

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

DESARROLLO DE UN SISTEMA DE SEGURIDAD DE ACCESO PARA APLICACIONES WEB. CASO DE ESTUDIO: SISTEMAS BANCARIOS

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

GABRIEL IGNACIO VILLAMAGUA ARMIJOS
igabo.nacho@gmail.com

DIRECTOR: ING. BOLÍVAR OSWALDO PALÁN TAMAYO. MSC
bolivar.palan@epn.edu.ec

Quito, Enero 2013

DECLARACIÓN

Yo, Gabriel Ignacio Villamagua Armijos, declaro bajo juramento que el trabajo aquí descrito es de mi autoría, que no ha sido previamente presentado para ningún grado o calificación profesional y que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo a la Escuela Politécnica Nacional, según lo establecido en la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad constitucional vigente.

Gabriel Ignacio Villamagua Armijos

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Gabriel Ignacio Villamagua Armijos bajo mi supervisión.

Ing. Bolívar Oswaldo Palán Tamayo. Msc

DIRECTOR DEL PROYECTO

AGRADECIMIENTOS

*Quiero agradecer primeramente a Dios
porque gracias a Él todas las cosas
existen.*

*A mi madre que siempre luchó para
que llegase a ser un hombre de bien, y
a mi papá que desde la memoria de
Dios ha sabido darme las fuerzas para
continuar.*

*A mis hermanos que siempre supieron
estar conmigo para ayudarme, en
especial a mi hermano Efrén que supo
ser mi segundo padre.*

*Finalmente, a la persona que amo más
en la vida... mi esposa Adriana L. que
supo darme su amor, apoyo y
comprensión y que gracias a ella llegué
a encontrar la felicidad.*

Gabriel

DEDICATORIA

*A mi madre y mis hermanos que
siempre me apoyaron en mis
decisiones aunque parecieran un
poco descabelladas.*

*A mi esposa porque ella es la que
soporta mis caprichos, mi locuras y
mis travesuras. Gracias por darle
sentido a mi vida.*

Gabriel

Tabla de Contenido

PRESENTACIÓN	2
RESUMEN	3
CAPÍTULO 1.	4
DEFINICIÓN DEL PROBLEMA Y SU ENTORNO DE SOLUCIÓN	4
1.1. ANÁLISIS DE LAS APLICACIONES BANCARIAS PARA WEB	4
1.1.1. APLICACIÓN BANCARIA.....	4
1.1.2. TECNOLOGÍAS INVOLUCRADAS.....	5
1.1.3. CADENA DE VALOR DE UN BANCO.....	6
1.2. ANÁLISIS DE LA WEB, SUS VULNERABILIDADES, RIESGOS, CONTROLES Y SEGURIDADES	12
1.3. SELECCIÓN DE LAS HERRAMIENTAS DE DESARROLLO	17
1.3.1. SELECCIÓN DEL DBMS.....	18
1.3.2. SELECCIÓN DEL ENTORNO DE DESARROLLO INTEGRADO.....	20
1.3.3. SELECCIÓN DEL SERVIDOR WEB	22
1.3.4. SELECCIÓN DE LA METODOLOGÍA DE DESARROLLO.....	24
CAPÍTULO 2.	28
ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DEL SISTEMA	28
2.1. ESPECIFICACIÓN DE REQUERIMIENTOS	28
2.2. ANÁLISIS DEL SISTEMA	36
2.2.1. ESPECIFICACIÓN DE SPRINTS	42
2.3. DISEÑO DEL SISTEMA.....	89
2.3.1. DISEÑO DE SPRINTS	89
2.4. CONSTRUCCIÓN DEL SISTEMA	119
SPRINT PRIMERO	119

CAPÍTULO 3.	134
SIMULACIÓN DEL SISTEMA EN EL CASO DE ESTUDIO	134
3.1. CONSTRUCCIÓN DE AMBIENTE DE SIMULACIÓN.....	134
3.1.1. HARDWARE UTILIZADO EN LA SIMULACIÓN.....	134
3.1.2. SOFTWARE UTILIZADO EN LA SIMULACIÓN	136
3.2. EJECUCIÓN DE SIMULACIÓN	139
3.2.1. CASO DE SIMULACIÓN # 01 “REGISTRO DE CLIENTE EN LA BANCA ELECTRÓNICA”	139
3.2.2. CASO DE SIMULACIÓN # 02 “INICIO SESIÓN DE CLIENTE”.....	143
3.2.3. CASO DE SIMULACIÓN #03 “FINALIZACIÓN DE SESIÓN”	145
3.2.4. CASO DE SIMULACIÓN #04 “ACTUALIZACIÓN DE DATOS DE AUTENTICACIÓN”	147
3.2.5. CASO DE SIMULACION #05 “ACTUALIZACIÓN DE DATOS PERSONALES”.....	149
3.2.6. CASO DE SIMULACIÓN #06 “BLOQUEAR USUARIO”	151
3.2.7. CASO DE SIMULACIÓN #07 “DESBLOQUEAR USUARIO”	153
3.2.8. CASO DE SIMULACIÓN #08 “TRANSFERENCIAS ENTRE CUENTAS PROPIAS”	155
3.2.9. CASO DE SIMULACIÓN #09 “RECUPERAR CONTRASEÑA”.....	158
3.2.10. CASO DE SIMULACIÓN #10 “RECORDAR USUARIO DE BANCA ELECTRÓNICA”	160
3.2.11. CASO DE SIMULACIÓN #11 “CREAR NUEVO USUARIO ADMINISTRADOR”.....	162
3.3. ANÁLISIS DE RESULTADOS.....	165
3.3.1. ANÁLISIS DE FUNCIONALIDAD	165
3.3.2. ANÁLISIS DE USABILIDAD	166
3.3.3. ANÁLISIS DE SEGURIDAD	171
CAPÍTULO 4.	174

CONCLUSIONES Y RECOMENDACIONES	174
4.1. CONCLUSIONES.....	174
4.2. RECOMENDACIONES	175
GLOSARIO.....	177
REFERENCIAS.....	179
ANEXOS	181

Lista de Ilustraciones (Figuras)

ILUSTRACIÓN 1.1 CADENA DE VALOR DE UN BANCO.	7
ILUSTRACIÓN 1.2 MODELO MVC.....	18
ILUSTRACIÓN 1.3 PANTALLA PRINCIPAL DE MYSQL ADMINISTRATOR.	20
ILUSTRACIÓN 1.4 ECLIPSE MARKETPLACE.....	22
ILUSTRACIÓN 2.1 DIAGRAMA DE GANTT DE SPRINT 01 (PARTE 1).....	47
ILUSTRACIÓN 2.2 DIAGRAMA DE GANTT DE SPRINT 01 (PARTE 2).....	48
ILUSTRACIÓN 2.3 ESFUERZO REALIZADO EN EL PRIMER SPRINT	49
ILUSTRACIÓN 2.4 ESFUERZO REALIZADO DEL PRIMER SPRINT	50
ILUSTRACIÓN 2.5 AVANCE DE TAREAS DURANTE EL PRIMER SPRINT	51
ILUSTRACIÓN 2.6 DIAGRAMA DE GANTT DE SEGUNDO SPRINT	56
ILUSTRACIÓN 2.7 ESFUERZO REALIZADO EN EL SEGUNDO SPRINT	58
ILUSTRACIÓN 2.8 ESFUERZO REALIZADO DEL SEGUNDO SPRINT	59
ILUSTRACIÓN 2.9 AVANCE DE TAREAS DURANTE EL SEGUNDO SPRINT..	59
ILUSTRACIÓN 2.10 DIAGRAMA DE GANTT DE TERCER SPRINT	64
ILUSTRACIÓN 2.11 ESFUERZO REALIZADO EN EL TERCER SPRINT	66
ILUSTRACIÓN 2.12 ESFUERZO REALIZADO DEL TERCER SPRINT	67
ILUSTRACIÓN 2.13 AVANCE DE TAREAS DURANTE EL TERCER SPRINT ...	67
ILUSTRACIÓN 2.14 DIAGRAMA DE GANTT DE CUARTO SPRINT	78
ILUSTRACIÓN 2.15 ESFUERZO REALIZADO EN EL CUARTO SPRINT	80
ILUSTRACIÓN 2.16 ESFUERZO REALIZADO DEL CUARTO SPRINT	81
ILUSTRACIÓN 2.17 AVANCE DE TAREAS DURANTE EL CUARTO SPRINT...	81
ILUSTRACIÓN 2.18 DIAGRAMA DE GANTT DE QUINTO SPRINT	86
ILUSTRACIÓN 2.19 ESFUERZO REALIZADO EN EL QUINTO SPRINT	87
ILUSTRACIÓN 2.20 ESFUERZO REALIZADO DEL QUINTO SPRINT	88
ILUSTRACIÓN 2.21 AVANCE DE TAREAS DURANTE EL QUINTO SPRINT....	88
ILUSTRACIÓN 2.22 DIAGRAMA E-R DEL SISTEMA	89
ILUSTRACIÓN 2.23 GRÁFICO DE ESTADOS DE UN HILO EN JAVA.....	95
ILUSTRACIÓN 2.24 PROCESO DE ENVIO DE CORREO ELECTRONICO	96
ILUSTRACIÓN 2.25 PROCESO DE ENVIO DE SMS.....	97
ILUSTRACIÓN 2.26 BOCETO DE INTERFAZ DE REGISTRO	99

ILUSTRACIÓN 2.27 BOCETO DE INTERFAZ DE INICIO DE SESIÓN	101
ILUSTRACIÓN 2.28 BOCETO DE INTERFAZ PARA PREGUNTAS Y SELECCIONAR IMAGEN.....	102
ILUSTRACIÓN 2.29 BOCETO DE INTERFAZ PRINCIPAL DE CLIENTE.....	103
ILUSTRACIÓN 2.30 BOCETO DE INTERFAZ PRINCIPAL DE ADMINISTRADOR	104
ILUSTRACIÓN 2.31 BOCETO DE INTERFAZ ACTUALIZAR INFORMACIÓN PERSONAL DE CLIENTE.....	106
ILUSTRACIÓN 2.32 BOCETO DE INTERFAZ DE REGISTRO	109
ILUSTRACIÓN 2.33 BOCETO DE INTERFAZ DE TRANSFERENCIAS ENTRE CUENTAS PROPIAS	110
ILUSTRACIÓN 2.34 BOCETO DE INTERFAZ DE MOVIMIENTOS	112
ILUSTRACIÓN 2.35 BOCETO DE INTERFAZ DE RECUPERACIÓN DE CONTRASEÑA	113
ILUSTRACIÓN 2.36 BOCETO DE RECORDATORIO DE USUARIO.....	114
ILUSTRACIÓN 2.37 BOCETO DE INTERFAZ DE BLOQUEO DE USUARIO...	115
ILUSTRACIÓN 2.38 BOCETO DE INTERFAZ DE DESBLOQUEO DE USUARIO	116
ILUSTRACIÓN 2.39 BOCETO DE INTERFAZ DE NUEVO USUARIO.....	117
ILUSTRACIÓN 3.1 TOPOLOGÍA DE RED.....	138
ILUSTRACIÓN 3.2 GRÁFICO PORCENTUAL DE USABILIDAD	166
ILUSTRACIÓN 3.3 GRÁFICO PORCENTUAL DE ACIERTOS Y FRACASOS .	167
ILUSTRACIÓN 3.4 GRÁFICO PORCENTUAL DE DISEÑO MINIMALISTA	170
ILUSTRACIÓN 3.5 GRÁFICO PORCENTUAL DE PATRÓN DE COLORES	171
ILUSTRACIÓN 3.6 GRÁFICO PORCENTUAL DE CUMPLIMIENTO DE SEGURIDAD	172

Lista de Tablas

TABLA 1.1 TECNOLOGÍAS INVOLUCRADAS EN EL DESARROLLO DE APLICACIONES WEB.....	5
TABLA 1.2 PORCENTAJE DE CUMPLIMIENTO DE LOS CORES BANCARIOS	11
TABLA 1.3 VULNERABILIDADES, RIESGOS, CONTROLES Y SEGURIDADES DE LA WEB.....	13
TABLA 1.4 CARACTERÍSTICAS DE ALGUNOS DBMS.....	19
TABLA 1.5 COMPARACIÓN ENTRE NETBEANS Y ECLIPSE	21
TABLA 1.6 COMPARACIÓN DE TRES SERVIDORES WEB.....	23
TABLA 2.1 PILA DEL PRODUCTO DE LA BANCA ELECTRÓNICA LIX	36
TABLA 2.2 SPRINT PRIMERO BANCA ELECTRÓNICA LIX.....	42
TABLA 2.3 DATOS DE CONFIGURACIÓN DE LA PILA DE ITERACIÓN.....	44
TABLA 2.4 SPRINT PLANNING MEETING PRIMERA ITERACIÓN.....	45
TABLA 2.5 SPRINT SEGUNDO BANCA ELECTRÓNICA LIX.....	52
TABLA 2.6 DATOS DE CONFIGURACIÓN DE LA PILA DE ITERACIÓN.....	53
TABLA 2.7 SPRINT PLANNING MEETING SEGUNDA ITERACIÓN	54
TABLA 2.8 SPRINT TERCERO BANCA ELECTRÓNICA LIX	60
TABLA 2.9 DATOS DE CONFIGURACIÓN DE LA PILA DE ITERACIÓN.....	61
TABLA 2.10 SPRINT PLANNING MEETING TERCERA ITERACIÓN.....	62
TABLA 2.11 SPRINT CUARTO BANCA ELECTRÓNICA LIX.....	68
TABLA 2.12 DATOS DE CONFIGURACIÓN DE LA PILA DE ITERACIÓN.....	70
TABLA 2.13 SPRINT PLANNING MEETING DE CUARTA ITERACIÓN	72
TABLA 2.14 SPRINT QUINTO BANCA ELECTRÓNICA LIX.....	82
TABLA 2.15 DATOS DE CONFIGURACIÓN DE LA PILA DE ITERACIÓN.....	83
TABLA 2.16 SPRINT PLANNING MEETING DE QUINTA ITERACIÓN	84

PRESENTACIÓN

La tecnología desde sus inicios y mucho más en la actualidad ha revolucionado la manera en como realizamos nuestras tareas diarias, tanto que podemos comunicarnos con personas que se encuentran al otro lado del mundo en unos cuantos segundos. Internet ha sido el mayor contribuyente para que todo esto sea posible, ahora podemos realizar transacciones desde la comodidad de nuestro hogar a cualquier momento del día mediante una pc, laptop, dispositivo móvil, etc.

Sin embargo, internet también permite la anonimidad de los usuarios lo que lo hace muy peligrosa si no se aplican controles y seguridades, un usuario puede ser víctima de fraude al realizar compras, realizar pagos de servicios básicos, etc. por lo que es necesario que la tecnología siga evolucionando para poder contrarrestar este tipo de delitos informáticos.

El presente trabajo busca minimizar el riesgo de ser víctima de un fraude informático al definir una solución web con tecnología actual y posteriormente al desarrollar una aplicación web bancaria que incorpora las seguridades y controles necesarios para que el usuario pueda realizar transacciones seguras en un medio inseguro como lo es internet.

RESUMEN

En el presente documento de titulación tiene como finalidad la elaboración de una aplicación web bancaria utilizando tecnologías actuales, basándose en una metodología de desarrollo ágil como lo es SCRUM.

El primer capítulo inicia con un análisis de las aplicaciones bancarias para web y sus características, además se especifican las tecnologías involucradas utilizadas en el desarrollo. Se analizan tres cores bancarios grandes utilizados en el mercado local y se determinan si cumplen con las necesidades de las entidades financieras. Debido a que la aplicación bancaria es orientada a la web, se realiza un análisis de las vulnerabilidades, riesgos, controles y seguridades de la misma para tener conocimiento del campo en que se va a desarrollar. Para el desarrollo de la aplicación se necesitan de herramientas y una metodología, por lo que se realiza un análisis y selección de un DBMS, IDE (Entorno de Desarrollo Integrado) y una metodología.

El segundo capítulo abarca el desarrollo de la aplicación web bancaria propuesta, que comprende la especificación de requerimientos, la especificación de la pila y análisis y diseño de SPRINTS del producto, y la construcción del producto final.

En el tercer capítulo se realiza la simulación de la aplicación en el caso de estudio, para esto se define el hardware y software necesario. Se realiza un análisis de funcionalidad, usabilidad y seguridad para verificar si la aplicación cumple con los objetivos propuestos inicialmente.

Finalmente, el cuarto capítulo contiene las conclusiones y recomendaciones obtenidas a lo largo del análisis, diseño y construcción de la aplicación propuesta.

CAPÍTULO 1.

DEFINICIÓN DEL PROBLEMA Y SU ENTORNO DE SOLUCIÓN

1.1. ANÁLISIS DE LAS APLICACIONES BANCARIAS PARA WEB

1.1.1. APLICACIÓN BANCARIA

Una aplicación bancaria es una aplicación que sirve para consumir los servicios que ofrece una entidad financiera.

Estas aplicaciones pueden realizar entre otras cosas las siguientes transacciones:

- Transferencias directas, entre bancos locales, o extranjeros.
- Pago de servicios como agua, luz, teléfono, etc.
- Consulta de saldos.
- Pago de tarjeta de crédito.
- Recargas electrónicas de telefonía celular.
- Compras como lotería, etc.

Estas aplicaciones son accesibles desde muchos equipos como dispositivos móviles, laptop, tablets, desktops, netbooks, etc. en general todo equipo que tenga acceso a internet.

Una aplicación bancaria debe cumplir al menos con las siguientes características:

- Eficiencia
- Seguridad
- Funcionalidad

Eficiencia: ya que la razón de utilizar una aplicación bancaria es para ahorrar tiempo, los tiempos de respuesta no deben ser muy grandes para lo cual se debe implementar sistemas de balanceo de carga con algoritmos que reduzcan este tiempo.

Seguridad: una aplicación bancaria debe brindar al usuario la seguridad de acceso para realizar transacciones de manera segura sin que peligre su información y los activos de su cuenta bancaria, para esto se deben implementar métodos de encriptación, certificados digitales, etc.

Funcionalidad: el sistema debe permitir la fácil mantenibilidad además de ser interoperable con otras aplicaciones para esto se debe utilizar estándares como el ISO8583 que especifica las transacciones financieras bancarias para transmisión de datos. Además, este debe ser altamente disponible para que el usuario realice sus transacciones en cualquier momento.

1.1.2. TECNOLOGÍAS INVOLUCRADAS

Para el desarrollo de aplicaciones web se necesita de tecnología, a continuación se realiza un resumen de los dos grandes grupos: front-end y back-end.

Tabla 1.1 Tecnologías involucradas en el desarrollo de aplicaciones web

ÁREA	TECNOLOGÍA	DESCRIPCIÓN	OBSERVACIÓN
Front-End (Cliente)	HTML	Lenguaje de marcado mediante etiquetas, cuyas etiquetas son encerradas por corchetes angulares (<>).	Se encuentra actualmente en su versión 5 pero será completada para el 2020 [1].
	CSS	Hojas de estilo en modo texto que describe la manera en que se va a mostrar una página web.	Se encuentra en su versión 3 [2].
	Javascript	Lenguaje interpretado y orientado a objetos, su uso principal es dentro de los navegadores web que permiten realizar tareas repetitivas y tediosas al	Funciona con un SANDBOX lo que hace que no se pueda ser manipulado y así ejecutar ataques.

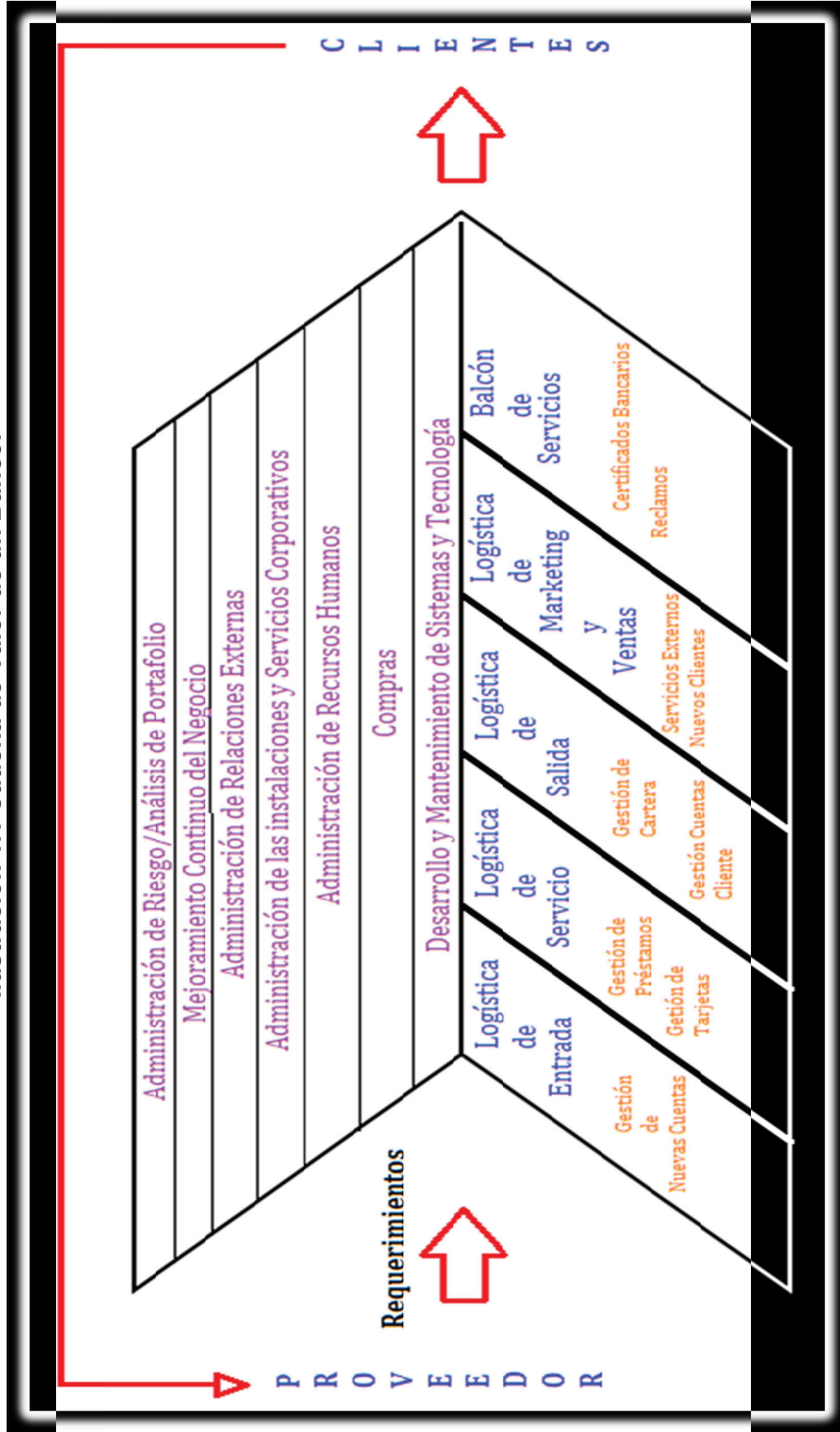
		servidor como las validaciones.	
	Plugins	Conjunto de componentes de software que añade funcionalidad a aplicaciones de software más grandes	Son libres para descarga y utilización pero no siempre son independientes de la plataforma.
Back-End (Servidor)	Servidor Web	Es una aplicación que está constantemente corriendo en un computador, estando a la espera de alguna petición de un usuario de acuerdo a la solicitud realizada.	Los servidores más conocidos son: Apache, Internet Information Services, Nginx y GWS (Google Web Server)
	Bases de datos	Una base de datos es un conjunto de datos, que están almacenados mediante diferentes modelos	Tipos de modelos: Jerárquicas, Red, Transaccionales, Relacionales, Multidimensionales Orientadas a objetos, Deductivas y Documentales [3].
	Lenguajes de programación	Sentencias de alto nivel que permite a los programadores emitir órdenes al computador.	Lenguajes más conocidos tenemos: JAVA, C, C++, C#, PHP, Objective-C, Visual Basic, Python, Javascript, Perl y Ruby [4].
	Frameworks	Permite desarrollar aplicaciones con mayor efectividad.	Existen muchos frameworks privativos y libres.

Elaborado por: Gabriel Ignacio Villamagua Armijos

1.1.3. CADENA DE VALOR DE UN BANCO

Una aplicación bancaria como su nombre lo explica debe estar en lineamiento de acuerdo a los procesos del banco, para esto a continuación tenemos la cadena de valor genérica de un banco.

Ilustración 1.1 Cadena de Valor de un Banco.



Elaborado por: Gabriel Ignacio Villamagua Armijos

1.1.3.1. Cooperative Open Banking Information System (COBIS)

Es un producto desarrollado por la compañía COBISCORP, este core está orientado a entidades financieras y bancos de primer y segundo piso. COBIS es una solución flexible que resuelve las necesidades de negocio financiero mediante un conjunto de módulos que lo conforman [5].

Algunas de las características importantes tenemos:

- Cliente único.
- Administración de usuarios centralizada.
- Autorización transaccional por rol de usuario.
- Log de las transacciones.
- Altamente parametrizable.
- Solución modular e integral.
- Contabilización automática: uso de interfaz contable.
- Funciona sobre distintos sistemas operativos Windows, Unix, AIX, Linux.

Los servicios que ofrece COBIS son:

- Servicio de pasivas.
 - Depósito a largo plazo.
 - Administración de tarjetas de débito.
 - Cuentas de ahorro.
 - Cuentas corrientes.
- Servicio de activas
 - Crédito
 - Factoring
 - Gestión de Cobranza
 - Cartera
- Servicio de intermediación financiera.

- Cámara y Remesas.
- Tesorería.
- Comercio Exterior.
- Remesas del exterior.
- Procesamiento de cheques.
- Servicios bancarios.

Los servicios web están directamente relacionados con las cuentas de ahorro y corrientes, todos estos servicios son accesibles por el usuario final mediante la web.

Algunas de las entidades financieras que utilizan COBIS en Ecuador son: Banco Amazonas, Banco Bolivariano, Banco Nacional de Fomento, Banco Promerica, Corporación Financiera Nacional y un grupo muy grande de cooperativas de ahorro y crédito.

1.1.3.2. TCS BαNCS CORE

TCS BαNCS es una solución que integra todos los aspectos de corebanking a través de entidades. Esta solución ofrece a las instituciones financieras a introducir nuevos productos con facilidad [6].

Esta solución ofrece los siguientes servicios:

- Conjunto de productos como cuentas de ahorros, cheques, cuentas corrientes y cuentas de depósito.
- Toda la gama de productos de crédito.
- Servicios transaccionales como las remesas, divisas, tarjetas y financiamiento.
- Accesibilidad mediante múltiples canales como banca móvil y web.
- Integración de procesos front, middle and bank-office.
- Vista única entre el banco y el cliente.

- Banca cuando sea, donde sea.

De los servicios ofrecidos por TCS BαNCS, los que están disponibles al cliente son banca móvil y web. Estos servicios proveen la facilidad de acceder a consultas, transferencia, pagos de servicios, compra de servicios, pago de tarjetas de crédito.

El banco más grande que utiliza TCS BαNCS en Ecuador es Banco Pichincha.

1.1.3.3. Prometeus (Produbanco)

Prometeus es la solución bancaria que utiliza el banco de la producción (Produbanco). Esta solución fue implementada con Windows Server System y ofrece los siguientes productos:

- Clientes
- Captaciones de renta fija (inversiones).
- Cuentas corrientes
- Cuentas de ahorros
- Administración de cuentas a la vista
- Cheques y chequeras
- Cambios
- Giros al exterior
- Cámara
- Remesa y tránsito.
- Cartera
- Comercio exterior
- Garantías
- Avales
- Custodia
- Mesa de dinero

- Mesa de cambios
- Contabilidad [7].

Además ofrece módulos de apoyo en el que se encuentran:

- Manejador de flujos
- Switch transaccional para canales electrónico, soporte web, call center, etc [7].

Los procesos accesibles al cliente final son mediante canales electrónicos y la web, en la que se tiene acceso a la información de la cuenta de ahorros, corriente, etc.

Tabla 1.2 Porcentaje de cumplimiento de los cores bancarios.

Core	Servicios no web	Servicios web	Canales electrónicos	Observación
COBIS	100% + servicios adicionales propios	100% + servicios parametrizables	100%	Incluye banca móvil
TCS BαNCS	100% + servicios adicionales	100%	100%	Incluye banca móvil
PROMETEUS	100% + servicios adicionales	100%	80%	Banca Móvil con servicios muy limitados.

Fuente: Secciones 1.1.3.1, 1.1.3.2 y 1.1.3.3 de este documento

Elaborado por: Gabriel Ignacio Villamagua Armijos

1.2. ANÁLISIS DE LA WEB, SUS VULNERABILIDADES, RIESGOS, CONTROLES Y SEGURIDADES

La web ha cambiado nuestro estilo de vida en la manera como nos comunicamos, compramos y realizamos tareas varias que antes no se podían realizar con la misma agilidad de ahora.

Aunque la web nos ha ayudado con muchas tareas ésta no es perfecta también tiene vulnerabilidades que hace que corramos riesgos, existen ciberdelicuentes intentando realizar crímenes cibernéticos como robo de información sensible como números de cuentas de banco, números de tarjetas de créditos, etc.

Para contrarrestar estos tipos de ataques existen controles y seguridades que intentan hacer la web un sitio más seguro para los usuarios que en su mayoría no tienen un amplio conocimiento de informática.

Tabla 1.3 Vulnerabilidades, riesgos, controles y seguridades de la web

VULNERABILIDAD	RIESGO	CONTROLES	SEGURIDADES
Cross Site Scripting (XSS)	Robo de información sensible (usuarios, contraseñas, números de cuenta, etc.)	Deshabilitar la opción HTTP TRACE en todos los servidores web.	Use el proyecto OWASP ESAPI que define controles para validaciones e inyección de ataques XSS [8].
Inyección de Código	Robo de usuarios y contraseñas.	No utilizar sentencias SQL construidas dinámicamente. No utilizar cuentas con privilegios administrativos.	Proteger las contraseñas con técnicas de encriptación muy segura como la encriptación 3DES [8].
Archivos maliciosos	Robo de información sensible. Molestosas publicidades. Bajo desempeño del computador. Modificación de archivos del sistema.	No abrir links que se encuentren en correos de dudosa procedencia.	Utilizar software anti-spam, antivirus y anti-malware [8].
Pérdida de información	Robo de información empresarial	Usar proxies de tráfico de salida para monitorear la	Utilizar mecanismos de autenticación y encriptación para

información que sale de la empresa.
 mover a través de una red.

Identificar qué información está disponible al público.

Software no autorizado	Comprometer máquinas para atacar a otras y robar información o realizar distintos tipos de ataques.	Hacer un inventario de todas las aplicaciones que se encuentran instaladas, además de servidores, computadores de escritorio, laptop y equipos a fines.	Configurar las estaciones de trabajo a un punto de restauración para que el equipo pueda ser restaurado en caso de haber sido vulnerado [8].
Referencia directa hacia un objeto de manera insegura	Información expuesta al público como archivos, directorios, registros de la base de datos etc. mediante una URL.	Hacer una revisión crítica de código y así identificar si hay información crítica susceptible de manipulación libre.	Evitar utilizar índices, o mapas indirectamente referenciados o cualquier método que apunte de manera directa hacia los objetos privados.
Sistemas operativos no seguros	Ataques para vulnerar al sistema mediante agujeros de seguridad.	Hacer una correcta administración de los sistemas operativos (cuentas de usuario,	Aplicar parches y actualizaciones oficiales del sistema operativo.

	establecer horarios de acceso)		Correcta configuración de los puertos, ruteadores, corta fuegos y demás dispositivos de la red. Cambiar contraseñas predeterminadas. Establecer políticas de contraseñas con mínimo de 12 caracteres alfanuméricos.
Red no segura	<p>Intrusión de software malicioso.</p> <p>Vulnerar equipos y realizar otros ataques.</p> <p>Robo de información empresarial.</p>	<p>Restringir el acoplamiento de a la red de nuevos equipos.</p>	
Phishing	Robo de toda la información posible al usuario (nombre de usuario, contraseñas, CI, etc.)	Utilizar filtro anti-spam.	Utilizar certificados digitales.
Captura de Teclado	Robo de información sensible (número de tarjetas de crédito,	No utilizar computadores de acceso al público	Utilizar teclados virtuales.
Usuarios	Robo de información	Limitar horarios de acceso.	Limitar los privilegios de acceso

Maliciosos	empresarial	Limitar la conexión de dispositivos extraíbles. Analizar el tráfico de la red como correos electrónicos, IM, etc.	a la información de acuerdo al perfil.
	Vulnerar seguridad de equipos.	No utilizar nombres, fechas, números de teléfono y en general información fácil de adivinar.	Contraseñas con caracteres como @\$*\$\$&() y con una longitud mínima de 12.
Diccionarios de	Robo de información.	Control remoto del equipo para realizar otros ataques.	Encriptación de paquetes con técnicas recientes como 3DES.
Datos			

Elaborado por: Gabriel Ignacio Villamagua Armijos

1.3. SELECCIÓN DE LAS HERRAMIENTAS DE DESARROLLO

La tecnología que se va a utilizar en el desarrollo de la aplicación es JSPs y SERVLETS con un patrón de desarrollo MVC. Además, se debe tener en cuenta las siguientes características de la aplicación:

- Seguridad.
- Desempeño.
- Escalabilidad.
- Persistencia de los datos.

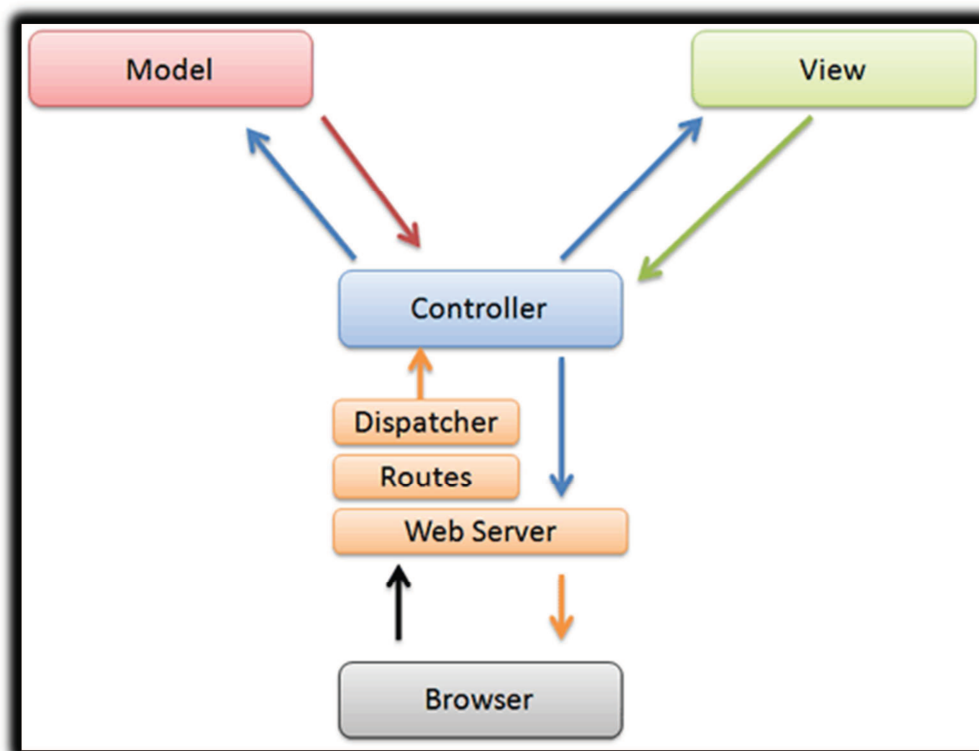
Para el desarrollo necesitamos de un DBMS, un servidor de aplicaciones, un IDE –para ayuda con el proceso de edición de código, el proceso de compilación, la depuración-, y una metodología de desarrollo.

El patrón de desarrollo a utilizar MVC ayuda a separar la lógica del negocio, de la interfaz de usuario y de la lógica de datos.

MVC tiene tres elementos que lo conforman:

- Modelo: datos y la lógica del negocio.
 - Accede a la capa de almacenamiento de datos.
 - Define las reglas del negocio.
- Vista: lo que el usuario visualiza.
 - Captura los datos y los envía al controlador.
 - Recibe los datos del modelo y los muestra al usuario.
- Controlador: realiza la parte dinámica.
 - Recibe los eventos de entrada.
 - Contiene reglas de gestión de eventos.

Ilustración 1.2 Modelo MVC



Fuente: <http://www.poignantcode.com/?p=70>

1.3.1. SELECCIÓN DEL DBMS

Las características en un DBMS que se está buscando son:

- Integridad.
- Escalabilidad.
- Bajo consumo de recursos.
- Alta Disponibilidad.
- Manejo de índices.
- Facilidad de acceso a los datos.
- Bajo costo.

Tabla 1.4 Características de algunos DBMS

	Oracle	Sybase	MySql	SQLServer	DB2	POSTGRESQL
Integridad	x	x	x	x	X	x
Bajo consumo de recursos		x	x	x	X	
Escalabilidad	x	x	x	x	X	x
Manejo de índices	x	x	x	x	X	x
Alta Disponibilidad	x	x	x	x	X	x
Bajo Costo			x			x

Fuente: <http://www.todoexpertos.com/categorias/tecnologia-e-internet/bases-de-datos/oracle/respuestas/14706/cuales-son-las-principales-ventajas-e-inconvenientes-de-las-diferentes-bases-de-datos>

Elaborado por: Gabriel Ignacio Villamagua Armijos

Tomando en cuenta los criterios anteriores y orientándonos por el software libre procedemos utilizar MySQL como DBMS.

1.3.1.1. MySQL

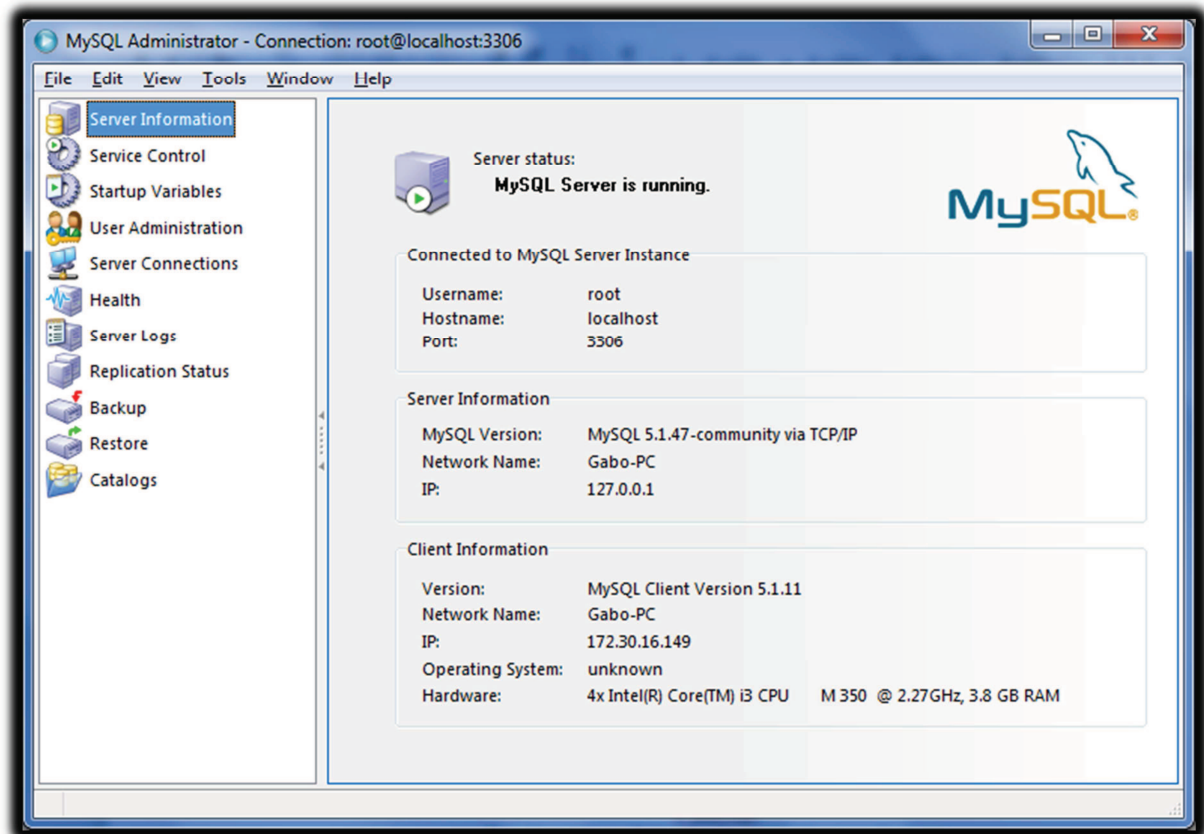
Es un DBMS perteneciente a ORACLE Corporation está liberado como software libre pero con licenciamiento dual ya que si se desea incorporar en un producto privativo se debe comprar una licencia específica.

Mysql es muy utilizado en desarrollo web debido a su simplicidad, además corre sobre múltiples plataformas UNIX, Linux, MAC y Windows. El consumo de recursos es muy bajo, los requerimientos del equipo para la instalación es muy baja por lo que se ahorra dinero al no gastar en equipos demasiado potentes.

Provee un módulo para realizar backups, restaurar bases de datos, administrar los usuarios, detener el servicio de MySQL y entre otras cosas llevar un registro

de las conexiones, número de consultas realizadas y el tráfico que existe en la red.

Ilustración 1.3 Pantalla principal de MySQL Administrator



Fuente: Instalación en PC de MySQL Server
Capturado por: Gabriel Ignacio Villamagua Armijos

1.3.2. SELECCIÓN DEL ENTORNO DE DESARROLLO INTEGRADO

Al lenguaje de programación JAVA (JSP's y SERVLET's) se lo ha tomado por su orientación a objetos, portabilidad, escalabilidad, corre sobre cualquier servidor web que soporte SERVLET's y JSP's, finalmente el consumo de recursos no es excesivo.

Para la edición de código, compilación, depuración y despliegue del proyecto web se procede a utilizar un entorno de desarrollo integrado (IDE) que nos facilitará todas las fases antes mencionadas.

Dos opciones de IDEs famosos que soportan JAVA y el patrón MVC son Eclipse y NetBeans. A continuación presentamos un cuadro con algunas características de estos IDE's.

Tabla 1.5 Comparación entre Netbeans y Eclipse

CARACTERÍSTICA	NETBEANS	ECLIPSE
Multiplataforma	X	X
Soporte Instalación Nuevos Plugins	X	X
Compatibilidad servidores web	X	X
Liviano (menor a los 500MB)	X	X
No necesita de instalación		X
Soporte SVN	X	X
Refactoring		X
Soporte MVC	X	X

Elaborado por: Gabriel Ignacio Villamagua Armijos

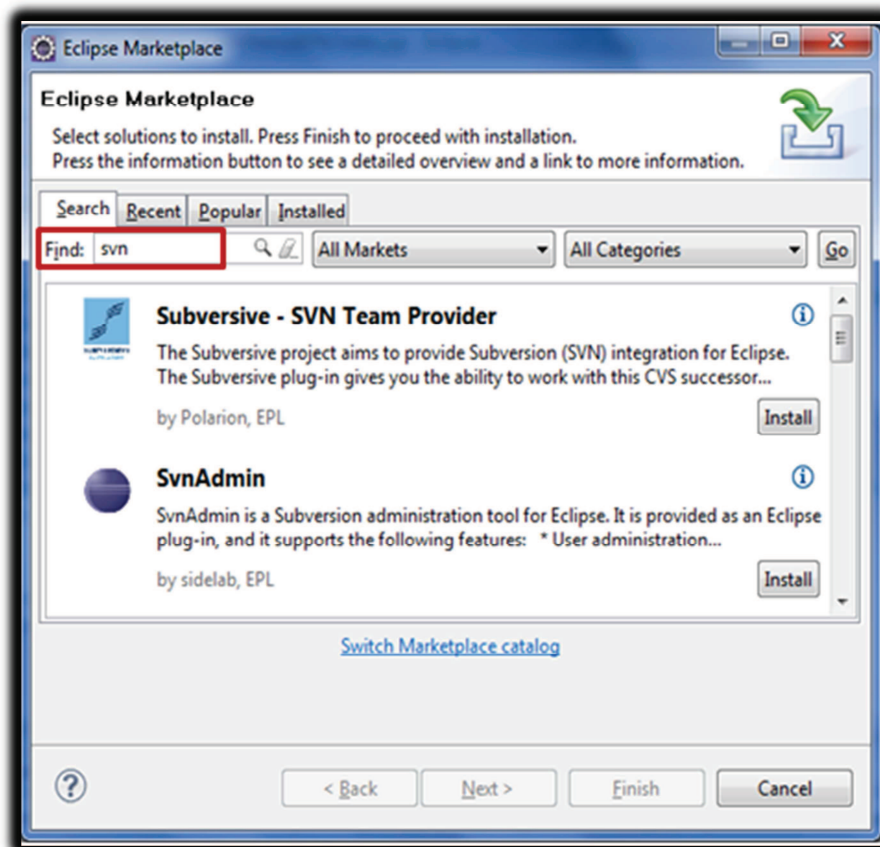
Por el cuadro comparativo mencionado anteriormente y por la estabilidad procedemos a orientarnos a ECLIPSE como IDE en nuestro desarrollo.

1.3.2.1. Eclipse

Actualmente se encuentra en su versión INDIGO liberada en 22 de Junio del 2011, es el IDE preferido por los desarrolladores de empresas grandes de desarrollo de software por su estabilidad.

Eclipse tiene un marketplace en el que se puede obtener nuevos plugins que se descargan desde un repositorio, un ejemplo es el plugin para cliente SVN. Está orientado principalmente a JAVA pero también es compatible con C, C++, Ada, Perl, Python, Ruby y PHP entre otros [9].

Ilustración 1.4 Eclipse Marketplace



Fuente: Eclipse IDE
Capturada por: Gabriel Ignacio Villamagua Armijos

1.3.3. SELECCIÓN DEL SERVIDOR WEB

Para que una aplicación web esté en línea se necesita de un servidor de aplicaciones web que esté constantemente escuchando las peticiones de los clientes. Como estamos orientados a JSP's y SERVLET's un servidor IIS no nos sería útil ya que este se encuentra optimizado para aplicaciones propiamente MICROSOFT.

Se necesita de un servidor que no ocupe muchos recursos, que sea rápido y soporte JSP y SERVLETS. Anteriormente ya definimos que se va a utilizar Eclipse

por lo que necesitamos de un servidor que sea compatible con este IDE (no obligatorio).

Los servidores soportados que vienen predeterminados en Eclipse son Apache Tomcat, IBM Websphere, JBoss, JOnAS y Oracle OC4J; de estos vamos a elegir a los 3 primeros ya que son los más conocidos en el mercado y poseen mayor soporte.

Tabla 1.6 Comparación de tres servidores web

Características	Apache Tomcat	IBM Websphere	JBoss
Bajo Consumo de recursos	x		
Soporte de JSP y SERVLETS	x	x	x

Elaborado por: Gabriel Ignacio Villamagua Armijos

Aunque los tres servidores cumplen con el soporte de JSP's y SERVLET's estos tienen un consumo más amplio de recursos que Apache Tomcat, esto se debe a que tienen más funcionalidades como soporte de EJB's las cuales son un consumo en vano de recursos ya que no se procederá a utilizarlas para el desarrollo, por estas razones nos orientamos por Apache Tomcat como servidor web.

1.3.3.1. Apache Tomcat

Es un servidor web escrito en Java, es multiplataforma por lo que no importa el sistema operativo solamente se debe disponer de la máquina virtual Java [10].

Apache Tomcat consta de los siguientes componentes:

- Catalina
- HTTP Connector
- JASPER

Catalina: es un contenedor de servlets [11].

Componente HTTP Connector: soporta el protocolo HTTP el cual escucha las conexiones entrantes de un puerto TCP específico en el servidor y reenvía la petición al motor de Tomcat y finalmente envía la respuesta al cliente que lo solicitó [12].

JASPER: es un motor de JSP's, el proceso que se realiza es parsear archivos JSP's para compilar a código JAVA como SERVLETS, en tiempo de ejecución JASPER detecta cambios en los JSP's y procede automáticamente a recompilarlos [12].

1.3.4. SELECCIÓN DE LA METODOLOGÍA DE DESARROLLO

Para el desarrollo de la aplicación web se ha decidido utilizar SCRUM, ya que se ajusta en la mayoría de sus aspectos al desarrollo de la aplicación bancaria.

1.3.4.1. SCRUM

SCRUM es una metodología para gestionar proyectos, la cual se basa en dos principios fundamentales:

- Desarrollo rápido muy cambiante.
- Ciclos de desarrollo cortos [13].

SCRUM evita la generación la excesiva documentación, esto no quiere decir que no se deba documentar más bien que la documentación no es una exigencia

previa para comenzar el proyecto. A lo que se orienta con la documentación es que se tenga un producto visible con el cual el cliente pueda evaluar el avance del proyecto y poder satisfacer los requerimientos en su totalidad.

Existen dos elementos:

- Actores: personas encargadas del proceso completo de desarrollo.
- Acciones: cada tarea a realizarse, ya sea en cada iteración (SPRINT) o una vez si el proyecto es relativamente pequeño.

Se definen 4 actores:

- Un propietario del producto (product owner), el cual define las prioridades del proyecto o producto.
- Un administrador SCRUM (SCRUM manager o SCRUM master), se asegura del seguimiento de la metodología por parte del equipo.
- El equipo de SCRUM (SCRUM team) es el encargado de implementar las funcionalidades del propietario del producto.
- Usuarios o clientes que son los que se benefician del producto, ellos pueden aportar con ideas, sugerencias o alguna necesidad que surja [14].

Las acciones básicas de la metodología SCRUM son:

- Pila del producto (Product Backlog): es una lista de tareas, funcionalidades o requerimientos que van a ser realizados, el propietario del producto es la persona encargada de mantener esta lista y marcar las propiedades del proyecto.
- Pila del Sprint (Sprint Backlog): es un listado de tareas obtenido de la del producto. Estas tareas deben ser realizadas en un periodo de 2 a 4 semanas. Cuando ya se ha iniciado la pila del Sprint, esta no se puede ser alterada o modificada. Se debe esperar al final del desarrollo del Sprint actual y luego modificar la tarea deseada, y esta pasaría a ser parte de una nueva pila del Sprint.

- La reunión diaria de SCRUM (Daily SCRUM Meeting), es una tarea realizada diariamente mientras dura el desarrollo de la pila Sprint, esta reunión es de máximo 30 minutos. Los temas principales a tratar en esta reunión diaria de SCRUM son:
 - Tareas que se han realizado desde la anterior reunión.
 - Tareas a realizar en ese día.
 - Problemas u obstáculos para realizar alguna tarea determinada.

Al final de cada SPRINT se realiza una reunión para realizar un análisis del cumplimiento de los objetivos.

Al final de la culminación del último SPRINT se tiene una reunión para realizar un análisis para saber si se cumplieron los objetivos iniciales que se fijaron, esto se lo realiza con el propietario del producto. No solo se marcarán los aspectos positivos sino también los negativos que sucedieron durante todos los SPRINT's, en este punto se debe tener un documento en el que el propietario del producto pueda evaluar.

Un proyecto se da por finalizado cuando todos los SPRINT's se han finalizado y no queda ningún requerimiento por cumplir.

1.3.4.2. Razones para usar SCRUM

Existen muchas metodologías orientadas al desarrollo de aplicaciones web sin embargo se ha seleccionado SCRUM por las siguientes razones:

- La web es muy cambiante a paso muy rápido por lo que se necesita de un desarrollo ágil para que las aplicaciones no queden en la obsolescencia incluso antes de que se ponga a producción. SCRUM plantea un desarrollo altamente veloz por lo que soluciona este inconveniente.

- Los proyectos muchas veces son informales por lo que a veces la documentación queda incompleta. SCRUM deja como una posibilidad hacer la documentación lo cual beneficia a un proyecto altamente veloz.
- Al desarrollar aplicaciones se tiene un equipo heterogéneo como son programadores, diseñadores, etc. Al tener una gran cantidad de reuniones planteadas por la metodología mejora la comunicación entre los mismos.
- SCRUM permite realizar Sprint en el que se definen tareas lo que lo hace iterativo, esto ayuda ya que un aplicación bancaria no puede esperar a estar desarrollada completamente para ponerla a producción más bien se la puede desarrollar por partes (Sprint) e ir poniéndola a producción.

CAPÍTULO 2.

ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DEL SISTEMA

Para el análisis, diseño y construcción del sistema de acuerdo a lo que se establece en el capítulo 1, se utilizará la metodología SCRUM.

2.1. ESPECIFICACIÓN DE REQUERIMIENTOS

La flexibilidad de SCRUM permite que se utilice cualquier herramienta para complementarse [13], en nuestro caso para la captura de requerimientos resultantes de la interacción con el propietario del producto utilizaremos el patrón conocido como Historia de Usuario.

Historia de Usuario	
Número: 1	Nombre: Registrar cliente en la banca electrónica.
Usuario: Cliente	
Prioridad en el negocio (alta, media, baja): alta	Riesgo en el desarrollo (alto, medio, bajo): alto
Responsable: Gabriel Villamagua	
Descripción: El sistema permitirá al cliente registrarse en la banca electrónica, para esto debe ser cliente del banco y poseer una cuenta, para esto debe llenar la información personal y de autenticación de la banca.	
¿Cómo probarlo?	
<ul style="list-style-type: none"> • Ingresar número de tarjeta de débito y contraseña. • El sistema realiza la lectura en la base de datos y valida la información. • Ingresar/Actualizar la información que se solicita. • Seleccionar un usuario y contraseña. • El sistema realiza la lectura en la base de datos y valida la información. 	

- Finalmente se realiza el proceso de almacenamiento en la base de datos.

Historia de Usuario	
Número: 2	Nombre: Iniciar Sesión de Cliente
Usuario: Cliente	
Prioridad en el negocio (alta, media, baja): alta	Riesgo en el desarrollo (alto, medio, bajo): alto
Responsable: Gabriel Villamagua	
Descripción: El sistema permitirá al cliente iniciar sesión en la banca electrónica, para esto el cliente previamente debe haberse registrado. El sistema consta de una doble autenticación es decir que a más del usuario y contraseña se pedirá que se conteste una pregunta y se seleccione una imagen que se registró anteriormente.	
¿Cómo probarlo? <ul style="list-style-type: none"> • Ingresar usuario y contraseña • El sistema realiza la lectura en la base de datos y valida la información. • Contestar el desafío de la pregunta y seleccionar una imagen ya ingresada en el registro. • El sistema realiza la lectura en la base de datos y valida la información. • Si la información es correcta se procede a iniciar la sesión del usuario, caso contrario se procede a bloquear. 	

Historia de Usuario	
Número: 3	Nombre: Finalización de sesión.
Usuario: Cliente	
Prioridad en el negocio (alta, media, baja): alta	Riesgo en el desarrollo (alto, medio, bajo): alto
Responsable: Gabriel Villamagua	
Descripción: El sistema permitirá al cliente terminar la sesión cuando lo desee. Para esto el usuario debe haber iniciado sesión exitosamente, el proceso de cierre de sesión evita que otro usuario pueda acceder a la información personal y financiera del	

propietario de la cuenta.
<p>¿Cómo probarlo?</p> <ul style="list-style-type: none"> • Hacer click en el botón de cierre de sesión. • El sistema procede a borrar la sesión y la caché del servidor web. • Finalmente se envía mensaje de cierre exitoso de sesión.
<p>Observación:</p> <ul style="list-style-type: none"> • La variable SESSION del servidor web tiene un ciclo de vida, en el cual si no se tiene un periodo de actividad por un tiempo, esta variable procede a eliminar las sesiones activas por lo que el cliente nuevamente debe iniciar sesión.

Historia de Usuario	
Número: 4	Nombre: Actualización de datos de autenticación
Usuario: Cliente	
Prioridad en el negocio (alta, media, baja): alta	Riesgo en el desarrollo (alto, medio, bajo): alto
Responsable: Gabriel Villamagua	
<p>Descripción:</p> <p>El sistema permitirá al cliente actualizar la información de autenticación (imagen y preguntas), información que se utilizará para iniciar sesión en la banca electrónica.</p>	
<p>¿Cómo probarlo?</p> <ul style="list-style-type: none"> • Iniciar sesión en la banca electrónica. • Seleccionar la opción actualizar información de autenticación. • El sistema procede a leer de la base de datos y obtener la información almacenada. • Los campos que no se pueden actualizar aparecen desactivados. • Ingresamos la nueva información en el formulario. • El sistema valida la información ingresada. • El sistema registra la nueva información en la base de datos. 	

Historia de Usuario	
Número: 5	Nombre: Actualización de información personal.
Usuario: Cliente	
Prioridad en el negocio (alta, media, baja): media	Riesgo en el desarrollo (alto, medio, bajo): alto
Responsable: Gabriel Villamagua	
Descripción: El sistema permitirá al cliente actualizar la información personal (dirección, correo electrónico, teléfono, número de celular, etc.), esta información no es necesaria para el proceso de inicio de sesión pero permite establecer la ubicabilidad del cliente.	
¿Cómo probarlo? <ul style="list-style-type: none"> • Iniciar sesión en la banca electrónica. • Seleccionar la opción actualizar información personal. • El sistema procede a leer de la base de datos y obtener la información almacenada. • Ingresamos/actualizamos la información en el formulario. • El sistema valida la información ingresada. • El sistema registra la nueva información en la base de datos. 	

Historia de Usuario	
Número: 6	Nombre: Bloquear usuario
Usuario: Administrador	
Prioridad en el negocio (alta, media, baja): alta	Riesgo en el desarrollo (alto, medio, bajo): media
Responsable: Gabriel Villamagua	
Descripción: El sistema permitirá al administrador bloquear a un usuario cuando este lo desee o por contravenciones a la ley (dinero proveniente de lavado de activos y	

cualquier acto ilícito) y de esta manera el usuario no podrá hacer uso de la banca electrónica.

¿Cómo probarlo?

- Iniciar sesión en la banca electrónica como usuario administrador.
- Seleccionar la opción de gestión de usuarios.
- Ingresar la cédula o apellidos del cliente a bloquear.
- El sistema procede a leer de la base de datos y obtener la información almacenada.
- Presionamos el botón bloquear.

Historia de Usuario	
Número: 7	Nombre: Desbloquear usuario
Usuario: Administrador	
Prioridad en el negocio (alta, media, baja): alta	Riesgo en el desarrollo (alto, medio, bajo): media
Responsable: Gabriel Villamagua	
<p>Descripción:</p> <p>El sistema permitirá al administrador desbloquear a un usuario cuando este lo desee ya sea que haya sido bloqueado por un acceso erróneo, por petición antes hecha del usuario o porque fue bloqueado por investigaciones contra actos ilícitos.</p>	
<p>¿Cómo probarlo?</p> <ul style="list-style-type: none"> • Iniciar sesión en la banca electrónica. • Seleccionar la opción de gestión de usuarios. • Ingresar la cédula o apellidos del cliente a bloquear. • El sistema procede a leer de la base de datos y obtener la información almacenada. • Presionamos el botón desbloquear. 	

Historia de Usuario	
Número: 8	Nombre: Transferencia
Usuario: Cliente	
Prioridad en el negocio (alta, media, baja): baja	Riesgo en el desarrollo (alto, medio, bajo): bajo
Responsable: Gabriel Villamagua	
Descripción: El sistema permitirá al cliente realizar transferencias entre cuentas propias, para lo cual el usuario debe haberse registrado e iniciado sesión.	
¿Cómo probarlo? <ul style="list-style-type: none"> • Iniciar sesión en la banca electrónica. • Seleccionar la opción de transferencia entre cuentas propias. • Ingresar la cuenta destino, el monto y el mail del destinatario. • El sistema procede a leer de la base de datos, obtener la información almacenada y validar que esté correcta. • Finalmente el sistema realiza el débito y crédito en las correspondientes cuentas. 	

Historia de Usuario	
Número: 8	Nombre: Transferencias entre cuentas propias
Usuario: Cliente	
Prioridad en el negocio (alta, media, baja): baja	Riesgo en el desarrollo (alto, medio, bajo): bajo
Responsable: Gabriel Villamagua	
Descripción: El sistema permitirá al cliente realizar transferencias entre cuentas propias, para lo cual el usuario debe haberse registrado e iniciado sesión.	
¿Cómo probarlo? <ul style="list-style-type: none"> • Iniciar sesión en la banca electrónica. • Seleccionar la opción de transferencia entre cuentas propias. • Ingresar la cuenta destino, el monto y el mail del destinatario. 	

- El sistema procede a leer de la base de datos, obtener la información almacenada y validar que esté correcta.
- Finalmente el sistema realiza el débito y crédito en las correspondientes cuentas.

Historia de Usuario	
Número: 9	Nombre: Recuperación de contraseña
Usuario: Cliente	
Prioridad en el negocio (alta, media, baja): alta	Riesgo en el desarrollo (alto, medio, bajo): bajo
Responsable: Gabriel Villamagua	
Descripción: El sistema permitirá al cliente recuperar la contraseña en caso de que se haya olvidado, para esto es necesario contar con la el número de la tarjeta de crédito y la contraseña, y pasar el desafío de las preguntas y selección de imagen.	
¿Cómo probarlo? <ul style="list-style-type: none"> • Seleccionar la opción de recuperación de contraseña. • Ingresar el número de la tarjeta de débito y contraseña. • El sistema procede a leer de la base de datos, obtener la información almacenada y validar que esté correcta. • Contestar la pregunta y seleccionar la pregunta. • El sistema procede a leer de la base de datos, obtener la información almacenada y validar que esté correcta. • Finalmente ingrese su nueva contraseña. • El sistema procede a almacenar la información en la base de datos. • El sistema procede a enviar un sms y un correo electrónico al cliente con la confirmación de cambio de contraseña. 	

Historia de Usuario	
Número: 10	Nombre: Recordar usuario banca electrónica.
Usuario: Cliente	

Prioridad en el negocio (alta, media, baja): baja	Riesgo en el desarrollo (alto, medio, bajo): bajo
Responsable: Gabriel Villamagua	
Descripción: El sistema permitirá al cliente recordar su usuario en caso de que este haya sido olvidado, la manera de recordarlo será mediante sms y correo electrónico.	
¿Cómo probarlo? <ul style="list-style-type: none"> • Seleccionar la opción de recordar usuario. • Ingresar el correo electrónico con el que se registró. • El sistema procede a leer de la base de datos, obtener la información almacenada y validar que esté correcta. • El sistema envía un correo electrónico y sms con el usuario. 	

Historia de Usuario	
Número: 11	Nombre: Crear nuevo usuario administrador
Usuario: Administrador	
Prioridad en el negocio (alta, media, baja): media	Riesgo en el desarrollo (alto, medio, bajo): bajo
Responsable: Gabriel Villamagua	
Descripción: El sistema permitirá al cliente administrador crear un nuevo cliente administrador, para lo cual necesita la identificación y el nombre.	
¿Cómo probarlo? <ul style="list-style-type: none"> • Iniciar sesión como administrador. • Selecciona la opción nuevo usuario. • Ingresar la identificación y el nombre. • El sistema procede a leer de la base de datos, obtener la información almacenada y validar que esté correcta (usuario no esté creado). 	

2.2. ANÁLISIS DEL SISTEMA

Ahora procedemos a definir la pila del producto y la planificación de cada SPRINT en base a las historias de usuario previamente definidas.

La pila del producto incluye los requerimientos y prioridades que se necesitan para obtener el producto deseado, en nuestro caso el producto deseado es la “Banca Electrónica LIX”.

A continuación se incluye la pila del producto representada en una tabla con los siguientes campos:

Nº: entero positivo secuencial para identificar a cada requerimiento.

Requerimiento: indica el nombre que se le asigna al requerimiento.

Descripción: breve párrafo que especifica que es lo que se va a realizar en ese requerimiento.

Prioridad: indica el orden con el que se van a implementar cada requerimiento.

Dificultad: un número entero entre 1 y 5 que especifica cual fácil o complejo es la implementación de dicho requerimiento, 1 significa fácil y 5 significa complejo [15].

Tabla 2.1 Pila del producto de la Banca Electrónica LIX

Nº	Requerimiento	Descripción	Prioridad	Dificultad
1	Diseño de Diagrama Entidad-Relación.	Gabriel Villamagua	1	3
2	Generación de modelo físico	Gabriel Villamagua	1	3
3	Generación código SQL para MySQL.	Gabriel Villamagua	1	2
4	Creación de índices	Gabriel Villamagua	1	3
5	Refinamiento de datos	Gabriel Villamagua	1	3
6	Importar usuarios base	Gabriel Villamagua	1	4

	del banco			
7	Verificar consistencia de datos	Gabriel Villamagua	1	4
8	Creación de procedimientos almacenados para transacciones	Gabriel Villamagua	1	3
9	Creación de procedimientos almacenados para autenticación	Gabriel Villamagua	1	3
10	Pruebas unitarias.	Gabriel Villamagua	1	3
11	Control de calidad	Ing. Bolívar Palán	1	4
12	Afinamiento de las estructuras y procedimientos de la base de datos.	Gabriel Villamagua	1	3
13	Selección del servidor de correo.	Gabriel Villamagua	2	3
14	Definición de la arquitectura (Definición de hilos)	Gabriel Villamagua	2	4
15	Cargar dependencias (archivos jar) para el envío de correo usando JAVA.	Gabriel Villamagua	2	2
16	Diseño de clase, implementación de métodos y constructores para manejo de hilos (Threads) en caso general	Gabriel Villamagua	2	5
17	Pruebas unitarias para funcionamiento de hilos	Gabriel Villamagua	2	4

18	Integración de correo electrónico con hilos (Threads)	Gabriel Villamagua	2	5
19	Pruebas unitarias.	Gabriel Villamagua	2	4
20	Control calidad	Ing. Bolívar Palán	2	4
21	Afinamiento primera versión de la banca electrónica.	Gabriel Villamagua	2	3
22	Definición de la infraestructura necesaria para el envío de sms.	Gabriel Villamagua	3	5
23	Instalación de gammu para gestión de dispositivo móvil (teléfono móvil).	Gabriel Villamagua	3	5
24	Selección y configuración del dispositivo móvil.	Gabriel Villamagua	3	5
25	Pruebas de envío de mensajes de texto mediante la terminal (GNU/Linux).	Gabriel Villamagua	3	4
26	Diseño de clase, implementación de métodos y constructores para comunicación con gammu desde JAVA.	Gabriel Villamagua	3	4
27	Integración de hilos en la clase de comunicación con gammu.	Gabriel Villamagua	3	4
28	Pruebas unitarias.	Gabriel Villamagua	3	4
29	Control de calidad	Ing. Bolívar Palán	3	4
30	Afinamiento de segunda versión de la banca	Gabriel Villamagua	3	2

	electrónica.			
31	Diseño de JSP de selección de imagen y contestar pregunta.	Gabriel Villamagua	3	3
32	Diseño de JSP de Inicio de sesión.	Gabriel Villamagua	3	3
33	Diseño de JSP de registro y actualización de datos personales	Gabriel Villamagua	3	3
34	Diseño de JSP de registro y actualización de datos de autenticación.	Gabriel Villamagua	3	3
35	Diseño de JSP recuperación de contraseña.	Gabriel Villamagua	3	3
36	Diseño de JSP principal de cliente.	Gabriel Villamagua	3	3
37	Diseño de JSP principal de administrador.	Gabriel Villamagua	3	3
38	Implementación de JSP y SERVLET de selección de imagen y pregunta.	Gabriel Villamagua	3	4
39	Implementación de servicio de selección de imagen y pregunta.	Gabriel Villamagua	3	4
40	Implementación de JSP y SERVLET de inicio de sesión.	Gabriel Villamagua	3	4
41	Implementación de servicio de inicio de sesión.	Gabriel Villamagua	3	4
42	Implementación de JSP y SERVLET de registro y	Gabriel Villamagua	3	4

	actualización de datos personales.			
43	Implementación de servicio de registro y actualización de datos personales	Gabriel Villamagua	3	4
44	Implementación de JSP y SERVLET de registro y actualización de datos de autenticación.	Gabriel Villamagua	3	4
45	Implementación de servicio de registro y actualización de datos de autenticación.	Gabriel Villamagua	3	4
46	Implementación de JSP y SERVLET de recuperación de contraseña.	Gabriel Villamagua	3	4
47	Implementación de servicio de registro de recuperación de contraseña.	Gabriel Villamagua	3	4
48	Implementación de JSP y SERVLET para transferencias de la pantalla principal de cliente.	Gabriel Villamagua	4	4
49	Implementación de servicio para transferencias interbancarias en la pantalla de cliente.	Gabriel Villamagua	4	4
50	Implementación de JSP y SERVLET para bloqueo y desbloqueo de clientes.	Gabriel Villamagua	4	4

51	Implementación de servicio para bloqueo y desbloqueo de clientes.	Gabriel Villamagua	4	4
52	Pruebas unitarias.	Gabriel Villamagua	4	4
53	Control de calidad.	Ing. Bolívar Palán	4	5
54	Afinamiento de cuarta versión de la banca electrónica.	Gabriel Villamagua	4	3
55	Implementación de patrón visual de tablas (color, bordes, fuente de las celdas, etc.).	Gabriel Villamagua	5	3
56	Implementación de patrón visual de títulos, leyendas, advertencias y links.	Gabriel Villamagua	5	3
57	Implementación de patrón visual de fondos de pantalla, fuente de letra y menús.	Gabriel Villamagua	5	3
58	Implementación de validación de ingreso de información en formularios.	Gabriel Villamagua	5	3
59	Prueba final de la banca electrónica.	Gabriel Villamagua	5	3
60	Control de calidad	Ing. Bolívar Palán	5	4
61	Afinamiento de reléase final.	Gabriel Villamagua	5	5

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1. ESPECIFICACIÓN DE SPRINTS

2.2.1.1. Primer Sprint

Este SPRINT comprende las tareas 1 a la 12 de la pila del producto especificada en la tabla 2.1, empezará el 2 de abril del 2012 y tiene una duración de 10 días y la fecha de la revisión será el 13 de Abril del 2012. Las reuniones diarias serán de máximo 15 minutos a primera hora de la mañana cuando todo el equipo se encuentre presente.

El objetivo del SPRINT es desarrollar el esquema de la base de datos y posteriormente generar el código SQL que será ejecutado en el DBMS MySQL.

En la tabla 2.2 se detalla las tareas a realizar en el SPRINT, los campos de la tabla son:

N°: corresponde a un secuencial de la tarea.

TAREA: nombre de la tarea.

RESPONSABLE: responsable de realizar la tarea.

TIEMPO ESTIMADO: número de horas estimadas para la realización de la tarea [15].

Tabla 2.2 SPRINT primero Banca Electrónica LIX

N°	TAREA	TIEMPO ESTIMADO
1	Diseño de Diagrama Entidad-Relación.	8 horas
2	Generación de modelo físico	4 horas
3	Generación código SQL para MySQL.	8 horas
4	Creación de índices	4 horas
5	Refinamiento de datos	8 horas
6	Importar usuarios base del	8 horas

	banco	
7	Verificar consistencia de datos	8 horas
8	Creación de procedimientos almacenados para transacciones	10 horas
9	Creación de procedimientos almacenados para autenticación	10 horas
10	Pruebas unitarias.	4 horas
11	Control de calidad	4 horas
12	Afinamiento de las estructuras y procedimientos de la base de datos.	4 horas
		80 horas

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.1.1. Elaboración y planificación

Para la gestión de los diferentes SPRINTS que se realizarán en el transcurso del desarrollo e implementación de la banca electrónica se utilizará una tabla, la cual consta de los campos que se detallan a continuación:

PROYECTO: nombre del proyecto.

Nº SPRINT: número de SPRINT.

INICIO: fecha de inicio de la iteración.

DÍAS: número de días que durará la iteración.

JORNADA: número de horas diarias de trabajo.

TAREAS: indica los tipos de tareas y los diferentes estados durante la iteración.

EQUIPO: indica los miembros del equipo.

FESTIVOS: indica días festivos que coinciden en el transcurso de la iteración.

Tabla 2.3 Datos de configuración de la pila de iteración

PROYECTO			
Banca Electrónica LIX			
Nº DE SPRINT	INICIO	DÍAS	JORNADA
1	2-abril-2012	10	8
TAREAS		EQUIPOS	FESTIVOS
TIPOS	ESTADOS		
Análisis	Pendiente	Ing. Bolívar Palán	
Codificación	En curso	Gabriel Villamagua	
Prototipado	Terminada		
Pruebas	Eliminada		
Reunión			

Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

En las reuniones diarias de máximo 15 minutos se debe tratar los siguientes aspectos:

- Tareas que se trabajaron el día anterior.
- Tareas a trabajar en ese día.
- Impedimentos para realizar su trabajo y si necesita de algún requerimiento para realizar la tarea.

Con esto procedemos a actualizar la pila de tareas de la iteración, así de esta manera la pila será actualizada diariamente.

Tabla 2.4 SPRINT Planning Meeting primera iteración

FECHA	TAREAS PENDIENTES	HORAS PENDIENTES	RESPONSABLE	ACTIVIDAD ANTERIOR	ACTIVIDAD ACTUAL	REQUERIMIENTOS EXTRAS
02/Abril/2012	11	80	Gabriel Villamagua	-----	Diseño de Diagrama Entidad-Relación.	NA
03/Abril/2012	10	72	Gabriel Villamagua	Diseño de Diagrama Entidad-Relación	Generación de modelo fisico	Requerimientos funcionales capturados en las historias de usuario
03/Abril/2012	9	68	Gabriel Villamagua	Generación del modelo fisico	Generación de código SQL para MySQL.	NA
04/Abril/2012	8	60	Gabriel Villamagua	Generación de código SQL para MySQL	Creación de índices	Instalación de MySQL
05/Abril/2012	7	56	Gabriel Villamagua	Creación de índices	Refinamiento de datos	NA
06/Abril/2012	6	48	Gabriel Villamagua	Refinamiento de datos	Importar usuarios base del banco	Tener los backups de la base de datos del banco

09/Abril/2012	5	40	Gabriel Villamagua	Importar usuarios base del banco	Verificar consistencia de datos	NA
10/Abril/2012	4	32	Gabriel Villamagua	Verificar consistencia de datos	Creación de procedimientos almacenados para transacciones	NA
11/Abril/2012	4	22	Gabriel Villamagua	Creación de procedimientos almacenados para transacciones	Creación de procedimientos almacenados para autenticación	NA
12/Abril/2012	3	12	Gabriel Villamagua	Creación de procedimientos almacenados para autenticación	Pruebas unitarias.	NA
13/Abril/2012	2	8	Ing. Bolívar Palán	-----	Control de calidad	NA
13/Abril/2012	1	4	Gabriel Villamagua	Pruebas unitarias	Afinamiento de las estructuras y procedimientos de la base de	NA

				datos.
--	--	--	--	--------

Elaborado por: Gabriel Ignacio Villamagua Armijos

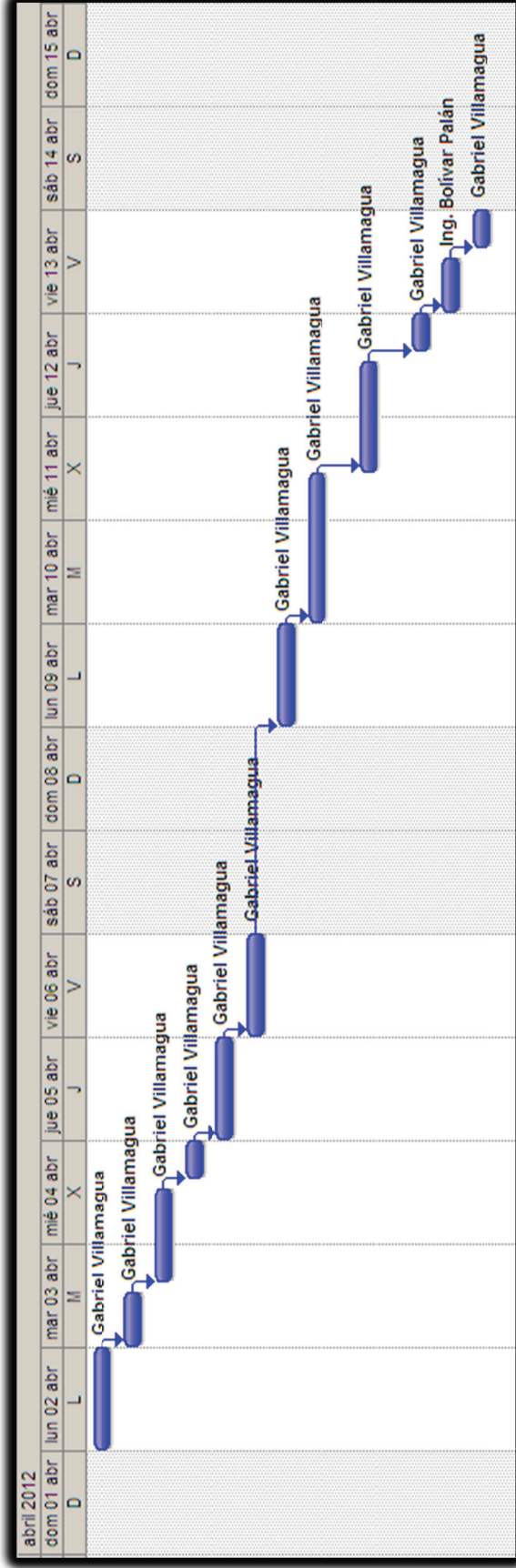
La siguiente figura permite visualizar el esfuerzo de la pila del SPRINT, se especifica tareas, el estado y la persona del equipo responsable de la iteración.

Ilustración 2.1 Diagrama de Gantt de Sprint 01 (Parte 1)

i	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	Diseño de Diagrama Entidad-Relación.	8 horas	lun 02/04/12	lun 02/04/12		Gabriel Villamagua
2	Generación de modelo físico	4 horas	mar 03/04/12	mar 03/04/12	1	Gabriel Villamagua
3	Generación código SQL para MySQL.	8 horas	mar 03/04/12	mié 04/04/12	2	Gabriel Villamagua
4	Creación de índices	4 horas	mié 04/04/12	mié 04/04/12	3	Gabriel Villamagua
5	Refinamiento de datos	8 horas	jue 05/04/12	jue 05/04/12	4	Gabriel Villamagua
6	Importar usuarios base del banco	8 horas	vie 06/04/12	vie 06/04/12	5	Gabriel Villamagua
7	Verificar consistencia de datos	8 horas	lun 09/04/12	lun 09/04/12	6	Gabriel Villamagua
8	Creación de procedimientos almacenados para transacciones	10 horas	mar 10/04/12	mié 11/04/12	7	Gabriel Villamagua
9	Creación de procedimientos almacenados para autenticación	10 horas	mié 11/04/12	jue 12/04/12	8	Gabriel Villamagua
10	Pruebas unitarias.	4 horas	jue 12/04/12	jue 12/04/12	9	Gabriel Villamagua
11	Control de calidad	4 horas	vie 13/04/12	vie 13/04/12	10	Ing. Bolívar Palán
12	Afinamiento de las estructuras y procedimientos de la base de datos.	4 horas	vie 13/04/12	vie 13/04/12	11	Gabriel Villamagua

Elaborado por: Gabriel Ignacio Villamagua Armijos

Ilustración 2.2 Diagrama de Gantt de Sprint 01 (Parte 2)



Elaborado por: Gabriel Ignacio Villamagua Armijos

Ilustración 2.3 Esfuerzo realizado en el primer SPRINT

SPRINT		INICIO	DURACIÓN																		
1		2-abr-12	10																		
PILA DEL SPRINT				Tarea	Tipo	Estad.	Responsable	ESFUERZO													
								L	M	X	J	V	L	L	M	X	J	V	L		
				Diseño de Diagrama E-R	Análisis	Terminada	Gabriel Villamagua	8													
				Generación Modelo Físico	Prototipado	Terminada	Gabriel Villamagua	4													
				Generación Código SQL para MySql	Pruebas	Terminada	Gabriel Villamagua	4	4												
				Creación de Índices	Codificación	Terminada	Gabriel Villamagua	4	4												
				Refinamiento de Datos	Codificación	Terminada	Gabriel Villamagua	8													
				Importar usuarios bdd del banco	Pruebas	Terminada	Gabriel Villamagua	8													
				Verificar consistencia de datos	Codificación	Terminada	Gabriel Villamagua	8													
				Creación de procedimientos almacenados para transacciones	Codificación	Terminada	Gabriel Villamagua	8	2												
				Creación de procedimientos almacenados para autenticación	Pruebas	Terminada	Gabriel Villamagua	6	4												
				Pruebas Unitarias	Pruebas	Terminada	Gabriel Villamagua	4													
				Control de Calidad	Codificación	Terminada	Ing. Bolívar Palán	4													
				Afinamiento de las estructuras y procedimientos de la bdd	Codificación	Terminada	Gabriel Villamagua	4													

Fuente: Plantilla tomada de www.navegapolis.net

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.1.2. Gráfico Burn Down Chart

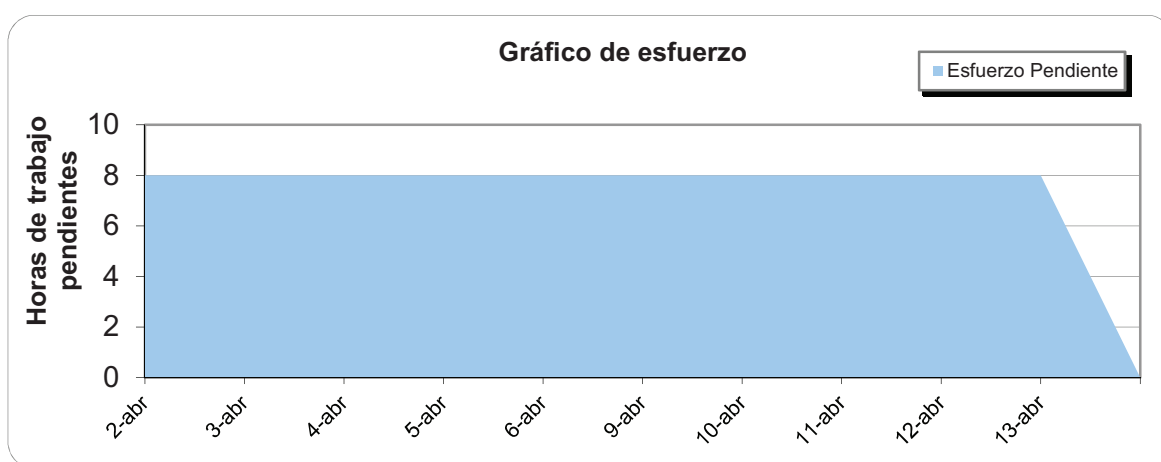
Burn Down Chart es un gráfico cartesiano que representa en el eje **x** los días laborables del SPRINT, y en el eje **y** la cantidad de esfuerzo estimada.

Para la construcción de este gráfico cada miembro del equipo al referirse al trabajo de ayer y el que tiene previsto de hacer hoy, y de esa manera se va actualizando el gráfico, con esto se muestra el esfuerzo que según el equipo falta para terminar.

La Ilustración 2.4 muestra las horas restantes a lo largo de la fecha de inicio hasta la fecha de finalización del SPRINT.

Ilustración 2.4 Esfuerzo realizado del primer SPRINT

Proyecto	SPRINT	INICIO	DÍAS
Banca Electrónica LIX	1	2-abr-12	10

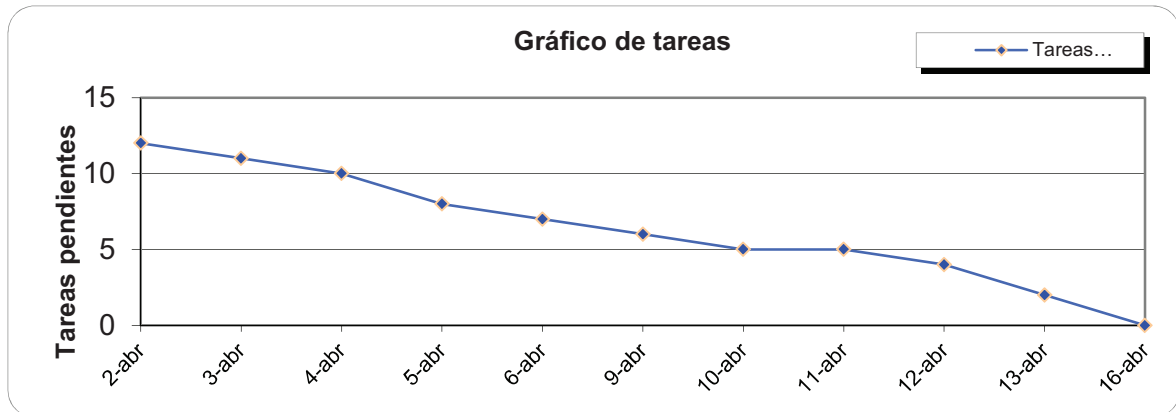


Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

Como podemos visualizar en el gráfico el decremento de horas se hace de forma constante, lo que indica que se hizo una buena estimación de la pila del producto.

En la ilustración 2.3 se muestra la lista de tareas y el avance realizado durante la iteración hasta su finalización.

Ilustración 2.5 Avance de tareas durante el primer SPRINT



Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

Como podemos visualizar en el gráfico algunas tareas tuvieron más dificultad que las demás tareas

2.2.1.2. Segundo sprint

Este SPRINT comprende las tareas del 13 a la 21 de la pila del producto especificada en la tabla 2.1, empezará el 16 de abril del 2012 y tiene una duración de 7 días y la fecha de la revisión será el 24 de Abril del 2012. Las reuniones diarias serán de máximo 15 minutos a primera hora de la mañana cuando todo el equipo se encuentre presente.

El objetivo del SPRINT es desarrollar el módulo encargado de realizar el envío de correos electrónicos a los usuarios de la banca electrónica.

En la tabla 2.5 se detalla las tareas a realizar en el SPRINT, esta tabla tiene el formato que anteriormente ya se especificó.

Tabla 2.5 SPRINT segundo Banca Electrónica LIX

N°	TAREA	RESPONSABLE	TIEMPO ESTIMADO
1	Selección del servidor de correo.	Gabriel Villamagua	6 horas
2	Definición de la arquitectura (Definición de hilos)	Gabriel Villamagua	8 horas
3	Cargar dependencias (archivos jar) para el envío de correo usando JAVA.	Gabriel Villamagua	2 horas
4	Diseño de clase, implementación de métodos y constructores para manejo de hilos (Threads) en caso general	Gabriel Villamagua	8 horas
5	Pruebas unitarias para funcionamiento de hilos	Gabriel Villamagua	8 horas
6	Integración de correo electrónico con hilos (Threads)	Gabriel Villamagua	8 horas
7	Pruebas unitarias.	Gabriel Villamagua	8 horas
8	Control calidad	Ing. Bolívar Palán	4 horas
9	Afinamiento primera versión de la banca electrónica.	Gabriel Villamagua	4 horas
		TOTAL	56 horas

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.2.1. Elaboración y seguimiento

Para la gestión de este SPRINT se utilizará la tabla con el formato especificado anteriormente.

Tabla 2.6 Datos de configuración de la pila de iteración

PROYECTO			
Banca Electrónica LIX			
Nº DE SPRINT	INICIO	DÍAS	JORNADA
2	16-abril-2012	7	8
TAREAS		EQUIPOS	FESTIVOS
TIPOS	ESTADOS		
Análisis	Pendiente	Ing. Bolívar Palán	
Codificación	En curso	Gabriel Villamagua	
Prototipado	Terminada		
Pruebas	Eliminada		
Reunión			

Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

La tabla 2.7 muestra las tareas y el avance diario del segundo SPRINT de la banca electrónica.

Tabla 2.7 SPRINT Planning Meeting segunda iteración

FECHA	TAREAS PENDIENTES	HORAS PENDIENTES	RESPONSABLE	ACTIVIDAD ANTERIOR	ACTIVIDAD ACTUAL	REQUERIMIENTOS EXTRAS
16/Abril/2012	9	56	Gabriel Villamagua	-----	Selección del servidor de correo	NA
16/Abril/2012	8	50	Gabriel Villamagua	Selección del servidor de correo	Definición de arquitectura (Definición de Hilos)	NA
17/Abril/2012	8	42	Gabriel Villamagua	Definición de arquitectura (Definición de Hilos)	Cargar dependencias (archivos jar) para el envío de correo usando JAVA	Haber descargado archivos jar desde los sitios oficiales de Oracle
18/Abril/2012	7	40	Gabriel Villamagua	Cargar dependencias (archivos jar)	Diseño de clase, implementación de métodos y	NA

					para el envío de correo usando JAVA	constructores para manejo de hilos (Threads) en caso general	
19/Abril/2012	6	32	Gabriel Villamagua	Diseño de clase, implementación de métodos y constructores para manejo de hilos en caso general	Pruebas unitarias para funcionamiento de hilos	NA	
20/Abril/2012	5	24	Gabriel Villamagua	Pruebas unitarias para funcionamiento de hilos	Integración de correo electrónico con hilos (threads)	NA	
23/Abril/2012	4	16	Gabriel Villamagua	Integración de correo electrónico con hilos (threads)	Pruebas unitarias	NA	
24/Abril/2012	3	12	Gabriel Villamagua	-----	Control de calidad	NA	

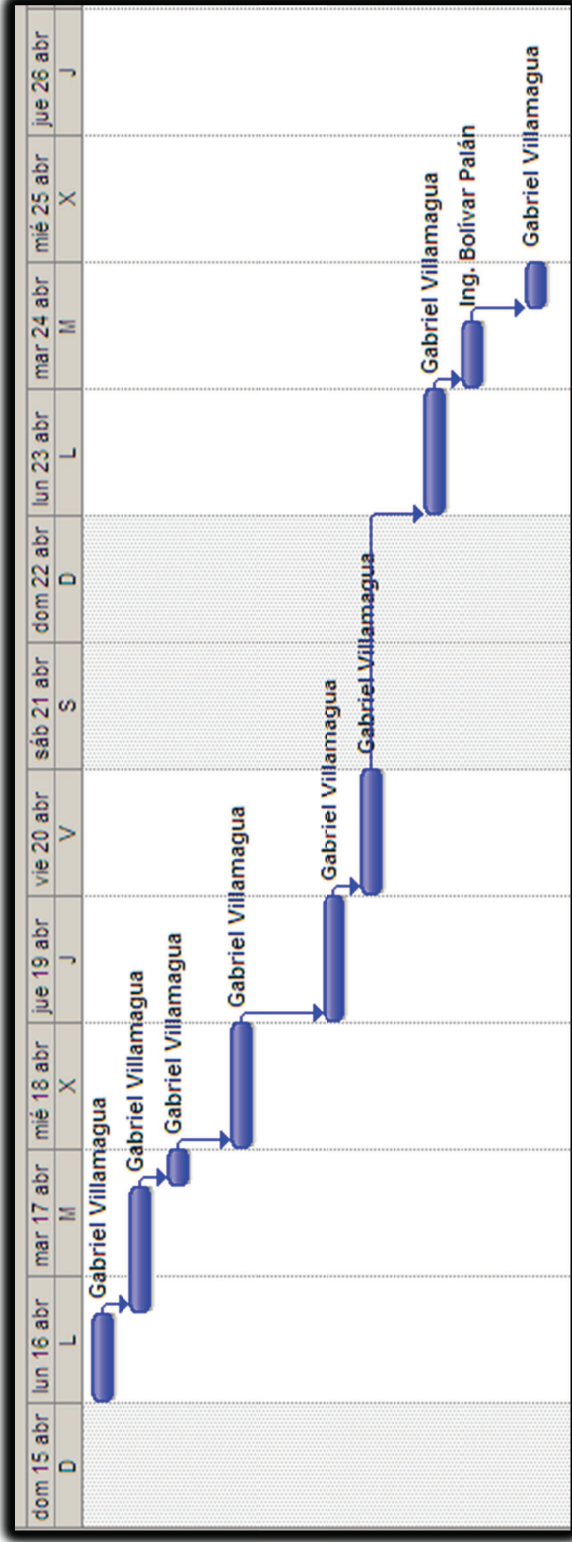
24/Abril/2012	2	8	Gabriel Villamagua	Pruebas unitarias	Afinamiento primera versión banca electrónica	NA
---------------	---	---	--------------------	-------------------	---	----

Elaborado por: Gabriel Ignacio Villamagua Armijos

La siguiente figura permite visualizar el esfuerzo de la pila del SPRINT, se especifica tareas, el estado y la persona del equipo responsable de la iteración.

Ilustración 2.6 Diagrama de Gantt de Segundo Sprint

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	Selección del servidor de correo.	6 horas	lun 16/04/12	lun 16/04/12		Gabriel Villamagua
2	Definición de la arquitectura (Definición de hilos)	8 horas	lun 16/04/12	mar 17/04/12	1	Gabriel Villamagua
3	Cargar dependencias (archivos jar) para el envío de correo usando JAVA.	2 horas	mar 17/04/12	mar 17/04/12	2	Gabriel Villamagua
4	Diseño de clase, implementación de métodos y constructores para manejo de hilos (Threads) en caso general	8 horas	mié 18/04/12	mié 18/04/12	3	Gabriel Villamagua
5	Pruebas unitarias para funcionamiento de hilos	8 horas	jue 19/04/12	jue 19/04/12	4	Gabriel Villamagua
6	Integración de correo electrónico con hilos (Threads)	8 horas	vie 20/04/12	vie 20/04/12	5	Gabriel Villamagua
7	Pruebas unitarias.	8 horas	lun 23/04/12	lun 23/04/12	6	Gabriel Villamagua
8	Control calidad	4 horas	mar 24/04/12	mar 24/04/12	7	Ing. Bolívar Palán
9	Afinamiento primera versión de la banca electrónica.	4 horas	mar 24/04/12	mar 24/04/12	8	Gabriel Villamagua



Elaborado por: Gabriel Ignacio Villamagua Armijos

Ilustración 2.7 Esfuerzo realizado en el segundo SPRINT

SPRINT		INICIO	DURACIÓN								
2	16-abr-12	7									
		Tareas pendientes			16-abr	17-abr	18-abr	19-abr	20-abr	23-abr	24-abr
		Horas de trabajo pendientes			9	8	6	5	4	3	2
					8	8	8	8	8	8	8
PILA DEL SPRINT											
Tarea	Tipo	Estado	Responsal	ESFUERZO							
Selección del servidor de correo	Análisis	Terminada	Gabriel Villamagua	6							
Definición de arquitectura	Prototipado	Terminada	Gabriel Villamagua	2	6						
Cargar dependencias para el envío de correo usando JAVA	Codificación	Terminada	Gabriel Villamagua		2						
Diseño de clase, implementación de métodos y constructores para	Prototipado	Terminada	Gabriel Villamagua			8					
Pruebas unitarias para funcionamiento de hilos	Pruebas	Terminada	Gabriel Villamagua				8				
Integración de correo electrónico con hilos	Codificación	Terminada	Gabriel Villamagua					8			
Pruebas unitarias	Pruebas	Terminada	Gabriel Villamagua						8		
Control de calidad	Pruebas	Terminada	Ing. Bolívar Palán							4	
Afinamiento primera versión banca electrónica	Codificación	Terminada	Gabriel Villamagua								4

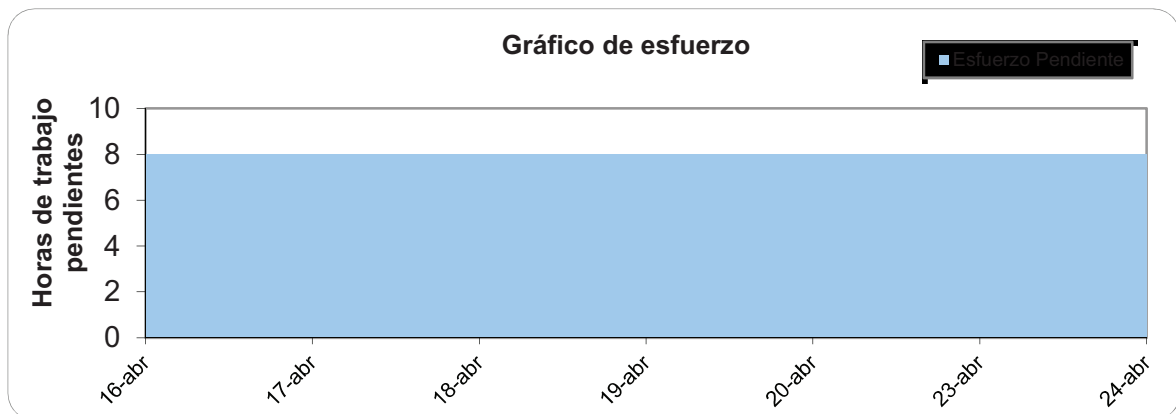
Fuente: Plantilla tomada de www.navegapolis.net
 Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.2.2. Gráfico Burn Down Chart

Las Ilustraciones 2.8 y 2.9 muestran el esfuerzo realizado y el avance de tareas respectivamente, durante el inicio y finalización del SPRINT.

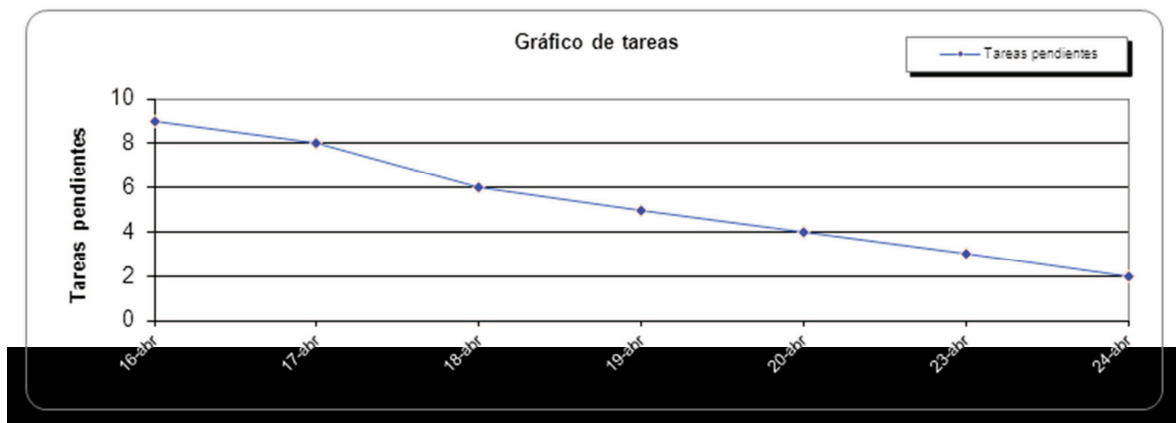
Ilustración 2.8 Esfuerzo realizado del segundo SPRINT

Proyecto	SPRINT	INICIO	DÍAS
Banca Electrónica LIX	2	16-abr-12	7



Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

Ilustración 2.9 Avance de tareas durante el segundo SPRINT



Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.3. Tercer sprint

Este SPRINT comprende las tareas 22 a la 30 de la pila del producto especificada en la tabla 2.1, empezará el 25 de abril del 2012 y tiene una duración de 6 días y la fecha de la revisión será el 3 de Mayo del 2012.

El objetivo del SPRINT es desarrollar el módulo encargado de realizar el envío de mensaje de texto (sms) a los usuarios de la banca electrónica.

En la tabla 2.8 se detalla las tareas a realizar en el SPRINT, esta tabla tiene el formato que anteriormente ya se especificó.

Tabla 2.8 SPRINT tercero Banca Electrónica LIX

N°	TAREA	RESPONSABLE	TIEMPO ESTIMADO
1	Definición de la infraestructura necesaria para el envío de sms.	Gabriel Villamagua	8 horas
2	Instalación de gammu para gestión de dispositivo móvil (teléfono móvil).	Gabriel Villamagua	2 horas
3	Selección y configuración del dispositivo móvil.	Gabriel Villamagua	8 horas
4	Pruebas de envío de mensajes de texto mediante la terminal (GNU/Linux).	Gabriel Villamagua	2 horas
5	Diseño de clase, implementación de métodos y constructores para comunicación con gammu desde JAVA.	Gabriel Villamagua	8 horas
6	Integración de hilos en la clase de comunicación con gammu.	Gabriel Villamagua	4 horas

7	Pruebas unitarias.	Gabriel Villamagua	8 horas
8	Control de calidad	Ing. Bolívar Palán	4 horas
9	Afinamiento de segunda versión de la banca electrónica.	Gabriel Villamagua	4 horas
		TOTAL	48 horas

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.3.1. *Elaboración y seguimiento*

Para la gestión de este SPRINT se utilizará la tabla con el formato especificado anteriormente.

Tabla 2.9 Datos de configuración de la pila de iteración

PROYECTO			
Banca Electrónica LIX			
Nº DE SPRINT	INICIO	DÍAS	JORNADA
3	25-abril-2012	6	8
TAREAS		EQUIPOS	FESTIVOS
TIPOS	ESTADOS		
Análisis	Pendiente	Ing. Bolívar Palán	
Codificación	En curso	Gabriel Villamagua	
Prototipado	Terminada		
Pruebas	Eliminada		
Reunión			

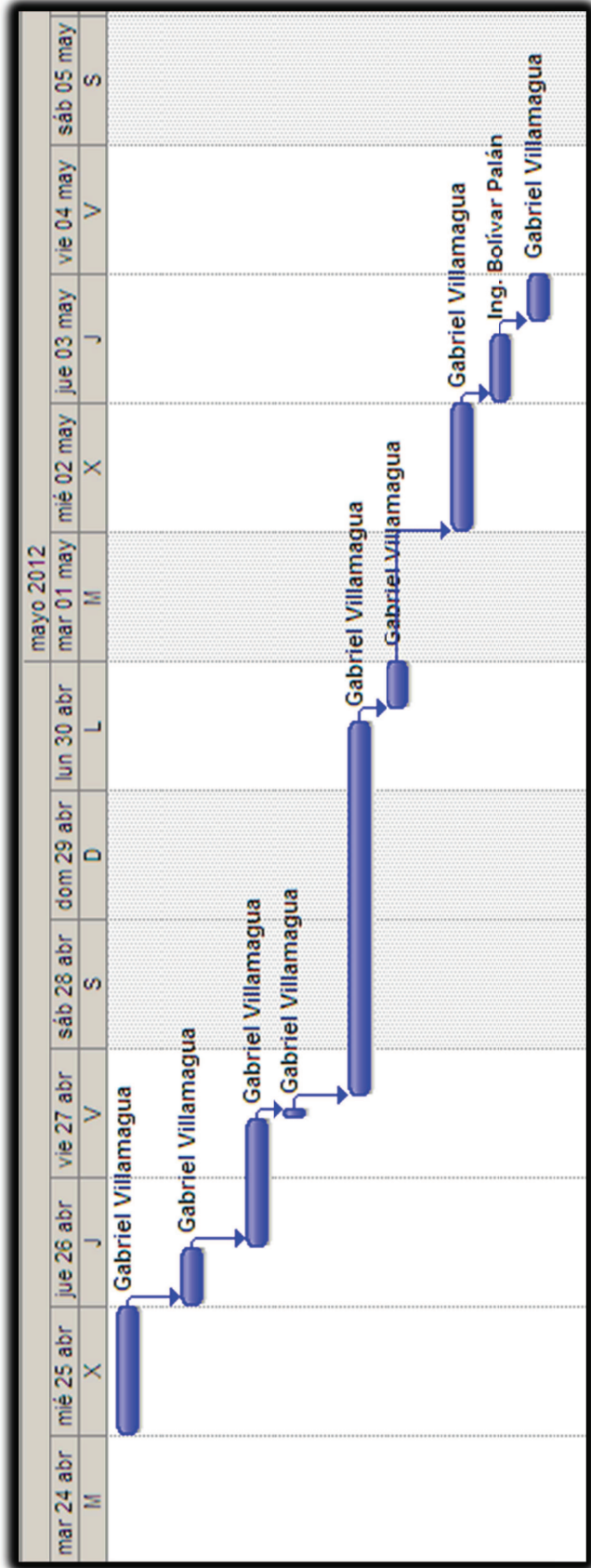
Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

La tabla 2.10 muestra las tareas y el avance diario del tercer SPRINT de la banca electrónica.

Tabla 2.10 SPRINT Planning Meeting tercera iteración

FECHA	TAREAS PENDIENTES	HORAS PENDIENTES	RESPONSABLE	ACTIVIDAD ANTERIOR	ACTIVIDAD ACTUAL	REQUERIMIENTOS EXTRAS
25Abril/2012	9	48	Gabriel Villamagua	-----	Definición de la infraestructura necesaria para el envío de sms.	NA
26/Abril/2012	8	40	Gabriel Villamagua	Definición de la infraestructura necesaria para el envío de sms.	Instalación de gammu para gestión de dispositivo móvil (teléfono móvil).	NA
26/Abril/2012	7	38	Gabriel Villamagua	Instalación de gammu para gestión de dispositivo móvil (teléfono móvil).	Selección y configuración del dispositivo móvil.	Descargado archivos jar desde los sitios oficiales de Oracle
27/Abril/2012	7	30	Gabriel Villamagua	Selección y configuración del dispositivo móvil.	Pruebas de envío de mensajes de texto mediante la terminal	NA

						(GNU/Linux).	
27/Abril/2012	6	28	Gabriel Villamagua	Pruebas de envío de mensajes de texto mediante la terminal (GNU/Linux).	Diseño de clase, implementación de métodos y constructores para comunicación con gammu desde JAVA.	NA	
30/Abril/2012	5	20	Gabriel Villamagua	Diseño de clase, implementación de métodos y constructores para comunicación con gammu desde JAVA.	Integración de hilos en la clase de comunicación con gammu.	NA	
02/Mayo/2012	4	16	Gabriel Villamagua	Integración de hilos en la clase de comunicación con gammu.	Pruebas unitarias.	NA	
03/Mayo/2012	3	8	Gabriel Villamagua	-----	Control de calidad	NA	
03/Mayo/2012	2	4	Gabriel Villamagua	Pruebas unitarias	Afinamiento de segunda versión	NA	



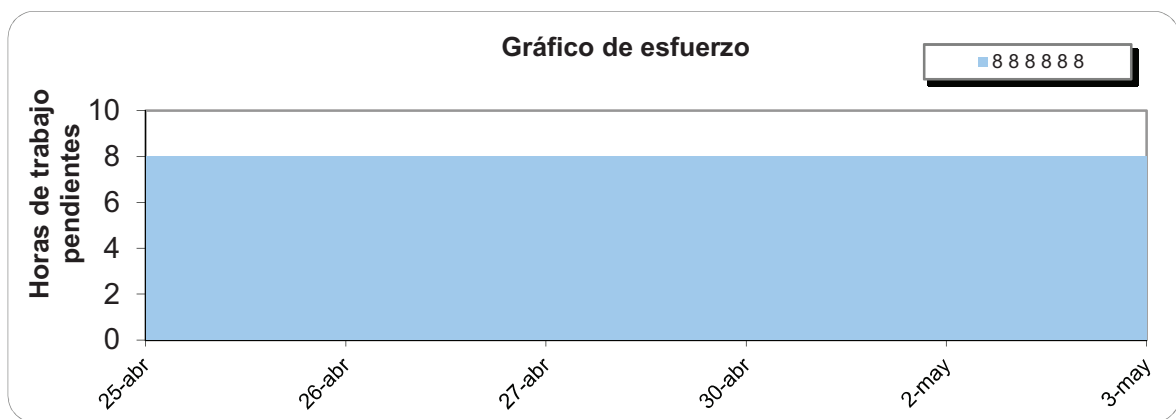
Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.4. Gráfico Burn Down Chart

Las ilustraciones 2.12 y 2.13 muestran el esfuerzo realizado y el avance de tareas respectivamente, durante el inicio y finalización del SPRINT.

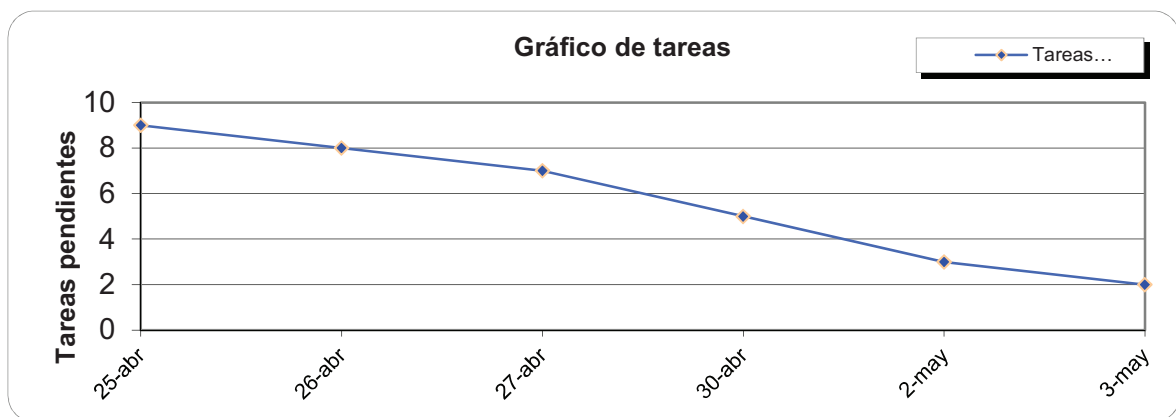
Ilustración 2.12 Esfuerzo realizado del tercer SPRINT

Proyecto	SPRINT	INICIO	DÍAS
Nombre del proyecto	3	25-abr-12	6



Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

Ilustración 2.13 Avance de tareas durante el tercer SPRINT



Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.5. Cuarto sprint

Este SPRINT comprende las tareas 31a la 52 de la pila del producto especificada en la tabla 2.1, empezará el 4 de mayo del 2012 y tiene una duración de 13 días y la fecha de la revisión será el 22 de Mayo del 2012. Las reuniones diarias serán de máximo 15 minutos a primera hora de la mañana cuando todo el equipo se encuentre presente.

El objetivo del SPRINT es desarrollar el módulo encargado de realizar el envío de mensaje de texto (sms) a los usuarios de la banca electrónica.

En la tabla 2.11 se detalla las tareas a realizar en el SPRINT.

Tabla 2.11 SPRINT cuarto Banca Electrónica LIX

N°	TAREA	RESPONSABLE	TIEMPO ESTIMADO
1	Diseño de JSP de selección de imagen y contestar pregunta.	Gabriel Villamagua	3 horas
2	Diseño de JSP de Inicio de sesión.	Gabriel Villamagua	2horas
3	Diseño de JSP de registro y actualización de datos personales	Gabriel Villamagua	3 horas
4	Diseño de JSP de registro y actualización de datos de autenticación.	Gabriel Villamagua	3 horas
5	Diseño de JSP re recuperación de contraseña.	Gabriel Villamagua	3 horas
6	Diseño de JSP principal de cliente.	Gabriel Villamagua	3 horas
7	Diseño de JSP principal de administrador.	Gabriel Villamagua	1 hora

8	Implementación de JSP y SERVLET de selección de imagen y pregunta.	Gabriel Villamagua	8 horas
9	Implementación de servicio de selección de imagen y pregunta.	Gabriel Villamagua	8 horas
10	Implementación de JSP y SERVLET de inicio de sesión.	Gabriel Villamagua	8 horas
11	Implementación de servicio de inicio de sesión.	Gabriel Villamagua	4 horas
12	Implementación de JSP y SERVLET de registro y actualización de datos personales.	Gabriel Villamagua	8 horas
13	Implementación de servicio de registro y actualización de datos personales	Gabriel Villamagua	3 horas
14	Implementación de JSP y SERVLET de registro y actualización de datos de autenticación.	Gabriel Villamagua	8 horas
15	Implementación de servicio de registro y actualización de datos de autenticación.	Gabriel Villamagua	3 horas
16	Implementación de JSP y SERVLET de recuperación de contraseña.	Gabriel Villamagua	5 horas
17	Implementación de servicio de registro de recuperación de contraseña.	Gabriel Villamagua	1 horas
18	Implementación de JSP y SERVLET para transferencias de la pantalla principal de cliente.	Gabriel Villamagua	12 horas

19	Implementación de servicio para transferencias interbancarias en la pantalla de cliente.	Gabriel Villamagua	5 horas
20	Implementación de JSP y SERVLET para bloqueo y desbloqueo de clientes.	Gabriel Villamagua	3 horas
21	Implementación de servicio para bloqueo y desbloqueo de clientes.	Gabriel Villamagua	1 hora
22	Implementación de JSP y SERVLET para creación de nuevos usuarios.	Gabriel Villamagua	4 horas
23	Implementación de servicio de creación de nuevo usuario.	Gabriel Villamagua	4 horas
24	Pruebas unitarias.	Gabriel Villamagua	5 horas.
25	Control de calidad.	Ing. Bolívar Palán	6 horas
26	Afinamiento de cuarta versión de la banca electrónica.	Gabriel Villamagua	4 horas
		TOTAL	118 horas

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.5.1. *Elaboración y seguimiento*

Para la gestión de este SPRINT se utilizará la tabla con el formato especificado anteriormente.

Tabla 2.12 Datos de configuración de la pila de iteración

PROYECTO			
Banca Electrónica LIX			
Nº DE SPRINT	INICIO	DÍAS	JORNADA
4	04-mayo-2012	13	8

TAREAS		EQUIPOS	FESTIVOS
TIPOS	ESTADOS		
Análisis	Pendiente	Ing. Bolívar Palán	
Codificación	En curso	Gabriel Villamagua	
Prototipado	Terminada		
Pruebas	Eliminada		
Reunión			

Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

La tabla 2.13 muestra las tareas y el avance diario del tercer SPRINT de la banca electrónica.

Tabla 2.13 SPRINT Planning Meeting de cuarta iteración

FECHA	TAREAS PENDIENTES	HORAS PENDIENTES	RESPONSABLE	ACTIVIDAD ANTERIOR	ACTIVIDAD ACTUAL	REQUERIMIENTOS EXTRAS
04/Mayo/2012	26	118	Gabriel Villamagua	-----	Diseño de JSP de selección de imagen y contestar pregunta.	NA
04/Mayo/2012	25	115	Gabriel Villamagua	Diseño de JSP de selección de imagen y contestar pregunta.	Diseño de JSP de Inicio de sesión.	NA
04/Mayo/2012	24	113	Gabriel Villamagua	Diseño de JSP de Inicio de sesión.	Diseño de JSP de registro y actualización de datos personales	NA
07/Mayo/2012	23	110	Gabriel Villamagua	Diseño de JSP de registro y actualización de datos personales	Diseño de JSP de registro y actualización de datos	NA

							autenticación.	
07/Mayo/2012	22	107	Gabriel Villamagua	Diseño de JSP de registro y actualización de datos de autenticación.	Diseño de JSP de recuperación de contraseña.	NA		
07/Mayo/2012	21	104	Gabriel Villamagua	Diseño de JSP de recuperación de contraseña.	Diseño de JSP principal de cliente.	NA		
08/Mayo/2012	21	101	Gabriel Villamagua	Diseño de JSP principal de cliente.	Diseño de JSP principal de administrador.	NA		
08/Mayo/2012	20	100	Gabriel Villamagua	Diseño de JSP principal de administrador.	Implementación de JSP y de SERVLET de selección de imagen y pregunta.	NA		
09/Mayo/2012	19	92	Gabriel Villamagua	Implementación de JSP y de SERVLET de selección de imagen y pregunta.	Implementación de servicio de selección de imagen y pregunta.	NA		

10/Mayo/2012	19	84	Gabriel Villamagua	Implementación de servicio de selección de imagen y pregunta.	Implementación de JSP y de SERVLET de inicio de sesión.			NA	
11/Mayo/2012	17	76	Gabriel Villamagua	Implementación de JSP y de SERVLET de inicio de sesión.	Implementación de servicio de inicio de sesión.			NA	
11/Mayo/2012	16	72	Gabriel Villamagua	Implementación de servicio de inicio de sesión.	Implementación de JSP y de SERVLET de registro y actualización de datos personales.			NA	
14/Mayo/2012	15	64	Gabriel Villamagua	Implementación de JSP y de SERVLET de registro y actualización de datos personales.	Implementación de servicio de registro y actualización de datos personales			NA	
15/Mayo/2012	14	61	Gabriel Villamagua	Implementación de servicio de	Implementación de JSP y			NA	

				registro y actualización de datos personales	SERVLET de registro y actualización de datos de autenticación.	
16/Mayo/2012	13	53	Gabriel Villamagua	Implementación de JSP y de SERVLET de registro y actualización de datos de autenticación.	Implementación de servicio de registro y actualización de datos de autenticación.	NA
16/Mayo/2012	12	50	Gabriel Villamagua	Implementación de servicio de registro y actualización de datos de autenticación.	Implementación de JSP y de SERVLET de recuperación de contraseña.	NA
17/Mayo/2012	11	45	Gabriel Villamagua	Implementación de JSP y de SERVLET de recuperación de contraseña.	Implementación de servicio de registro de recuperación de contraseña.	NA
17/Mayo/2012	10	44	Gabriel Villamagua	Implementación	Implementación	NA

				de servicio de registro de recuperación de contraseña.	de JSP y de SERVLET para transferencias de la pantalla principal de cliente.	
18/Mayo/2012	8	32	Gabriel Villamagua	Implementación de JSP y SERVLET para transferencias de la pantalla principal de cliente.	Implementación de servicio para transferencias interbancarias en la pantalla de cliente.	NA
21/Mayo/2012	7	27	Gabriel Villamagua	Implementación de servicio para transferencias interbancarias en la pantalla de cliente.	Implementación de JSP y SERVLET para bloqueo y desbloqueo de clientes.	NA
21/Mayo/2012	6	24	Gabriel Villamagua	Implementación de JSP y SERVLET para bloqueo y desbloqueo de clientes.	Implementación de servicio para bloqueo y desbloqueo de clientes.	NA

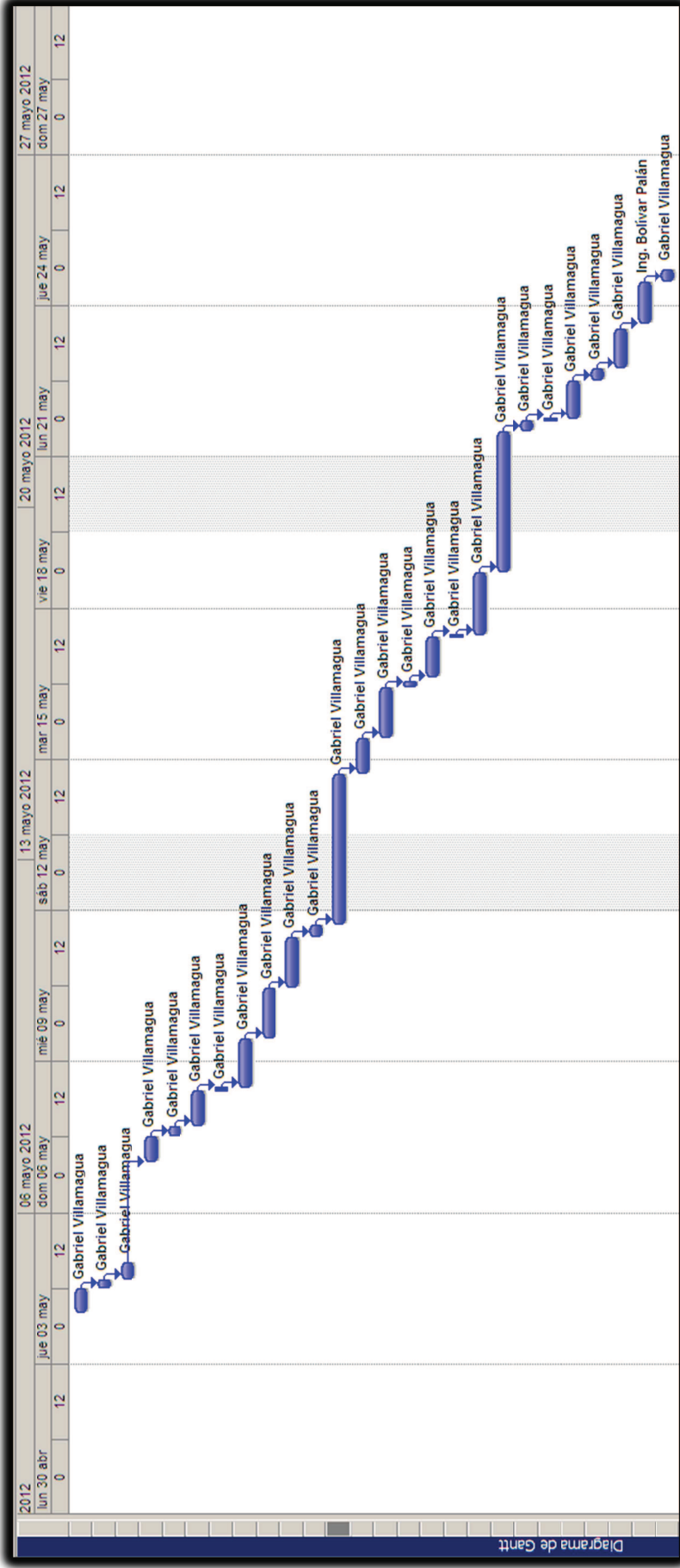
21/Mayo/2012	5	23	Gabriel Villamagua	Implementación de servicio para bloqueo y desbloqueo de clientes.	Implementación de JSP y SERVLET para creación de nuevos usuarios.	NA
22/Mayo/2012	4	19	Gabriel Villamagua	Implementación de JSP y SERVLET para creación de nuevos usuarios.	Implementación de servicio de creación de nuevo usuario.	NA
22/mayo/2012	3	15	Gabriel Villamagua	Implementación de servicio de creación de nuevo usuario.	Pruebas Unitarias	NA
23/Mayo/2012	2	10	Gabriel Villamagua	-----	Control de calidad.	NA
24/Mayo/2012	1	4	Gabriel Villamagua	Pruebas unitarias.	Afinamiento de tercera versión de la banca electrónica.	NA

Elaborado por: Gabriel Ignacio Villamagua Armijos

La siguiente figura permite visualizar el esfuerzo de la pila del SPRINT, se especifica tareas, el estado y la persona del equipo responsable de la iteración.

Ilustración 2.14 Diagrama de Gantt de Cuarto Sprint

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	Diseño de JSP de selección de imagen y contestar pregunta.	3 horas	vie 04/05/12	vie 04/05/12		Gabriel Villamagua
2	Diseño de JSP de inicio de sesión.	2 horas	vie 04/05/12	vie 04/05/12 1		Gabriel Villamagua
3	Diseño de JSP de registro y actualización de datos personales	3 horas	vie 04/05/12	vie 04/05/12 2		Gabriel Villamagua
4	Diseño de JSP de registro y actualización de datos de autenticación.	3 horas	lun 07/05/12	lun 07/05/12 3		Gabriel Villamagua
5	Diseño de JSP de recuperación de contraseña.	3 horas	lun 07/05/12	lun 07/05/12 4		Gabriel Villamagua
6	Diseño de JSP principal de cliente.	3 horas	lun 07/05/12	mar 08/05/12 5		Gabriel Villamagua
7	Diseño de JSP principal de administrador.	1 hora	mar 08/05/12	mar 08/05/12 6		Gabriel Villamagua
8	Implementación de JSP y SERVLET de selección de imagen y pregunta.	8 horas	mar 08/05/12	mié 09/05/12 7		Gabriel Villamagua
9	Implementación de servicio de selección de imagen y pregunta.	8 horas	mié 09/05/12	jue 10/05/12 8		Gabriel Villamagua
10	Implementación de JSP y SERVLET de inicio de sesión.	8 horas	jue 10/05/12	vie 11/05/12 9		Gabriel Villamagua
11	Implementación de servicio de inicio de sesión.	4 horas	vie 11/05/12	vie 11/05/12 10		Gabriel Villamagua
12	Implementación de JSP y SERVLET de registro y actualización de datos personales.	8 horas	vie 11/05/12	lun 14/05/12 11		Gabriel Villamagua
13	Implementación de servicio de registro y actualización de datos personales	3 horas	lun 14/05/12	mar 15/05/12 12		Gabriel Villamagua
14	Implementación de JSP y SERVLET de registro y actualización de datos de autenticación.	8 horas	mar 15/05/12	mié 16/05/12 13		Gabriel Villamagua
15	Implementación de servicio de registro y actualización de datos de autenticación.	3 horas	mié 16/05/12	mié 16/05/12 14		Gabriel Villamagua
16	Implementación de JSP y SERVLET de recuperación de contraseña.	5 horas	mié 16/05/12	jue 17/05/12 15		Gabriel Villamagua
17	Implementación de servicio de registro de recuperación de contraseña.	1 hora	jue 17/05/12	jue 17/05/12 16		Gabriel Villamagua
18	Implementación de JSP y SERVLET para transferencias de la pantalla principal de cliente.	12 horas	jue 17/05/12	vie 18/05/12 17		Gabriel Villamagua
19	Implementación de servicio para transferencias interbancarias en la pantalla de cliente.	5 horas	vie 18/05/12	lun 21/05/12 18		Gabriel Villamagua
20	Implementación de JSP y SERVLET para bloqueo y desbloqueo de clientes.	3 horas	lun 21/05/12	lun 21/05/12 19		Gabriel Villamagua
21	Implementación de servicio para bloqueo y desbloqueo de clientes.	1 hora	lun 21/05/12	lun 21/05/12 20		Gabriel Villamagua
22	Implementación de JSP y SERVLET para creación de nuevos usuarios.	4 horas	lun 21/05/12	mar 22/05/12 21		Gabriel Villamagua
23	Implementación de servicio de creación de nuevo usuario.	4 horas	mar 22/05/12	mar 22/05/12 22		Gabriel Villamagua
24	Pruebas unitarias.	5 horas	mar 22/05/12	mié 23/05/12 23		Gabriel Villamagua
25	Control de calidad.	6 horas	mié 23/05/12	jue 24/05/12 24		Ing. Boívar Palán
26	A finamiento de cuarta versión de la banca electrónica.	4 horas	jue 24/05/12	jue 24/05/12 25		Gabriel Villamagua



Elaborado por: Gabriel Ignacio Villamagua Armijos

Ilustración 2.15 Esfuerzo realizado en el cuarto SPRINT

SPRINT	INICIO	DURACIÓN	PILA DEL SPRINT							ESFUERZO												
4	4-may-12	15	Tarea	Tipo	Estado	Responsable	V	L	M	X	J	V	L	M	X	J	V	L	M	X	J	
			Diseño de JSP de selección de imagen y contestar pregunta.	Análisis	Terminada	Gabriel Villamagua																
			Diseño de JSP de inicio de sesión.	Análisis	Terminada	Gabriel Villamagua																
			Diseño de JSP de registro y actualización de datos personales	Análisis	Terminada	Gabriel Villamagua																
			Diseño de JSP de registro y actualización de datos de autenticación.	Análisis	Terminada	Gabriel Villamagua																
			Diseño de JSP de recuperación de contraseña.	Análisis	Terminada	Gabriel Villamagua																
			Diseño de JSP principal de cliente.	Análisis	Terminada	Gabriel Villamagua	2	1														
			Diseño de JSP principal de administrador.	Análisis	Terminada	Gabriel Villamagua	1															
			Implementación de JSP y SERVLET de selección de imagen y pregunta.	Codificación	Terminada	Ing. Bolívar Palán	6	2														
			Implementación de servicio de selección de imagen y pregunta.	Codificación	Terminada	Gabriel Villamagua	6	2														
			Implementación de JSP y SERVLET de inicio de sesión.	Codificación	Terminada	Gabriel Villamagua	6	2														
			Implementación de servicio de inicio de sesión.	Codificación	Terminada	Gabriel Villamagua		4														
			Implementación de JSP y SERVLET de registro y actualización de datos personales.	Codificación	Terminada	Gabriel Villamagua	2	6														
			Implementación de servicio de registro y actualización de datos personales	Codificación	Terminada	Gabriel Villamagua	2	1														
			Implementación de JSP y SERVLET de registro y actualización de datos de autenticación.	Codificación	Terminada	Gabriel Villamagua	7	1														
			Implementación de servicio de registro y actualización de datos de autenticación.	Codificación	Terminada	Gabriel Villamagua		3														
			Implementación de JSP y SERVLET de recuperación de contraseña.	Codificación	Terminada	Gabriel Villamagua		4	1													
			Implementación de servicio de recuperación de contraseña.	Codificación	Terminada	Gabriel Villamagua		1														
			Implementación de JSP y SERVLET para transferencias de la pantalla principal del cliente.	Codificación	Terminada	Gabriel Villamagua		6	6													
			Implementación de servicio para transferencias directas en la pantalla del cliente	Codificación	Terminada	Gabriel Villamagua		2	3													
			Implementación de JSP y SERVLET para bloqueo y desbloqueo de clientes	Codificación	Terminada	Gabriel Villamagua		3														
			Implementación de JSP y SERVLET para creación de nuevos usuarios.	Codificación	Terminada	Gabriel Villamagua		2	2													
			Implementación de servicio de creación de nuevo usuario.	Codificación	Terminada	Gabriel Villamagua		4														
			Implementación de servicio para bloqueo y desbloqueo de clientes	Codificación	Terminada	Gabriel Villamagua		1	1													
			Pruebas unitarias	Pruebas	Terminada	Gabriel Villamagua		1	4													
			Control de calidad	Pruebas	Terminada	Ing. Bolívar Palán		4	2													
			Afinamiento de tercera versión de la banca electrónica	Codificación	Terminada	Gabriel Villamagua		4														

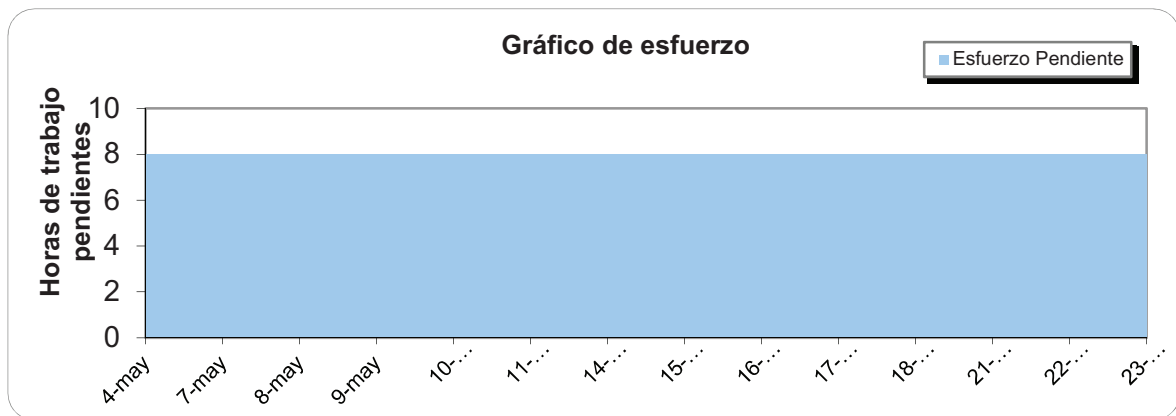
Fuente: Plantilla tomada de www.navegapolis.net
 Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.5.2. Gráfico Burn Down Chart

Las ilustraciones 2.16 y 2.17 muestran el esfuerzo realizado y el avance de tareas respectivamente, durante el inicio y finalización del SPRINT.

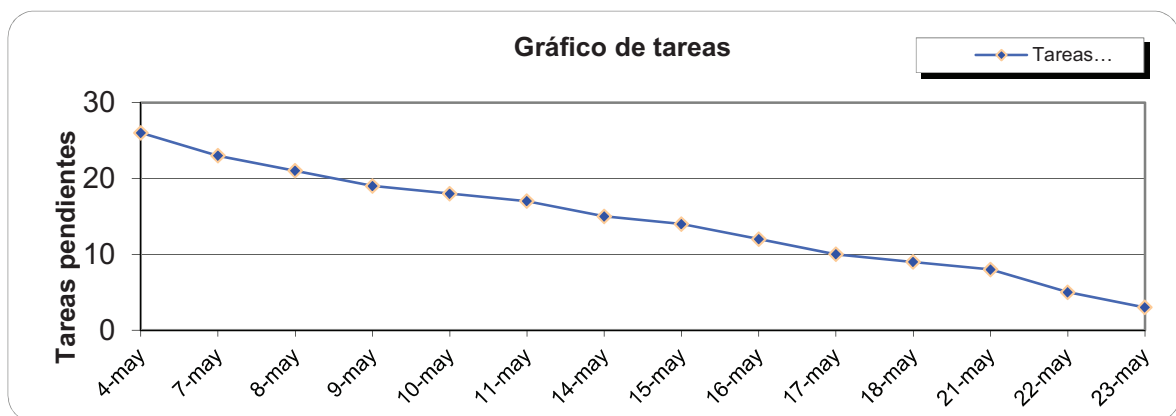
Ilustración 2.16 Esfuerzo realizado del cuarto SPRINT

Proyecto	SPRINT	INICIO	DÍAS
Nombre del proyecto	4	4-may-12	15



Fuente: Plantilla tomada de www.navegapolis.net
 Elaborado por: Gabriel Ignacio Villamagua Armijos

Ilustración 2.17 Avance de tareas durante el cuarto SPRINT



Fuente: Plantilla tomada de www.navegapolis.net
 Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.6. Quinto sprint

Este SPRINT comprende las tareas 53 a la 60 de la pila del producto especificada en la tabla 2.1, empezará el 24 de mayo del 2012 y tiene una duración de 5 días y la fecha de la revisión será el 31 de Mayo del 2012.

Los objetivos de este SPRINT son:

- Establecer un patrón visual para todas las interfaces (JSP's) de la banca virtual.
- Realizar las validaciones de la información previamente a realizar cualquier cambio de estado del sistema.

En la tabla 2.14 se detalla las tareas a realizar en el SPRINT.

Tabla 2.14 SPRINT quinto Banca Electrónica LIX
Realizado por: Gabriel Villamagua

Nº	TAREA	RESPONSABLE	TIEMPO ESTIMADO
1	Implementación de patrón visual de tablas (color, bordes, fuente de las celdas, etc.).	Gabriel Villamagua	4 horas
2	Implementación de patrón visual de títulos, leyendas, advertencias y links.	Gabriel Villamagua	4.5 horas
3	Implementación de patrón visual de fondos de pantalla, fuente de letra y menús.	Gabriel Villamagua	4.5 horas
4	Implementación de validación de ingreso de información en formularios.	Gabriel Villamagua	12 horas
5	Prueba final de la banca electrónica.	Gabriel Villamagua	15 horas

6	Control de calidad	Ing. Bolívar Palán	15 horas
7	Afinamiento de reléase final.	Gabriel Villamagua	15 horas
		TOTAL	70 horas

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.6.1. *Elaboración y seguimiento*

Para la gestión de este SPRINT se utilizará la tabla con el formato especificado anteriormente.

Tabla 2.15 Datos de configuración de la pila de iteración

PROYECTO			
Banca Electrónica LIX			
Nº DE SPRINT	INICIO	DÍAS	JORNADA
5	28-mayo-2012	14	8
TAREAS		EQUIPOS	FESTIVOS
TIPOS	ESTADOS		
Análisis	Pendiente	Ing. Bolívar Palán	
Codificación	En curso	Gabriel Villamagua	
Prototipado	Terminada		
Pruebas	Eliminada		
Reunión			

Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

La tabla 2.16 muestra las tareas y el avance diario del tercer SPRINT de la banca electrónica.

Tabla 2.16 SPRINT Planning Meeting de quinta iteración

FECHA	TAREAS PENDIENTES	HORAS PENDIENTES	RESPONSABLE	ACTIVIDAD ANTERIOR	ACTIVIDAD ACTUAL	REQUERIMIENTOS EXTRAS
28/Mayo/2012	7	70	Gabriel Villamagua	-----	Implementación de patrón visual de tablas (color, bordes, fuente de las celdas, etc.).	NA
28/Mayo/2012	6	66	Gabriel Villamagua	Implementación de patrón visual de tablas (color, bordes, fuente de las celdas, etc.).	Implementación de patrón visual de títulos, leyendas, advertencias y links.	NA
29/Mayo/2012	4	61.5	Gabriel Villamagua	Implementación de patrón visual de títulos, leyendas, advertencias y links.	Implementación de patrón visual de fondos de pantalla, fuente de letra y menús.	NA

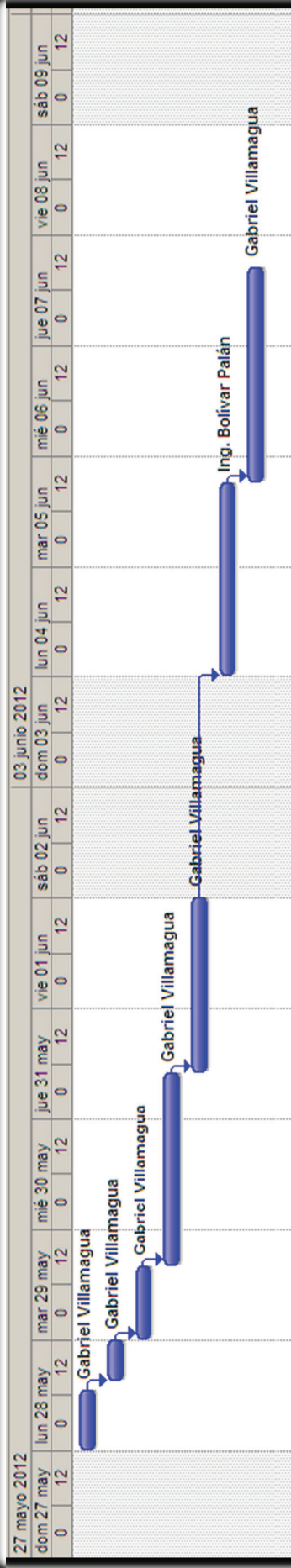
29/Mayo/2012	3	57	Gabriel Villamagua	Implementación de patrón visual de fondos de pantalla, fuente de letra y menús.	Implementación de validación de ingreso de información en formularios.	NA
31/Mayo/2012	2	45	Gabriel Villamagua	Implementación de validación de ingreso de información en formularios.	Prueba final de la banca electrónica.	NA
04/Junio/2012	2	30	Gabriel Villamagua	-----	Control de calidad	NA
07/Junio/2012	1	15	Gabriel Villamagua	Prueba final de la banca electrónica.	Afinamiento de release final.	NA

Elaborado por: Gabriel Ignacio Villamagua Armijos

La siguiente figura permite visualizar el esfuerzo de la pila del SPRINT, se especifica tareas, el estado y la persona del equipo responsable de la iteración.

Ilustración 2.18 Diagrama de Gantt de Quinto Sprint

ID	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	Implementación de patrón visual de tablas (color, bordes, fuente de las celdas, etc.).	4 horas	lun 28/05/12	lun 28/05/12		Gabriel Villamagua
2	Implementación de patrón visual de títulos, leyendas, advertencias y links.	4 horas	lun 28/05/12	lun 28/05/12	1	Gabriel Villamagua
3	Implementación de patrón visual de fondos de pantalla, fuente de letra y menús.	5 horas	mar 29/05/12	mar 29/05/12	2	Gabriel Villamagua
4	Implementación de validación de ingreso de información en formularios.	12 horas	mar 29/05/12	jue 31/05/12	3	Gabriel Villamagua
5	Prueba final de la banca electrónica.	15 horas	jue 31/05/12	vie 01/06/12	4	Gabriel Villamagua
6	Control de calidad	15 horas	lun 04/06/12	mar 05/06/12	5	Ing. Bolívar Palán
7	A finamiento de reléase final.	15 horas	mar 05/06/12	jue 07/06/12	6	Gabriel Villamagua



Elaborado por: Gabriel Ignacio Villamagua Armijos

Ilustración 2.19 Esfuerzo realizado en el quinto SPRINT

SPRINT	INICIO	DURACIÓN	ESFUERZO									
5	28-may-12	14	L	M	X	J	V	L	M	X	J	
			7	6	4	4	3	2	2	1	1	
		Tareas pendientes	8	8	8	8	8	8	8	8	7	
		Horas de trabajo pendientes										
PILA DEL SPRINT			Tarea	Tipo	Estado	Responsal						
			Implementación de patrón visual de tablas (color, bordes, fuente de las celdas, etc.).	Codificación	Terminada	Gabriel Villamagua	4					
			Implementación de patrón visual de títulos, leyendas, fuente de letra y menús.	Codificación	Terminada	Villamagua	4	0,5				
			Implementación de patrón visual de fondos de pantalla, fuente de letra y menús.	Codificación	Terminada	Gabriel Villamagua		4,5				
			Implementación de validación de ingreso de información en formularios.	Codificación	Terminada	Gabriel Villamagua	3	8	1			
			Prueba final de la banca electrónica.	Pruebas	Terminada	Villamagua		7	7			
			Control de calidad	Pruebas	Terminada	Villamagua		1	8	6		
			Afinamiento de reléase final.	Codificación	Terminada	Villamagua				2	8	7

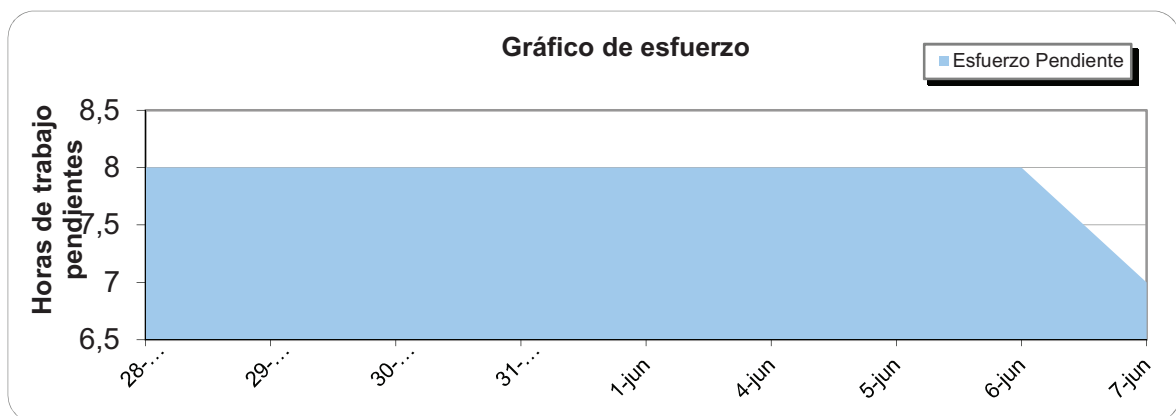
Fuente: Plantilla tomada de www.navegapolis.net
 Elaborado por: Gabriel Ignacio Villamagua Armijos

2.2.1.6.2. Gráfico Burn Down Chart

Las Ilustraciones 2.20 y 2.21 muestran el esfuerzo realizado y el avance de tareas respectivamente, durante el inicio y finalización del SPRINT.

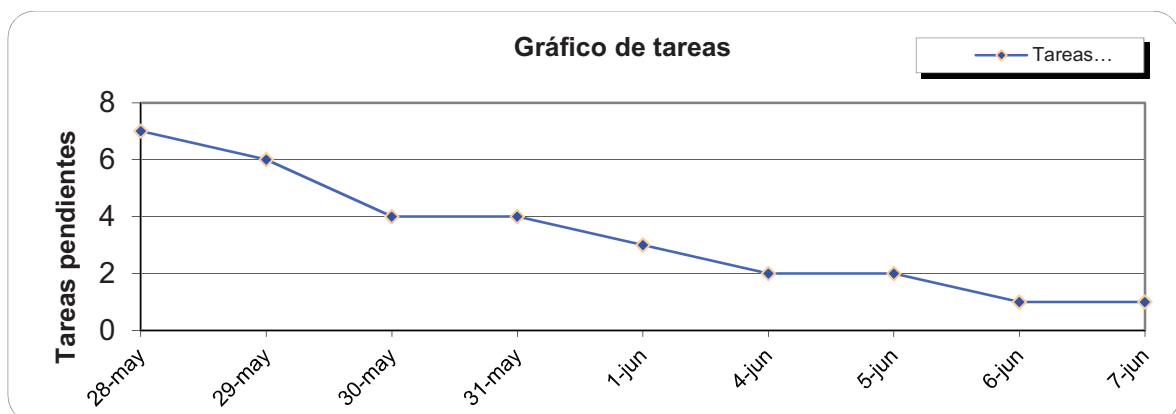
Ilustración 2.20 Esfuerzo realizado del quinto SPRINT

Proyecto	SPRINT	INICIO	DÍAS
Nombre del proyecto	5	28-may-12	14



Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

Ilustración 2.21 Avance de tareas durante el quinto SPRINT



Fuente: Plantilla tomada de www.navegapolis.net
Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3. DISEÑO DEL SISTEMA

A continuación procedemos a definir las interfaces que serán visualizadas por los usuarios de la banca electrónica.

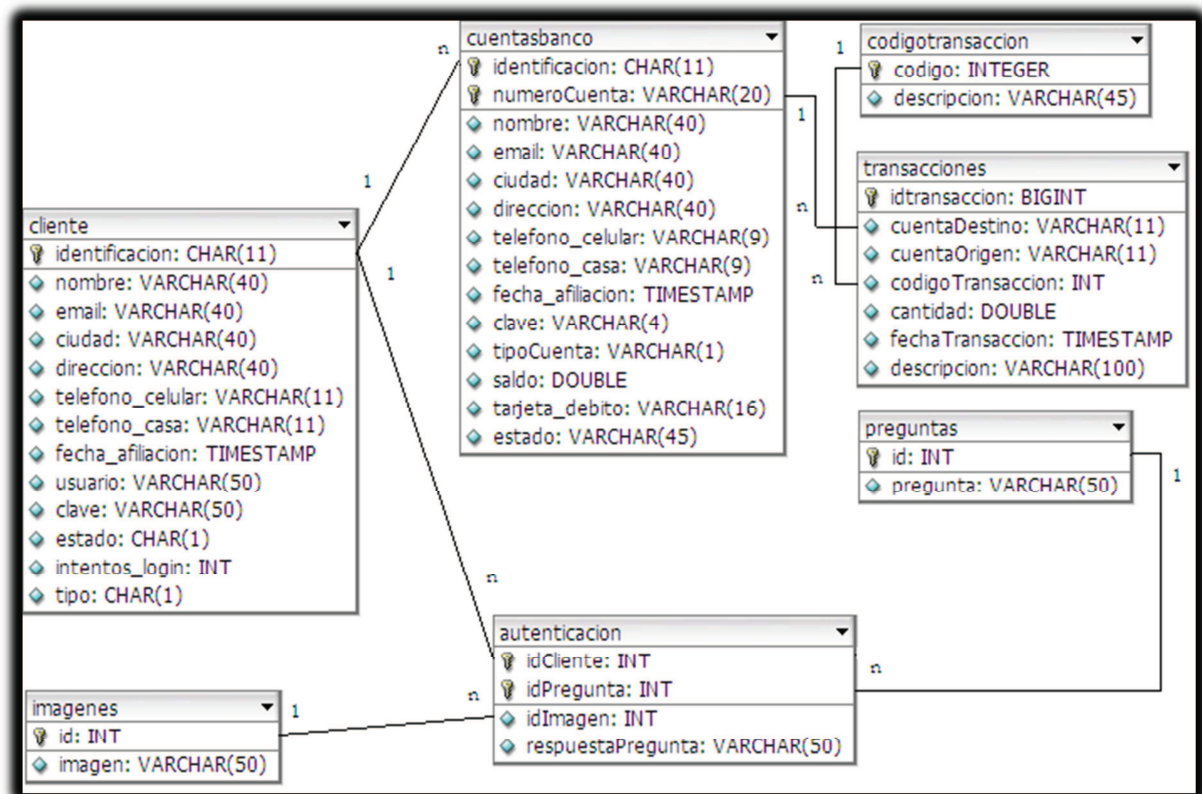
2.3.1. DISEÑO DE SPRINTS

Una vez realizado el análisis y planning de cada sprint ahora procedemos a realizar el diseño.

2.3.1.1. Primer Sprint

En este SPRINT procedemos a definir el diagrama entidad relación sobre el cual el sistema de la banca electrónica va a correr.

Ilustración 2.22 Diagrama E-R del Sistema



Elaborado por: Gabriel Ignacio Villamagua Armijos

Para definir la capa de datos se utilizará la siguiente tabla que tiene los siguientes campos:

Nombre: Nombre de la estructura.

Objetivo: Especifica el objetivo de la creación de la estructura.

Nombre Campo: especifica los nombres de los campos de las estructuras.

Tipo de Dato: especifica el tipo de dato de cada campo de las estructuras.

Longitud: especifica la longitud de cada campo de las estructuras.

Descripción: especifica breve descripción de cada campo de las estructuras.

2.3.1.1.1. *Diseño de estructura cliente*

Nombre: Cliente			
Objetivo: Permite persistir la información de los clientes de la banca electrónica.			
Nombre Campo	Tipo de Dato	Longitud	Descripción
identificacion	Caracteres numéricos	11	Cédula o pasaporte.
nombre	Texto		Nombre del cliente.
email	texto		Correo electrónico personal de cliente.
ciudad	texto		Ciudad de domicilio del cliente
direccion	Alfanumérico		Dirección de domicilio del cliente.
telefono_celular	Caracteres numéricos	10	Número de teléfono celular.
telefono_casa	Caracteres numéricos	9	Número de teléfono fijo.
fecha_afiliacion	Fecha con hora		Fecha y hora en que se registró a la banca electrónica el cliente.
usuario	Alfanumérico	>=12	Usuario seleccionado por el

			cliente.
clave	Alfanumérico	>=12	Clave seleccionada por el cliente.
estado	Caracter (A o B)	1	Estado activo o bloqueado del cliente en la banca electrónica.
intentos_login	Entero positivo		Número de intentos fallidos de autenticación del cliente en la banca electrónica.
tipo	Caracter (A o C)	1	Privilegios de administrador o cliente.

2.3.1.1.2. Diseño de estructura cuentasbanco

Nombre: cuentasbanco			
Objetivo: Permite almacenar todas las cuentas del banco.			
Nombre Campo	Tipo de Dato	Longitud	Descripción
identificacionCliente	Caracteres numéricos	11	Cedula o pasaporte.
numeroCuenta	Big int	20	Número de cuenta.
nombre	Caracteres numéricos	40	Nombre cliente del banco.
email	texto		Ciudad de domicilio del cliente.
Dirección	Alfanumérico		Dirección de domicilio del cliente.
telefono_casa	Caracteres numéricos	10	Número de teléfono fijo.
telefono_celular	Caracteres numéricos	9	Número de teléfono celular.
fecha_afiliacion	Fecha con hora		Fecha y hora en que se creó la cuenta en el banco.
Clave	Alfanumérico	4	Clave de la tarjeta de débito

tipoCuenta	Carácter (A o C)	1	Ahorros o corriente.
Saldo	Decimal	15 con 2 de precisión	Saldo disponible en la cuenta.
Tarjeta_debito	Caracteres numéricos	14	Número de tarjeta de débito.
estado	Caracter (A o B)	1	Bloqueado o activo.

2.3.1.1.3. *Diseño de estructura imágenes*

Nombre: imágenes			
Objetivo: Permite almacenar los nombres de las imágenes para el desafío y registro del cliente.			
Nombre Campo	Tipo de Dato	Longitud	Descripción
Id	entero		Id de imagen.
imagen	Alfanumérico	50	Nombre de la imagen.

2.3.1.1.4. *Estructura codigotransaccion*

Nombre: codigotransaccion			
Objetivo: Permite almacenar en los tipos de transacciones.			
Nombre Campo	Tipo de Dato	Longitud	Descripción
codigo	entero		Secuencial de transacción.
descripcion	Alfanumérico	45	Nombre de la transacción.

2.3.1.1.5. *Diseño de estructura preguntas*

Nombre: preguntas			
Objetivo: Permite almacenar las preguntas el desafío y registro del cliente.			
Nombre Campo	Tipo de Dato	Longitud	Descripción
Id	entero		Secuencial de pregunta.
pregunta	Alfanumérico	50	Pregunta.

2.3.1.1.6. *Diseño de estructura transacciones*

Nombre: transacciones			
Objetivo: Permite almacenar los movimientos realizados por un cliente en la banca electrónica.			
Nombre Campo	Tipo de Dato	Longitud	Descripción
IdTransaccion	entero		Secuencial de transacción.
cuentaDestino	Caracteres numéricos	11	Cuenta de destino de la transacción.
CuentaOrigen	Caracteres numéricos	11	Cuenta de débito en caso de ser transferencia.
codigoTransaccion	int		Id de transacción (referencia de estructura transacciones)
Cantidad	decimal	15 con precisión 2	Cantidad de dinero que se utiliza en la transacción.
fechaTransaccion	Fecha con hora		Fecha y hora en que se realizó la transacción.
descripcion	Alfanumérico	100	Razón de la transacción.

2.3.1.1.7. *Diseño de estructura autenticación*

Nombre: autenticación			
Objetivo: Permite almacenar las información de autenticación del cliente.			
Nombre Campo	Tipo de Dato	Longitud	Descripción
IdCliente	Caracteres numéricos.		Identificación Cliente (referencia estructura clientes).
Idpregunta	int		Id de pregunta (referencia en estructura preguntas).
idImagen	Int		Id de imagen (referencia en estructura imágenes).
respuestaPregunta	Alfanumérico	50	Respuesta de la pregunta seleccionada por el cliente.

2.3.1.1.8. *Diseño de procedimiento almacenado realizarTransferencia*

Nombre: realizar transferencias			
Objetivo: permite realizar transferencias entre cuentas propias.			
Parámetros que recibe			
Nombre	Tipo de Dato	Longitud	Descripción
cuentaOrigenTemp	Caracteres numéricos	12	Variable temporal para cuenta origen.
cuentaDestinoTemp	Caracteres numéricos	12	Variable temporal para cuenta destino.
montoTemp	Decimal	15 con precisión 2	Variable temporal para monto.
descripcionTemp	Alfanumérico	100	Variable temporal para razón de transferencia.

saldoCuentaDestino	Decimal	15 con precisión 2	Variable temporal para saldo en cuenta destino.
saldoCuentaOrigen	Decimal	15 con precisión 2	Variable temporal para saldo en cuenta origen.

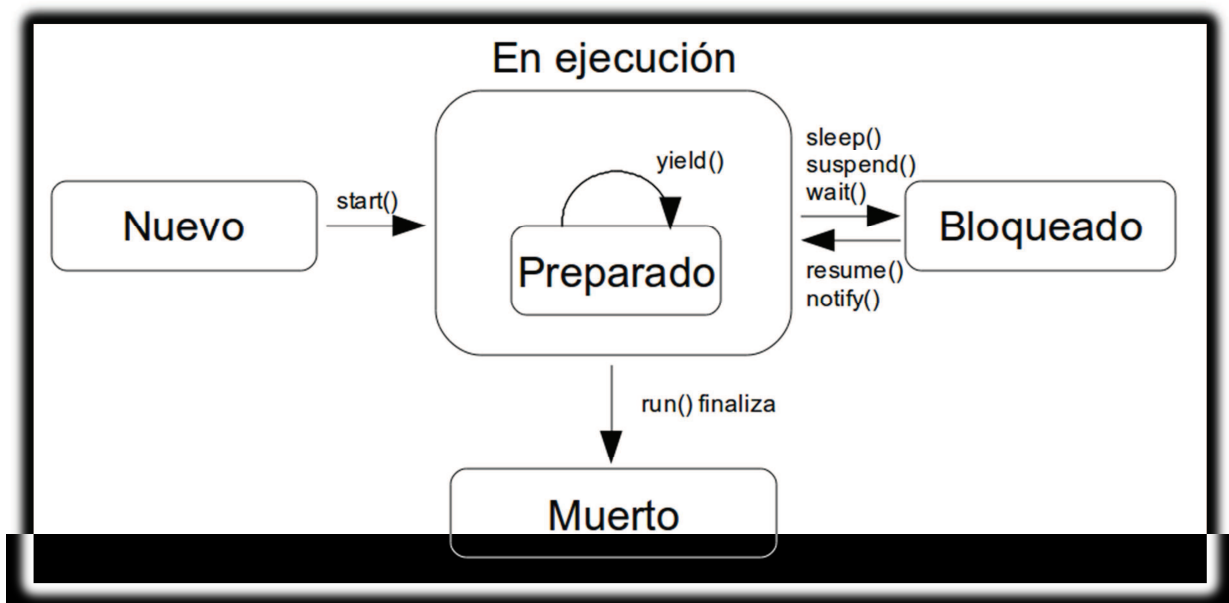
2.3.1.2. Segundo Sprint

En este SPRINT procedemos a definir las clases, métodos y constructores para implementar el servicio de correo electrónico.

Nuestro servidor de correo electrónico a utilizar es Google Mail (GMAIL), este servidor de correo es un servicio al público de la compañía Google.

Los procesos complementarios como el servicio de correo electrónico no debe afectar en el tiempo de respuesta del sistema, por lo que es necesario utilizar hilos (threads) para que realicen el envío de correo electrónico ya que un hilo consume menos recursos que un proceso.

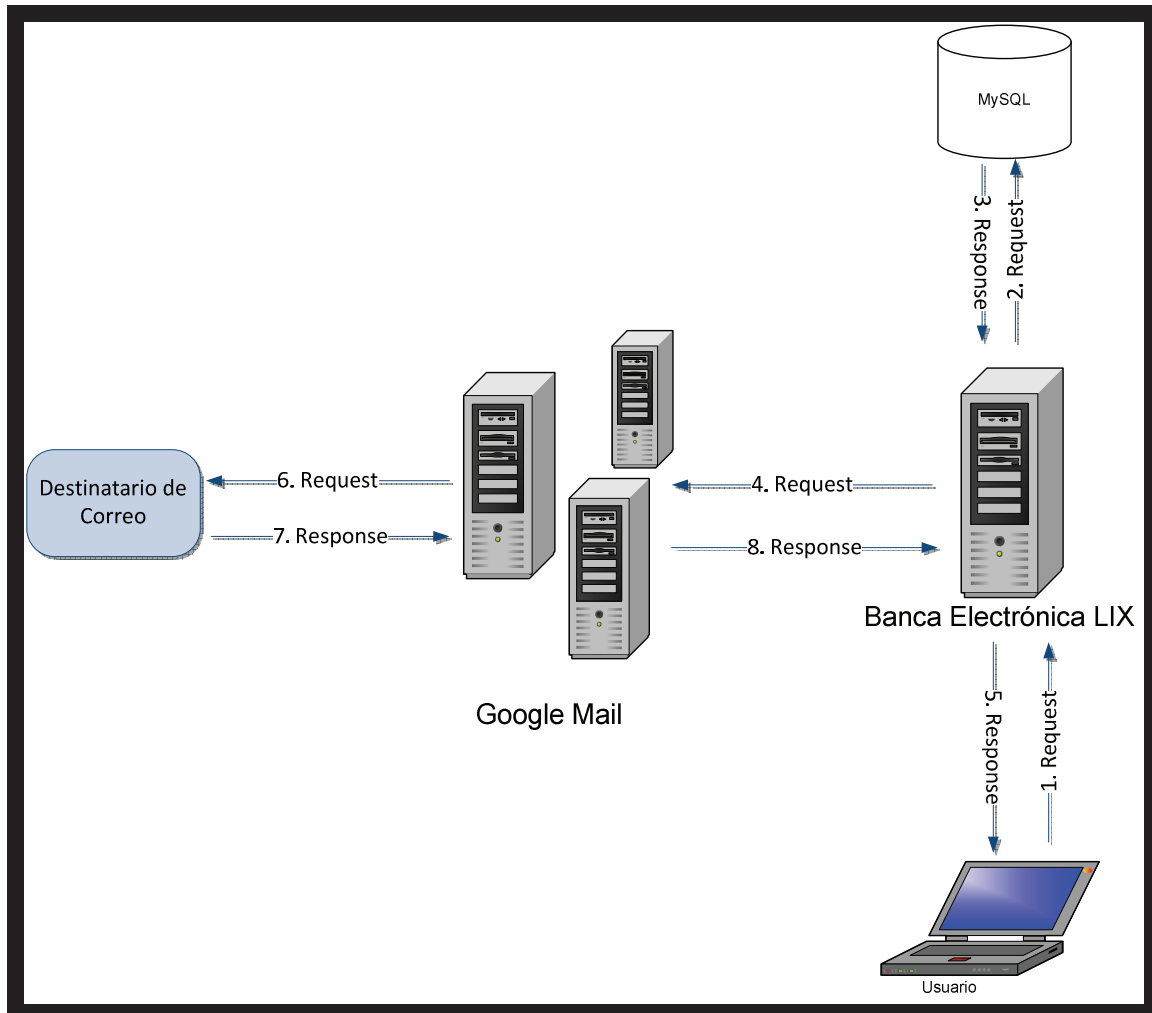
Ilustración 2.23 Gráfico de estados de un hilo en java



Fuente: <http://eaddfsi.wordpress.com/category/11-concurrencia-en-java/>

A continuación presentamos un gráfico que especifica de mejor manera cómo se va a realizar el envío de correo electrónico.

Ilustración 2.24 Proceso de envío de correo electrónico.



Elaborado por: Gabriel Ignacio Villamagua Armijos

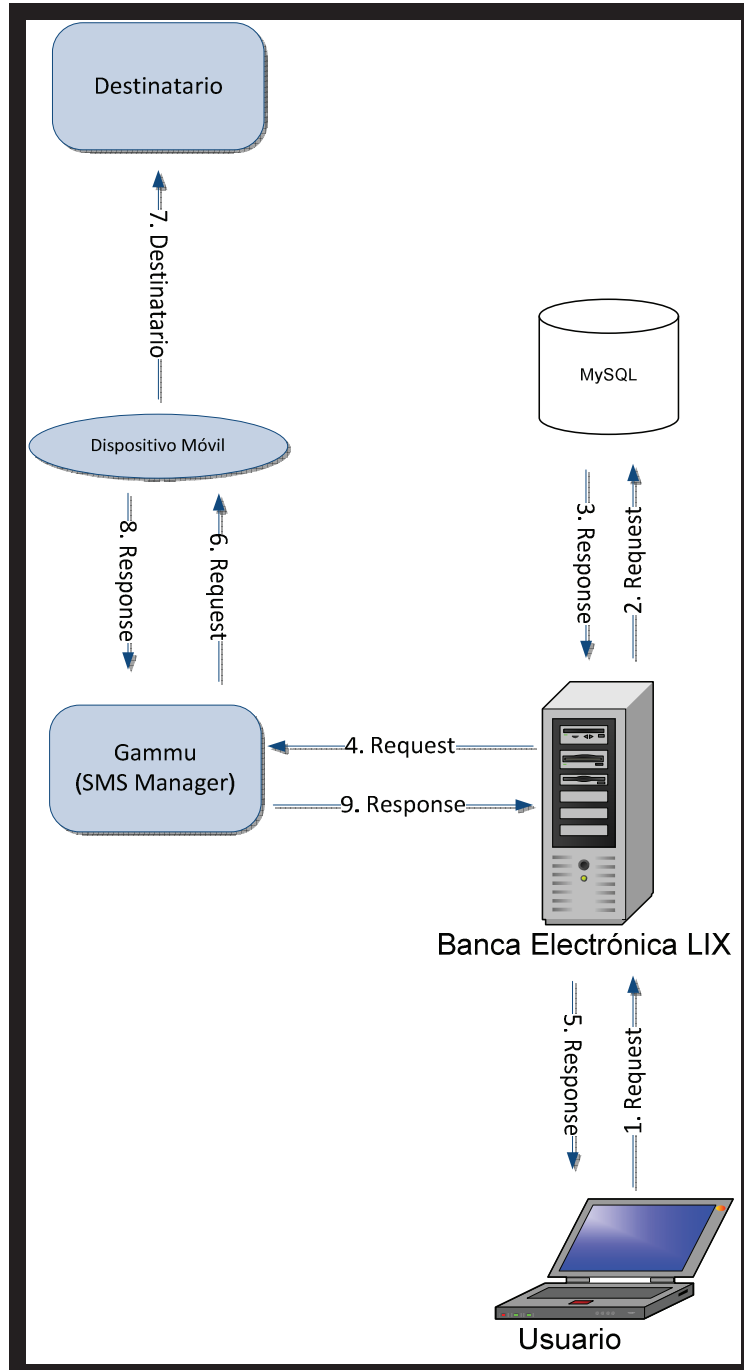
Cómo podemos visualizar en el gráfico una vez que accedemos a la banca electrónica, el sistema procede a enviar un mensaje al usuario inmediatamente sin esperar a que se realice el envío de correo electrónico ya que este proceso se lleva a cabo mediante un hilo (thread) sin que afecte a los procesos principales del negocio, incluso si el código que el hilo ejecuta llegase a fallar no se afectaría en el desempeño del sistema.

2.3.1.3. Tercer Sprint

Procedemos a definir el envío de mensajes de texto a los teléfonos móviles.

Para esto se va a utilizar un teléfono móvil Nokia modelo 3500c y una aplicación (Gammu) que nos ayudará a gestionar el teléfono móvil. Gammu recibirá una orden de terminal enviada desde java a la vez gestionada por un controlador (servlet).

Ilustración 2.25 Proceso de envío de SMS'



Elaborado por: Gabriel Ignacio Villamagua Armijos

Como podemos visualizar en el gráfico el sistema envía una respuesta al usuario sin que se haya finalizado el proceso de envío de SMS, el hilo (thread)

se encarga de realizar esta funcionalidad de manera paralela a los procesos principales del negocio.

2.3.1.4. Cuarto Sprint

En este SPRINT procedemos a diseñar las interfaces que utilizaremos en el sistema.

2.3.1.4.1. Interfaz de Registro

Nombre: Interfaz de registro																	
Objetivo: Permite al usuario cliente registrarse como usuario en la banca electrónica, para esto se debe ingresar el número de la tarjeta de débito y la contraseña.																	
<table border="1"> <thead> <tr> <th>Componente</th> <th>Descripción</th> </tr> </thead> <tbody> <tr> <td>Etiqueta (título)</td> <td>Permite establecer y visualizar el título de la banca electrónica.</td> </tr> <tr> <td>Cabecera HTML (h2)</td> <td>Permite establecer y visualizar un mensaje informativo al usuario.</td> </tr> <tr> <td>Cabecera HTML (h4)</td> <td>Permite establecer y visualizar un mensaje informativo para que el usuario ingrese el número de la tarjeta de crédito.</td> </tr> <tr> <td>Campo de Texto 1</td> <td>Permite capturar el número de la tarjeta de crédito ingresada por el cliente.</td> </tr> <tr> <td>Cabecera HTML (h4)</td> <td>Permite establecer un mensaje informativo para que el usuario ingrese la contraseña de la tarjeta de débito.</td> </tr> <tr> <td>Campo de Texto 2</td> <td>Permite capturar la contraseña de la tarjeta de crédito ingresada por el cliente.</td> </tr> <tr> <td>Botón</td> <td>Permite enviar la información capturada en los campos de texto a un controlador (servlet) para que se realice el almacenamiento en la base de datos. Este botón envía la información una vez que esta ha sido correctamente</td> </tr> </tbody> </table>		Componente	Descripción	Etiqueta (título)	Permite establecer y visualizar el título de la banca electrónica.	Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario.	Cabecera HTML (h4)	Permite establecer y visualizar un mensaje informativo para que el usuario ingrese el número de la tarjeta de crédito.	Campo de Texto 1	Permite capturar el número de la tarjeta de crédito ingresada por el cliente.	Cabecera HTML (h4)	Permite establecer un mensaje informativo para que el usuario ingrese la contraseña de la tarjeta de débito.	Campo de Texto 2	Permite capturar la contraseña de la tarjeta de crédito ingresada por el cliente.	Botón	Permite enviar la información capturada en los campos de texto a un controlador (servlet) para que se realice el almacenamiento en la base de datos. Este botón envía la información una vez que esta ha sido correctamente
Componente	Descripción																
Etiqueta (título)	Permite establecer y visualizar el título de la banca electrónica.																
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario.																
Cabecera HTML (h4)	Permite establecer y visualizar un mensaje informativo para que el usuario ingrese el número de la tarjeta de crédito.																
Campo de Texto 1	Permite capturar el número de la tarjeta de crédito ingresada por el cliente.																
Cabecera HTML (h4)	Permite establecer un mensaje informativo para que el usuario ingrese la contraseña de la tarjeta de débito.																
Campo de Texto 2	Permite capturar la contraseña de la tarjeta de crédito ingresada por el cliente.																
Botón	Permite enviar la información capturada en los campos de texto a un controlador (servlet) para que se realice el almacenamiento en la base de datos. Este botón envía la información una vez que esta ha sido correctamente																

	ingresada y validada mediante código JQUERY.
Cabecera HTML (h6)	Permite establecer y visualizar un mensaje informativo para que el usuario conozca que puede redireccionar la página actual a otra página de inicio de sesión si ya se ha realizado el proceso de registro anteriormente.
Link (<a/>)	Al realizar click este permite redireccionar la página actual a la página de inicio de sesión.

Para el proceso de validación de la información ingresada en los campos de los formularios se utiliza JQUERY. Éste valida la longitud y estructura de los datos.

Ilustración 2.26 Boceto de Interfaz de Registro



Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.2. Interfaz inicio de sesión

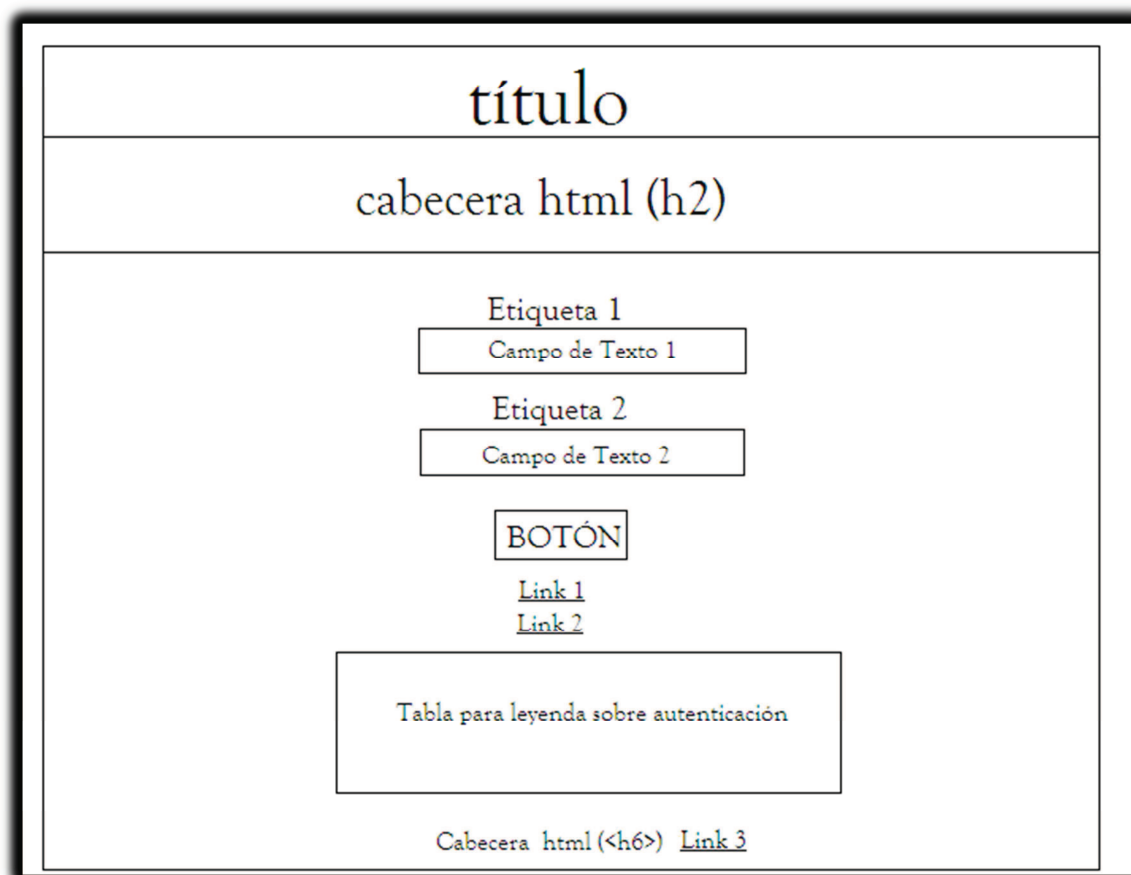
Nombre: Interfaz de inicio de sesión.

Objetivo: Permite al usuario cliente iniciar sesión en la banca electrónica, para

esto previamente se debe haber registrado mediante la interfaz de registro.

Componente	Descripción
Etiqueta (título)	Permite establecer y visualizar el título de la banca electrónica.
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario.
Cabecera HTML (h4)	Permite establecer y visualizar un mensaje informativo al cliente para que se percate que debe ingresar el nombre de usuario de la banca electrónica.
Campo de Texto 1	Permite capturar el nombre del usuario ingresado por el cliente.
Cabecera HTML (h4)	Permite establecer y visualizar un mensaje informativo al cliente para que se percate que debe ingresar su contraseña.
Campo de Texto 2	Permite capturar la contraseña correspondiente al nombre de usuario de la banca electrónica.
Botón	Permite enviar la información capturada en los campos de texto a un controlador (servlet) para que se realice el proceso de validación de la información y permitir el acceso o no a la banca electrónica.
Link (<a/>) 1	Al hacer click se redirecciona a una nueva interfaz que permite recordar el nombre de usuario de la banca electrónica.
Link (<a/>) 2	Al hacer click se redirecciona a una nueva interfaz que permite obtener una nueva contraseña cuando esta haya sido olvidada.
Tabla (<table><tr><td></td></tr></table>)	Permite establecer y visualizar una leyenda sobre el número de intentos fallidos de autenticación y el correspondiente bloqueo de la cuenta.
Link (<a/>) 3	Al hacer click se redirecciona a la interfaz de registro.

Ilustración 2.27 Boceto de Interfaz de Inicio de Sesión



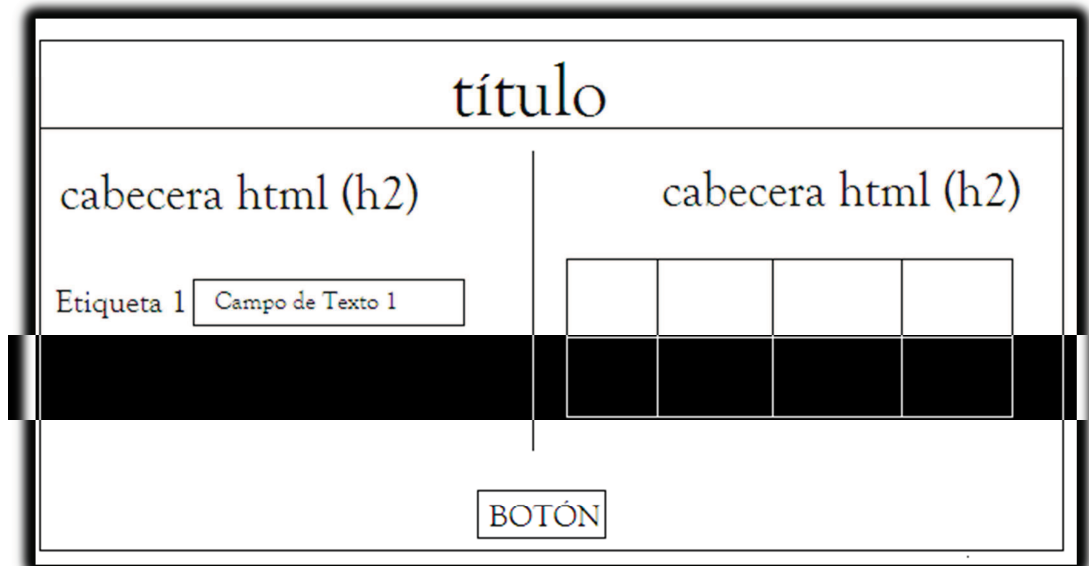
Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.3. Interfaz para contestar preguntas y seleccionar imagen

Nombre: Interfaz para contestar pregunta y seleccionar imagen.	
Objetivo: Permite al usuario cliente contestar una de las preguntas seleccionadas al azar de la base de datos y seleccionar la imagen que escogió en el proceso de registro.	
Componente	Descripción
Etiqueta (título)	Permite establecer y visualizar el título “preguntas de verificación” en la interfaz.
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario para se percate de que debe contestar una pregunta de verificación.
Cabecera HTML	Permite establecer y visualizar la pregunta que el cliente

(h4)	debe contestar en el campo de texto 1.
Campo de Texto 1	Permite capturar la respuesta a la pregunta de verificación.
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario para que se percate de que debe seleccionar una pregunta.
Tabla	Permite ordenar las 6 imágenes presentadas al usuario de las cuales debe seleccionar solamente una. Al hacer click sobre una imagen, se emite un evento onClick que llama a una función que dibuja un marco alrededor de la imagen la cual significa que ha sido seleccionada.
Botón (Enviar)	Permite enviar la información capturada en los campos de texto y la imagen en la tabla a un controlador (servlet) para que se realice el proceso de almacenamiento en la base de datos., para esto se llama a la función SUBMIT la cual se ejecuta siempre que la información esté correctamente ingresada.

Ilustración 2.28 Boceto de Interfaz para preguntas y seleccionar imagen

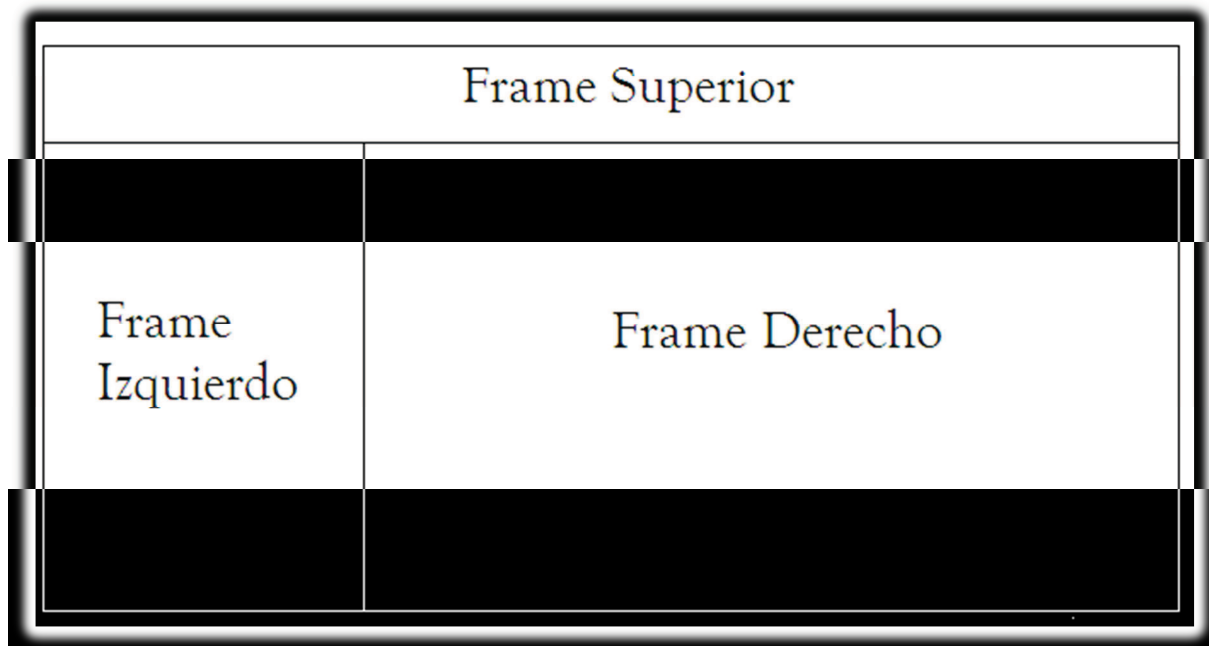


Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.4. *Interfaz principal de cliente*

Nombre: Interfaz principal cliente.	
Objetivo: Permite al cliente acceder a los servicios de la banca electrónica.	
Componente	Descripción
Frame Superior	Permite establecer y visualizar el nombre de la banca electrónica.
Frame Izquierdo	Contiene un menú con todas las funcionalidades de la banca electrónica LIX.
Frame Derecho	Contiene todas las interfaces redireccionadas como parte de la selección del menú en el frame izquierdo.

Ilustración 2.29 Boceto de Interfaz principal de Cliente



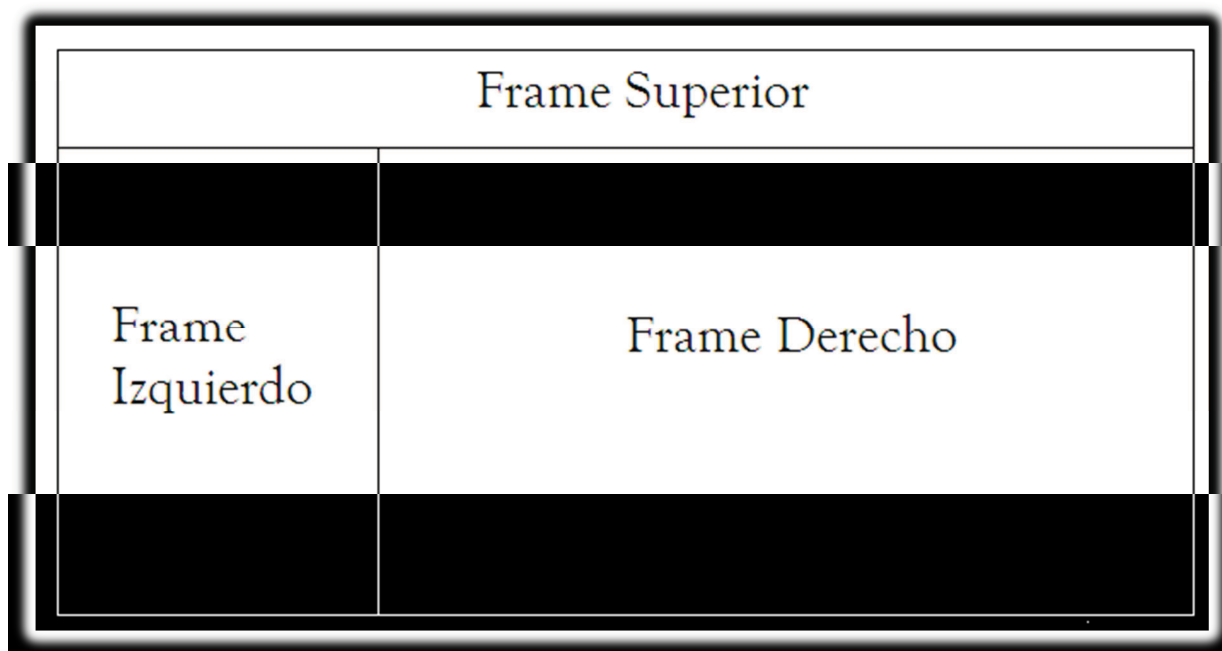
Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.5. *Interfaz principal de administrador*

Nombre: Interfaz principal administrador.
Objetivo: Permite al cliente acceder a los servicios de la banca electrónica.

Componente	Descripción
Frame Superior	Permite establecer y visualizar el nombre de la banca electrónica.
Frame Izquierdo	Contiene un menú con todas las funcionalidades de la banca electrónica LIX.
Frame Derecho	Contiene todas las interfaces redireccionadas como parte de la selección del menú en el frame izquierdo.

Ilustración 2.30 Boceto de Interfaz principal de administrador



Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.6. Interfaz actualizar información personal de cliente

Nombre: Actualizar información personal cliente	
Objetivo: Permite al usuario actualizar la información personal que ya había ingresada al momento de realizar el proceso de registro.	
Componente	Descripción
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario.
Tabla	Permite en su primera columna especificar todos las

	etiquetas de la información del cliente y en la segunda especificar los campos de texto donde se va a ingresar la información a actualizar.
Etiqueta 1	Permite establecer y visualizar información para que el usuario ingrese la identificación.
Campo de Texto 1	Permite capturar la identificación ingresada por el cliente.
Etiqueta 2	Permite establecer y visualizar información para que el usuario ingrese el nombre.
Campo de Texto 2	Permite visualizar el nombre del cliente. Este campo no es modificable
Etiqueta 3	Permite establecer y visualizar información para que el usuario ingrese su mail.
Campo de Texto 3	Permite capturar el mail ingresado por el cliente.
Etiqueta 4	Permite establecer y visualizar información para que el usuario seleccione una ciudad.
Combo Box	Permite seleccionar de una lista de ciudades predefinida, la ciudad de domicilio del cliente.
Etiqueta 5	Permite establecer y visualizar información para que el usuario ingrese la dirección.
Campo de Texto 4	Permite capturar la información de la dirección ingresada por el cliente.
Etiqueta 6	Permite establecer y visualizar información para que el usuario ingrese su número de teléfono celular.
Campo de Texto 5	Permite capturar la información del número de celular ingresada por el cliente.
Etiqueta 7	Permite establecer y visualizar información para que el usuario ingrese el teléfono fijo.
Campo de Texto 6	Permite capturar la información del número fijo ingresada por el cliente.
Botón (Actualizar)	Envía la información a un controlador (servlet) que realiza el proceso de actualización de la información en

	la base de datos, para esto se llama a la función SUBMIT la cual se ejecuta siempre que la información esté correctamente ingresada.
--	--

Ilustración 2.31 Boceto de Interfaz Actualizar información personal de cliente

Cabecera html (h2)

Etiqueta 1

Etiqueta 2

Etiqueta 3

Etiqueta 4

Etiqueta 5

Etiqueta 6

Etiqueta 7

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.7. *Interfaz actualizar información de autenticación cliente*

Nombre: Actualizar información de autenticación de cliente	
Objetivo: Permitir al usuario actualizar la información de autenticación que ya había ingresada al momento de realizar el proceso de registro.	
Componente	Descripción
Cabecera HTML (h2) número 1	Permite establecer y visualizar un mensaje informativo indicando que se debe ingresar/actualizar al menos 3 preguntas.
Cabecera HTML (h4) número 1	Permite establecer y visualizar la primera pregunta.
Campo de Texto 1	Permite capturar la respuesta de la primera pregunta.
Cabecera HTML (h4) número 2	Permite establecer y visualizar la segunda pregunta.
Campo de Texto 2	Permite capturar la respuesta de la segunda pregunta.
Cabecera HTML (h4) número 3	Permite establecer y visualizar la tercera pregunta.
Campo de Texto 3	Permite capturar la respuesta de la tercera pregunta.
Cabecera HTML (h4) número 4	Permite establecer y visualizar la cuarta pregunta.
Campo de Texto 4	Permite capturar la respuesta de la cuarta pregunta.
Cabecera HTML (h4) número 5	Permite establecer y visualizar la quinta pregunta.
Campo de Texto 5	Permite capturar la respuesta de la quinta pregunta.
Cabecera HTML (h4) número 6	Permite establecer y visualizar la sexta pregunta.
Campo de Texto 6	Permite capturar la respuesta de la sexta pregunta.
Cabecera HTML (h4) número 7	Permite establecer y visualizar la séptima pregunta.
Campo de Texto 7	Permite capturar la respuesta de la séptima pregunta.
Cabecera HTML (h2) número 2	Permite establecer y visualizar un mensaje informativo indicando que se puede seleccionar otra imagen a la ya

	seleccionada en caso que el usuario lo desee.
Tabla	Permite ordenar en cada celda una imagen para que el usuario seleccione una imagen.
Cabecera HTML (h2) número 3	Permite establecer y visualizar un mensaje informativo indicando que se puede actualizar la contraseña si lo desea.
Cabecera HTML (h4) numero 8	Permite establecer y visualizar un mensaje indicando al usuario que debe ingresar la contraseña actual.
Campo de texto 8	Permite capturar la contraseña actual del cliente.
Cabecera HTML (h4) numero 9	Permite establecer y visualizar un mensaje indicando al usuario que debe ingresar su nueva contraseña.
Campo de texto 9	Permite capturar la nueva contraseña.
Cabecera HTML (h4) numero 10	Permite establecer y visualizar un mensaje indicando al usuario que debe reingresar su nueva contraseña.
Campo de texto 10	Permite capturar el reingreso de la nueva contraseña.
Botón (Actualizar)	Permite enviar el formulario completo para guardarlo en la base de datos, para esto se llama a la función SUBMIT la cual se ejecuta siempre que la información esté correctamente ingresada.

Ilustración 2.32 Boceto de Interfaz de Registro

<p>Cabecera html (h2)</p> <p>Etiqueta 1</p> <p>Campo Texto 1</p> <p>Etiqueta 2</p> <p>Campo Texto 2</p> <p>Etiqueta 3</p> <p>Campo Texto 3</p> <p>Etiqueta 4</p> <p>Campo Texto 4</p> <p>Etiqueta 5</p> <p>Campo Texto 5</p> <p>Etiqueta 6</p> <p>Campo Texto 6</p> <p>Etiqueta 7</p> <p>Campo Texto 7</p>	<p>Cabecera html (h2)</p> <table border="1" style="width: 100%; height: 100px;"> <tr> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> </tr> </table>								
<p>Cabecera html (h2)</p> <p>Etiqueta 8 Campo de Texto 8</p> <p>Etiqueta 9 Campo de Texto 9</p> <p>Etiqueta 10 Campo de Texto 10</p> <p style="text-align: center;">BOTÓN</p>									

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.8. *Interfaz de transferencias entre cuentas propias*

Nombre: Actualizar transferencias entre cuentas propias	
Objetivo: Permitir al usuario realizar transferencias a otras cuentas del mismo banco.	
Componente	Descripción
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario.
Cabecera HTML	Permite visualizar un mensaje informativo en indique

(h4) numero 1	que el número en el campo de texto corresponde al número de cuenta origen.
Campo de Texto 1	Permite visualizar el número de cuenta origen. Este campo es no modificable,
Cabecera HTML (h4) numero 2	Permite visualizar un mensaje informativo indicando que se debe ingresar el número de cuenta destino.
Campo de Texto 2	Permite capturar el número de cuenta destino.
Cabecera HTML (h4) numero 3	Permite visualizar un mensaje informativo indicando que se debe ingresar el monto a transferir.
Campo de Texto 3	Permite capturar el monto a transferir.
Cabecera HTML (h4) numero 4	Permite visualizar un mensaje informativo indicando que se debe ingresar el correo electrónico
Campo de Texto 4	Permite capturar el correo electrónico.
Botón (Aceptar)	Envía la información a un controlador (servlet) que realiza a su vez llama a un procedimiento almacenado para realizar la transferencia.

Ilustración 2.33 Boceto de Interfaz de transferencias entre cuentas propias

Cabecera html (h2)

Etiqueta 1

Etiqueta 2

Etiqueta 3

Etiqueta 4

Etiqueta 5

Etiqueta 6

Elaborado por: **Gabriel Ignacio Villamagua Armijos**

2.3.1.4.9. *Interfaz de movimientos*

Nombre: Movimientos	
Objetivo: Permitir al usuario consultar los movimientos efectuados en su cuenta.	
Componente	Descripción
Cabecera HTML (h2) número 1	Permite establecer y visualizar un mensaje informativo especificando que se encuentra en la interface de movimientos.
Cabecera HTML (h2) numero 2	Permite establecer y visualizar un mensaje informativo al usuario especificando que se seleccione el rango de fechas a consultar.
Tabla	Permite ordenar los elementos del rango de fechas
Columna 1 (<td>) Etiquetas	Permite establecer texto en las filas (<tr>) indicando la fecha inicial y final de consulta de movimientos.
Columna 2 (<td>) combo box	Permite establecer dos combo boxes en las filas (<tr>) indicando el día inicial y final de movimientos a consultar. El número de días en cada combo box que se puede seleccionar es de 31.
Columna 3 (<td>) combo box	Permite establecer dos combo boxes en las filas (<tr>) indicando el mes inicial y final de movimientos a consultar. El número de días en cada combo box que se puede seleccionar es de 12.
Columna 4 (<td>) combo box	Permite establecer dos combo boxes en las filas (<tr>) indicando el año inicial y final de movimientos a consultar. El número de años en cada combo box que se puede seleccionar es 1.
Botón (Consultar)	Envía la información a un controlador (servlet) que ejecuta una sentencia sql contra la base de datos y obtiene la información a presentar.
Tabla	Permite visualizar todos los movimientos realizados durante un rango de fechas.

Ilustración 2.34 Boceto de Interfaz de Movimientos

The wireframe shows a rectangular interface with a double border. At the top left, it says 'Cabecera html (h2)'. In the center, there is another 'Cabecera html (h2)'. Below this, there are three labels: 'Etiqueta 1', 'Etiqueta 2', and 'Etiqueta 3'. Underneath these labels are two rows of input fields. The first row has three fields labeled 'Etiqueta 4', 'Etiqueta 2', and 'Etiqueta 3'. The second row has three fields labeled 'Etiqueta 5', 'Etiqueta 2', and 'Etiqueta 3'. Below the input fields is a rectangular button labeled 'BOTÓN'. At the bottom of the interface, there is a dark grey rectangular area containing three rows of seven small rectangular boxes, representing a table or grid.

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.10. Interfaz de recuperación de contraseña

Nombre: Interfaz de recuperación de contraseña	
Objetivo: Permite al usuario cliente recuperar su contraseña en caso de que haya sido olvidada.	
Componente	Descripción
Etiqueta (título)	Permite establecer y visualizar el título “restauración de contraseña”.
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario.
Cabecera HTML (h4)	Permite establecer y visualizar un mensaje informativo para que el usuario ingrese el número de la tarjeta de débito.
Campo de Texto 1	Permite capturar el número de la tarjeta de débito ingresada por el cliente.
Cabecera HTML	Permite establecer un mensaje informativo para que el

(h4)	usuario ingrese la contraseña de la tarjeta de débito.
Campo de Texto 2	Permite capturar la contraseña de la tarjeta de débito ingresada por el cliente.
Botón (Recuperar contraseña)	Envía la información capturada en los campos de texto a un controlador (servlet) para que se valide con la información en la base de datos. Este botón envía la información una vez que esta ha sido correctamente ingresada y validada mediante código JQUERY.
Cabecera HTML (h6)	Permite establecer y visualizar un mensaje informativo para que el usuario conozca que puede redireccionar la página actual a otra página de inicio de sesión si ya se ha realizado el proceso de registro anteriormente.
Link (<a/>)	Al realizar click este permite redireccionar la página actual a la página de inicio de sesión.

Ilustración 2.35 Boceto de Interfaz de Recuperación de Contraseña

título

cabecera html (h2)

Etiqueta 1

Etiqueta 2

Cabecera html (<h6>) [Link 1](#)

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.11. *Interfaz de recordatorio de usuario*

Nombre: Interfaz de recordatorio de usuario	
Objetivo: Permite al usuario cliente recordarle cuál es su usuario ingresando su correo electrónico.	
Componente	Descripción
Etiqueta (título)	Permite establecer y visualizar el título “restauración de contraseña”.
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario.
Campo de Texto 1	Permite capturar el correo electrónico ingresado por el cliente.
Botón (Recordar usuario)	Envía la información capturada en el campo de texto a un controlador (servlet) para que se valide con la información en la base de datos. Este botón envía la información una vez que esta ha sido correctamente ingresada y validada mediante código JQUERY.

Ilustración 2.36 Boceto de recordatorio de usuario

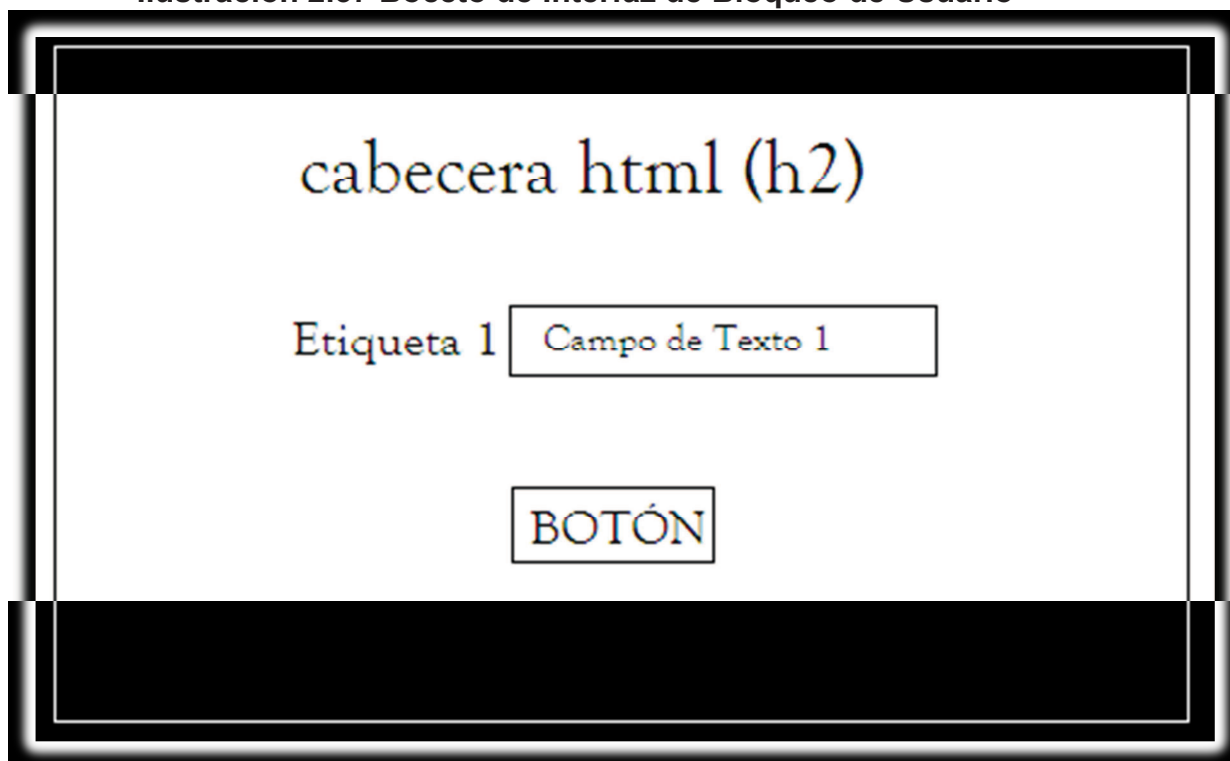


Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.12. *Interfaz de bloqueo de usuario*

Nombre: Bloqueo de usuario	
Objetivo: Permite al administrador bloquear un usuario.	
Componente	Descripción
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario.
Cabecera HTML (h4)	Permite establecer y visualizar un mensaje informativo para que el usuario ingrese la identificación.
Campo de Texto 1	Permite capturar el número de identificación del usuario
Botón (Bloquear)	Envía la información capturada en el campo de texto a un controlador (servlet) para que ejecute una sentencia SQL que realiza la actualización del estado del usuario.

Ilustración 2.37 Boceto de Interfaz de Bloqueo de Usuario

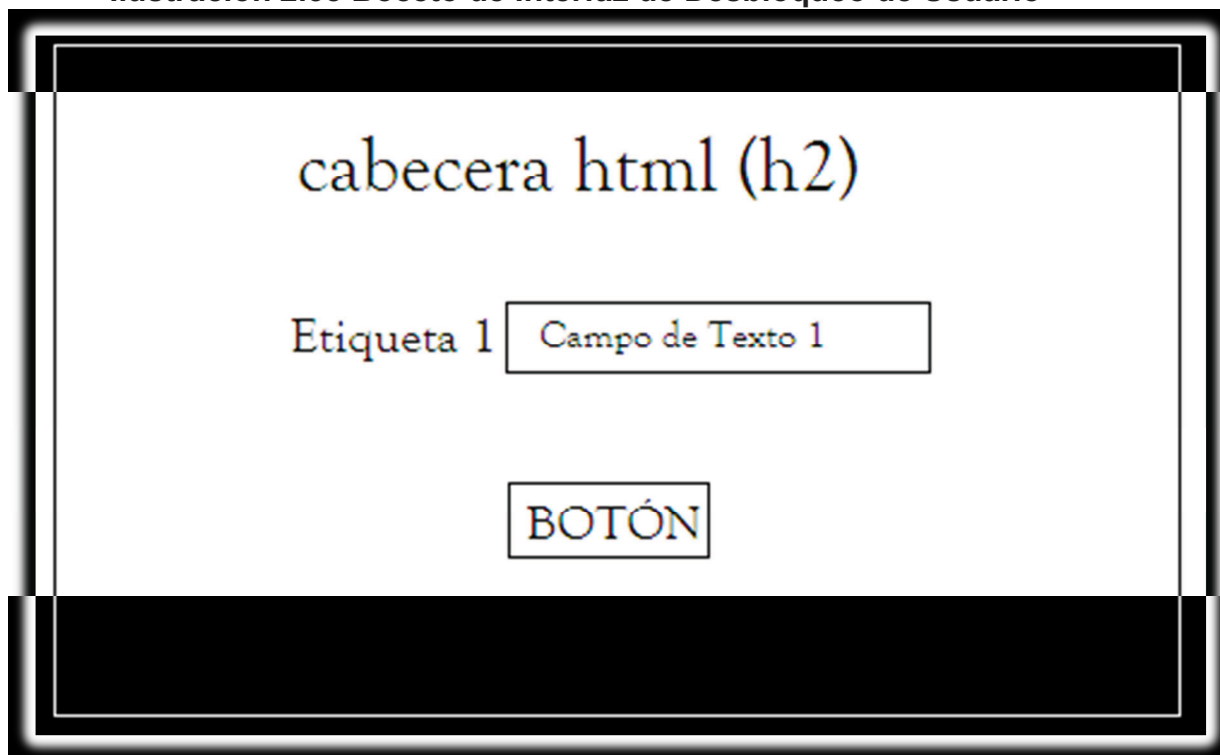


Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.13. *Interfaz de desbloqueo de usuario*

Nombre: Desbloqueo de usuario	
Objetivo: Permite al administrador desbloquear un usuario.	
Componente	Descripción
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario.
Cabecera HTML (h4)	Permite establecer y visualizar un mensaje informativo para que el usuario ingrese la identificación.
Campo de Texto 1	Permite capturar el número de identificación del usuario.
Botón (Desbloquear)	Envía la información capturada en el campo de texto a un controlador (servlet) para que ejecute una sentencia SQL que realiza la actualización del estado del usuario.

Ilustración 2.38 Boceto de Interfaz de Desbloqueo de Usuario



Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.4.14. Interfaz de nuevo usuario

Nombre: Nuevo Usuario	
Objetivo: Permite al administrador crear un nuevo usuario administrador	
Componente	Descripción
Cabecera HTML (h2)	Permite establecer y visualizar un mensaje informativo al usuario.
Cabecera HTML (h4)	Permite establecer y visualizar un mensaje informativo para que el usuario ingrese la identificación.
Campo de Texto 1	Permite capturar el número de identificación del usuario.
Cabecera HTML (h4)	Permite establecer y visualizar un mensaje informativo para que el usuario ingrese el nombre del nuevo usuario.
Campo de Texto 2	Permite capturar el nombre del nuevo usuario.
Botón (Crear Usuario)	Envía la información capturada en los campos de texto a un controlador (servlet) para que ejecute una sentencia SQL que almacena la información en la base de datos.

Ilustración 2.39 Boceto de Interfaz de Nuevo Usuario

cabecera html (h2)

Etiqueta 1

Etiqueta 2

Elaborado por: Gabriel Ignacio Villamagua Armijos

2.3.1.5. Quinto SPRINT

En este SPRINT procedemos a definir los patrones visuales para las JSP's (GUI's), es decir colores, fuente, tamaños de letras. Para describir de mejor manera procedemos a utilizar una tabla que consta de los siguientes campos:

Componente: tipo de componente HTML.

Fuente: fuente de la letra.

Color: color de la fuente.

Tamaño Letra: tamaño de la fuente.

Componente	Fuente	Color	Tamaño Letra
Título	Arial, Helvetica, sans-serif	Blanco	20px
Etiqueta	Arial, Helvetica, sans-serif	Negro	12px
Botón	Arial, Helvetica, sans-serif	Fuente Blanca, Fondo Gris	12px
Fondo frame superior	NA	#a369c4	NA
Fondo frame izquierdo	NA	Blanco	NA
Fondo frame derecho	NA	Blanco	NA
Tabla	NA	Borde: 1px solid #BBAA99	NA
Link	Arial, Helvetica, sans-serif	#a369c4	12px
Cabecera h1	Arial, Helvetica, sans-serif	#fff	40px

Cabecera h2	Arial, Helvetica, sans-serif	#4e4e4e	28px
Cabecera h3	Arial, Helvetica, sans-serif	#fff	24px

2.4. CONSTRUCCIÓN DEL SISTEMA

Una vez realizado la gerencia y seguimiento de los SPRINT's procedemos a detallar los elementos que se desarrollaron.

SPRINT PRIMERO

Como resultado final de este SPRINT se desarrolló un script que será ejecutado en el DBMS (MySQL) y permitirá que se almacene la información de los usuarios, sus transacciones, información crediticia, etc., además de crear los procedimientos almacenados de las transferencias.

A continuación presentamos un pequeño código representativo del script resultante.

TABLA CLIENTE

```

-----
CREATE TABLE `cliente` (
  `identificacion` char(11) NOT NULL,
  `nombre` varchar(40) NOT NULL,
  `email` varchar(40) default NULL,
  `ciudad` varchar(40) default NULL,
  `direccion` varchar(40) default NULL,
  `telefono_celular` varchar(11) default NULL,
  `telefono_casa` varchar(11) default NULL,
  `fecha_afiliacion` timestamp NOT NULL default CURRENT_TIMESTAMP on
update CURRENT_TIMESTAMP,
  `preg_resp_img` varchar(5000) default NULL,
  `usuario` varchar(5000) default NULL,
  `clave` varchar(5000) default NULL,
  `estado` char(1) default 'A',
  `intentosLogin` int(10) unsigned default '0',
  `tipo` char(1) NOT NULL default 'C',
  PRIMARY KEY USING BTREE (`identificacion`)
)

```

PROCEDIMIENTO ALMACENADO PARA TRANSFERENCIAS

```

-----
DELIMITER $$

/*!50003 SET @TEMP_SQL_MODE=@@SQL_MODE,
SQL_MODE='STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER' */ $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `realizarTransferencia`(in
cuentaOrigen varchar(12), in cuentaDestino varchar(12), in monto
double, in descripcion varchar(100))
BEGIN
DECLARE cuentaOrigenTemp varchar(12) DEFAULT cuentaOrigen;
DECLARE cuentaDestinoTemp varchar(12) DEFAULT cuentaDestino;
DECLARE montoTemp double DEFAULT monto;
DECLARE descripcionTemp varchar(100) DEFAULT descripcion;
DECLARE saldoCuentaOrigen double DEFAULT 0;
DECLARE saldoCuentaDestino double DEFAULT 0;
START TRANSACTION;
SET saldoCuentaOrigen = (select saldo from cuentasbanco where
numeroCuenta = cuentaOrigenTemp);
SET saldoCuentaDestino = (select saldo from cuentasbanco where
numeroCuenta = cuentaDestinoTemp);
SET saldoCuentaOrigen = (saldoCuentaOrigen - montoTemp);
SET saldoCuentaDestino = (saldoCuentaDestino + montoTemp);
update cuentasbanco set saldo=saldoCuentaOrigen where numeroCuenta =
cuentaOrigenTemp;
update cuentasbanco set saldo=saldoCuentaDestino where numeroCuenta =
cuentaDestinoTemp;
insert into transacciones(cuentaOrigen,
cuentaDestino,codigoTransaccion, cantidad, fechaTransaccion,
descripcion) values(cuentaOrigenTemp, cuentaDestinoTemp, 1, montoTemp,
now(), descripcionTemp);
COMMIT;
END $$
/*!50003 SET SESSION SQL_MODE=@TEMP_SQL_MODE */ $$

DELIMITER ;

```

SPRINT SEGUNDO

Una vez finalizado el segundo SPRINT se desarrollaron los siguientes elementos:

- Se creó una clase GestorHiloCorreo.java que extiende de la clase Thread que se encarga de realizar el envío de correo electrónico, el funcionamiento es simple esta clase inicia sesión con el servidor Google y procede a enviar el correo electrónico, el éxito o fracaso al enviar el correo electrónico no afecta a los procesos principales tanto en funcionamiento y en tiempo de respuesta.

A continuación presentamos el código de la clase java que realiza el proceso de envío de correo electrónico.

```

import java.util.Properties;
import javax.mail.Message;
import javax.mail.MessagingException;
import javax.mail.NoSuchProviderException;
import javax.mail.Session;
import javax.mail.Transport;
import javax.mail.internet.AddressException;
import javax.mail.internet.InternetAddress;
import javax.mail.internet.MimeMessage;
import com.epn.tesis.bancapersonal.interfaces.IMailParameters;

public class GestorHiloCorreo extends Thread {

    String mailDestinatario = "";
    String asunto = "";
    String mensaje = "";

    public GestorHiloCorreo(String mailDestinatario, String asunto,
String mensaje){
        this.mailDestinatario = mailDestinatario;
        this.asunto = asunto;
        this.mensaje = mensaje;
    }

    public void run() {
        Properties props = new Properties();

        props.setProperty("mail.smtp.host",
IMailParameters.MAIL_SMTP_HOST);
        props.setProperty("mail.smtp.starttls.enable",
IMailParameters.MAIL_TLS_ENABLE);
        props.setProperty("mail.smtp.port",
IMailParameters.MAIL_SMTP_PORT);
        props.setProperty("mail.smtp.user",
IMailParameters.MAIL_SMTP_USER);
        props.setProperty("mail.smtp.auth",
IMailParameters.MAIL_SMTP_AUTH);

        Session session = Session.getDefaultInstance(props);
        session.setDebug(true);

        MimeMessage message = new MimeMessage(session);
        try {
            message.setFrom(new
InternetAddress(IMailParameters.MAIL_SMTP_USER));
            message.addRecipient(Message.RecipientType.TO, new
InternetAddress(mailDestinatario));
            message.setSubject(asunto);
            message.setText(mensaje);

        } catch (AddressException e) {
            e.printStackTrace();
        } catch (MessagingException e) {
            e.printStackTrace();
        }
    }
}

```

```

    }

    Transport t = null;
    try {
        t = session.getTransport("smtp");
    } catch (NoSuchProviderException e) {
        e.printStackTrace();
    }
    try {
        t.connect(IMailParameters.MAIL_SMTP_USER, IMailParameters.MAIL_SM
TP_PASSWORD);
    } catch (MessagingException e) {
        e.printStackTrace();
    }

    try {
        t.sendMessage(message, message.getAllRecipients());
    } catch (MessagingException e) {
        e.printStackTrace();
    }
    try {
        t.close();
    } catch (MessagingException e) {
        e.printStackTrace();
    }
}
}
}

```

SPRINT TERCERO

En este SPRINT una vez concluido se desarrollaron los siguientes elementos:

- Se creó una clase GestorHiloSMS.java que extiende de la clase Thread que se encarga de realizar el envío de mensajes de texto a los teléfonos celulares de los clientes, el funcionamiento es el siguiente:
 - Esta clase envía una orden mediante terminal a Gammu.
 - Gammu envía una orden al teléfono móvil.
 - El teléfono móvil realiza el proceso de enviar el mensaje de texto.

A continuación presentamos el código de la clase java que realiza el proceso de envío de mensajes de texto.

```

import java.io.BufferedReader;
import java.io.InputStreamReader;

public class GestorHiloSMS extends Thread {

    String mensaje = "";

```

```

String numeroCelular = "";
String s = "";

public GestorHiloSMS(String mensaje, String numeroCelular) {
    this.mensaje = mensaje;
    this.numeroCelular = numeroCelular;
}

public void run() {
    String[] commands = new String[] { "gammu", "sendsms",
"TEXT",
        numeroCelular, "-text", mensaje };

    try {
        Process p = Runtime.getRuntime().exec(commands);
        BufferedReader stdInput = new BufferedReader(new
InputStreamReader(p.getInputStream()));
        BufferedReader stdError = new BufferedReader(new
InputStreamReader(p.getErrorStream()));

        while ((s = stdInput.readLine()) != null) {
            System.out.println(s);
        }

        while ((s = stdError.readLine()) != null) {
            System.out.println(s);
        }

    } catch (Exception e) {
        try {
            Runtime.getRuntime().exec(commands);
        } catch (Exception e2) {
            System.out.print("No se ha podido enviar el SMS
al número "
                + numeroCelular);
        }
    }
}
}

```

SPRINT CUARTO

En este SPRINT se desarrollaron los siguientes elementos:

- GUI's (GraphicalUser Interfaces). Se desarrollaron GUI's (JSP's) entre los cuales constan los siguientes formularios: registro, inicio de sesión,

actualización de datos personales, actualización de datos de autenticación, recuperación de contraseña, vista principal de cliente y administrador y transferencias entre cuentas propias.

- Controladores. Se desarrollaron controladores que realizan el proceso dinámico de capturar los proveniente de la vista, utiliza los modelos y llama a los servicios.
- Servicios. Se desarrollaron servicios utilizados por los controladores y la vista para la realizar la lógica del negocio. Los servicios creados son: actualización de información cliente, autenticación cliente, consulta de movimientos, obtención información de usuario, envío correo electrónico, envío mensaje de texto, recordar usuario, realización de transferencias, consultar saldo, bloqueo y desbloqueo de usuarios.

A continuación presentamos una parte de código java de algunas GUI, controladores y servicios (los más representativos):

```
<form
    action="/BancaPersonalSistemaBiometrico/ValidarRespuestaIngresad
aClienteControlador"
    method="post">

<table align="center">
    <tr>
        <td>

            <table>
                <h2>Conteste la pregunta de verificación</h2>
                <tr>
                    <td><%=preguntaEImagen.get(1)%></td>
                    <td><input type="text"
name="preguntaSeleccionada"
                                id="preguntaSeleccionada"
onkeyup="ocultarBoton(this.value);"></td>
                </tr>
            </table>

        </td>

        <td><br>
        </td>

        <td>

        <h2>Seleccione una Imagen</h2>

        <table>
            <tr align="center">

                <td></td>

                                <td></td>

                                <td></td>

                                <td></td>
                                </tr>

                                <tr align="center">

                                <td></td>

                                <td></td>

                                <td></td>

                                <td></td>

                                </tr>
                                </table>
                                </td>
                                </tr>

```

```

</table>

<br>
<br>

<div><input type="hidden" name='imagenSeleccionada' id='image'></div>
<div id="btnAceptar"><input type="submit" value="Aceptar"></div>

</form>

```

EL siguiente código muestra la GUI de pregunta de autenticación.

```

public Vector<String> seleccionarPreguntaCliente (String
identificacion ) throws SQLException, ClassNotFoundException{

    // CONEXION
    Class.forName(IDatabaseParameters.DRIVER);
    String url = "jdbc:mysql://" + IDatabaseParameters.SERVER + ":"
+ IDatabaseParameters.PORT + "/tesis";
    Connection con = DriverManager.getConnection(url,
IDatabaseParameters.USER, IDatabaseParameters.PASSWORD);
    // FIN CONEXION

    int idPregunta;

    idPregunta = (int) (1 + Math.random()*7);

    PreparedStatement ps = con
    .prepareStatement("select idPregunta from autenticacion
where idCLiente like ? ");
    ps.setString(1, identificacion);
    ResultSet rs = ps.executeQuery();

    int valor = 0;

    Vector idPreguntas = new Vector();

    while(rs.next()){
        valor =
Integer.parseInt((rs.getString("idPregunta")));
        idPreguntas.add(String.valueOf(valor));
    }

    while(Integer.parseInt((String)
(idPreguntas.get((int) (Math.random()*idPreguntas.size()))))!=
idPregunta){
        for (int i = 0; i < idPreguntas.size(); i++) {

            if
(idPreguntas.get(i).equals(idPregunta)) {
                break;
            }
        }
    }
}

```

```

                else {
                    idPregunta = (int) (1 +
Math.random()*7);
                }
            }
        }

        PreparedStatement ps2 = con
        .prepareStatement("select a.idCliente, p.pregunta," +
            " a.respuestaPregunta, a.idImagen from
autenticacion a, " +
            "preguntas p, imagenes i where a.idCliente like
? " +
            "and a.idPregunta like ? " +
            "and p.id = a.idPregunta and i.id = a.idImagen ");
        ps2.setString(1, identificacion);
        ps2.setInt(2, idPregunta);
        ResultSet rs2 = ps2.executeQuery();
        Vector preguntaEImagen = new Vector();

        while (rs2.next()) {

            preguntaEImagen.add(rs2.getString("idCliente"));
            preguntaEImagen.add(rs2.getString("pregunta"));

            preguntaEImagen.add(rs2.getString("respuestaPregunta"));
            preguntaEImagen.add(rs2.getString("idImagen"));
        }

        return preguntaEImagen;
    }

```

EL siguiente código muestra el controlador de autenticación.

```

import java.io.IOException;
import java.sql.SQLException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import com.epn.tesis.bancapersonal.objects.Cliente;
import com.epn.tesis.bancapersonal.services.registrarCliente.GestorActualizar
Cliente;

@WebServlet("/ActualizarClienteControlador")
public class ActualizarClienteControlador extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public ActualizarClienteControlador() {

```

```

        super();
    }

    protected void doGet(HttpServletRequest request,
        HttpServletResponse response) throws
ServletException, IOException {
        try {
            doIt(request, response);
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (InstantiationException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (IllegalAccessException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (ClassNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    protected void doPost(HttpServletRequest request,
        HttpServletResponse response) throws
ServletException, IOException {
        try {
            doIt(request, response);
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (InstantiationException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (IllegalAccessException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (ClassNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    public void doIt(HttpServletRequest request, HttpServletResponse
response)
        throws SQLException, InstantiationException,
        IllegalAccessException, ClassNotFoundException,
ServletException,
        IOException {

        Cliente cliente = new Cliente();

        // obtener y validar los parámetros

        cliente.setIdentificacion(request.getParameter("identificacion")
);

        cliente.setNombre(request.getParameter("nombre"));
        cliente.setEmail(request.getParameter("email"));
        cliente.setCiudad(request.getParameter("ciudad"));
    }

```



```

        cliente.setDireccion(request.getParameter("direccion"));
        cliente.setTelefonoCelular(request.getParameter("movil"));

        cliente.setTelefonoConveccional(request.getParameter("fijo"));
        cliente.setFechaAfiliacion(null);

        //cedula necesaria para guardar en la BDD
        String identificador =
request.getParameter("identificacion");

        HttpSession sessionIdentificador = request.getSession();
        sessionIdentificador.setAttribute("IDENTIFICADOR"
,identificador);

        // generar registro
        GestorActualizarCliente gesActCli = new
GestorActualizarCliente();

        // invocar al servicio
        Boolean exito = gesActCli.actualizarCliente(cliente);

        if (exito) {

            RequestDispatcher rd =
request.getRequestDispatcher("Banca/exitoActualizacionDatosPersonalesC
liente.jsp");
            rd.forward(request, response);

        } else {

            RequestDispatcher rd =
request.getRequestDispatcher("Banca/errorActualizacionDatosPersonales.
jsp");
            rd.forward(request, response);

        }

    }

}

```

SPRINT QUINTO

En este SPRINT se desarrollaron los siguientes elementos:

- CSS. Se desarrollaron los archivos style.css y style_tablas.css los cuales dan un patrón visual (color, bordes, fuente de las celdas, interlineados, etc.) a todas las GUI's (JSP's). Además, se definieron los patrones visuales de fondos de pantalla, fuente de letra y menús.
- JAVASCRIPT (JQUERY). Se implementó validaciones de ingreso de datos en formularios para mantener la integridad de la base de datos.

Además, se implementó seguridad de inicio y finalización de sesión de la banca electrónica.

- Finalmente, se creó un archivo `bancaElectronicaLIX.war` que será deployado en el servidor TOMCAT para la puesta en producción de la banca electrónica LIX.

A continuación presentamos una parte representativa del código CSS generado:

```
/* Left & Right alignment */
.left {
    float: left
}

.right {
    float: right
}

.wrapper {
    width: 100%;
    overflow: hidden
}

.inner_copy, .inner_copy a {
    border: 0;
    float: right;
    background: #fff;
    color: #f00;
    width: 50%;
    line-height: 10px;
    font-size: 10px;
    margin: -220% 0 0 0;
    overflow: hidden;
    padding: 0
}

body {
    background: #fff;
    border: 0;
    font: 13px Arial, Helvetica, sans-serif;
    color: #4e4e4e;
    line-height: 20px
}

/* Global Structure */
.body1 {
    background: url(../images_css/bg_img2.png) top center no-repeat;
    height: 680px;
    position: absolute;
    width: 100%;
    z-index: 1
}

.body2 {
```

```
        background: url(../images_css/top.jpg) top center no-repeat
#2d213c
    }

    .body3 {
        background: url(../images_css/line_hor1.jpg) bottom repeat-x
#f5f5f5;
        border-top: 1px solid #fff
    }

    .body4 {
        background: url(../images_css/line_hor2.jpg) top repeat-x
#f5f5f5
    }

    .body5 {
        background: url(../images_css/bot.jpg) top center no-repeat
#38294a
    }

    .main {
        margin: 0 auto;
        width: 900px;
        position: relative;
        z-index: 2
    }

    /* main layout */
    a {
        color: #a369c4;
        text-decoration: underline;
        outline: none
    }

    a:hover {
        text-decoration: none
    }

    h1 {
        padding: 21px 0 0 20px
    }

    h2 {
        font-size: 28px;
        font-weight: bold;
        font-weight: 300;
        color: #4e4e4e;
        padding: 29px 0 27px 0;
        line-height: 38px
    }

    h2 span {
        display: block;
        margin-top: -10px
    }

    h3 {
        background: url(../images_css/box2.jpg) no-repeat;
        font-size: 24px;
        padding: 0 20px;
        margin: 17px 0 20px 0;
```

```

        color: #fff;
        line-height: 51px;
        font-weight: 300;
        text-align: center;
    }

    .text1 {
        font-size: 40px;
        font-weight: 300;
        color: #fff;
        line-height: 39px;
        padding: 30px 0 30px 0;
        text-align: center;
    }

```

Ilustración 2.30 CSS para menus, links, títulos, cabeceras, etc. excepto para tablas.

```

a,abbr,acronym,address,applet,article,aside,audio,b,blockquote,big,bod
y,center,canvas,caption,cite,code,command,datalist,dd,del,details,dfn,
dl,div,dt,em,embed,fieldset,figcaption,figure,font,footer,h1,h2,h3,h4,
h5,h6,header,hgroup,html,i,iframe,img,ins,kbd,keygen,label,legend,li,m
eter,nav,object,ol,output,p,pre,progress,q,s,samp,section,small,span,s
ource,strike,strong,sub,sup,tbody,tfoot,thead,th,tr,tdvideo,tt,u,ul,va
r
{
    background: transparent;
    border: 0 none;
    font-size: 100%;
    margin: 0;
    padding: 0;
    border: 0;
    outline: 0;
    vertical-align: top;
    text-align: center;
}

table,td,tr {
    vertical-align: middle;
}

```

EL siguiente código JQUERY/JAVASCRIPT sirve para validación de campos (correo electrónico.)

```

<script type="text/javascript">
    //campos no vacios
    jQuery.fn.emailTest = function() {

        if ($("#emailDisponible").html() == 'EMAIL YA REGISTRADO
        POR OTRO USUARIO') {
            alert("¡EMAIL YA REGISTRADO POR OTRO USUARIO, por
            favor ingrese otro email!");
            return false;
        } else {
            return true;
        }
    };

```

```

$(document).ready(function() {
    $("#registroForm").submit(function() {
        if ($("#sugerenciaEmailUnico").emailTest()) {
            return true;
        } else
            return false;
    });
});
</script>

<!-- comprobar correo unico -->
<script type="text/javascript">
    function verificarEmail(email) {
        $.post("CheckEmail.jsp", {
            email : "" + email + ""
        }, function(data) {
            $('#sugerenciaEmailUnico').html(data).show();
        });
    }
</script>

$(document).ready(function() {

    $('form').keypress(function(e) {
        if (e == 13) {
            return false;
        }
    });

    $('input').keypress(function(e) {
        if (e.which == 13) {
            return false;
        }
    });
});

$(document).ready(function() {

    $('form').keypress(function(e) {
        if (e == 13) {
            return false;
        }
    });

    $('input').keypress(function(e) {
        if (e.which == 13) {
            return false;
        }
    });
});

```

CAPÍTULO 3.

SIMULACIÓN DEL SISTEMA EN EL CASO DE ESTUDIO

El sistema ha sido construido y ahora procedemos a realizar la simulación en el caso de estudio para asegurar el correcto funcionamiento. El proceso de simulación comprende el levantamiento del ambiente, la ejecución y un análisis de los resultados.

3.1. CONSTRUCCIÓN DE AMBIENTE DE SIMULACIÓN

Procedemos a realizar el levamiento de ambiente de simulación, esta simulación nos permitirá determinar si el sistema está en condiciones de ser implementado en producción.

3.1.1. HARDWARE UTILIZADO EN LA SIMULACIÓN

3.1.1.1. Servidor 1

En este servidor estará cargado el sistema (banca electrónica) en el cual los clientes enviaran sus solicitudes para ser atendidas.

Características

IP: 192.168.0.107

RAM: 4 GB

PROCESADOR: Intel Core I3 CPU M 350 2.27GHz

DISCO: 500GB

TARJETA INALÁMBRICA: Realtek RTL8 802.11b/g/n

3.1.1.2. Servidor 2

En este servidor estará cargado el servicio externo (agua potable) con el cual se deberá conectar el sistema (banca electrónica) para realizar el pago del servicio externo.

Características

IP: 192.168.0.108

RAM: 2 GB

PROCESADOR: Intel Core 2 Duo CPU 2.20GHz

DISCO: 320GB

TARJETA ETHERNET: Realtek PCIe FE Family Controller

3.1.1.3. Servidor 3

En este servidor se encuentra levantada la base de datos en la cual se encuentra todos los datos con los que va a trabajar el sistema (banca electrónica LIX).

Características

IP: 192.168.0.109

RAM: 4 GB

PROCESADOR: Intel Core 2 Duo CPU 2.20GHz

DISCO: 500GB

TARJETA ETHERNET: Realtek PCIe FE Family Controller

3.1.1.4. Cliente

El cliente puede ser un dispositivo móvil o cualquier dispositivo que tenga conexión a internet, en nuestra simulación el cliente es una pc con las siguientes características:

Características

IP: 192.168.0.110

RAM: 1GB

PROCESADOR: Intel Pentium 1.8GHz

DISCO: 160GB

TARJETA ETHERNET: Realtek PCIe FE Family Controller

3.1.1.5. Dispositivo Móvil

Este dispositivo móvil es el encargado de enviar los SMS's a los números de celulares de los clientes.

Características

SISTEMA OPERATIVO: Nokia OS S40 v3

CONECTIVIDAD BANDA: 900/1800/1900

MENSAJES: SMS/MMS

DIMENSIONES: 107x45x13.1 mm

PESO: 81 gr

MEMORIA INTERNA: 2000MB

3.1.2. SOFTWARE UTILIZADO EN LA SIMULACIÓN

3.1.2.1. Servidor 1

Características

SISTEMA OPERATIVO: Windows 7 Professional Service Pack 1

SERVIDOR WEB: Apache Tomcat 7.0

SDK: JDK 1.6

3.1.2.2. Servidor 2

Características

SISTEMA OPERATIVO: Windows 7 Professional Service Pack 1

SERVIDOR WEB: JBOSS 5.0.0

SDK: JDK 1.6

3.1.2.3. Servidor 3

Características

SISTEMA OPERATIVO: Windows 7 Professional Service Pack 1

DBMS: MySQL Server 5.0

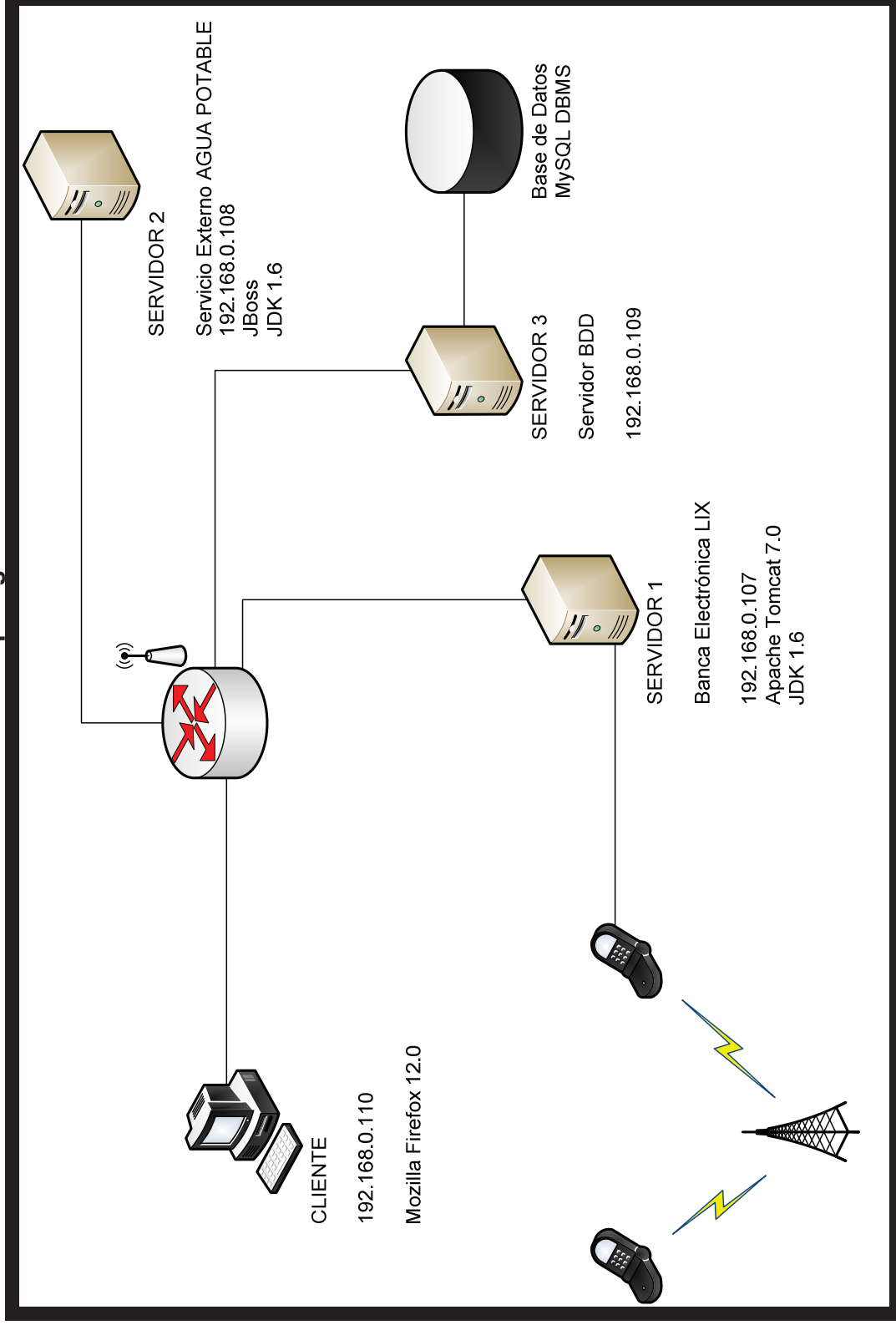
3.1.2.4. Cliente

Características

NAVEGADOR WEB: Mozilla Firefox 14.0.1

En el gráfico a continuación se presenta la topología de red que se va a utilizar para la simulación en el caso de estudio (Bancos).

Ilustración 3.1 Topología de Red



Elaborado por: Gabriel Ignacio Villamagua Armijos

3.2. EJECUCIÓN DE SIMULACIÓN

Una vez implementado toda la infraestructura de la banca electrónica procedemos a realizar la ejecución, para esto definimos casos de prueba que nos servirán para comprobar si el sistema cumple con los objetivos deseados.

En SCRUM no se especifica un método para realizar simulaciones, sin embargo se sugiere realizar la simulación (prueba) mediante la comprobación de las historias de usuario.

3.2.1. CASO DE SIMULACIÓN # 01 “REGISTRO DE CLIENTE EN LA BANCA ELECTRÓNICA”

Para realizar la simulación vamos a utilizar el siguiente formato de tabla con los siguientes campos:

Número: secuencial de la historia del caso de simulación (prueba).

Historia de Usuario: nombre y número de la historia de usuario de lo que se quiere simular (probar).

Nombre caso de simulación: Nombre que se le asigna al caso de simulación (prueba).

Descripción: breve descripción de lo que se desea simular.

Prerrequisitos: condiciones previas que deben existir antes de realizar la simulación.

Entrada/Pasos de Ejecución: pasos y entradas necesarias para realizar la simulación.

Resultado esperado: resultado que se espera luego de realizar la simulación.

Resultado Simulación: resultado real obtenido luego de haberse realizado la simulación [13].

CASO DE SIMULACIÓN	
Número: 1	Historia de usuario: 1 - Registrar cliente en la banca electrónica.
Nombre Caso de Simulación: Registro de cliente en la banca electrónica.	
Descripción: Se procede a realizar la simulación del proceso de registro del cliente en la banca electrónica LIX.	
Prerrequisitos: El usuario de la banca electrónica debe ser cliente del banco por lo que debe contar con una tarjeta de débito, contraseña y número de cuenta.	
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1) El usuario ingresa a la banca y selecciona la opción de registrarse. 2) Ingresar número de tarjeta de débito y contraseña. 3) Seleccionar un usuario y contraseña para acceder a la banca. 4) Ingresar respuestas a mínimo 3 preguntas y seleccionar una imagen. 5) Se produce redirección hacia la GUI de inicio de sesión. 	
Resultado Esperado: <ul style="list-style-type: none"> • El cliente ya puede ingresar a la banca electrónica. • La información de autenticación y datos personales ha sido registrada en las estructuras de la base de datos. 	
Resultado Simulación: Excelente. El sistema registró en la base de datos la información de autenticación en las estructuras cliente y autenticación. El cliente obtuvo su usuario y contraseña por lo que ya puede iniciar sesión para utilizar los servicios de la banca electrónica.	

Demostración:

- 1) El usuario ingresa a la banca y selecciona la opción de registrarse.

INGRESO BANCA PERSONAL

Ingrese sus datos para iniciar sesión

Usuario:

Contraseña:

[¿Recordar su usuario?](#)
[¿Olvidó su contraseña?](#)

¡RECUERDA que el número máximo de intentos fallidos es TRES, luego tu cuenta será bloqueada y deberás acercarte a la oficina del banco más cercana!

Si no estas registrado haz click [aquí](#)

2) Ingresar número de tarjeta de débito y contraseña.

LIX BANCA PERSONAL

Ingrese sus datos para realizar el proceso de registro

Número de Tarjeta de Débito:

OK

Contraseña:

Si ya estas registrado [Inicia Sesión](#)

3) Seleccionar un usuario y contraseña para acceder a la banca.

REGISTRO BANCA PERSONAL

Ingrese/Actualice la siguiente información

Identificación:

Nombre:

Email:

Ciudad:

Dirección:

Móvil:

Fijo:

Ingrese nueva información

Usuario:

USUARIO DISPONIBLE

Clave:

Reingrese su clave:

4) Ingresar respuestas a mínimo 3 preguntas y seleccionar una imagen.

PREGUNTAS DE SEGURIDAD

Conteste al menos 3 preguntas

¿Cuál es el sabor de su helado favorito?

¿Cuál es su deporte favorito?

¿Cuál es su cantante favorito?


¿Año de nacimiento de su madre?

¿Cuál es su bebida favorita?

¿Cuál es su mascota favorita?

"Ingrese una frase propia"

Seleccione una Imagen



5) Se produce redirección hacia la GUI de inicio de sesión.

aquí'."/>

3.2.2. CASO DE SIMULACIÓN # 02 “INICIO SESIÓN DE CLIENTE”

CASO DE SIMULACIÓN	
Número: 2	Historia de usuario: 2 – Iniciar sesión cliente
Nombre Caso de Simulación: Inicio de sesión cliente.	
Descripción: Una vez registrado se procede a iniciar sesión con el usuario y contraseña que seleccionamos en el proceso de registro.	
Prerrequisitos: El usuario debe haberse registrado en la banca electrónica.	
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1) Ingresar a la GUI de inicio sesión de la banca electrónica. 2) Ingresar usuario y contraseña. 3) Ingresar respuesta a la pregunta y seleccionar imagen previamente escogida en el proceso de registro. 	
Resultado Esperado:	

- El cliente puede acceder a la banca electrónica y realizar las transacciones disponibles.

Resultado Simulación: Excelente.

El usuario puede acceder y visualizar todas las transacciones disponibles para su perfil como cliente.

Demostración:

- 1) Ingresar a la GUI de inicio sesión de la banca electrónica.

- 2) Ingresar usuario y contraseña.

- 3) Ingresar respuesta a la pregunta y seleccionar imagen previamente escogida en el proceso de registro.



La GUI de inicio de sesión es presentada.



3.2.3. CASO DE SIMULACIÓN #03 “FINALIZACIÓN DE SESIÓN”

CASO DE SIMULACIÓN	
Número: 3	Historia de usuario: 3 – Finalización de Sesión.
Nombre Caso de Simulación: Finalización de Sesión.	
Descripción: Una vez realizado el proceso de simulación de inicio de sesión, se procede a realizar la simulación del proceso de finalización de sesión.	

Prerrequisitos:

El usuario debe haber iniciado sesión.

Entrada/Pasos de Ejecución:

- 1) Iniciar sesión en la banca electrónica.
- 2) Dar click en el link "Cerrar Sesión" en la parte superior derecha de la pantalla principal del cliente o administrador.

Resultado Esperado:

- La sesión del usuario será eliminada y la pantalla en la que se encuentra será redirigida a la pantalla de inicio de sesión. Al finalizar la sesión el usuario no podrá regresar a la pantalla principal mientras realice el proceso de inicio de sesión.

Resultado Simulación: Buena.

El usuario realizó el proceso de inicio de sesión y procedió a finalizar sesión, la pantalla principal fue redireccionada a la pantalla de inicio de sesión.

No se pudo comprobar la sesión que la sesión haya sido eliminada.

Demostración:

- 1) Iniciar sesión en la banca electrónica.

The screenshot shows the LIX BANCA PERSONAL interface. At the top right, there is a link labeled "Cerrar Sesión". Below the header, the account details are displayed in a table format:

NÚMERO CUENTA	NOMBRE CLIENTE	SALDO	ESTADO	TIPO
12345678905	Galo Villamagua	80	A	Ahorros

On the left side, there are several menu items: "Actualización de Datos", "Posición Consolidada", "Transferencias", and "Movimientos".

- 2) Dar click en el link "Cerrar Sesión" en la parte superior derecha de la pantalla principal del cliente o administrador.

This screenshot is identical to the previous one, but the "Cerrar Sesión" link in the top right corner is highlighted with a red rectangular box, indicating the action to be performed.

Se produce redirección a la GUI de inicio de sesión.

INGRESO BANCA PERSONAL

Ingrese sus datos para iniciar sesión

Usuario:
GaloVillamagua

Contraseña:
.....

Ingresar

[¿Recordar su usuario?](#)
[¿Olvidó su contraseña?](#)

¡RECUERDA que el número máximo de intentos fallidos es TRES, luego tu cuenta será bloqueada y deberás acercarte a la oficina del banco más cercana!

Si no estas registrado haz click [aquí](#)

3.2.4. CASO DE SIMULACIÓN #04 “ACTUALIZACIÓN DE DATOS DE AUTENTICACIÓN”

CASO DE SIMULACIÓN	
Número: 4	Historia de usuario: 4 – Actualización de Datos de Autenticación.
Nombre Caso de Simulación: Actualización de datos de autenticación.	
Descripción: El usuario procede a actualizar sus datos con los que realiza el proceso de autenticación (inicio de sesión).	
Prerrequisitos: El usuario debe estar registrado en la banca electrónica.	
Entrada/Pasos de Ejecución:	

- 1) Iniciar sesión en la banca electrónica.
- 2) En el menú lateral izquierdo, en la opción datos de autenticación damos click.
- 3) Ingresamos/Actualizamos la nueva información (preguntas, imagen y/o contraseña).
- 4) Finalmente hacemos click en el botón “Actualizar”.

Resultado Esperado:

- La información de autenticación es actualizada en la base de datos y la nueva información puede ser vista.

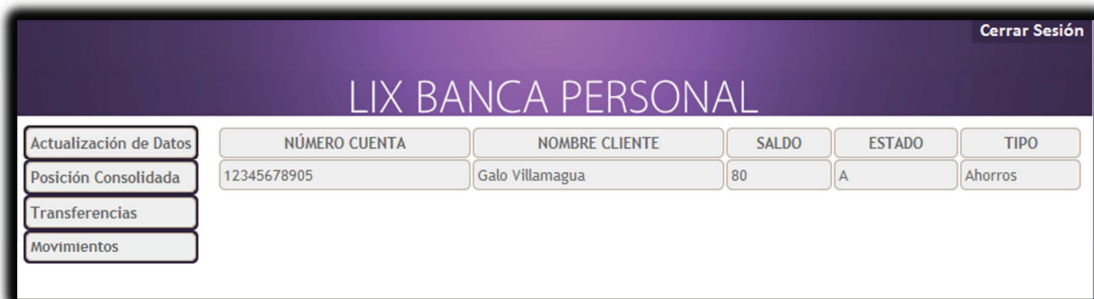
Resultado Simulación: Excelente

Se pudo realizar la actualización de la información de autenticación.

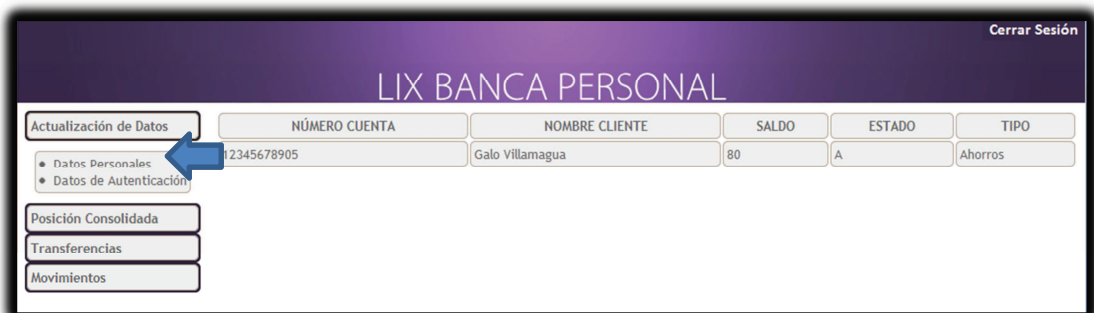
Al desplegar la nueva información, ésta ya reflejó los cambios realizados.

Demostración:

- 1) Iniciar sesión en la banca electrónica.



- 2) En el menú lateral izquierdo, en la opción datos de autenticación damos click.



- 3) Ingresamos/Actualizamos la nueva información (preguntas, imagen y/o contraseña).

LIX BANCA PERSONAL

Conteste al menos 3 preguntas

¿Cuál es el sabor de su helado favorito?

¿Cuál es su deporte favorito?

¿Cuál es su cantante favorito?


¿Año de nacimiento de su madre?

¿Cuál es su bebida favorita?

¿Cuál es su mascota favorita?

"Ingrese una frase propia"

Seleccione una Imagen




Actualización Contraseña

Contraseña Actual:

Nueva Contraseña:

Reingrese su nueva contraseña:



4) Finalmente hacemos click en el botón “Actualizar”.

3.2.5. CASO DE SIMULACION #05 “ACTUALIZACIÓN DE DATOS PERSONALES”

CASO DE SIMULACIÓN	
Número: 5	Historia de usuario: 5 – Actualización de datos personales.
Nombre Caso de Simulación: Actualización de datos personales.	
Descripción: El usuario procede a actualizar sus datos personales como dirección, número de celular, ciudad, etc.	
Prerrequisitos: El usuario debe estar registrado en la banca electrónica.	
Entrada/Pasos de Ejecución:	

- 1) Iniciar sesión en la banca electrónica.
- 2) En el menú lateral izquierdo, en la opción datos de autenticación damos click.
- 3) Ingresamos/Actualizamos la nueva información (dirección, número celular, ciudad, etc).
- 4) Finalmente hacemos click en el botón “Actualizar”.

Resultado Esperado:

- La información personal es actualizada en la base de datos y la nueva información puede ser vista.

Resultado Simulación: Excelente

Se pudo realizar la actualización de la información personal del usuario.

Al desplegar la nueva información, ésta ya reflejó los cambios realizados.

Demostración:

- 1) Iniciar sesión en la banca electrónica.



- 2) En el menú lateral izquierdo, en la opción datos de autenticación damos click.



- 3) Ingresamos/Actualizamos la nueva información (dirección, número celular, ciudad, etc).

LIX BANCA PERSONAL

Ingrese/Actualice la siguiente información

Identificación:

Nombre:


Email:

Ciudad:

Dirección:

Móvil:

Fijo:



4) Finalmente hacemos click en el botón “Actualizar”.

Se emite el siguiente mensaje.

LIX BANCA PERSONAL

3.2.6. CASO DE SIMULACIÓN #06 “BLOQUEAR USUARIO”

CASO DE SIMULACIÓN	
Número: 6	Historia de usuario: 6 – Bloquear Usuario.
Nombre Caso de Simulación: Bloquear Usuario.	
Descripción: Un administrador podrá bloquear un usuario mediante permiso expreso del mismo, cuando instituciones gubernamentales o el banco lo decida.	
Prerrequisitos: El usuario debe estar registrado en la banca electrónica. El usuario debe estar desbloqueado.	

Entrada/Pasos de Ejecución:

- 1) Iniciar sesión en la banca electrónica como administrador.
- 2) En el menú lateral izquierdo, damos click en la opción “bloquear usuario”.
- 3) Ingresamos la cédula de identidad del cliente
- 4) Damos click en el botón “Bloquear”.

Resultado Esperado:

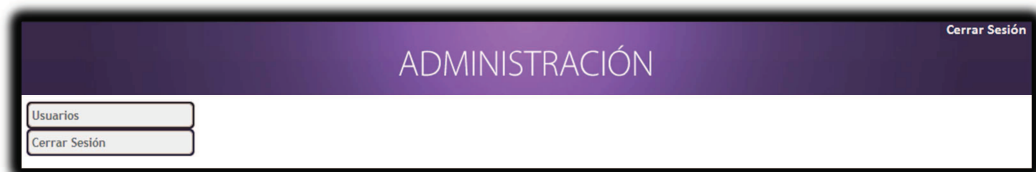
- El usuario no podrá acceder a la banca electrónica.

Resultado Simulación: Excelente

El usuario no puede iniciar sesión ya que ha sido bloqueado.

Demostración:

- 1) Iniciar sesión en la banca electrónica como administrador.



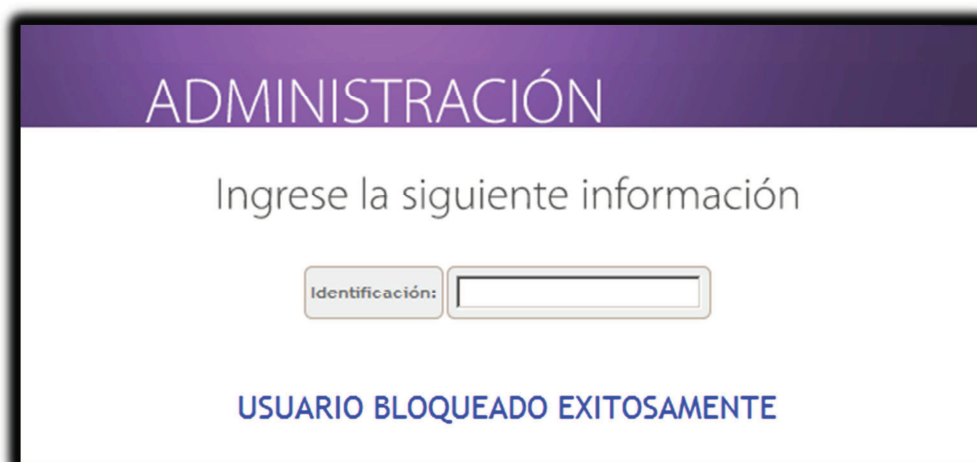
- 2) En el menú lateral izquierdo, damos click en la opción “bloquear usuario”.



- 3) Ingresamos la cédula de identidad del cliente.

 A screenshot of the 'ADMINISTRACIÓN' interface showing a form titled 'Ingrese la siguiente información'. The form includes an 'Identificación:' field with the value '1104698636'. Below this are five input fields for 'NOMBRE', 'CIUDAD', 'DIRECCIÓN', 'FECHA AFILIACIÓN', and 'ESTADO'. The values entered are 'Adriana Aguirre', 'Quito', 'Martha', '2012-12-02 22:15:05.0', and 'A' respectively. At the bottom of the form is a 'BLOQUEAR' button.

- 4) Damos click en el botón “Bloquear”.
El siguiente mensaje es desplegado.



3.2.7. CASO DE SIMULACIÓN #07 “DESBLOQUEAR USUARIO”

CASO DE SIMULACIÓN	
Número: 7	Historia de usuario: 7 – Desbloquear Usuario.
Nombre Caso de Simulación: Desbloquear Usuario.	
Descripción: Un administrador podrá bloquear un usuario mediante permiso expreso del mismo.	
Prerrequisitos: El usuario debe estar registrado en la banca electrónica. El usuario debe estar bloqueado.	
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1) Iniciar sesión en la banca electrónica como administrador. 2) En el menú lateral izquierdo, damos click en la opción “desbloquear usuario”. 3) Ingresamos la cédula de identidad del cliente 4) Damos click en el botón “Desbloquear”. 	
Resultado Esperado: <ul style="list-style-type: none"> • El usuario podrá acceder a la banca electrónica. 	

Resultado Simulación: Excelente

El usuario puede iniciar sesión ya que ha sido bloqueado.

Demostración:

- 1) Iniciar sesión en la banca electrónica como administrador.



- 2) En el menú lateral izquierdo, damos click en la opción “desbloquear usuario”.



- 3) Ingresamos la cédula de identidad del cliente.

 A screenshot of the web application interface showing a form titled 'Ingrese la siguiente información'. The form has a field labeled 'Identificación:' with the value '1104698636'. Below this are five fields: 'NOMBRE', 'CIUDAD', 'DIRECCIÓN', 'FECHA AFILIACIÓN', and 'ESTADO'. The values for these fields are: 'Adriana Aguirre', 'Quito', 'Martha', '2012-12-02 22:17:17.0', and 'B'. At the bottom of the form is a button labeled 'DESBLOQUEAR'.

- 4) Damos click en el botón “desbloquear”.
El siguiente mensaje es desplegado.

ADMINISTRACIÓN

Ingrese la siguiente información

Identificación:

USUARIO DESBLOQUEADO EXITOSAMENTE

3.2.8. CASO DE SIMULACIÓN #08 “TRANSFERENCIAS ENTRE CUENTAS PROPIAS”

CASO DE SIMULACIÓN	
Número: 8	Historia de usuario: 8 – Transferencias entre cuentas propias.
Nombre Caso de Simulación: Transferencias entre cuentas propias.	
Descripción: El cliente puede realizar transferencias entre cuentas propias del banco.	
Prerrequisitos: El usuario debe estar registrado en la banca electrónica. El usuario debe iniciar sesión.	
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1) Iniciar sesión en la banca electrónica. 2) En el panel lateral izquierdo seleccionar la opción “Transferencias→Cuentas Propias”. 3) Ingresar la siguiente información: cuenta destino, monto, razón de la transferencia, correo electrónico del beneficiario. Presionar el botón “Aceptar”. 4) Confirmar si la información desplegada en la pantalla es correcta y presionar el botón “Aceptar”. 	
Resultado Esperado: <ul style="list-style-type: none"> • El dinero será debitado y acreditado en las cuentas de origen y destino respectivamente. 	

Resultado Simulación: Excelente

El saldo en dinero en la cuenta origen y destino fueron debitados correctamente.

Demostración:

- 1) Iniciar sesión en la banca electrónica.

The screenshot shows the 'LIX BANCA PERSONAL' interface. At the top right is a 'Cerrar Sesión' button. The main header is 'LIX BANCA PERSONAL'. Below it is a table with account information:

	NÚMERO CUENTA	NOMBRE CLIENTE	SALDO	ESTADO	TIPO
	12345678905	Galo Villamagua	80	A	Ahorros

On the left side, there is a vertical menu with the following options: 'Actualización de Datos', 'Posición Consolidada', 'Transferencias', and 'Movimientos'.

- 2) En el panel lateral izquierdo seleccionar la opción "Transferencias→Cuentas Propias".

This screenshot shows the same interface as the previous one, but with the 'Transferencias' menu expanded. A blue arrow points to the 'Cuentas Propias' option. The expanded menu includes:

- Cuentas Propias
- Bancos Nacionales
- Bancos Internacionales

The rest of the interface, including the account details table and the 'Cerrar Sesión' button, remains the same.

- 3) Ingresar la siguiente información: cuenta destino, monto, razón de la transferencia, correo electrónico del beneficiario. Presionar el botón "Aceptar".

Cerrar Sesión

LIX BANCA PERSONAL

Actualización de Datos

Posición Consolidada

Transferencias

- Cuentas Propias
- Bancos Nacionales
- Bancos Internacionales

Movimientos

Transferencias entre cuentas propias

Cuenta Origen:	<input type="text" value="12345678905"/>
Cuenta Destino:	<input type="text" value="12345678902"/> OK
Monto (Ej: 250.25):	<input type="text" value="10"/>
Razón de la transferencia:	<input type="text" value="Préstamo impresora"/>
Notificar al correo electrónico:	<input type="text" value="n2902v@live.com"/>
Notificar al beneficiario (correo electrónico):	<input type="text" value="politecnica.epn@gmail.com"/>
<input type="button" value="Aceptar"/>	

- 4) Confirmar si la información desplegada en la pantalla es correcta y presionar el botón “Aceptar”.

Cerrar Sesión

LIX BANCA PERSONAL

Actualización de Datos

Posición Consolidada

Transferencias

- Cuentas Propias
- Bancos Nacionales
- Bancos Internacionales

Movimientos

Transferencias entre cuentas propias

Se va a realizar la transferencia con la siguiente información, por favor compruebe que la información esté correcta.

CUENTA ORIGEN: 12345678905
CUENTA DESTINO: 12345678902
BENEFICIARIO: Adriana Aguirre
MONTO: 10
RAZÓN DE TRANSFERENCIA: Préstamo impresora
NOTIFICAR A: n2902v@live.com
NOTIFICAR AL BENEFICIARIO: politecnica.epn@gmail.com
<input type="button" value="Aceptar"/>

El siguiente mensaje es presentado.

Cerrar Sesión

LIX BANCA PERSONAL

Actualización de Datos

Posición Consolidada

Transferencias

- Cuentas Propias
- Bancos Nacionales
- Bancos Internacionales

Movimientos

TRANSFERENCIA EXITOSA

CUENTA ORIGEN: 12345678905
CUENTA DESTINO: 12345678902
BENEFICIARIO: Adriana Aguirre
MONTO: 10
RAZÓN DE TRANSFERENCIA: Préstamo impresora
<input type="button" value="Aceptar"/>

El saldo en la cuenta origen ha sido debitado.

LIX BANCA PERSONAL					Cerrar Sesión
NÚMERO CUENTA	NOMBRE CLIENTE	SALDO	ESTADO	TIPO	
12345678905	Galo Villamagua	70	A	Ahorros	

3.2.9. CASO DE SIMULACIÓN #09 “RECUPERAR CONTRASEÑA”

CASO DE SIMULACIÓN	
Número: 9	Historia de usuario: 9 – Recuperar Contraseña.
Nombre Caso de Simulación: Recuperar Contraseña.	
Descripción: El cliente tiene la opción de recuperar su contraseña de acceso en caso de no recordarla o desee cambiarla por motivos de seguridad.	
Prerrequisitos: El usuario debe estar registrado en la banca electrónica.	
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1) En la pantalla de inicio de sesión, seleccionar la opción “¿olvidó su contraseña?” 2) Ingresar número de tarjeta de débito y contraseña. 3) Responder a la pregunta y seleccionar la imagen de desafío. 4) Ingresar una nueva contraseña y presionar el botón “Aceptar”. 	
Resultado Esperado: <ul style="list-style-type: none"> • El usuario cambia su contraseña actual por una nueva contraseña. 	
Resultado Simulación: Excelente Mediante dos niveles de seguridad (clave tarjeta débito y desafío de pregunta e imagen) el usuario pudo cambiar su contraseña.	

Demostración:

- 1) En la pantalla de inicio de sesión, seleccionar la opción “¿olvidó su contraseña?”

INGRESO BANCA PERSONAL

Ingrese sus datos para iniciar sesión

Usuario:

Contraseña:

[¿Recordar su usuario?](#)
[¿Olvidó su contraseña?](#)

¡RECUERDA que el número máximo de intentos fallidos es TRES, luego tu cuenta será bloqueada y deberás acercarte a la oficina del banco más cercana!

Si no estas registrado haz click [aquí](#)

2) Ingresar número de tarjeta de débito y contraseña.

RESTAURACIÓN DE CONTRASEÑA

Ingrese sus datos

Número de Tarjeta de Débito:

Contraseña:


3) Responder a la pregunta y seleccionar la imagen de desafío.

PREGUNTAS DE VERIFICACIÓN

Consteste la pregunta de verificación

¿Cuál es su deporte favorito?

Seleccione una Imagen











4) Ingresar una nueva contraseña y presionar el botón “Aceptar”.

Se redirecciona a la GUI principal de inicio de sesión.

3.2.10. CASO DE SIMULACIÓN #10 “RECORDAR USUARIO DE BANCA ELECTRÓNICA”

CASO DE SIMULACIÓN	
Número: 10	Historia de usuario: 10 – Recordar usuario de banca electrónica.
Nombre Caso de Simulación:	

Recordar usuario de banca electrónica.
Descripción: El cliente puede recordar su usuario en caso de no recordarlo.
Prerrequisitos: El usuario debe estar registrado en la banca electrónica.
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1) En la pantalla de inicio de sesión, en la parte inferior seleccionar la opción “recordar usuario”. 2) Ingresar la dirección de correo electrónico con la que se registró el cliente. 3) Presionar el botón “Recordar Usuario”
Resultado Esperado: <ul style="list-style-type: none"> • Un SMS y correo electrónico serán enviados al cliente, este contiene el usuario con el que se ha registrado en la banca electrónica.
Resultado Simulación: Excelente El usuario ha sido receptado mediante SMS y correo electrónico.

Demostración:

- 1) En la pantalla de inicio de sesión, en la parte inferior seleccionar la opción “Recordar su usuario”.

INGRESO BANCA PERSONAL

Ingrese sus datos para iniciar sesión

Usuario:

Contraseña:

[¿Recordar su usuario?](#) 

[¿Olvidó su contraseña?](#)

¡RECUERDA que el número máximo de intentos fallidos es TRES, luego tu cuenta será bloqueada y deberás acercarte a la oficina del banco más cercana!

Si no estas registrado haz click [aquí](#)

- 2) Ingresar la dirección de correo electrónico con la que se registró el cliente.
Presionar el botón “Recordar Usuario”.



Se redirecciona a la GUI de inicio de sesión.

aquí'. At the bottom, there is a green message: '¡Se ha enviado un email y un SMS con su usuario de la BANCA LIX!'." data-bbox="211 356 799 690"/>

3.2.11. CASO DE SIMULACIÓN #11 “CREAR NUEVO USUARIO ADMINISTRADOR”

CASO DE SIMULACIÓN	
Número: 11	Historia de usuario: 11 – Crear nuevo usuario administrador.
Nombre Caso de Simulación:	

Crear nuevo usuario administrador.
Descripción: Nuevos administradores pueden ser creados por el usuario administrador.
Prerrequisitos: El usuario debe ser administrador.
Entrada/Pasos de Ejecución: <ol style="list-style-type: none"> 1) Iniciar sesión como usuario administrador. 2) En el panel lateral izquierdo de la página principal, seleccionar la opción “Nuevo Usuario”. 3) Ingresar la identificación del nuevo usuario. 4) Ingresar el nombre del nuevo usuario. 5) Dar click en el botón “Crear usuario”.
Resultado Esperado: <ul style="list-style-type: none"> • El nuevo usuario debe estar registrado en la base de datos y debe permitir autenticarse, la primera vez debe pedir cambio de clave.
Resultado Simulación: Excelente El nuevo usuario fue reflejado en la base de datos. La primera vez el usuario debió establecer una nueva contraseña. El usuario puede iniciar sesión correctamente.

Demostración:

- 1) Iniciar sesión como usuario administrador.



- 2) En el panel lateral izquierdo de la página principal, seleccionar la opción “Nuevo Usuario”.



3) Ingresar la identificación y el nombre del nuevo usuario.

The screenshot shows the 'ADMINISTRACIÓN' header. Below it, the text 'Ingrese la siguiente información' is centered. There are two input fields: 'Identificación:' with the value '1104698639' and 'Nombre:' with the value 'Paulina Villamagua'. Below the fields is a button labeled 'Crear Usuario'.

4) Dar click en el botón "Crear usuario".

The screenshot shows the 'ADMINISTRACIÓN' header. Below it, the text 'Ingrese la siguiente información' is centered. There are two empty input fields: 'Identificación:' and 'Nombre:'. Below the fields is a button labeled 'Crear Usuario'. At the bottom of the page, the text 'USUARIO CREADO EXITOSAMENTE' is displayed in green.

3.3. ANÁLISIS DE RESULTADOS.

Luego de realizar la simulación, procedemos a analizar los resultados obtenidos.

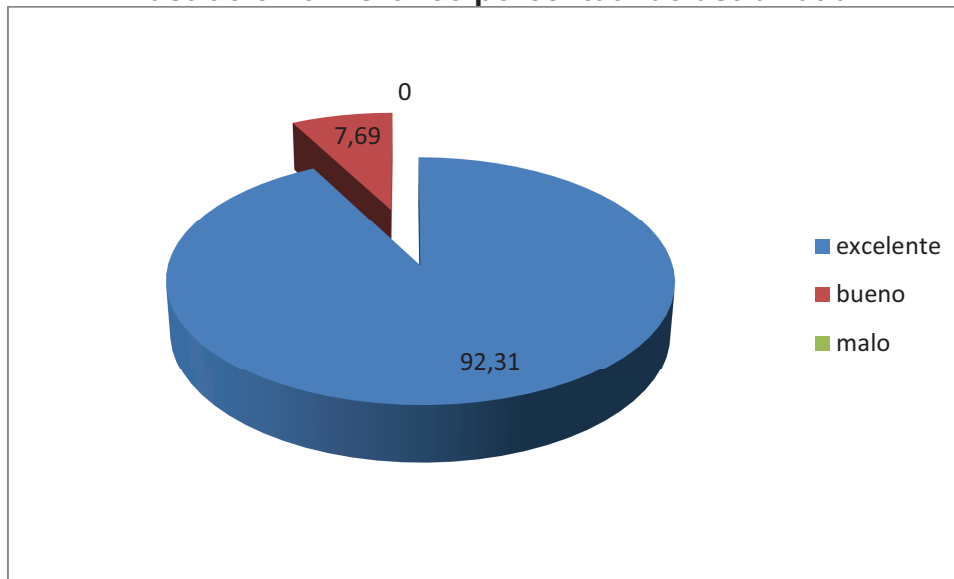
3.3.1. ANÁLISIS DE FUNCIONALIDAD

De acuerdo a los resultados obtenidos en la simulación procedemos a realizar la siguiente tabla, en ella catalogamos el resultado de las simulaciones en Excelente, Bueno y Malo.

	Excelente	Bueno	Malo
Registrar Cliente	X		
Iniciar Sesión	X		
Finalización de Sesión		X	
Actualización de datos de autenticación	X		
Actualización de datos personales	X		
Bloquear Usuario	X		
Desbloquear Usuario	X		
Transferencia Bancaria	X		
Recuperar contraseña	X		
Recordar Usuario	X		
Nuevo Usuario Administrador	X		
Envío de SMS	X		
Envío de Correo Electrónico	X		

De un total de 13 tareas, 12 estuvieron excelentes y 1 en una tarea el resultado fue bueno, realizando un ponderado tenemos el siguiente gráfico.

Ilustración 3.2 Gráfico porcentual de usabilidad



Elaborado por: Gabriel Ignacio Villamagua Armijos

Como podemos ver en el gráfico 92,31% de las funcionalidades estuvieron excelentes, el 7,69% funcionaron bueno y un 0% malo. El 7.69% se debe a que la funcionalidad “finalizar sesión” no permite verificar si la variable **session** del apache tomcat se ha eliminado.

3.3.2. ANÁLISIS DE USABILIDAD

La usabilidad es la facilidad con la que usuarios utilizan el software bancario, para esto se ha tomado una muestra de 20 personas que utilicen la banca, de esta manera se pueden simular en un ambiente real.

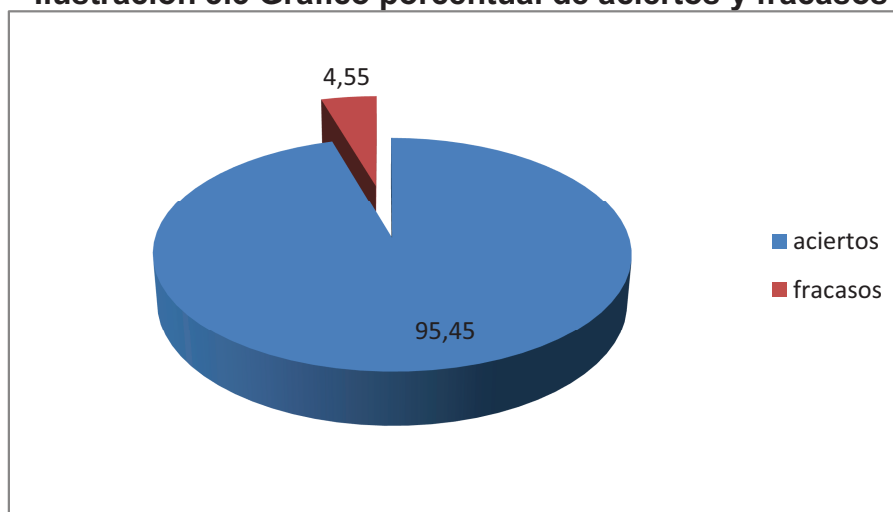
Se han tomado 5 indicadores que nos servirán para medir el nivel de usabilidad que tiene el sistema.

Ratio de compleción de tarea: el resultado final de las tareas asignadas a los usuarios fueron los siguientes.

	# Éxitos	# Fracayos
Registrar Cliente	18	2
Iniciar Sesión	20	4
Finalización de Sesión	20	0
Actualización de datos de autenticación	17	2
Actualización de datos personales	18	2
Bloquear Usuario	20	0
Desbloquear Usuario	20	0
Transferencia Bancaria	19	0
Recuperar contraseña	17	0
Recordar Usuario	20	0
Nuevo Usuario Administrador	20	0
TOTAL	210	10

Como resultado de la tabla anterior tenemos: 210 aciertos y 10 fracasos.

Ilustración 3.3 Gráfico porcentual de aciertos y fracasos



Elaborado por: Gabriel Ignacio Villamagua Armijos

Visualizando el gráfico ponderado tenemos 95.45% de aciertos y un 4.55% de fracasos, el valor ponderado de fracasos se debió a errores humanos entre los cuales tenemos: mal ingreso de los datos personales y de autenticación y una vez

realizado el registro en el software bancario el recordar los datos ingresados fue el principal problema.

Tiempo para completar las tareas: el tiempo utilizado para completar la tarea más larga en promedio fue de 4 minutos, la cual corresponde al proceso de registro ya que este proceso conlleva que se actualice o ingrese información nueva al sistema. La tarea más corta en tiempo fue “recordar usuario” y tomó en promedio 36 segundos desde que se inició el proceso hasta que se recibió el SMS y el correo electrónico.

	Tiempo MIN. (minutos)	Tiempo MAX (minutos)	Promedio (minutos)
Registrar Cliente	1.5	4.5	3
Iniciar Sesión	0.6	2	1.3
Finalización de Sesión	0.1	0.1	0.1
Actualización de datos de autenticación	1	3.5	2.25
Actualización de datos personales	1	3	2
Bloquear Usuario	0.5	1	0.75
Desbloquear Usuario	0.5	1	0.75
Transferencia Bancaria	1	3	2
Recuperar contraseña	1	3	2
Recordar Usuario	0.6	0.7	0.65
Nuevo Usuario Administrador	0.6	1	0.8
PROMEDIO	0.37	1.27	0.82

El tiempo total para realizar todas las tareas mencionadas fue de 15 minutos un aproximado de 0.82 minutos por cada tarea, hay que tomar en cuenta que existen tareas que se realizan cada vez que se utiliza a la banca electrónica como el proceso de autenticación (inicio de sesión), lo que hace que el usuario tome experiencia y le tome cada vez menos tiempo para realizarla.

Las tareas con mayor duración en realizarse por parte del usuario se debieron a que hubo diferentes niveles de experticia en manejo de computadoras y navegación por internet por parte de los usuarios, algunos eran expertos, otros tenían nivel intermedio y finalmente algunos solamente habían utilizado alguna vez un procesador de palabras.

Dificultad de la tarea: las tareas más complicadas para los usuarios son “registrarse” y “actualizar datos de autenticación” ya que el proceso conlleva que se validen los datos, y de los cuales los datos correspondientes a autenticación deben ser recordados con exactitud.

Promedio de calificaciones de dificultad de realización de una tarea

(1 fácil 5 muy difícil)	Calificación
Registrar Cliente	4
Iniciar Sesión	1
Finalización de Sesión	1
Actualización de datos de autenticación	4
Actualización de datos personales	2
Bloquear Usuario	1
Desbloquear Usuario	1
Transferencia Bancaria	2
Recuperar contraseña	4
Recordar Usuario	1
Nuevo Usuario Administrador	1
Promedio Final	2

Como podemos visualizar el promedio de dificultad de uso del software es 2, lo que significa que el software es fácil de utilizar.

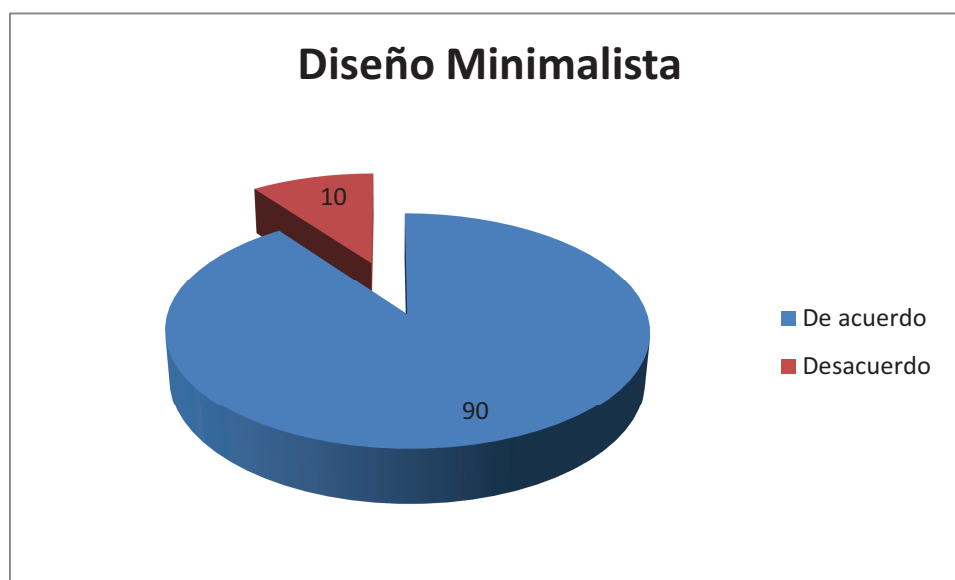
Errores: se registró un error, ahora ya corregido que corresponde a la validación de correo electrónico ya que no admitía direcciones como gvillamaqua@correo.com.ec es decir con doble dominio (com.ec).

Expectativas: los usuarios esperaron un sistema de autenticación con una banca electrónica completa en su totalidad, pero cabe mencionar que el software es un prototipo.

Diseño Visual: con respecto al diseño visual tenemos los siguientes resultados:

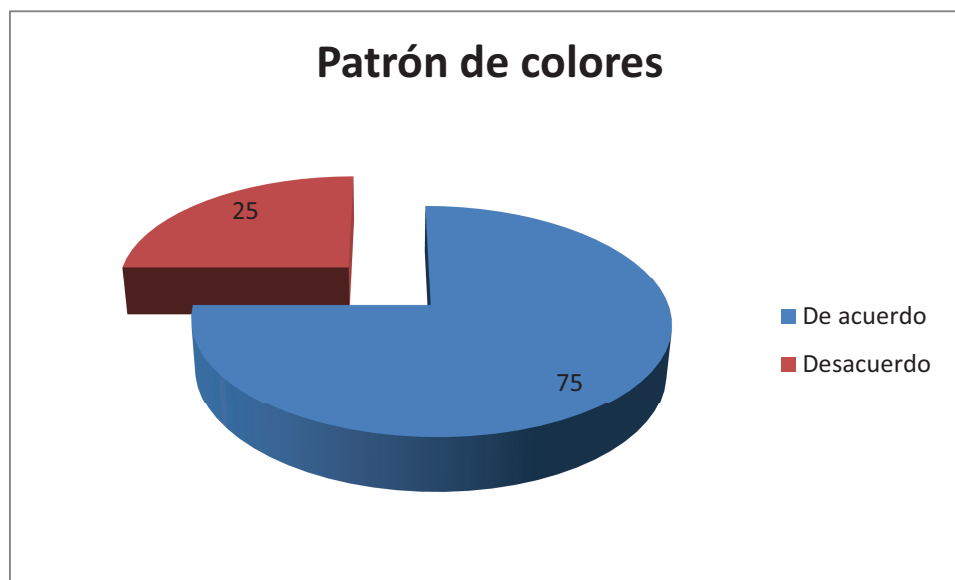
	De acuerdo	Desacuerdo
Diseño Minimalista	18	2
Patrón de colores	15	5

Ilustración 3.4 Gráfico porcentual de diseño minimalista



Elaborado por: Gabriel Ignacio Villamagua Armijos

Ilustración 3.5 Gráfico porcentual de patrón de colores



Elaborado por: Gabriel Ignacio Villamagua Armijos

El diseño minimalista fue acogido por el 90% de los usuarios, pero el patrón de colores primarios (negro, púrpura y blanco) no tuvo mucha acogida solamente un 75% lo hizo, se recomendó cambiar el color púrpura por otro color no tan fuerte como amarillo o azul.

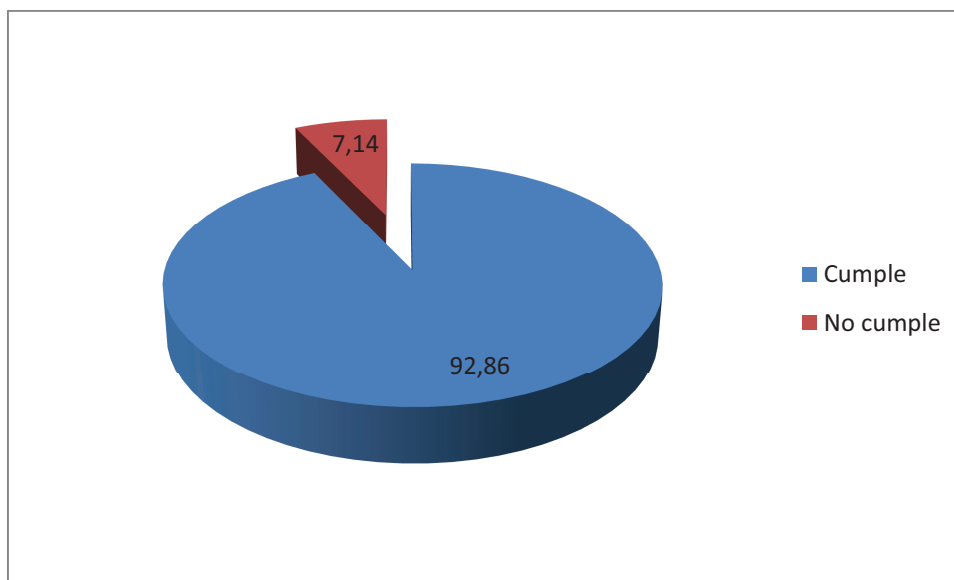
3.3.3. ANÁLISIS DE SEGURIDAD

Se ha realizado una tabla con características relevantes que debe tener un sistema de software de aplicaciones bancarias.

	Sí	No
Encriptación Contraseñas en BDD		X
Privilegios por Usuario	X	
Seguridad SQL	X	
Borrado Cookies	X	
Borrado Variables Session	X	

Tiempo Inactividad	X	
Finalización de Sesión	X	
Bloqueo Usuarios	X	
Políticas Contraseñas	X	
Políticas Usuarios	X	
Seguridad Autenticación	X	
Métodos POST	X	
Doble Autenticación	X	
Desactivación de autocompletar	X	

Ilustración 3.6 Gráfico porcentual de cumplimiento de seguridad



Elaborado por: Gabriel Ignacio Villamagua Armijos

El prototipo del sistema de autenticación cumple con un nivel de seguridad aproximada del 93%, teniendo deficiencia en la encriptación a nivel de BDD. Cabe señalar que este valor porcentual es en base al análisis de vulnerabilidades mencionadas en la tabla 1.3 del capítulo 1.

En el desarrollo de aplicaciones se han tomado cuenta vulnerabilidades que se pueden proteger a nivel de lenguaje de programación y se han intentado

minimizar, pero debemos considerar que la mayoría de las vulnerabilidades analizadas en el capítulo 1 corresponden al sistema operativo, red y falta de políticas de seguridad, por lo que para que el riesgo de robo o pérdida de información sea menor se deben implementar controles como los especificados anteriormente en la tabla 1.3.

CAPÍTULO 4.

CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

Luego del proceso de desarrollo, simulación y análisis del software desarrollado “Banca Electrónica Lix” podemos concluir:

- El riesgo de sufrir ataques por inyección SQL se minimizó al aplicar correctamente el patrón de desarrollo MVC (Model-View-Controller), ya que en una de sus especificaciones se menciona que en la vista (front-end) no deben existir conexiones con usuarios y/o contraseñas ya que son procesadas a nivel de cliente y por lo tanto visibles al usuario final.
- La doble autenticación no permitió que un intruso pueda entrar a la banca electrónica LIX ya que aunque sepa la contraseña de inicio de sesión, el desafío de la imagen y pregunta logró que no se pueda iniciar.
- Se evitó el ataque por diccionario de datos, ya que el intento de autenticación errónea por tercera vez bloquea automáticamente al usuario al ingreso de la banca electrónica.
- Se comprobó que el uso del software mejoró la manera de realizar las transacciones de los usuarios de manera segura, el tiempo de cada transacción es muy corto en relación al tiempo que toma en acercarse a una agencia del banco.

El utilizar SCRUM como metodología de desarrollo permite concluir que:

- SCRUM ayudó de manera exitosa al manager a realizar la gestión del proyecto al definir correctamente la pila y los sprints del sistema y asignar correctamente los recursos.
- Se realizó un correcto seguimiento de las tareas asignadas a cada recurso mediante las reuniones diarias, al poder identificar los obstáculos que pueden impedir que una tarea se complete de manera exitosa.

- Se comprobó que SCRUM a excepción de las metodologías tradicionales permite que en la marcha se puedan ingresar nuevos requerimientos, además de poder realizar planes de contingencia debido al principio de la flexibilidad.

En cuestión del servidor de aplicaciones web y el lenguaje de desarrollo utilizado podemos concluir:

- El uso de java y Apache Tomcat permitió que la aplicación bancaria sea independiente de la plataforma en la que se ejecuta, esto porque JAVA utiliza su bytecode y Tomcat está desarrollado con JAVA. Nuestra aplicación se encuentra actualmente corriendo sobre Ubuntu una distribución GNU/Linux.
- Se comprobó que la aplicación puede crecer de acuerdo al ritmo del banco, esto es debido al nivel de escalabilidad que nos proporciona el lenguaje java.

4.2. RECOMENDACIONES

Aunque la aplicación web bancaria es altamente segura y su funcionalidad es correcta, no queda demás realizar las siguientes recomendaciones:

- La aplicación web bancaria es un prototipo y no puede llegar a ser un producto final, se recomienda que al decidir implementar el sistema en un ambiente de producción se utilice una versión final estable.
- Es recomendable utilizar un balanceador de carga para distribuir las peticiones desde internet hacia múltiples servidores, de esa manera el tiempo de respuesta será menor.
- En cuanto a configuraciones, es recomendable que no se utilicen directorios predeterminados por Apache Tomcat ya que esto hace que el sistema sea más vulnerable, debido a que el primer paso de un atacante sería intentar obtener privilegios en esos directorios.

- Es recomendable desactivar módulos innecesarios en el servidor web, ya que hace que se consuman mayor número de recursos en vano.
- Configurar el servidor para que emita información en el log de la menor manera posible, ya que la excesiva información da guías a los atacantes de cómo está conformada la aplicación web, además de que los logs pueden llegar a saturar la memoria produciéndose DOS.
- Implementar seguridades como cortafuegos, que bloqueen el tráfico entrante y saliente no correspondiente a llamadas http, https y llamadas en el puerto 25.
- Implementar usuarios con diferentes privilegios de acuerdo a su rol y no permitir el acceso anónimo a la aplicación bancaria, para que información sensible de los usuarios no pueda ser comprometida.

GLOSARIO

AJAX

Asynchronous JavaScript And XML

BDD

Base de Datos

DBMS

Database Management System

CSS

Cascading Style Sheets

DOM

DocumentObjectModel

HTML

HyperTextMarkupLanguage

IDE

IntegratedDevelopmentEnviroment

OOP

Oriented Object Programming

GUI

Graphical User Interface

OWASP

Open Web Application Security Project

MVC

Model View Controller

JSP

JavaServer Page

HTTP

Hypertext Transfer Protocol

SPRINT

Oriented Object Programming

SQL

Structured Query Language

SMS

Short Message Service

JDK

Java Development Kit

REFERENCIAS

- [1] World Wide Web Consortium, "World Wide Web Consortium," Octubre 2012. [Online]. Available: <http://www.w3.org/TR/html5/>. [Accessed Diciembre 2012].
- [2] Wikipedia, "Wikipedia," Octubre 2012. [Online]. Available: http://es.wikipedia.org/wiki/Hojas_de_estilo_en_cascada#CSS3. [Accessed Diciembre 2012].
- [3] Wikipedia, "Wikipedia," Octubre 2012. [Online]. Available: http://es.wikipedia.org/wiki/Modelo_de_base_de_datos. [Accessed Diciembre 2012].
- [4] DesarrolladoresWeb.net, "DesarrolladoresWeb.net," Noviembre 2011. [Online]. Available: http://www.desarrolloweb.com/de_interes/ranking-lenguajes-programacion-noviembre-2011-6170.html. [Accessed Noviembre 2012].
- [5] COBISCORP SA, "Cobiscorp Banking Technology Partners," Agosto 2012. [Online]. Available: <http://www.cobiscorp.com/SolucionesInform%C3%A1ticas/COBISCoreBanking/tabid/121/Default.aspx>. [Accessed Noviembre 2012].
- [6] TCS, "AGILITY EXPLAINED IN TRANSACTION BANKING," [Online]. Available: http://www.tcs.com/offerings/bancs/banking/core_banking/Pages/default.aspx.
- [7] Microsoft, "Produbanco logra manejar todas sus transacciones de una forma transparente e independiente, gracias a la implementación de su nuevo aplicativo bancario "Prometeus" basado en Windows Server System," [Online]. Available: <http://www.microsoft.com/ecuador/casosdeexito/produbanco2.aspx>. [Accessed Diciembre 2012].
- [8] SANS Institute, "CSIS: 20 Critical Security Controls Version 4.0" [Online]. Available: <http://www.sans.org/critical-security-controls/> [Accessed Diciembre 2012]
- [9] Eclipse Foundation, "Eclipse IDE," [Online]. Available: <http://www.eclipse.org/downloads/index-developer.php>. [Accessed Noviembre 2012].

- [1 Wikipedia, "Wikipedia," Octubre 2012. [Online]. Available:
0] <http://es.wikipedia.org/wiki/Tomcat>. [Accessed Diciembre 2012].
- [1 Wikipedia, "Catalina Tomcat," 2012. [Online]. Available:
1] http://es.wikipedia.org/wiki/Tomcat_Catalina.
- [1 Wikipedia, "Apache Tomcat," Diciembre 2012. [Online]. Available:
2] http://en.wikipedia.org/wiki/Apache_Tomcat. [Accessed Diciembre 2012].
- [1 P. Juan, in *Flexibilidad con Scrum*, 2008.
3]
- [1 P. Juan, "El modelo scrum," in *Flexibilidad con Scrum*, 2008.
4]
- [1 J. Revelo, "Desarrollo prototipo de juego 3D (utilizando SCRUM)," in *Tesis*:
5] http://www.navegapolis.net/files/Flexibilidad_con_Scrum.pdf, Quito, 2012.

ANEXOS

Los anexos que se numeran a continuación se encuentran en el disco adjunto.

ANEXO 1 – Manual de Instalación (Caso de Estudio)

ANEXO 2 – Manual de Usuario (Caso de Estudio)