recuperado el 3 de abril de 2016 <http://tecnologiaedu.us.es/tecnoedu/images/stories/jca107>
17. CAC Santa María Magdalena (2014). *Memoria Anual*. Recuperado el 31 de enero de 2016 de [www.coopsantamaria.com.pe/memoria\\_anual/memoria2014/](http://www.coopsantamaria.com.pe/memoria_anual/memoria2014/)
18. CAC. "Santa María Magdalena" (2006). *Estatuto 2006*. Ayacucho, Perú.
19. Descripción de la arquitectura de Android, por Google (2008). Recuperado el 28 de diciembre de 2015 <http://es.youtube.com/watch?v=QBGfUs9mQYY>
20. Dalvik VM Internals, especificaciones de la máquina virtual (s. f.). Recuperado el 28 de diciembre de 2015 de <http://sites.google.com/site/io/dalvik-vm-internals>
21. Descripción del ciclo de vida de una aplicación Android, por Google (s. f.). Recuperado el 29 de diciembre de 2015 de <http://es.youtube.com/watch?v=fL6gSd4ugSI>
22. Estrada, A. (2004). *Protocolos Tcp/Ip de Internet*, México D. F., México: Revista Digital Universitaria - DGSCA-UNAM.
23. Escobar, J. (2008), *Validez de contenido y juicio de expertos: una aproximación a su utilización*. Recuperado el 3 de abril de 2016 [http://www.humanas.unal.edu.co/psicometria/files/7113/8574/5708/Articulo3\\_Juicio\\_de\\_expertos\\_27-36.pdf](http://www.humanas.unal.edu.co/psicometria/files/7113/8574/5708/Articulo3_Juicio_de_expertos_27-36.pdf)
24. Fernández, G. (2002) *Introducción a Extreme Programming, Ingeniería de Software II*, Madrid, España.
25. Fernández J. (1983). *Población y muestra. El muestreo*. Madrid, España: UNED
26. Fondo de Empleados del Banco de la Nación (s. f). *Manual de procedimiento de refinanciamiento, cobranzas y recuperaciones de los créditos otorgados por el feban*. Lima, Perú.
27. Galio, M. (2010). *Sistema para la gestión y control de cobranzas a través de dispositivos móviles*, Escuela Superior Politécnica del Litoral. Guayaquil, Ecuador
28. García, T. (2005). *Población y Muestra*. Madrid, España: Editorial Almendralejo.
29. Gonzales, M. (2000). *El protocolo TCP/IP*, Recuperado el 19 de febrero de 2016 de <http://www.lacompu.com/soporte/hardware/red/tcp/index.php>
30. Hernández, R., Fernández, C. y Baptista, P. (2010). *Metodología de la Investigación (4º ed.)*. México, D.F., México: Mc Graw Hill.

31. Joskowicz, J. (2008). *Reglas y Prácticas en eXtreme Programming*. Universidad de Vigo, Madrid, España.
32. Joyanes, L. (1998). *Programación orientada a objetos* (2ª Ed.). Madrid, España: Mcgraw W-Hill/Interamericana De España.
33. Letelier, p., Penadés, C., (2004). *Métodologías ágiles para el desarrollo de software*, Valencia, España: Universidad Politécnica de Valencia.
34. Lujan, S. (2002). *Programación en Internet: Cliente Web*. Madrid: España. Editorial Club Universitario
35. Lucas, A., Romera, P., Fraile, M., Argente, F. y Alfaro, A. (1993). *Diseño y Gestión de Sistema de Base de Datos*. Madrid, España: Editorial Paraninfo.
36. López, A. (2002), *Hacia una Cobranza Profesional*. Buenos Aires, Argentina: Ediciones Novel.
37. Medina M., Cubides D. (1999). *Deudor* (4ª Ed). Bogotá, Colombia: Pontificia Universidad Javeriana.
38. Mateu, C. (2004). *Desarrollo de Aplicaciones Web*. Barcelona España: Universidad Oberta de Catalunya.
39. Mullo, R. (2014). *Manual de crédito genérico para cooperativas de ahorro y crédito*. Asunción, Paraguay: Die Genossenschaften
40. Mukhar, K., Lavinger, T. y Carnell, J. (2002). *Fundamentos de base de datos con Java*. Madrid, España: Ediciones Anaya Multimedia.
41. Noticia de la sección El Navegante (2007). Recuperado el 28 de diciembre de 2015 <http://www.elmundo.es/navegante/2007/11/05/tecnologia>
42. Página oficial del proyecto Android (2008). Recuperado el 28 de diciembre de 2015 de <http://code.google.com/android/index.html>
43. Página oficial de Open Handset Alliance (2008). Recuperado el 28 de diciembre de 2015 <http://www.openhandsetalliance.com/index.html>
44. Superintendencia de Banca y Seguro y Administradoras Privadas de Fondos de Pensiones (2008). *Resolución S.B.S. N° 11356 - 2008*. Lima, Perú.
45. Pires, P., Mário, R., Benavides F., and Mattoso, (s. f.). M. *Building Reliable Web Services Compositions*. Recuperado el 3 de enero de 2016. <http://www.netobjectdays.org/pdf/02/papers/ws-rsd/1251.pdf>.
46. Portillo, M. (2009), *Manual de Administración de Cartera*. Fundación de Capacitación y Asesoría en Microfinanzas San Salvador. El Salvador: C.A.

47. Peñafiel, M. (s.f.). *Aplicaciones web*. Recuperado el 18 de abril de 2016, de <http://www.extremeprogramming.org/>
48. Rial, A (2004), *Las Mejores prácticas en las cobranzas al Consumidor*. Londres, Inglaterra: VRL Publishing, Ltd.
49. Sociedad Financiera de Objeto Múltiple (2010). *Modelo Genérico para Intermediario Financiero*. México D. F. México.
50. Susano, A. (2014), *Estrategia de la Recuperación de la Cartera de Créditos*. Lima, Perú. Investigación y Análisis Financiero SRL.
51. Sierra, B. (1988). *Técnicas de investigación Social. Teoría y Ejercicios*. Madrid, España: Paraninfo.
52. Teskiewicz, A (2007). *Modelos predictivos para cobranza y refinanciación*, Buenos Aires, Argentina: First Collection Summit
53. Tamayo, M. (1997) *El Proceso de la Investigación científica*. México D. F., México: Editorial Limusa S.A.
54. Vera, A. (2011). *Diseño de un sistema de gestión aplicado al área de crédito y cobranzas*, Ecuador. Escuela Superior Politécnica del Litoral.
55. Vasquez, M. (2013), *Análisis, Diseño e Implementación de un Sistema de recaudación de deudas*. Lima, Perú.
56. Weitzenfeld, A. (2005). *Ingeniería de Software orientada a objetos con UML Java e Internet*. Buenos Aires, Argentina: Thomson editores
57. Wells, D. (2006). *Extreme Programming: A gentle introduction*. Recuperado el 20 de diciembre de 2015, de <http://www.extremeprogramming.org/>
58. Wesley, A. (2001). *Interview with Kent Beck and Martin Fowler*. Recuperado el 10 de enero del 2016 de <http://www.awprofessional.com/articles/article.asp?>
59. Wells, D. (s. f.). *The Rules and Practices of Extreme Programming*. Recuperado el 21 de diciembre de 2015 de <http://www.extremeprogramming.org/rules.html>

## ANEXOS

### ANEXO A

#### INTRUMENTOS PARA RECOLECTAR INFORMACION

<b>GUÍA DE ENTREVISTA AL JEFE DE LA OFICINA DE RECUPERACIONES</b>	
Nombres y Apellidos:	
Cargo:	
<b>PREGUNTAS</b>	
1.- ¿Qué información se necesita de los deudores para hacer el cobro?	
2.- ¿Cuáles son las actividades de los gestores de cobranza?	
3.- ¿Cuáles son los tipos de notificaciones de pago y de qué manera son entregadas?	
5.- ¿De acuerdo al tipo de notificación, cada que tiempo son entregadas las notificaciones?	
6.- ¿Cuáles son los tipos de deudores?	
7.- ¿De qué forma se les debería de clasificar a los deudores si es que no existe un método?	
8.- ¿Cómo se determina el monto de la deuda?	
9.- ¿Cómo se obtiene el porcentaje del interés moratorio?	
10.- ¿Cómo se clasifica al deudor de acuerdo a los días de atraso?	
11.- ¿Cuál es el promedio de deudores al mes?	
12.- ¿Cuál es el número de gestores de cobranzas?	
13.- ¿Cuál es el promedio de deudores que tienen en cartera?	
14.- ¿Cuál es el promedio de notificaciones entregadas al mes?	

Tabla N° A.1: Guía de Entrevista al Jefe de la Oficina de Recuperaciones

<b>ENCUESTA SOBRE COMO DETERMINAR LA VOLUNTAD DE PAGO Y LA CAPACIDAD DE PAGO</b>	
<b>¿De las siguientes preguntas cuál cree usted que son más importantes para determinar la voluntad de pago del deudor? (Responder Si o No)</b>	
<b>PREGUNTAS</b>	<b>(SI/NO)</b>
1. ¿Tiene actitudes por salvar el crédito?	
2. ¿Muestra disciplina en sus pagos?	
3. ¿Se le ubica fácilmente?	
4. ¿Es amable y cordial con sus acreedores?	
<b>¿De las siguientes preguntas cuál cree usted que son más importantes para determinar la capacidad de pago del deudor? (Responder con Si o No)</b>	
<b>PREGUNTAS</b>	<b>(SI/NO)</b>
1. ¿Su documentación y registros de la deuda se encuentran accesibles?	
2. ¿Financieramente cuenta con solvencia económica?	
3. ¿Está en capacidad de atender las obligaciones?	
4. ¿Muchas veces ha llegado a extremos de disponer de las garantías?	

Tabla N° A.2: Encuesta sobre como determinar la capacidad de pago

<b>REGISTRO PARA ANALISIS DOCUMENTAL DE LOS REPORTES DE COBRANZA</b>	
Nombre o número del reporte:	
Tipo de reporte:	
Para qué sirve el reporte:	
Información que contiene el reporte:	

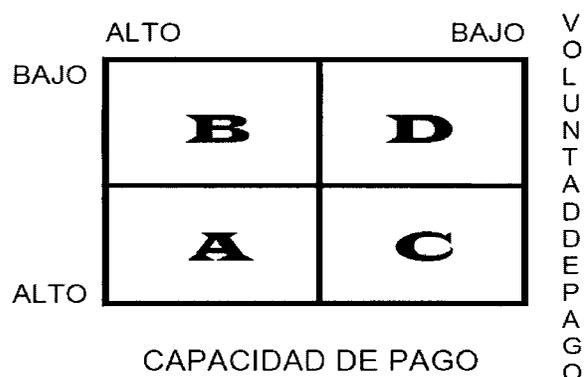
Tabla n° A.3: Registro para análisis documental de los reportes de cobranza

<b>REGISTRO PARA ANALISIS DOCUMENTAL DE LAS NORMAS Y POLÍTICAS DE RECUPERACION DE CRÉDITOS</b>	
Nombre del documento:	
Para qué sirve:	
Nombre del proceso:	
Actividad:	
Tareas:	

Tabla nº A.4: Registro para análisis documental de las normas y políticas de recuperación de créditos

**ANEXO B**  
**CLASIFICACION DEL DEUDOR**

**Ubicación y Clasificación**  
**de Prestatarios**



De acuerdo a las encuestas para determinar la voluntad de pago y la capacidad de pago se obtuvieron como resultado las siguientes preguntas a las cuales se les asigno un peso por la importancia que tienen:

PREGUNTAS	peso	CLASE A	CLASE B	CLASE C	CLASE D
¿Tiene actitudes por salvar el crédito?	2	SI	NO	SI	NO
¿Es amable y cordial con sus acreedores?	1	SI	NO	SI	NO
¿Está en capacidad de atender las obligaciones?	1	SI	SI	NO	NO
¿Financieramente cuenta con solvencia económica?	2	SI	SI	NO	NO

Tabla nº B.1: Preguntas para determinar la capacidad y voluntad de pago

**PROCESOS FINALES**

Los cinco procesos finales que se tomaron en cuenta para el desarrollo de la investigación son los siguientes:

1. Notificaciones de pago
2. Compromisos de pago
3. Asignación de gestores
4. Consulta de deudores
5. Reportes de recuperación

## ANEXO C

### OPERACIONALIZACION DE LAS VARIABLES

VARIABLES	DIMENSIONES	INDICADORES	ITEM	INSTRUMENTO
Gestión de cobranza	Cartera en cobranza	Deudor	1. ¿Qué información se necesita de los deudores para hacer el cobro?	Entrevista
			2. ¿Cuál es el promedio de deudores al mes?	Entrevista
		Gestor de cobranza	3. ¿Cuál es el número de los gestores de cobranzas?	Entrevista
			4. ¿Cuáles son las actividades de los gestores de cobranza?	Entrevista
			5. ¿Qué formatos son llenados por los gestores de cobranza?	Registro para Análisis documental
		Normas y políticas de cobranza	6. ¿Cuáles son los procesos de cobranza?	Registro para Análisis documental
	Reporte de cobranza	7. ¿Qué reportes de cobranza existen?	Registro para Análisis documental	
	Notificaciones de pago	Tipo de notificación	8. ¿Cuáles son los tipos de notificaciones de pago que se entrega a los deudores?	Entrevista
		Notificaciones entregadas	9. ¿Cuál es el promedio de notificaciones entregadas al mes?	Entrevista
			10. ¿De qué manera son entregadas las notificaciones de pago a los deudores?	Entrevista
			11. ¿De acuerdo al tipo de notificación, cada que tiempo son entregadas las notificaciones?	Entrevista
	Clasificación del deudor	Tipo de deudor	12. ¿Existe alguna manera de clasificar a los tipos de deudores?	Entrevista
		Voluntad de pago	13. ¿Tiene actitudes por salvar el crédito?	Encuesta
			14. ¿Muestra disciplina en sus pagos?	
			15. ¿Se le ubica fácilmente?	
		16. ¿Su actitud es hostil con sus acreedores?		
Capacidad de pago		17. ¿Su documentación y registros de la deuda se encuentran accesibles?	Encuesta	
		18. ¿Financieramente cuenta con solvencia económica?		
		19. ¿Está en capacidad de atender las obligaciones?		
		20. ¿Muchas veces ha llegado a extremos de disponer de las garantías?		
Créditos morosos	Deuda	Monto de la deuda	21. ¿Cómo se determina el monto de la deuda?	Entrevista
	Interés moratorio	Taza de interés moratorio	22. ¿Cómo se obtiene el porcentaje del interés moratorio?	Entrevista
	Días de atraso	Clasificación de acuerdo a los días de atraso	23. ¿Cómo se clasifica al deudor de acuerdo a los días de atraso?	Entrevista