

ESCUELA DE INGENIERÍA EN SISTEMAS

Tema:

"DESARROLLO DE UNA APLICACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO SAN FRANCISCO LIMITADA USANDO LA TECNOLOGÍA OBJECTIVE C, PARA EL DISPOSITIVO MÓVIL IPHONE EN EL PERÍODO 2011 - 2012"

Disertación de grado previo a la obtención del título de Ingeniero de Sistemas y Computación

Línea de Investigación:

Implementación de soluciones para ambientes Windows, Web y Móviles.

Autor:

EDWIN FABRICIO MUÑOZ DÍAZ

Asesor:

MG. VERÓNICA MARIBEL PAILIACHO MENA

Ambato – Ecuador

Abril – 2015

PONTIFICIA UNIVERSIDAD CATOLICA DEL ECUADOR
SEDE AMBATO

HOJA DE APROBACIÓN

Tema:

"DESARROLLO DE UNA APLICACIÓN DE BANCA MÓVIL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO SAN FRANCISCO LIMITADA USANDO LA TECNOLOGÍA OBJECTIVE C, PARA EL DISPOSITIVO MÓVIL IPHONE EN EL PERÍODO 2011 - 2012"

Línea de Investigación:

Implementación de soluciones para ambientes Windows, Web y Móviles.

Autor:

EDWIN FABRICIO MUÑOZ DÍAZ

Verónica Maribel Pailiacho Mena, Mg. **f.** _____
CALIFICADORA

Enrique Xavier Garcés Freire, Msc. . **f.** _____
CALIFICADOR

Teresa Milena Freire Aillón, Mg. **f.** _____
CALIFICADORA

Galo Mauricio López Sevilla, Msc. **f.** _____
DIRECTOR DE LA ESCUELA DE INGENIERIA EN SISTEMAS

Hugo Rogelio Altamirano Villarroel, Dr. **f.** _____
SECRETARIO GENERAL PUCESA

Ambato – Ecuador

Abril – 2015

DECLARACIÓN DE AUTENTICIDAD
Y RESPONSABILIDAD

Yo, Edwin Fabricio Muñoz Díaz portador de la cédula de ciudadanía N°. 180300578-2 declaro que los resultados obtenidos en la investigación que presento como informe final, previo a la obtención del título de Ingeniero de Sistemas y Computación son absolutamente originales, auténticos y personales.

En tal virtud, declaro que el contenido, las conclusiones y los efectos legales y académicos que se desprenden del trabajo propuesto de investigación y luego de la redacción de este documento son y serán de mi sola y exclusiva responsabilidad legal y académica.

Edwin Fabricio Muñoz Díaz
CI. 180300578-2

Dedicatoria

A Dios por haberme permitido llegar hasta esta meta de mi vida y haberme dado salud para lograr mis objetivos, además de su infinita bondad amor y amparo en toda mi carrera estudiantil.

Agradecimiento

A Dios por toda la ayuda incondicional que me ha sabido brindar al momento de cumplir las debidas metas que me he propuesto en mi vida.

A toda mi familia que siempre estuvo pendiente en todo mi lapso estudiantil y supo ser partícipe para la culminación del mismo.

Al Ing. Diego Torres por ser un valioso amigo, ejemplo y un apoyo incondicional al momento de realizar este proyecto.

A mi asesora Ing. Verónica Pailiacho por su tiempo, preocupación y buenas enseñanzas en el periodo universitario.

RESUMEN

El objetivo del proyecto es el desarrollo una aplicación de banca móvil para la cooperativa de ahorro y crédito San Francisco Limitada usando la tecnología objective C para el dispositivo móvil Iphone; debido a que esta prestigiosa institución carece de este servicio. La metodología para el análisis que se empleó es cualitativa mediante observaciones y entrevistas e investigación bibliográfica y de campo. Para la fabricación se implementó la metodología de desarrollo UML; llegando al resultado de crear una aplicación denominada SFPUCEmóvil, que permite realizar consultas confidenciales bancarias de un cliente que tenga cuenta de ahorros en mencionada entidad. Como conclusión las aplicaciones móviles en el ámbito financiero son el negocio del hoy y del mañana, innovando y mejorando las políticas de competencia en el área financiera y el mercado bancario, por lo tanto SFPUCEmóvil, pone a disposición la consulta de información bancaria de un socio a cualquier hora del día y en cualquier parte que el usuario se encuentre, obteniendo fluidez y velocidad de servicio en atención al cliente.

Palabras claves: Banca móvil, COAC San Francisco, Xcode, Iphone, ObjectiveC.

ABSTRACT

The aim of this project is to develop a mobile banking application for the credit union San Francisco Limited using Objective C technology for iPhone mobile devices due to the fact that this prestigious institution lacks this service. The methodology applied in the analysis is qualitative by using observations and interviews; and bibliographic and field research. To develop it, UML development methodology was implemented, resulting in the creation of an application named SFPUCEmóvil which allows confidential banking enquiries to be made for a client who has a savings account in the aforementioned bank. In conclusion, mobile applications in the financial world are the business of today and tomorrow, innoving and improving competition policies in the financial area and banking market. Therefore, SFPUCEmóvil offers the enquiry service of banking information to the members at any time of the day and from anywhere the user is, thus obtaining fluidity and speed in customer service.

***Key words**:* mobile banking, COAC San Francisco, xcode, Iphone, ObjectiveC.

ÍNDICE DE CONTENIDOS

PRELIMINARES

DECLARACIÓN DE AUTENTICIDAD Y RESPONSABILIDAD iii

Dedicatoria iv

Agradecimiento v

RESUMEN vi

ABSTRACT vii

CAPÍTULO I 1

1.1 ANTECEDENTES. 1

1.2 DEFINICION DEL PROBLEMA. 2

1.3 PLANTEAMIENTO DEL TEMA. 3

1.4 DELIMITACIÓN DEL PROBLEMA. 3

1.5 JUSTIFICACIÓN. 4

1.6 OBJETIVOS. 6

1.7 METODOLOGÍA DE TRABAJO. 7

1.7.1 TIPOS DE INVESTIGACIÓN. 7

1.7.1.1 INVESTIGACIÓN CUALITATIVA. 7

1.7.2 MODALIDAD BÁSICA DE LA INVESTIGACIÓN. 7

CAPÍTULO II 8

2.1 COOPERATIVA. 8

2.1.1 ¿Qué es una Cooperativa?. 8

2.1.2 Ley de Cooperativas del Ecuador. 9

2.1.3 Diferencia entre una empresa Capitalista de una empresa Cooperativa. 9

2.1.4 Cooperativismo. 10

2.1.4.1 Valores del Cooperativismo. 11

2.2 BANCA MÓVIL. 12

2.2.1 Beneficios de Banca Móvil. 13

2.2.2 Estrategia de Banca Móvil. 13

2.2.3 Aplicaciones de Banca Móvil. 14

2.2.4 Banca. 14

2.2.5 Banca Electrónica 15

2.3 TECNOLOGÍA OBJECTIVE C. 17

2.3.1 XCode. 17

2.3.2 Interface Builder..... 19
2.3.3 COCOA..... 20
2.4 DISPOSITIVO MÓVIL IPhone..... 22
2.4.1 Iphone 3GS..... 22
2.4.2 Iphone 4..... 23
2.4.3 Iphone 4S. 23
2.4.4 Conectividad de Internet y accesibilidad en Iphone. 24
2.5 DESARROLLO DE SOFTWARE. 25
2.5.1 UML..... 25
2.5.1.1 ¿Qué es UML?. 25
2.5.1.2 ¿Qué fases comprende UML?..... 25
2.5.1.3 ¿Qué ofrece UML?..... 26
2.5.1.4 ¿Cuáles son las Utilidades del Lenguaje UML?. 26
2.6 WEB SERVICE (Servicio Web)..... 28
2.6.1 Ejemplo de conexión de un web service con dispositivos móviles. 29
2.6.2 El lenguaje de intercambio XML..... 29
2.6.2.1 XML Web Services..... 30
2.6.3 Requisitos de un Web Service..... 30
2.7 PROTOCOLO DE CONEXIÓN ENTRE APLICACIÓN SFPUCE MÓVIL HACIA EL WEB SERVICE INSTITUCIONAL (SOAP)..... 31
2.7.1 VENTAJAS DEL PROTOCOLO SOAP..... 31
2.8 BASE DE DATOS..... 34
2.8.1 Características de base de datos. 34
2.9 SQL. 36
2.10 VIRTUALIZACIÓN..... 37
2.10.1 Mac OS X Snow Leopard. 37
2.11 Isimulator (SDK de Iphone)..... 40
2.12 VIRTUAL BOX..... 42
2.13 SEGURIDAD ELECTRÓNICA..... 44
2.14 MANEJO DE PROTOCOLOS DE SEGURIDAD..... 46
2.14.1 FIREWALL..... 46
2.14.2 CHECKPOINT. 47
2.14.3 ENCUESTA DE CHECKPOINT A INSTITUCIÓN FINANCIERA SAN FRANCISCO. 48

2.15 IPTABLES DE LINUX. 49
CAPÍTULO III 50
3.1 DESARROLLO. 50
3.1.1 METODOLOGÍA DE INVESTIGACIÓN. 50
3.1.2 POBLACIÓN Y MUESTRA. 51
3.1.2.1 Definición de Población. 51
3.1.2.2 Definición Muestra 51
3.1.2.3 Fórmula para calcular el tamaño de la Muestra. 52
3.1.2.4 Especificación de Caracteres de Fórmula. 52
3.1.2.5 Resumen Población y Muestra. 53
3.2 METODOLOGÍA DE DESARROLLO. 54
3.2.1 Metodología RUP. 54
3.2.2 ANÁLISIS. 55
3.2.2.1 Encuesta. 55
3.2.2.2 Entrevista 56
3.2.2.3 Análisis de la información recopilada. 58
3.2.3 ENTREVISTA. 64
3.2.3.1 Análisis de la Entrevista. 66
3.3 funcionales y no funcionales de la aplicación. 67
3.3.1 FUNCIONES. 67
3.3.2 METAS. 67
3.4 RESUMEN GENERAL. 69
3.5 CASOS DE USO. 70
3.5.1 Nombre: Autentificación de Usuario. 70
3.5.2 Nombre: Consulta de Saldos. 72
3.5.3 Nombre: Consulta de préstamos. 74
3.5.4 Nombre: Consulta fecha de pago de cuota. 76
3.5.5 Nombre: Consulta fecha de vencimiento de cuota. 79
3.5.6 Nombre: Consulta de últimos movimientos. 82
3.5.7 Nombre: Transferencia de dinero entre cuentas. 84
3.6 MODELO CONCEPTUAL. 88
3.7 DIAGRAMAS DE SECUENCIA. 89
3.7.1 AUTENTIFICACIÓN DE USUARIO. 89
3.7.2 CONSULTA DE SALDOS. 90

3.7.3 CONSULTA DE PRÉSTAMOS..... 91
3.7.4 CONSULTA DE FECHA PAGO DE CUOTA. 92
3.7.5 CONSULTA FECHA DE VENCIMIENTO DE CUOTA. 93
3.7.6 CONSULTA DE ÚLTIMOS MOVIMIENTOS. 94
3.7.7 TRANSFERENCUA DE DINERO. 95
3.8 DIAGRAMA DE ACTIVIDADES..... 96
3.9 DISEÑO. 97
3.9.1 DISEÑO DE BASE DE DATOS INSTITUCIONAL. 97
3.9.2 DICCIONARIO DE DATOS..... 98
3.9.2.1 Tabla Cuentas..... 98
3.9.2.2 Tabla MOVIMIENTOS CUENTAS. 99
3.9.2.3 Tabla USUARIOS..... 100
3.9.2.4 Tabla CLIENTES. 101
3.9.3 DISEÑO DE INTERFACES..... 102
3.9.3.1 ICONO DE LA PLICACIÓN. 102
3.9.3.2 FORMULARIO DE AUTENTIFICACIÓN DE USUARIO. 102
3.9.3.3 FORMULARIO MENU SFPUCE móvil..... 103
3.9.3.4 FORMULARIO CONSULTA DE SALDOS. 104
3.9.3.5 FORMULARIO CONSULTA DE PRÉSTAMOS 105
3.9.3.6 FORMULARIO TRANSFERENCIA DE DINERO 106
3.9.3.7 FORMULARIO ACERCA DE. 107
3.9.3.8 FORMULARIO SALIR. 108
3.10 CONTRATOS. 109
3.10.1 AUTENTIFICACIÓN DE USUARIO. 109
3.10.2 CONSULTA DE SALDOS..... 110
3.10.3 CONSULTA DE PRÉSTAMOS..... 111
3.10.4 CONSULTA DE FECHA DE PAGO DE CUOTA. 112
3.10.5 CONSULTA DE FECHA DE VENCIMIENTO DE CUOTA. 113
3.10.6 CONSULTA DE ÚLTIMOS MOVIMIENTOS. 114
3.10.7 TRANSACCIÓN DE DINERO ENTRE CUENTAS. 115
3.11 FUNCIONALIDAD. 117
3.12 ARQUITECTURA. 118
3.12.1 ARQUITECTURA 3 CAPAS. 118
3.12.1.1 Capa de Presentación (SFPUCE móvil): 118

3.12.1.2 Capa de Negocio (Web Service Institucional): 118
3.12.1.3 Capa de Datos (Base de Datos Institucional): 118
3.13 IMPLEMENTACION. 119
3.13.1 Requerimientos Minimos. 119
3.14 HERRAMIENTAS. 120
3.14.1 XCODE. 120
3.14.2 INTERFACE BUILDER. 121
3.15 PROGRAMACION 122
3.15.1 Web Service Institucional 122
3.15.1.1 Programación de Web Service para conexión con base de datos Institucional. .. 122
3.15.1.2 Programación de web service institucional para la aplicación SFPUCE móvil... 123
3.15.1.3 Programación de web service institucional para autentificación de usuario de SFPUCEmóvil. 125
3.15.1.4 Programación de web service institucional para consulta de préstamos vigentes y su saldo en SFPUCEmóvil. 130
3.15.1.5 Programación de web service institucional para consulta saldo cuenta de ahorros de SFPUCEmóvil. 132
3.15.1.6 Programación de web service institucional para consulta de 5 últimos movimientos de SFPUCEmóvil. 134
3.15.1.7 Programación de web service institucional para consulta de fecha de último pago de cuota de préstamo y su valor en SFPUCEmóvil. 136
3.15.1.8 Programación de web service institucional para consulta de fecha próximo pago de cuota de préstamo y su valor en SFPUCEmóvil. 137
3.15.1.9 Programación de web service institucional para transferencia de dinero entre cuentas de SFPUCEmóvil. 138
3.15.1.10 Programación de web service institucional para bloqueo de usuario en SFPUCEmóvil. 139
3.15.2 PROGRAMACIÓN DE LA APLICACIÓN SFPUCE móvil. 140
3.15.2.1 AUTENTIFICACIÓN DE USUARIO. 140
3.15.2.2 CONSULTA DE SALDOS. 150
3.15.2.3 CONSULTA DE PRESTAMOS 155
3.15.2.4 CONSULTA DE FECHA DE ULTIMO PAGO DE CUOTA 165
3.15.2.5 CONSULTA FECHA DE VENCIMIENTO DE CUOTA 179
3.15.2.6 CONSULTA ÚLTIMOS MOVIMIENTOS. 193
3.15.2.7 TRANSFERENCIA DE DINERO. 203
3.15.3 PRUEBAS. 217

3.15.4 RESULTADOS. 218
3.15.4.1 AUTENTIFICACIÓN DE USUARIO. 218
3.15.4.2 CONSULTA DE SALDOS. 221
3.15.5 VIRTUALIZACIÓN. 223
3.15.5.1 VIRTUAL BOX. 223
3.15.5.2 MAC OS X SNOW LEOPARD 10.6.6. 224
CAPÍTULO IV 225
4.1 CONCLUSIONES. 225
4.2 RECOMENDACIONES. 226
4.3 BIBLIOGRAFÍA. 227
4.4 GLOSARIO DE TERMINOS. 232
4.5 ANEXOS. 236
4.5.1 MANUALES. 236
4.5.1.1 MANUAL INSTALACIÓN DE APLICACIÓN (A). 236
4.5.1.2 MANUAL INSTALACIÓN DE APLICACIÓN (B). 237
4.5.1.3 MANUAL DE USUARIO. 239
4.5.2 ANEXO A ENCUESTA BANCA MOVIL 252
4.5.3 ANEXO B VIRTUALIZACIÓN. 254

ÍNDICE DE GRÁFICOS

Figura # 2.1 Diferencias empresa capitalista y cooperativa. 9
Figura # 2.2 Logotipo institución financiera cooperativa San Francisco. 10
Figura # 2.3 Banca Móvil. 12
Figura # 2.4 Estrategia Banca Móvil. 13
Figura # 2.5 Banca Electrónica. 15
Figura # 2.6 Icono Herramienta XCode. 17
Figura # 2.7 Icono herramienta interface builder. 19
Figura # 2.8 Programación cocoa. 20
Figura # 2.9 Dispositivo móvil iphone. 22
Figura # 2.10 Web Service. 28
Figura # 2.11 Conexión Web Service mediante dispositivo móvil. 29
Figura # 2.12 Mensaje soap. 33
Figura # 2.13 Base de datos. 34
Figura # 2.14 SQL. 36
Figura #2.15 Mac OS X Snow Lepard. 37
Figura # 2.16 Simulador de Iphone Isimulator. 40
Figura # 2.17 Presentación software virtual box. 42
Figura # 2.18 Protocolo de seguridad firewall. 46
Figura # 2.19 Protocolo de seguridad checkpoint. 47
Figura. # 3.1 Gráfico de resultados y porcentaje pregunta 1. 58
Figura # 3.2 Gráfico de resultados y porcentaje pregunta 2. 59
Figura # 3.3 Gráfico de resultados y porcentaje pregunta 3. 60
Figura # 3.4 Gráfico de resultados y porcentaje pregunta 4. 61
Figura # 3.5 Gráfico de resultados y porcentaje pregunta 5. 63
Figura #3.6 Diagrama de caso de uso 71
Autentificación de usuario. 71
Figura # 3.7 Diagrama de caso de uso 73
Consulta de saldos en ahorros. 73
Figura # 3.8 Diagrama de caso de uso 75
Consulta de préstamos. 75
Figura # 3.9 Diagrama de caso de uso 78
Consulta fecha de pago de cuota. 78

Figura # 3.10 Diagrama de caso de uso 81
Consulta fecha de vencimientos de cuota. 81
Figura # 3.11 Diagrama de caso de uso 83
Consulta de últimos movimientos. 83
Figura # 3.12 Diagrama de caso de uso 87
Transferencia de dinero entre cuentas. 87
Figura # 3.13 Modelo conceptual 88
Figura # 3.14 Diagrama de secuencia 89
Autentificación de usuario. 89
Figura # 3.15 Diagrama de secuencia 90
Consulta de saldos. 90
Figura # 3.16 Diagrama de secuencia 91
Consulta de préstamos 91
Figura # 3.17 Diagrama de secuencia 92
Consulta de fecha de pago de cuota. 92
Figura # 3.18 Diagrama de secuencia 93
Consulta de fecha de vencimientos de cuota 93
Figura # 3.19 Diagrama de secuencia 94
Consulta de últimos movimientos. 94
Figura # 3.20 Diagrama de secuencia 95
Transferencia de dinero entre cuentas. 95
Figura # 3.21 Diagrama de actividades estados de la aplicación. 96
Figura # 3.22 Esquema de base de datos institucional. 97
Figura # 3.23 Icono aplicación SFPUCE móvil. 102
Figura # 3.24 Formulario Autentificación de usuario. 102
Figura # 3.25 Formulario MENU SFPUCE móvil 103
Figura # 3.26 Formulario Consulta de saldos. 104
Figura # 3.27 Formulario Consulta de préstamos 105
Figura # 3.28 Formulario Transferencia de dinero. 106
Figura # 3.29 Formulario acerca de. 107
Figura # 3.30 Formulario salir 108
Figura 3.31 Funcionalidad aplicación 3 capas. 117
Figura #3.32 Prueba apertura de app SFPUCEmóvil. 218
Figura # 3.33 Prueba ingreso correcto datos en autentificación de usuario. 219

Figura # 3.34 Prueba ingreso incorrecto de datos en autentificación de usuario. 220
Figura # 3.35 Prueba consulta de saldos en cuenta de ahorro 221
Figura #4.2 Ícono Aplicación SFPUCE móvil 239
Figura #4.3 Menú de aplicaciones Iphone. 239
Figura #4.4 Ingreso de usuario y contraseña en pantalla principal de app 240
Figura #4.5 Consulta de saldo en cuenta de ahorros en SFPUCEmovil. 241
Figura #4.6 Consulta de préstamos y saldo préstamo en SFPUCE móvil. 242
Figura #4.7 Consulta de fecha de última cuota pagada y valor en SFPUCE móvil. 243
Figura #4.8 Consulta fecha de su próxima cuota y valor en SFPUCEmovil 244
Figura #4.9 Consulta de 5 últimos movimientos en la aplicación SFPUCE móvil. 245
Figura #4.10 Transacción de dinero en la aplicación SFPUCE móvil. 246
Figura #4.11 Alerta antes de realizar transferencia en SFPUCE móvil. 247
Figura #4.12 Transferencia de dinero satisfactoria en SFPUCE móvil. 248
Figura # 4.14 Ejemplo cabecera de email, tras transferencia 249
Figura # 4.15 Ejemplo cuerpo de email, tras transferencia 249
Figura #4.16 Acerca de en la aplicación SFPUCE móvil. 250
Figura #4.17 Salir de la aplicación SFPUCE móvil. 251
Figura # 4.18 Bienvenida de Instalación de VirtualBox. 254
Figura # 4.19 Instalación Personalizada de Virtual Box. 254
Figura # 4.20 Creación de Accesos Directos de Virtual Box. 255
Figura # 4.21 Mensaje de Alerta en el asistente de Instalación de Virtual Box 255
Pérdida de Conexiones de red de la máquina anfitrión. 255
Figura # 4.22 Confirmación del asistente VirtualBox listo para instalarse. 256
Figura # 4.23 Proceso de Instalación de VirtualBox. 256
Figura # 4.24 Asistente de Instalación de VirtualBox terminado Herramienta de Virtualización lista para usarse. 257
Figura # 4.25 Asistente de Instalación de Mac Snow Leopard. 258
Figura # 4.26 Inicio de Instalación de Mac Snow Leopard. 258
Figura # 4.27 Configuración de Idioma en la Instalación de Mac Snow Leopard. 259
Figura # 4.28 Preparación de Instalación de Mac Snow Leopard. 259
Figura # 4.29 Inicio de Configuración de Mac Snow Leopard. 260
Figura # 4.30 Términos de contrato de Licencia para Mac Snow Leopard. 260
Figura # 4.31 Asistente de Creación de Imagen de Disco de Mac Snow Leopard. 261
Figura # 4.32 Selección de Imagen de Disco para instalación de Mac Snow Leopard. ... 261

Figura # 4.33 Instalación de Mac Snow Leopard en disco duro. 262
Figura # 4.34 Seguimiento de Instalación de Mac Snow Leopard. 262
Figura # 4.35 Confirmación de Instalación de Mac Snow Leopard. 263
Figura # 4.36 Notificación instalación correcta de Mac Snow Leopard. 263
Figura # 4.37 Reinicio después de instalación de Mac Snow Leopard. 264
Figura # 4.38 Carga de Sistema Operativo Mac Snow Leopard. 264
Figura # 4.39 Visualización de Disco Duro Virtual. 265
Figura # 4.40 Configuración de Sistema Operativo Selección de Idioma. 265
Figura #4.41 Selección de idioma español ISO sistema estandarizado. 266
Figura #4.42 Bienvenida de Sistema Operativo Mac OS x. 266
Figura #4.43 Creación de Cuenta en Sistema Operativo Mac Snow Leopard. 267
Figura #4.44 Registro de Datos con cuenta. 267
Figura #4.45 Registro de Datos con cuenta en Mac Snow Leopard. 268
Figura #4.46 Creación de Datos en Mac Snow Lepard. 268
Figura #4.47 Confirmación de registro de Cuenta en Mac Snow Leopard. 269
Figura # 4.48 Configuración de Teclado de Sistema Operativo Mac Snow Leopard. 269
Figura # 4.49 Entorno listo Mac Snow Leopard 10.6.6. 270
Figura # 4.50 Registro en pagina Ios Dev Center. 270
Figura # 4.51 Registro para descarga de herramientas. 271
Figura # 4.52 Creación de nuevo ID en página de Aple. 271
Figura # 4.53 Ingreso de datos en nuevo ID. 272
Figura # 4.54 Completando datos en perfin profesional. 272
Figura # 4.55 Confirmación de Registro. 273
Figura # 4.56 E mail activación. 273
Figura # 4.57 Activación de cuenta 274
Figura # 4.58 Cuenta creada en Apple Developers. 274
Figura # 4.59 Servicio de Apple. 275
Figura # 4.60 Descarga de Herramientas de desarrollo. 276
Figura # 4.61 Introduccion de Xcode. 276
Figura # 4.62 Estado de descarga de Xcode. 277
Figura # 4.63 Archivos de descarga. 277
Figura # 4.64 Acceso al archivo xcode. 278
Figura # 4.65 Download Xcode. 278
Figura # 4.66 Inicio de instalación SDK de desarrollo de Aplicaciones Iphone. 279

Figura # 4.67 Carga de Instalador SDK. 279
Figura # 4.68 Seguridades de SDK. 280
Figura # 4.69 Asistente de Instalación de SDK. 280
Figura # 4.70 Políticas de acuerdo de SDK. 281
Figura # 4.71 Licencia SDK. 281
Figura # 4.72 Direccionamiento de SDK en imagen de Disc. 282
Figura # 4.73 Informe de espacio de SDK en imagen de Disc. 282
Figura # 4.74 Confirmación de administración en instalación. 283
Figura # 4.75 Inicio de Instalación de Xcode. 283
Figura # 4.76 Xcode Instalado. 284
Figura # 4.77 Ruta de Instalación de Xcode. 284
Figura # 4.78 Bienvenida de Xcode. 286
Figura # 4.79 Bienvenida de Interface Builder. 286
Figura # 4.80 Bienvenida de Isimulator. 287

Tablas

Tabla. # 3.1 Tabla de resultados y porcentaje pregunta 1. 58
Tabla # 3.2 Tabla de resultados y porcentaje pregunta 2. 59
Tabla # 3.3 Tabla de resultados y porcentaje pregunta 3. 60
Tabla # 3.4 Tabla de resultados y porcentaje pregunta 4. 61
Tabla # 3.5 Tabla de resultados y porcentaje pregunta 4. 62
Tabla #3.6 Diseño de tablas 98
Tabla CUENTAS. 98
Tabla # 3.7 Diseño de tablas 99
Tabla MOVIMIENTOS CUENTAS. 99
Tabla # 3.8 Diseño de tablas 100
Tabla USUARIO. 100
Tabla # 3.9 Diseño de tablas 101
Tabla CLIENTES. 101

CAPÍTULO I

GENERALIDADES

1.1 ANTECEDENTES.

La Cooperativa Financiera de Ahorro y Crédito San Francisco Ltda., es una empresa cimentada en principios cooperativos, orientada a ofrecer servicios financieros y no financieros ejecutados con calidad, para contribuir al bienestar de sus socios, clientes y a la sociedad en general.

La visión de la empresa es llegar a ser el mejor aliado de sus socios, generando soluciones financieras integrales de calidad, basadas en prácticas éticas, transparentes y rentables.

El ámbito de las Cooperativas de ahorro y crédito en estos días, es muy grande, y de alta competitividad, pero la Cooperativa Financiera San Francisco es una empresa que se encuentra en el primer lugar entre las medianas.

Realizando el seguimiento respectivo se estima que la empresa contiene alrededor de 120.000 socios, de los cuales 64.865 están activos

El proyecto surge por la necesidad y carencia de una aplicación de Banca Móvil en la Institución Financiera San Francisco, que permita consultar toda la información bancaria de un socio afiliado, ya que ésta solo contiene un sistema que es cerrado y local (sistema solo de intranet), que limita al socio a no poder acceder a su información financiera desde cualquier parte que se encuentre y en cualquier horario. Ante esta escasez de servicio nace la idea de creación de la aplicación, para ofrecer una mejor comodidad a los socios de la Cooperativa.

1.2 DEFINICION DEL PROBLEMA.

¿De qué manera una aplicación de banca móvil puede generar información confidencial bancaria de un socio aliado a la empresa Financiera San Francisco Ltda.?

El problema parte por cierta incomodidad de los socios al perder tiempo debido a las largas filas de gente que esperan por consultar su información.

Este fenómeno estresa mucho a todas las personas, y cabe resaltar que en varias ocasiones, se realizan consultas pequeñas, cortas y sencillas, que no justifican y merecen el largo lapso de espera consecuente por la cantidad de gente.

La mejor vía de consulta es acercarse personalmente a la Cooperativa para realizar sus respectivas preguntas y aclarar la información de sus cuentas, pero, al realizarlo se encuentran con este fenómeno molesto que ya es algo frecuente.

De aquí nace la idea de querer brindar a los socios, este nuevo servicio de banca móvil, que les permitirá consultar toda su información bancaria privada, desde cualquier parte donde se encuentren y en cualquier momento, obteniendo fluidez y velocidad de servicio en atención al cliente.

1.3 PLANTEAMIENTO DEL TEMA.

Desarrollo de una aplicación de banca móvil para la Cooperativa de ahorro y crédito San Francisco Limitada usando la tecnología Objective C implementado en el dispositivo móvil IPhone para el periodo 2011 - 2012.

1.4 DELIMITACIÓN DEL PROBLEMA.

El proyecto se desarrolló en la Cooperativa de Ahorro y Crédito San Francisco Limitada en la ciudad de Ambato, con una duración estimada de 18 meses, para su correcto análisis, estudio, capacitación, programación, compilación, diseño, pruebas, mantenimiento y documentación, llegando a la culminación generando un nuevo servicio para los socios de la cooperativa Financiera San Francisco Ltda.

La aplicación SFPUCEmóvil permite realizar las siguientes acciones:

- Validación y seguridad de usuarios
- Consulta de saldo en cuenta de ahorros.
- Consulta de préstamos vigentes y saldo de los mismos.
- Consulta de fecha de ultimo pago de cuota y valor de la misma.
- Consulta de fecha de próxima cuota y el valor de la misma.
- Consulta de 5 últimos movimientos en cuenta de ahorros.
- Transferencia de dinero entre cuentas pertenecientes a la misma Institución.

1.5 JUSTIFICACIÓN.

La creación de la aplicación SFPUCE móvil, de Banca Móvil, parte de la idea de innovar el negocio y la inexistencia de este tipo de software en la Institución Financiera San Francisco, ya que esta aplicación le permitirá a los socios realizar transacciones seguras en movimiento, acceder a su respectiva información bancaria, y realizar consultas personales las 24 horas del día, utilizando un sistema móvil, claro y sencillo brindando así una mejor comodidad de servicio e innovación para mejorar políticas de competencia dentro del mercado bancario.

Se ha pensado implementar esta aplicación en el dispositivo de comunicaciones IPhone ya que debido a sus funcionalidad, seguridad y tecnología, se ha transformado en una gran herramienta del mundo moderno, ya que proporciona beneficios y características durante el desarrollo de una aplicación.
En el aspecto de visualización y diseño, se puede gestionar con un agradable interfaz de usuario, exelente definición de pantalla, manipulación touchScreen con las opciones de la interfaz, y un tamaño de pantalla de 16 millones de colores.

Con respecto a navegación, este dispositivo tiene una exelente velocidad de conexión WIFI (característica más importante por la conexión necesaria al web service de la Cooperativa).
Además Iphone nos ofrece ventajas en el trabajo con aplicaciones, como es la ejecución de una sola aplicación, en una ventana principal, tiempo de respuesta limitado antes, durante, y después de la ejecución, dimensiones de pantalla limitadas físicamente a 320x480 pixeles, limitaciones importantes en la memoria disponible, no garbage recollector, lo que quiere decir más limitaciones de memoria, se tiene la característica innata que es un trabajo sin ratón y sin teclado físico, y lo mas importante una magnifica Interacción con Internet.

IPhone tiene compatibilidad con Objective C, el cual es un lenguaje de programación orientado a objetos creado como un súper conjunto de C para el desarrollo de aplicaciones.

Después de tener claro los mejores y necesarios puntos de hardware y software para la aplicación móvil, es factible desarrollarlo en la Institución Financiera de Ahorro y crédito San Francisco por los siguientes motivos:

- Principalmente porque la empresa no cuenta con ningún servicio electrónico usando un dispositivo como un teléfono celular.
- La Cooperativa en este año actualizó su infraestructura tecnológica implementando un servidor de dominio BLADE HP ProLiant DL360 G7 que ha sido optimizado para instalaciones con limitaciones de espacio y combina potencia informática, con la característica de poseer sistemas flexibles inteligentes preparados para entornos dinámicos complejos, brindando mayor rendimiento, capacidad y eficacia en menor espacio con un consumo óptimo de energía; por lo que de esta manera existe toda una completa seguridad confiable y legible como para crear una aplicación de banca móvil, que permita conseguir información desde cualquier parte que se encuentre el cliente sin afectar la integridad del sistema principal.
- Hay que tomar en cuenta que el nuevo plan estratégico institucional de este año es contar con nuevos canales de comunicación para los socios. Sistema de innovación claramente aprobado.
- La institución se encuentra obligada a cumplir con las mismas obligaciones que la banca por que se ha convertido en una competencia directa.
- Un punto muy importante por agregar es que existe la autorización para acceder a la base de datos con una confianza muy grande convirtiéndose este en un punto muy importante para llevar a cabo la creación de la aplicación de SFPUCE móvil.

1.6 OBJETIVOS.

GENERAL.

Desarrollar una aplicación de banca móvil para la cooperativa de ahorro y crédito San Francisco Limitada usando la tecnología Objective C para el dispositivo móvil IPhone en el Periodo 2011 - 2012.

ESPECIFÍCOS.

- Realizar un estudio acerca del problema y la carencia de aplicaciones de banca móvil en la Institución Financiera San Francisco.
- Diseñar una interfaz de usuario para IPhone, que sea legible y fácil de utilizar al realizar las respectivas consultas bancarias.
- Programar la aplicación de banca móvil y conectarla mediante el internet al web service institucional.
- Realizar la documentación respectiva del proyecto.

1.7 METODOLOGÍA DE TRABAJO.

1.7.1 TIPOS DE INVESTIGACIÓN.

1.7.1.1 INVESTIGACIÓN CUALITATIVA.

Observación.

Se estudiarán todos los protocolos de seguridad dentro de la Cooperativa San Francisco para poder acceder sin ningún problema a los datos del web service Institucional y poder ejecutar las consultas requeridas por el usuario.

Se visitó la Institución para ser partícipe y testigo del desarrollo diario que ésta realiza, e investigar lo necesario que son las consultas bancarias en tiempo real.

Entrevistas (Encuestas a los propietarios, lluvia de ideas).

Se realizará una encuesta a todos los socios dentro de la Cooperativa San Francisco que cumpla las expectativas y las necesidades del tema propuesto.

La encuesta tratará de estudiar acerca de que tanto conocen los socios acerca de una aplicación de banca móvil (lo que es un punto muy importante ya que se trata de un tema innovador), si es que les gustaría este servicio y que tipo de consultas les interesaría realizar dentro de la comodidad de este servicio.

1.7.2 MODALIDAD BÁSICA DE LA INVESTIGACIÓN.

Campo.- La aplicación de Banca Móvil se estima será de gran impacto en la gente joven aliada a la Institución, ya que de hoy en día ellos tienen más relación con la tecnología y con ésta clase de dispositivos de comunicación, siendo este un aspecto importante para que más clientes jóvenes sean parte de la Cooperativa Financiera San Francisco.

Bibliográfica.- La investigación se llevará a cabo basada en libros, folletos, manuales, cursos, proyectos realizados, estimaciones, y en internet.

CAPÍTULO II

2. MARCO TEÓRICO.

2.1 COOPERATIVA.

2.1.1 ¿Qué es una Cooperativa?.

"Una cooperativa es una asociación autónoma de personas agrupadas voluntariamente para satisfacer sus necesidades económicas, sociales y culturales comunes, por medio de una empresa que se posee en conjunto y se controla democráticamente. Las cooperativas se crean por necesidades económicas y sociales, operan sin fines de lucro. Su gobierno se establece sobre bases democráticas, cada socio puede llegar a ser un director. No se especula con las acciones de los socios, las economías se les devuelven a los socios a base del patrocinio de cada uno. En efecto, la identidad del sistema cooperativo se fundamenta en los principios y valores. El cooperativismo es la puesta en práctica de los valores de igualdad, justicia, ayuda mutua, esfuerzo propio, honestidad, responsabilidad social, democracia, construcción, sacrificio y solidaridad. Lo más importante en la organización cooperativa es el ser humano.

Las personas forman una cooperativa no sólo con propósitos económicos sino sociales y espirituales. En la cooperativa el aspecto humano se hace presente en el espíritu de servicio, de servir a los demás. Esta motivación es principalísima en el cooperativismo. La cooperación como sistema económico y social está basado en ideas que van desde la ayuda mutua hasta la prioridad del ser humano sobre el dinero, desde la participación equitativa en las ganancias hasta lograr una sociedad igualitaria. La cooperativa es democrática. Se mide no tan sólo por la participación de los socios en las decisiones de la empresa sino por su derecho al voto en las asambleas sin importar el dinero que tenga invertido en la cooperativa. Se organiza entre personas con intereses comunes para

servirse mutuamente. No se otorgan favores especiales o privilegios a funcionarios y directivos, la autoridad suprema está en el socio". (Liga Coop, 2011)

2.1.2 Ley de Cooperativas del Ecuador.

Dice: "Son cooperativas las sociedades de derecho privado, formadas por personas naturales o jurídicas que, sin perseguir finalidades de lucro, tienen por objeto planificar y realizar actividades o trabajos de beneficio social y colectivo, a través de una empresa manejada en común y formada con la aportación económica, intelectual y moral de sus miembros". (Coopjep, 2008)

2.1.3 Diferencia entre una empresa Capitalista de una empresa Cooperativa.

Empresa capitalista	Empresa cooperativa
Las personas buscan obtener ganancias y beneficiarse unos sobre otros	Las personas buscan dar servicios y el beneficio común
Con la ganancia se beneficia el propietario del capital	Con la ganancia se beneficia la prestación de servicios
Principal objetivo: ensanchar los márgenes hasta hacerlos lo más provechosos posibles para el accionista	Principal objetivo: ofrecer servicios de calidad y económicos, y reportar beneficios a los socios
El beneficio logrado se distribuye entre los accionistas	El excedente disponible se devuelve a los socios en proporción a sus actividades o servicios
El capital dirige, la persona no	La persona dirige, el capital no
La persona no tiene ni voz ni voto	La persona tiene voz y voto
El número de socios es limitado	El número de socios es ilimitado. Pueden ser socios todas las personas que lo deseen, según estatutos
Los objetivos son independientes del socio	Los objetivos son dependientes de las necesidades de los socios
Administrada por un número reducido de personas	Se gobierna con la participación de todos los socios
Se organiza internamente por medio de la competencia	Se organiza internamente por medio del apoyo mutuo

Figura # 2.1 Diferencias empresa capitalista y cooperativa.

Cooperativa Financiera San Francisco Limitada.



Figura # 2.2 Logotipo institución financiera cooperativa San Francisco.

2.1.4 Cooperativismo.

"Cualquiera sea la situación, lo cierto es que a través de los tiempos la cooperación ha tomado formas, estilos y matices diferentes, acordes con cada realidad social, y también es incuestionable que sus estructuras, métodos y concepciones se han ido perfeccionando y al mismo tiempo ganando su propio espacio, reconocimiento y aceptación comunitaria, de modo que hoy los sistemas de cooperación han alcanzado niveles muy interesantes de competitividad, frente al sistema productivo imperante, al menos en las áreas donde dispone de ventajas competitivas a su favor.

El Cooperativismo es una doctrina socio-económica que promueve la organización de las personas para satisfacer, de manera conjunta sus necesidades. El Cooperativismo está presente en todos los países del mundo. Le da la oportunidad a los seres humanos de escasos recursos, tener una empresa de su propiedad junto a otras personas. Uno de los propósitos de este sistema es eliminar la explotación de las personas por los individuos o empresas dedicados a obtener ganancias.

La participación del ser humano común, asumiendo un rol protagonista en los procesos socio-económico en la sociedad en que vive, es la principal fortaleza de la doctrina cooperativista.

El cooperativismo se rige por valores y principios basados en el desarrollo integral del ser humano". (Coac-Sanfra, s.f.)

2.1.4.1 Valores del Cooperativismo.

"Los valores son cualidades que las personas cultivan para ser dignos y respetables en la sociedad, los mismos que se adquieren y desarrollan con tesón y esfuerzo. Las Cooperativas adoptan los siguientes valores": (COOPJEP, s.f.)

- Responsabilidad.
- Democracia.
- Igualdad.
- Equidad.
- Solidaridad.
- Honestidad.
- Transparencia.
- Responsabilidad Social.
- Preocupación por los demás.

2.2 BANCA MÓVIL.



Figura # 2.3 Banca Móvil.

"Grupos financieros apuestan a plataformas a través del celular, de las cuales podrá sacar ventaja; pagar cuentas, consultar saldos o hacer depósitos vía remota generará ahorros en tiempo y dinero.

En el mundo de hoy la banca móvil es un término popular. Banca móvil significa una transacción financiera llevada a cabo por iniciar sesión en el sitio Web de una institución financiera mediante un teléfono celular, como "visualización de saldos", hacer "transferencias entre cuentas" o pago de facturas. Es un término utilizado para realizar el balance de cheques, transacciones de cuenta, pagos etc. a través de un dispositivo móvil como un teléfono móvil.

Los servicios no basados en transacciones de carácter informativo son sin embargo esencial para llevar a cabo transacciones por ejemplo, consultas de saldo podrían ser necesario antes de comprometerse" (Davivienda, s.f.)

2.2.1 Beneficios de Banca Móvil.

"1. Olvidar filas.
2. Menor Costo.
3. Aprendizaje de Tecnología.
4. Seguridad.
5. Atención inmediata y disponible las 24 horas.
6. Sistema simple.
8. Adiós del 'token' de los dispositivos móviles.
9. Organización de presupuesto". (Cnn Expansion, 2011)

2.2.2 Estrategia de Banca Móvil.



Figura # 2.4 Estrategia Banca Móvil.

- La tecnología en su gran potencial, es manejada por gente joven lo que no necaesariamente significa que es un público bancario.
- La estrategia de banca móvil radica que por medio de esta tecnología se genere soluciones de respuestas bancarias por medio de mensajes, conexiones web, y despliegue de resultados en dispositivos que cualquier tipo de cliente lo requiera.
- Además que abarque manipulación simple y comprensiva, que brinde al cliente fácil manejo de uso y exelente confidencialidad de datos.

2.2.3 Aplicaciones de Banca Móvil.

Las aplicaciones de banca móvil engloban esta combinación, tanto con sus beneficios como con la estratégia, pero todo mediante la tecnología.

Tecnología que hoy abarca dispositivos móviles, que generan respuesta limitada en un tiempo de espera oportuno.

2.2.4 Banca.

Término que se refiere al conjunto de entidades y empresas que constituyen el Sistema Bancario, es decir, el conjunto de entidades financieras dedicadas a realizar operaciones con capital (depósitos, préstamos, inversiones, etc.)

2.2.5 Banca Electrónica.



Figura # 2.5 Banca Electrónica.

"La banca electrónica hace referencia al tipo de banca que se realiza por medios electrónicos como puede ser cajeros electrónicos, teléfono y otras redes de comunicación. Tradicionalmente, este término ha sido atribuido a la banca por Internet o banca online, pero conviene aclarar su significado.

Algunos autores lo consideran como un constructor de orden superior que supone varios canales que incluyen también la banca telefónica, la banca por teléfono móvil (basada en tecnología Wireless Application Protocol WAP que traslada Internet al teléfono móvil) y la basada en televisión interactiva (iNet-television)". (Wikipedia, 2015)

Beneficios:

- Agilidad y eficacia en sus operaciones bancarias.
- Tecnología de punta a su servicio.
- Total confidencialidad.
- Acceso permanente.

"La banca, por supuesto, no ha sido ajena a esta revolución de la información. Por una parte, ha abaratado los costes de su gestión interna, al ser más rápido y barato mover información, y por otra parte ha ahorrado horas de trabajo de sus

empleados al existir ahora gran cantidad de trámites que los clientes realizan por Internet. Por tanto, Internet genera unos beneficios adicionales al sistema bancario porque incrementa su productividad y baja sus costes de gestión". (Wikipedia, 2015)

2.3 TECNOLOGÍA OBJECTIVE C.

2.3.1 XCode.



Figura # 2.6 Icono Herramienta XCode.

"Xcode es el entorno de desarrollo integrado (IDE, en sus siglas en inglés) de Apple Inc. y se suministra gratuitamente junto con Mac OS X. Xcode trabaja conjuntamente con Interface Builder, una herramienta gráfica para la creación de interfaces de usuario.

Xcode incluye la colección de compiladores del proyecto GNU (GCC), y puede compilar código C, C++, Objective-C, Objective-C++, Java y AppleScript mediante una amplia gama de modelos de programación, incluyendo, pero no limitado a Cocoa, Carbón y Java.

Xcode (Pronunciado: ex-co-ud) es una serie de herramientas de desarrollo en Mac OS X. Este es desarrollado por Apple e incluye las herramientas necesarias para programar en el ambiente de Mac OS o iOS (iPhone, iPod Touch, iPad).

Xcode es el entorno de desarrollo integrado de Apple Inc. Xcode es el motor que le brinda poder al ambiente integrado de desarrollo de Apple para Mac OS X y para iPhone OS. Xcode es completamente funcional para el desarrollo de aplicaciones Mac e incluye un editor de código de clase mundial, un depurador gráfico, e integrado en Objective-C, C y C + + compiladores. Xcode posee un profundo conocimiento de los marcos que el poder del Cacao Mac OS X, e incluso es capaz de identificar los errores mediante el análisis del código que

escriba sin ejecutar la aplicación. La experiencia de desarrollo intuitiva hace que sea fácil de obtener su primera aplicación en funcionamiento, sin embargo, Xcode es tan poderoso que propio Mac OS X está construido utilizando el conjunto de herramientas de exactamente la misma.

CARACTERÍSTICAS.

- Ofrece una interfaz de usuario atractiva y potente.
- Es capaz de organizar y editar los archivos fuente, ver la documentación, crear el producto, limpiar el código y optimizar el rendimiento del programa.
- Xcode además te ayuda a programar para tecnologías nuevas. Se trata de una plataforma de desarrollo integrada con enormes posibilidades de personalización que ofrece prestaciones con las que podrás crear un entorno de trabajo eficiente y fácil de usar.
- Se suministra gratuitamente junto con Mac OS X." (Bermudez, 2011)

2.3.2 Interface Builder.



Figura # 2.7 Icono herramienta interface builder.

"Interface Builder es la aplicación del entorno de programación para Mac OS X e iPhone OS que se usa para hacer interfaces gráficas de usuario.

Es una aplicación muy potente, como podremos comprobar en detalles tán pequeños como el uso inteligente de las guías para colocar componentes. La programación en Cocoa es un tanto peculiar en cuanto a la unión de eventos y elementos de la interfaz gráfica.

El paradigma de programación que sigue Cocoa enlaza eventos de elementos de la interfaz gráfica a objetos (implementaciones en código). De esta interpretación de las interfaces aparecen los outlets (IBOutlet) y las acciones (IBActions).

Interface Builder es una herramienta de desarrollo visual que ayuda a crear rápidamente la parte de interfaz gráfica de la aplicación.

Esta herramienta además de código, genera ficheros .nib que son ficheros con una descripción de los components de la interfaz gráfica, que se cargan al arrancar la aplicación y que se usa Cocoa para pintar la ventana y sus components en pantalla, es decir el equivalente a los recursos en otros sistemas." (Gumus, s.f.)

2.3.3 COCOA.



Figura # 2.8 Programación cocoa.

"La API de programación de Mac OS X se llama Cocoa y es un api totalmente orientado a objetos, es decir formada básicamente por clases. A esta API se la considera un framework (entorno de trabajo de desarrollo con todas las funciones básicas), en el sentido de que sus clases ayudan al programador a montar el esqueleto de su aplicación.

Una característica importante de esta API es que es accesible tanto desde Objective-C como desde java, lo cual es muy útil ya que reduce considerablemente el tiempo de aprendizaje del framework.

Hay que tener en cuenta que en la actualidad las librerías de programación son tan grandes, que su aprendizaje nos lleva mucho más tiempo que el aprendizaje del lenguaje de programación en sí.

La librería Cocoa está dividida en 2 kits:

- **Application Kit.-** Este Kit contiene todas las clases relacionadas con la interfaz gráfica (ventanas, botones, cajas de texto, etc). Pero el Application Kit es mucho más que un conjunto de componentes de interfaz gráfica, o tiene clases para gestión de documentos (mediante el patrón vista-documento usado también en otros muchos frameworks de programación), o por ejemplo,

también trae clases que encapsulan fuentes, colores, impresión corrección ortográfica o drag and drop (arrastrar y soltar).

- **Foundation Kit.-** Este Kit contiene todas las demás clases no visuales como son las clases de acceso a ficheros, de programación multihilo, de trabajo en red, de utilidad, de gestión de cadenas, etc." (Personal.Ya, s.f.)

2.4 DISPOSITIVO MÓVIL IPhone.



Figura # 2.9 Dispositivo móvil iphone.

"IPhone es una familia de teléfonos inteligentes multimedia con conexión a Internet, pantalla táctil capacitiva y escasos botones físicos diseñados por la compañía Apple Inc.

Ya que carecen de un teclado físico, integran uno en la pantalla táctil con orientaciones tanto vertical como horizontal.

2.4.1 Iphone 3GS.

El iPhone 3GS dispone de una cámara de fotos de 3 megapíxeles y un reproductor de música (equivalente al del iPod) además de software para enviar y recibir mensajes de texto y mensajes de voz. También ofrece servicios de Internet como leer correo electrónico, cargar páginas web y conectividad por Wi-Fi. La primera generación de teléfonos eran GSM cuatribanda.

Apple Inc. anunció el iPhone en 2007, tras varios rumores y especulaciones que circulaban desde hacía meses. El iPhone se introdujo inicialmente en los Estados Unidos el 29 de junio de 2007. Fue nombrado "Invento del año" por la revista "Time" en el 2009.

Se presentó el iPhone 3GS (S de Speed, velocidad), presentando el mismo diseño que el 3G, pero con hardware mejorado. El iPhone 3GS es hasta **dos veces más rápido que su predecesor cargando páginas web o ejecutando aplicaciones**, ES 2.0 para ofrecer mejores gráficos 3D, soporta HSDPA a 7,2 MB (también conocido como 3.5G), cuenta con una nueva cámara de 3 megapíxeles con enfoque automático y grabación de vídeo, control por voz, y brújula.

Mientras que el día 7 de junio del 2010 Apple presentó el iPhone 4, con un diseño renovado tanto estéticamente, abandonando su estética redondeada, como a nivel Hardware.

Contiene el System of Chip Apple A4, creado por Intrensity y Samsung, aunque posteriormente comprado por Apple, que es el mismo que contiene el iPad (excepto el iPad 2 que contiene un chip Apple A5), aunque reducido de frecuencia de trabajo, 512 MiB de memoria RAM, un panel LCD IPS de alta resolución autodenominado Retina Display, 2 cámaras, una de 5 megapíxeles con opción para grabar en HD 720p y la otra VGA.

2.4.2 Iphone 4.

El iPhone 4 es la 4ª generación del dispositivo de telefonía móvil de Apple que fue lanzado al mercado en 2010.

Destaca la renovación estética frente al iPhone 3G y 3GS con una forma más poligonal y un frontal y trasera planos realizados en vidrio sobre un chasis de acero. Y destaca el estreno del sistema operativo iOS 4. EL cual incluye multitasking, fondos de pantalla en pantalla de inicio, cámara de 5 MP.

2.4.3 Iphone 4S.

El iPhone 4S es el último teléfono de la familia iPhone. Entre sus novedades están: Cámara de 8 megapixeles con 5 lentes, grabación y edición en Full HD

(1080p), System on Chip A5 de doble núcleo a 800 MHz con nueva GPU, duplica las velocidades máximas de datos HSDPA hasta los 14,4 Mb/s, control por voz "SIRI", soportará redes CDMA y GSM, entre otras.

Por otro lado, la estética sigue siendo idéntica a la del iPhone 4 tanto en forma, en dimensiones, en peso, y en pantalla, aún esto, el iPhone 4S ha sido el iPhone más vendido de toda la historia, consiguiendo vender 4 millones de unidades en solo un fin de semana, con lo que ha conseguido una fuerte demanda en el mercado.

2.4.4 Conectividad de Internet y accesibilidad en Iphone.

El primer iPhone puede conectarse a un área local Wi-Fi o global GSM o la red EDGE, ambas normas 2G. Como justificación a esta limitación frente a otros dispositivos de la competencia, Steve Jobs declaró en septiembre de 2008 que la tecnología 3G tendría que extenderse en los Estados Unidos y que los chipsets 3G tendrían que hacerse mucho más eficientes antes de incluirlo en el iPhone.

El iPhone, por defecto pedirá una nueva conexión Wi-Fi descubierta y preguntará por la contraseña cuando se necesite, también apoyando manualmente la conexión de redes Wi-Fi cerradas.

Cuando Wi-Fi está activo, el iPhone automáticamente cambiará la red de EDGE a cualquier red Wi-Fi cercana y que haya sido probada antes. La versión 2.0 del Sistema Operativo del iPhone soporta 802.1X, que es utilizada por muchas universidades y redes Wi-Fi corporativas.

La conexión a Internet universal ofrecida por el iPhone ha sido utilizada por los usuarios. Según Google, el iPhone genera 50 veces más búsqueda que cualquier otro microteléfono móvil.” (Wikipedia, s.f.)

2.5 DESARROLLO DE SOFTWARE.

2.5.1 UML.

2.5.1.1 ¿Qué es UML?.

"Las siglas UML significan Lenguaje Unificado de Modelado y es el que prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación.

2.5.1.2 ¿Qué fases comprende UML?.

UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. UML ofrece nueve diagramas que consisten en modelar sistemas.

- Diagramas de Casos de Uso para modelar los procesos 'business'.
- Diagramas de Secuencia para modelar el paso de mensajes entre objetos.
- Diagramas de colaboración para modelar interacciones entre objetos.
- Diagramas de Estado para modelar el comportamiento de los objetos en el sistema.
- Diagramas de Actividad para modelar el comportamiento de los Casos de Uso, objetos u operaciones.
- Diagramas de Clases para modelar la estructura estática de las clases en el sistema.
- Diagramas de Objetos para modelar la estructura estática de los objetos en el sistema.
- Diagramas de Componentes para modelar componentes.
- Diagramas de Implementación para modelar la distribución del sistema.

- UML es una consolidación de muchas de las notaciones y conceptos más usadas orientados a objetos.

2.5.1.3 ¿Qué ofrece UML?.

UML prescribe una notación estándar y semánticas esenciales para el modelado de un sistema orientado a objetos. Previamente, un diseño orientado a objetos podría haber sido modelado con cualquiera de la docena de metodologías populares, causando a los revisores tener que aprender las semánticas y notaciones de la metodología empleada antes que intentar entender el diseño en sí. Ahora con UML, diseñadores diferentes modelando sistemas diferentes pueden sobradamente entender cada uno los diseños de los otros.

2.5.1.4 ¿Cuáles son las Utilidades del Lenguaje UML?.

El intercambio de información de diseño e ideas usando la notación UML sería hecho en los medios que siempre han sido populares: pizarras, cuadernos y trozos de papel por nombrar algunos. Pero UML se sirve mejor por una herramienta de modelado, la cual puede ser usada para capturar, guardar, rechazar, integrar automáticamente información, y diseño de documentación.

Una característica que beneficia a los modeladores, UML también hace más fácil escoger una herramienta de modelado. Hace tiempo, el modelador primero tenía que seleccionar una notación de metodología, y después estaba limitado a seleccionar una herramienta que la soportara. Ahora con UML como estándar, la elección de notación ya se ha hecho para el modelador. Y con todas las herramientas de modelado soportando UML, el modelador puede seleccionar la herramienta basada en las áreas claves de funcionalidad soportadas que permiten resolver los problemas y documentar las soluciones.

Como una buena caja de herramientas, una buena herramienta de

modelado ofrece todas las herramientas necesarias para conseguir hacer eficientemente varios trabajos, sin dejarle nunca sin la herramienta correcta.” (Geofisica, 2010)

2.6 WEB SERVICE (Servicio Web).



Figura # 2.10 Web Service.

"Un Web Service es un componente de software que se comunica con otras aplicaciones codificando los mensaje en XML y enviando estos mensajes a través de protocolos estándares de Internet tales como el Hypertext Transfer Protocol (HTTP). Intuitivamente un Web Service es similar a un sitio web que no cuenta con un interfaz de usuario y que da servicio a las aplicaciones en vez de a las personas. Un Web Service, en vez de obtener solicitudes desde el navegador y retornar páginas web como respuesta, lo que hace es recibir solicitudes a través de un mensaje formateado en XML desde una aplicación, realiza una tarea y devuelve un mensaje de respuesta también formateado en XML.

Microsoft y otras empresas líderes están promocionando SOAP como estándar de los mensajes para los Web Services. Un mensaje SOAP se parece mucho a una carta, es un sobre que contiene una cabecera con la dirección del receptor del mensaje, un conjunto de opciones de entrega (tal como la información de encriptación), y un cuerpo o body con la información o data del mensaje.

2.6.1 Ejemplo de conexión de un web service con dispositivos móviles.

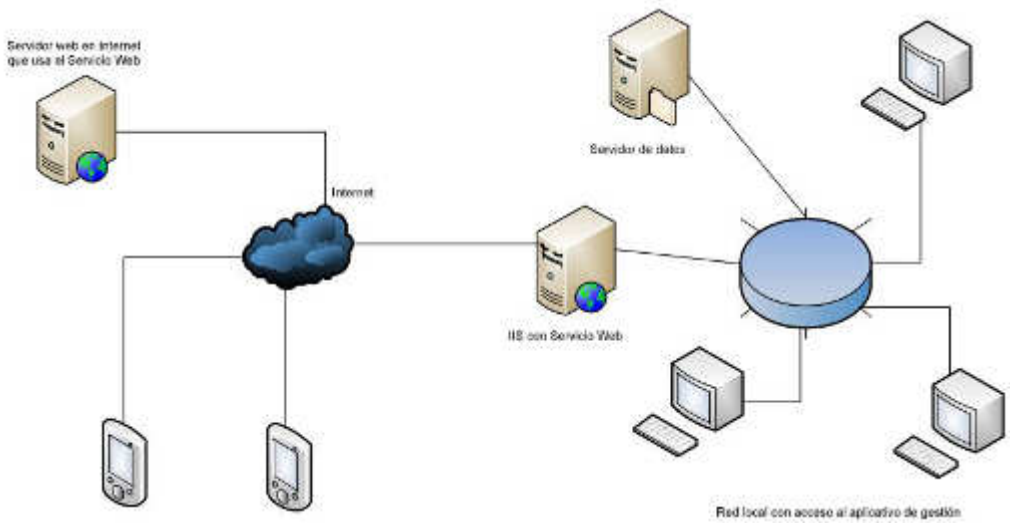


Figura # 2.11 Conexión Web Service mediante dispositivo móvil.

2.6.2 El lenguaje de intercambio XML.

Se define como:

"Toda la información disponible para cualquier persona, en cualquier lugar, a través de cualquier dispositivo."

Debido a la gran masificación de Internet a niveles insospechables y al gran impacto causado por las tecnologías de la información en las últimas dos décadas del siglo pasado, la manera de hacer negocios y la comunicación entre las personas y las empresas cambió de una manera rotunda. Bajo este contexto se hacía cada vez mayor la necesidad de integrar y compartir información entre distintas plataformas de software y hardware.

2.6.2.1 XML Web Services.

Los Web Services son un modelo de programación para la comunicación entre aplicaciones. La conexión de aplicaciones a través de la Internet mejorará la capacidad de las empresas para trabajar conjuntamente con sus socios de negocio, proveedores y clientes. Creando una capa de Web Services sobre una aplicación corporativa existente, las organizaciones podrán permitir que sistemas externos puedan invocar las funciones de la aplicación a través de Internet (o una intranet corporativa) sin tener que modificar la aplicación misma.

2.6.3 Requisitos de un Web Service.

- Interoperabilidad.
- Amigabilidad con Internet.
- Interfaces fuertemente tipadas.
- Posibilidad de aprovechar los estándares de Internet existentes.
- Soporte para cualquier lenguaje:.
- Soporte para cualquier infraestructura de componente distribuida. (C., DesarrolloWeb, 2004)

2.7 PROTOCOLO DE CONEXIÓN ENTRE APLICACIÓN SFPUCE MÓVIL HACIA EL WEB SERVICE INSTITUCIONAL (SOAP).

"Son las siglas de Simple Object Access Protocol. Con este protocolo se pedían realizar RPC o remote procedure calls, es decir, podíamos bien en cliente o servidor realizar peticiones mediante http a un servidor web. Los mensajes debían tener un formato determinado empleando XML para encapsular los parámetros de la petición. Con el paso del tiempo el proyecto interesó a Importantes multinacionales entre las que se encuentran IBM y Microsoft y de este interés por XML-RPC se desarrolló SOAP."

En el núcleo de los servicios Web se encuentra el protocolo simple de acceso a datos SOAP, que proporciona un mecanismo estándar de empaquetar mensajes. SOAP ha recibido gran atención debido a que facilita una comunicación del estilo RPC (llamadas a procedimientos remotos) entre un cliente y un servidor remoto.

Una de las razones principales es que SOAP ha recibido un increíble apoyo por parte de la industria. SOAP es el primer protocolo de su tipo que ha sido aceptado prácticamente por todas las grandes compañías de software del mundo. Compañías que en raras ocasiones cooperan entre sí están ofreciendo su apoyo a este protocolo. ALGÚNas de las mayores Compañías que soportan SOAP son Microsoft, IBM, SUN, Microsystems, SAP y Ariba.

2.7.1 VENTAJAS DEL PROTOCOLO SOAP.

Algunas de las ventajas de SOAP son:

- **No está asociado con ningún lenguaje**:

Los desarrolladores involucrados en nuevos proyectos pueden elegir desarrollar con **el último y mejor lenguaje de programación que exista** pero los

desarrolladores responsables de mantener antiguas aflicciones heredadas podrían no poder hacer esta elección sobre el lenguaje de programación que utilizan. SOAP no especifica una API, por lo que la implementación de la API se deja al lenguaje de programación, como en Java, y la plataforma como Microsoft .Net.

- **No se encuentra fuertemente asociado a ningún protocolo de transporte**:

  La especificación de SOAP no describe como se deberían asociar los mensajes de SOAP con HTTP. Un mensaje de SOAP no es más que un documento XML, por lo que puede transportarse utilizando cualquier protocolo capaz de transmitir texto.

- **No está atado a ninguna infraestructura de objeto distribuido**:

  La mayoría de los sistemas de objetos distribuidos se pueden extender, y ya lo están algunos de ellos para que admitan SOAP.

- **Aprovecha los estándares existentes en la industria**:

  Los principales contribuyentes a la especificación SOAP evitaron, intencionadamente, reinventar las cosas. Optaron por extender los estándares existentes para que coincidieran con sus necesidades. Por ejemplo, SOAP aprovecha XML para la codificación de los mensajes, en lugar de utilizar su propio sistema de tipo que ya están definidas en la especificación esquema de XML. Y como ya se ha mencionado SOAP no define un medio de trasporte de los mensajes; los mensajes de SOAP se pueden asociar a los protocolos de transporte existentes como HTTP y SMTP.

- **Permite la interoperabilidad entre múltiples entornos**:

  SOAP se desarrolló sobre los estándares existentes de la industria, por lo que las aplicaciones que se ejecuten en plataformas con dicho estándares pueden

comunicarse mediante mensaje SOAP con aplicaciones que se ejecuten en otras plataformas. Por ejemplo, una aplicación de escritorio que se ejecute en una PC puede comunicarse con una aplicación del back-end ejecutándose en un mainframe capaz de enviar y recibir XML sobre HTTP.

- **Anatomía de un mensaje de SOAP:**

SOAP proporciona un mecanismo estándar de empaquetar un mensaje.

Un mensaje SOAP se compone de un sobre que contiene el cuerpo del mensaje y cualquier información de cabecera que se utiliza para describir el mensaje. A continuación tiene un ejemplo:" (Maestros del Web, 2009)

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <!--Optional header information goes here. -->
    <To>Scott</To>
    <From>Suzanne</From>
  </soap:Header>
  <soap:Body>
    <!--Message goes here. -->
    Please pick up some milk on your way home from work.
  </soap:Body>
</soap:Envelope>
```

Figura # 2.12 Mensaje soap.

2.8 BASE DE DATOS.



Figura # 2.13 Base de datos.

"Una base de datos es un "almacén" que nos permite guardar grandes cantidades de información de forma organizada para que luego podamos encontrar y utilizar fácilmente.
Una base de datos se puede definir como un conjunto de información relacionada que se encuentra agrupada o estructurada.

Desde el punto de vista informático, la base de datos es un sistema formado por un conjunto de datos almacenados en discos que permiten el acceso directo a ellos y un conjunto de programas que manipulen ese conjunto de datos.

Cada base de datos se compone de una o más tablas que guarda un conjunto de datos. Cada tabla tiene una o más columnas y filas. Las columnas guardan una parte de la información sobre cada elemento que queramos guardar en la tabla, cada fila de la tabla conforma un registro.

2.8.1 Características de base de datos.

Entre las principales características de los sistemas de base de datos podemos mencionar:

- Independencia lógica y física de los datos.
- Redundancia mínima.

- Acceso concurrente por parte de múltiples usuarios.
- Integridad de los datos.
- Consultas complejas optimizadas.
- Seguridad de acceso y auditoría.
- Respaldo y recuperación.
- Acceso a través de lenguajes de programación estándar." (Maestros del Web, 2009)

2.9 SQL.

"Debido a la diversidad de lenguajes y de bases de datos existentes, la manera de comunicar entre unos y otras sería realmente complicada a gestionar de no ser por la existencia de estándares que nos permiten el realizar las operaciones básicas de una forma universal.

Es de eso de lo que trata el Structured Query Language que no es más que un lenguaje estándar de comunicación con bases de datos.

SQL es un lenguaje universal dedicado a las bases de datos. Es un lenguaje de sintaxis simple y muy potente. Mediante él se puede recorrer, modificar o borrar registros de las tablas de datos.

El lenguaje se utiliza para definir, consultar y actualizar la base de datos, y es el más popular de su estilo.

El SQL posibilita la realización de consultas para recuperar información de las bases de datos de forma sencilla.



Figura # 2.14 SQL.

El SQL está considerado como un lenguaje declarativo de alto nivel ya que, al manejar conjuntos de registros y no registros individuales, ofrece una elevada productividad en la codificación y en la orientación a objetos. Una sentencia de SQL puede resultar equivalente a más de un programa que utilice un lenguaje de bajo nivel." (Alegsa, s.f.)

2.10 VIRTUALIZACIÓN.

2.10.1 Mac OS X Snow Leopard.



Figura #2.15 Mac OS X Snow Lepard.

"Mac OS X es fácil de usar y sumamente potente. Todo en él desde el escritorio al arrancar el Mac hasta las aplicaciones que se usa a diario ha sido diseñado pensando en la simplicidad y en la elegancia. Por eso, tareas como navegar por Internet, mirar el correo o entablar videoconferencias con un amigo de otro continente resultan fáciles y divertidas. Naturalmente, hacer simples las tareas más increíbles requiere una tecnología avanzada, por eso Mac OS X está construido sobre robustos cimientos UNIX que proporcionan niveles inigualables de estabilidad y compatibilidad con los estándares de Internet.

Optimizado, no reinventado.

Mac OS X es famoso por su sencillez, fiabilidad y facilidad de uso. Así que cuando hubo que diseñar Snow Leopard, los ingenieros de Apple solo tenían un reto, convertir un sistema operativo genial en algo aún mejor. Buscaron áreas que se pudieran perfeccionar, simplificar y acelerar, desde pequeños detalles como expulsar discos externos hasta aspectos importantes como la instalación del OS. En muchos casos han conseguido convertir lo asombroso en algo literalmente espectacular.

Hardware y software, integrados a la perfección.

Los programas de todos los Mac están creados por la misma empresa que fabrica el equipo, por lo que el resultado es un ordenador completamente integrado.
Por ejemplo, detecta automáticamente las redes Wi-Fi disponibles y se conecta a ellas y cuando el sistema operativo sale de una hibernación se vuelve a conectar a las redes para que se retome el trabajo rápidamente. Además hace que el disco duro gire más lento cuando está inactivo, decidiendo de forma inteligente qué procesador (el de la CPU o la GPU) es más adecuado para realizar una tarea.

Entre las características más importantes de Mac OS X Snow Leopard para el desarrollo de la aplicación SFPUCE móvil se destaca lo siguiente:

Diseñado para la seguridad.

Sin que apenas tenga que poner nada de su parte, Mac OS X se protege a sí mismo le protege a usted de virus, programas maliciosos y otras amenazas. Este

sistema operativo ha sido creado para Internet en la era de Internet, y ofrece numerosas formas de defensa frente a las amenazas de la Web.

Como todos los Mac vienen con una configuración segura, no tiene que preocuparse por entender ajustes complejos, y lo que es mejor, no aparecerán constantes avisos de seguridad que entorpezcan sus tareas. Además, Apple responde con rapidez a las amenazas de Internet y ofrece actualizaciones de seguridad de forma directa y automática.

Defensa contra virus y programas dañinos.

Sin que apenas Ud. tenga que intervenir, Mac OS X ofrece un sistema multicapa de defensa contra virus y otros programas dañinos peligrosos.
Por ejemplo, impide que los intrusos dañen sus programas mediante la técnica de "jaulas". Para ello, restringe las acciones que los programas pueden ejecutar en el Mac, los archivos a los que pueden acceder y el resto de programas que pueden iniciar.

Entre otras prestaciones automáticas de seguridad se encuentran Library Randomization, que impide que los comandos dañinos encuentren su objetivo, y Execute Disable, que protege a la memoria del Mac de ataques." *(Mundomacpunta, s.f.)*

2.11 Isimulator (SDK de Iphone).



Figura # 2.16 Simulador de Iphone Isimulator.

"El iPhone SDK es un Kit de desarrollo de software para la plataforma iOS de Apple Inc., con el objetivo de permitir a terceros desarrollar aplicaciones nativas para el iOS (sistema operative de Iphone). Fue liberado en Febrero del 2008.

El SDK es un kit de herramientas que le permitirán a los desarrolladores de software independiente crear y vender programas para el iPhone y el iPod Touch y además, va a dejar que puedan llegar los videojuegos para la plataforma de Apple.

Otro punto importante de este kit es la facilidad con la que se puede manejar. Además, la versatilidad, lo que significa que se van a aprovechar al máximo todas las posibilidades que tiene el iPhone como por ejemplo su tecnología de localización, su interfaz táctil, etc.

Cuando se anunció por primera vez, el iPhone sólo permite aplicaciones basadas en web, pero no las aplicaciones que los usuarios puedan instalar en los propios dispositivos. El impulso para crear aplicaciones nativas era fuerte, sin embargo, y muy pronto los desarrolladores han roto la seguridad del iPhone y comenzó a crear sus propias aplicaciones, con diversas respuestas de Apple.

El SDK incluye un entorno de desarrollo, de escritorio simulador de iOS para la optimización de las herramientas de prueba, y de arrastrar y soltar constructor de interfaces, y el acceso a la documentación y soporte.” (Analia, 2008)

2.12 VIRTUAL BOX.



Figura # 2.17 Presentación software virtual box.

"Por medio de esta aplicación es posible instalar sistemas operativos adicionales, conocidos como "sistemas invitados", dentro de otro sistema operativo "anfitrión", cada uno con su propio ambiente virtual.
VirtualBox es básicamente un programa de virtualización.

La virtualización en informática es un concepto que está creciendo mucho en estos días y es la posibilidad de estar corriendo más de un sistema operativo simultáneamente. Esto quiere decir que se puede usar algún sistema operativo base como Linux / Windows o Mac OS y sobre él con una máquina virtual, creando un disco virtual y usando una parte de la memoria y vídeo del anterior podemos instalar cualquier otro sistema operativo.
Las herramientas de virtualización se han puesto de moda de manera casi exponencial durante los últimos años, como VirtualBox; esta corre sobre diferentes sistemas operativos y para los desarrolladores e ingenieros en cómputo que desean realizar pruebas de productos sin afectar sus máquinas, esta es una de las mejores alternativas. Adicionalmente hay algo que tiene VirtualBox que aún es mejor, es un program Open Source, eso significa que se podrá usar en una máquina teniendo un costo de $0.

VENTAJAS:

- Virtualiza la mayoría de sistemas operativos.
- Buena integración entre la máquina virtual y el PC.
- Soporte para USB y otras conexiones.
- Comparte ratón/teclado y carpetas.
- Entorno seguro para realizar pruebas.

DESVENTAJAS:

- A veces la conexión USB da problemas.
- Exige un equipo potente.

VirtualBox es el software de virtualización más populares entre los escritorios de miles de usuarios. Su facilidad de uso unido a su potencia hacen de éste software un referente para poder probar y correr sistemas operativos en nuestros PC sin tener que particionar, redimensionar o formatear nuestros discos. Todo de manera muy intuitiva y con la ayuda de un interfaz visual muy logrado." (Hardsoftgeek, 2007)

2.13 SEGURIDAD ELECTRÓNICA.

"En este caso trataremos acerca de la seguridad electrónica de un sitio Web, en este caso, seguridad electrónica de información confidencial de la aplicación de banca móvil.

Seguridad Web.

Hasta el momento, se ha presentado un web que ofrece un acceso abierto a un conjunto de información que explícitamente se hace pública. Sin embargo, en determinadas circunstancias, es interesante poder limitar el acceso a documentos reservados o útiles para un conjunto restringido de personas. Se pueden establecer dos tipos de restricciones:

1. Limitación de acceso en función de direcciones IP o dominio. Sólo los usuarios de un dominio u organización tendrán acceso a la información.

2. Limitación de acceso por nombres de usuario y claves de acceso. Sólo los usuarios que conozcan una clave de acceso válida pueden acceder a la información.

Otro aspecto que está cobrando especial importancia es la seguridad de la información que se intercambia en el Web. La explotación comercial de Internet exige disponer de sistemas de comunicación seguros, capaces de adaptarse a las necesidades de los nuevos servicios, como la compra electrónica o la banca a distancia. En estos servicios, se manejan dos conceptos fundamentales, la autentificación (garantizar que tanto el usuario de un cliente Web como un determinado servidor de información son quienes dicen ser) y la confidencialidad (hacer que la información intercambiada no pueda ser interceptada por terceros).

Control de acceso a la información.

Se utiliza para limitar el acceso a determinados documentos de un servidor Web, en función del origen y tipo de petición.

Ningún lenguaje de programación puede prever código inseguro, aunque las características del lenguaje puedan ayudar a bloquear revelación de información confidencial, que para nuestro proyecto es importante.

La programación deficiente conduce a este tipo de ataques, por lo que proporcionare ALGÚNas medidas de seguridad para prevenir vulnerabilidades futuras, como en este caso la implementación del lenguaje conocido Objetive C. (Castellanos, 2010)

2.14 MANEJO DE PROTOCOLOS DE SEGURIDAD.

2.14.1 FIREWALL.



Figura # 2.18 Protocolo de seguridad firewall.

Es una parte de un sistema o una red que está diseñada para bloquear el acceso no autorizado, permitiendo al mismo tiempo comunicaciones autorizadas. Se trata de un dispositivo o conjunto de dispositivos configurados para permitir, limitar, cifrar, descifrar, el tráfico entre los diferentes ámbitos sobre la base de un conjunto de normas y otros criterios.

Los cortafuegos pueden ser implementados en hardware o software, o una combinación de ambos. Los cortafuegos se utilizan con frecuencia para evitar que los usuarios de Internet no autorizados tengan acceso a redes privadas conectadas a Internet, especialmente intranets. Todos los mensajes que entren o salgan de la intranet pasan a través del cortafuego, que examina cada mensaje y bloquea aquellos que no cumplen los criterios de seguridad especificados.

Un firewall es un dispositivo que filtra el tráfico entre redes, como mínimo dos. El firewall puede ser un dispositivo físico o un software sobre un sistema operativo. En general debemos verlo como una caja con DOS o más interfaces de red en la que se establecen una reglas de filtrado con las que se decide si una

conexión determinada puede establecerse o no. Incluso puede ir más allá y realizar modificaciones sobre las comunicaciones. (Slideshare, 2010)

2.14.2 CHECKPOINT.



Figura # 2.19 Protocolo de seguridad checkpoint.

¿Qué tipo de empresas eligen CheckPoint?

Empresas de todas las industrias y todos los tamaños confían la seguridad de su información y de sus equipos a CheckPoint, ya que diseñó una amplia gama de soluciones flexibles, móviles y fácilmente adaptables a cada necesidad.

CheckPoint desarrolló avanzados dispositivos de seguridad para pequeñas empresas, que protegen desde 5 hasta 150 usuarios conectados a la red. Por otro lado, cuenta con una avanzada línea de soluciones, como por ejemplo, Software Blade, para satisfacer las necesidades de las corporaciones que trabajan con más de 1000 PCs.

2.14.3 ENCUESTA DE CHECKPOINT A INSTITUCIÓN FINANCIERA SAN FRANCISCO.

¿Al trabajar con Internet, a qué riesgos está expuesta la Cooperativa San Francisco?

"Si se trabaja con una aplicación de Banca Móvil por medio de Internet para acceder a un Web Service, y no se cuenta con ninguna solución de seguridad, la empresa estará vulnerable frente a atentados de hackers, virus y hasta puede padecer estafas en línea."

¿Qué clase de vulnerabilidades puede presentar la Aplicación de Banca Móvil SFPUCE Móvil dentro de la Institución Financiera San Francisco?

"En todo lo relacionado con el software siempre tendremos fallas en el desarrollo o comprobaciones mal hechas en el comportamiento del mismo."

"Esto hará ante cierta acción del usuario o atacante sobre el software haga que este reaccione de una forma inesperada, como por ejemplo otorgándole al usuario mayores permisos de acceso, que la aplicación se congele, se corrompan datos, se quede expuesta información confidencial, y en el peor de los casos se manipule de mala manera cantidades de dinero etc." (Torres, 2010)

2.15 IPTABLES DE LINUX.

"IPtables es un sistema de firewall vinculado al kernel de Linux.
Un firewall de iptables no es como un servidor que se lo inicia o se lo detiene o que se pueda caer por un error de programación (esto es una pequeña mentira, ha tenido alguna vulnerabilidad que permite DOS, pero nunca tendrá tanto peligro como las aplicaciones que escuchan en determinado puerto TCP); iptables está integrado con el kernel, es parte del sistema operativo.

Se pone en marcha aplicando reglas.
Para ellos se ejecuta el comando iptables, con el que añadimos, borramos, o creamos reglas. Por ello un firewall de iptables no es sino un simple script de Shell en el que se van ejecutando las reglas de firewall." (Izura, s.f.)

CAPÍTULO III

3.1 DESARROLLO.

3.1.1 METODOLOGÍA DE INVESTIGACIÓN.

Enfoque Cuantitativo

El proyecto utiliza un enfoque cuantitativo ya que utiliza la recolección y el análisis de datos para contestar preguntas de investigación y probar hipótesis establecidas previamente y confía en la medición numérica, el conteo y frecuentemente en el uso de la estadística para establecer con exactitud patrones de comportamiento de una población

TIPO

Investigación Bibliográfica

El proyecto se desarrolla con una investigación bibliográfica que es aquella etapa donde se explora sobre un determinado tema o problema.

La investigación bibliográfica constituye una excelente introducción a todos los otros tipos de investigación, además de que constituye una necesaria primera etapa de todas ellas, puesto que ésta proporciona el conocimiento de las investigaciones ya existentes teorías, hipótesis, experimentos, resultados, instrumentos y técnicas usadas acerca del tema o problema que el investigador se propone investigar o resolver.

3.1.2 POBLACIÓN Y MUESTRA.

3.1.2.1 Definición de Población.

Es el conjunto de individuos, organismo o entes inanimados de los cuales queremos conocer alguna o algunas características.

Población.

Número de Clientes actuales en la Cooperativa San Francisco: 135.996 clientes (incluidos quienes tienen cuentas activas e inactivas).

Número de Clientes actuales en la Cooperativa San Francisco que tienen cuentas activas: 64.865 clientes con cuentas activas.

3.1.2.2 Definición Muestra.

La muestra estadística representa al universo, a la serie infinita de valores a los que nunca tendremos acceso. Si esto se logra, nos permite predecir las propiedades del universo a partir de los valores de la muestra.

Por lo tanto al mencionar el término muestra debemos tener en cuenta 2 conceptos básicos para la comprensión y captación a lo que se quiere llegar a calcular.

Muestra.

La muestra estadística representa al universo, a la serie infinita de valores a los que nunca tendremos acceso. Si esto se logra, nos permite predecir las propiedades del universo a partir de los valores de la muestra.

Por lo tanto al mencionar el término muestra debemos tener en cuenta 2 conceptos básicos para la comprensión y captación a lo que se quiere llegar a calcular.

Parte de una población seleccionada por técnicas estadísticas, en la que a cada uno de sus miembros se toma la característica que se quiere estudiar.



3.1.2.3 Fórmula para calcular el tamaño de la Muestra.

$$N = \frac{z^2 N\, G^2}{(N-1)E^2 + G^2\, Z^2}$$

3.1.2.4 Especificación de Caracteres de Fórmula.

N= Tamaño de Población (64.865 clientes con cuentas activas)

G^2= 0.25

E= Error (margen de error 0.07)

Z= Nivel Confiable 1.96 (95%)

Cálculo de Muestra Implementando Fórmula.

$$N = \frac{3.8416 * 64865 * 0.25}{63999 * 0.0049 + 0.25 * 3.8416}$$

$$N = \frac{62396.3}{314.55}$$

N= 198.366

3.1.2.5 Resumen Población y Muestra.

Población.

135.996 clientes (incluidos quienes tienen cuentas activas e inactivas).

64.865 clientes con cuentas activas.

Muestra.

195.4080 clientes

3.2 METODOLOGÍA DE DESARROLLO.

3.2.1 Metodología RUP.

Como metodología de desarrollo se va a utilizar el proceso unificado de desarrollo de software (RUP) el cual es un proceso de desarrollo de software genérico que puede ser utilizado para una gran cantidad de tipos de sistemas, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de competencia y diferentes tamaños de proyectos.

Provee un enfoque disciplinado en la asignación de tareas y responsabilidades dentro de una organización de desarrollo. Su meta es asegurar la producción de software de muy alta calidad que satisfaga las necesidades de los usuarios finales, dentro de un calendario y presupuesto predecible y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

UML es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

Además ofrece un estándar para describir un plano del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

3.2.2 ANÁLISIS.

Recopilación de información.

Para la recopilación de información y obtener el equivalente exacto a la población y muestra que se citó anteriormente, se utilizó como técnica la siguiente encuesta:

3.2.2.1 Encuesta.

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR SEDE AMBATO
SERÉIS MIS TESTIGOS

SAN FRANCISCO

ENCUESTA DE BANCA MOVIL PARA LOS SOCIOS DE LA COOPERATIVA SAN FRANCISCO

Objetivo:

Determinar las preferencias de los socios en cuanto a modelo de teléfono celular y servicios financieros de banca móvil requeridos.

ESTIMADO SOCIO:

Indique con una X su respuesta:

1.- ¿CONOCE USTED ACERCA DE LAS VENTAJAS Y SERVICIOS DE BANCA MOVIL?

SI ( ) NO ( )

2.- ¿LA COOPERATIVA FINANCIERA SAN FRANCISCO OFRECE ALGÚN SERVICIO DE CONSULTA DE DATOS A TRAVES DE TELEFONIA CELULAR?

SI ( ) NO ( )

3.- ¿LE GUSTARIA CONTAR CON UN NUEVO SERVICIO, DENTRO DE LA COOPERATIVA, QUE LE PERMITA CONSULTAR TODA SU RESPECTIVA INFORMACION BANCARIA CONFIDENCIAL A TRAVES DE SU TELEFONO CELULAR?

SI ( ) NO ( )

4.- ¿CONTANDO CON ESTE SERVICIO EN QUE TELEFONO CELULAR LE GUSTARIA IMPLEMENTARLO?

BLACKBERRY ( )

IPHONE ( )

OTROS ( ) ……

5.- ¿QUE TIPO DE CONSULTAS LE GUSTARIA REALIZAR A TRAVES DE SU TELEFONO CELULAR?

( ) Consulta de Saldos

( ) Consulta de Préstamos

( ) Fecha de pago de cuota

( ) Fecha de vencimiento de cuota

( ) Consulta de últimos movimientos

( ) Transferencia entre cuentas de la misma institución

GRACIAS POR SU PARTICIPACION ☺

3.2.2.2 Entrevista.

¿La Institución cuenta actualmente con una aplicación de consulta de información bancaria de un socio a través de telefonía móvil?

¿Qué ventajas brinda el Kiosco Informático de la Institución?

¿Qué desventaja brinda el Kiosco Informático de la Institución?

¿Se ha recibido quejas acerca del servicio del Kiosco informático de la Institución?

¿El plan estratégico de innovación de mercado Institucional ha pensado en algún momento en implementar una tecnología de consulta Móvil?

¿Actualmente la Institución Financiera San Francisco cuenta con ALGÚNa aplicación de Banca Móvil?

¿Cuál es la misión institucional en cuanto a satisfacer en atención al Cliente?

¿Le agradaría implementar una aplicación de banca móvil de consulta en la institución para mejorar la atención al cliente y comodidad de los usuarios?
¿La infraestructura tecnológica de la Institución es adecuada para trabajar de una manera favorable con una aplicación de Banca Móvil?

¿Cuál cree UD. que serian las nuevas ventajas de trabajar en una empresa con tecnología móvil?

Análisis de la Información.

3.2.2.3 Análisis de la información recopilada.

Se realizó la encuesta a 198 socios activos de la Cooperativa San Francisco basándose en el resultado de la muestra, obteniendo el siguiente análisis en cada pregunta.

1.- ¿CONOCE USTED ACERCA DE LAS VENTAJAS Y SERVICIOS DE BANCA MÓVIL?.

RESULTADOS PREGUNTA 1			
SI	**NO**	**%SI**	**%NO**
77	118	39%	61%

Tabla. # 3.1 Tabla de resultados y porcentaje pregunta 1.

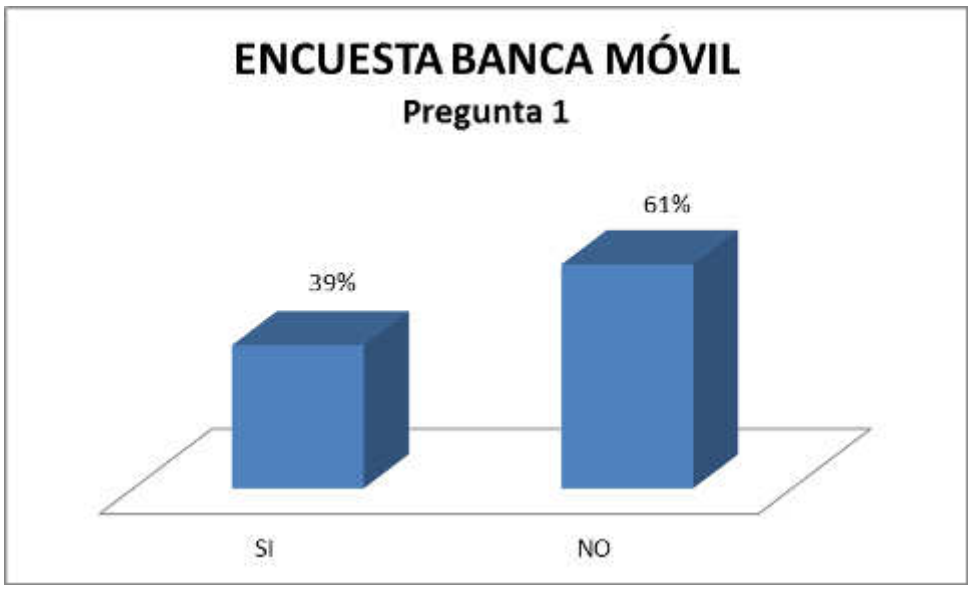


Figura. # 3.1 Gráfico de resultados y porcentaje pregunta 1.

ANÁLISIS PREGUNTA #1.

Claramente esta pregunta muestra que la mayoría de los socios no tienen conocimiento del beneficio que abarca el tema banca móvil.

Por ende, esto quiere decir que nuestro entorno social de clientes carece de capacitación de tecnología Banca Móvil.

2.- ¿LA COOPERATIVA FINANCIERA SAN FRANCISCO OFRECE ALGÚN SERVICIO DE CONSULTA DE DATOS A TRAVES DE TELEFONÍA CELULAR?.

RESULTADOS PREGUNTA 2			
SI	**NO**	**%SI**	**%NO**
7	188	3%	97%

Tabla # 3.2 Tabla de resultados y porcentaje pregunta 2.

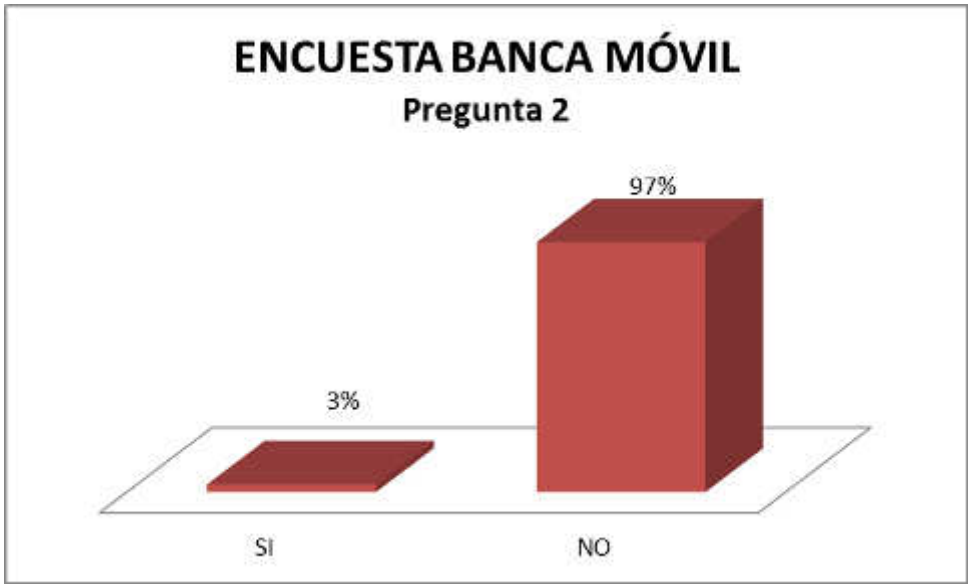


Figura # 3.2 Gráfico de resultados y porcentaje pregunta 2.

ANÁLISIS PREGUNTA #2.

En la actualidad la Cooperativa San Francisco cuenta con un kiosco informático para consultas pero no a través de telefonía celular, se evidencia en la respuesta dada por los socios.

3.- ¿LE GUSTARÍA CONTAR CON UN NUEVO SERVICIO, DENTRO DE LA COOPERATIVA, QUE LE PERMITA CONSULTAR TODA SU RESPECTIVA INFORMACIÓN BANCARIA CONFIDENCIAL A TRAVES DE SU TELÉFONO CELULAR?.

RESULTADOS PREGUNTA 3			
SI	**NO**	**%SI**	**%NO**
182	13	93%	7%

Tabla # 3.3 Tabla de resultados y porcentaje pregunta 3.

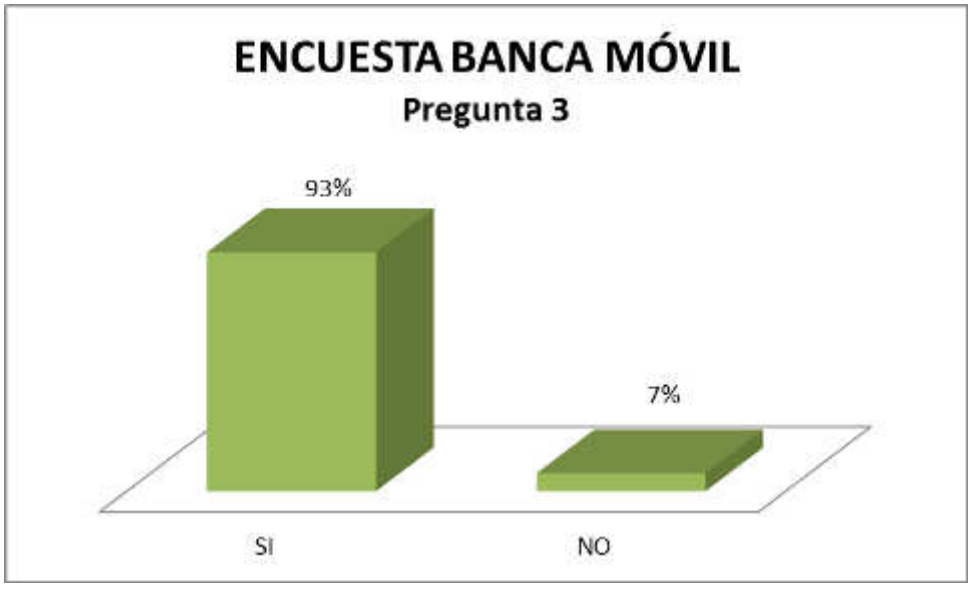


Figura # 3.3 Gráfico de resultados y porcentaje pregunta 3.

ANÁLISIS PREGUNTA #3.

Los resultados de esta pregunta indican, que si existe la aprobación y el interés de la gran mayoría de los socios, para implementar un nuevo servicio de consultas bancarias a través de telefonía celular.

4.- ¿CONTANDO CON ESTE SERVICIO EN QUE TELÉFONO CELULAR LE GUSTARÍA IMPLEMENTARLO?

RESULTADOS PREGUNTA 4					
BLACKBERRY	**IPHONE**	**OTROS**	**% BLACKBERRY**	**% IPHONE**	**% OTROS**
89	90	38	45%	46%	9%

Tabla # 3.4 Tabla de resultados y porcentaje pregunta 4.

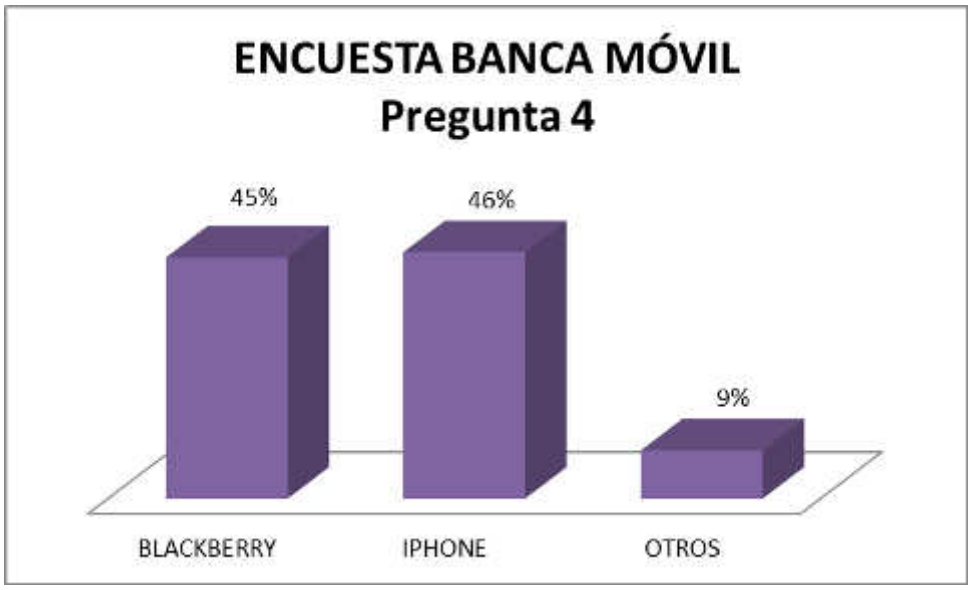


Figura # 3.4 Gráfico de resultados y porcentaje pregunta 4.

ANÁLISIS PREGUNTA #4.

Esta pregunta demuestra que la gran mayoría de los socios están interesados por el teléfono Blackberry y por el dispositivo móvil Iphone, lo cual indica que la aplicación a implementarse en el celular Iphone es idóneo en su gran mayoría.

5.- ¿QUE TIPO DE CONSULTAS LE GUSTARÍA REALIZAR A TRAVES DE SU TELÉFONO CELULAR?

( ) Consulta de Saldos.

( ) Consulta de Préstamos.

( ) Fecha de pago de cuota.

( ) Fecha de vencimiento de cuota.

( ) Consulta de últimos movimientos.

( ) Transferencia entre cuentas de la misma institución.

RESULTADOS PREGUNTA 5					
Consulta de Saldos	**Consulta de Préstamos**	**Fecha de pago de cuota**	**Fecha de vencimiento de cuota**	**Consulta de últimos movimientos**	**Transferencia entre cuentas de la misma institución**
177	157	127	129	140	142
91%	80%	65%	66%	71%	73%

Tabla # 3.5 Tabla de resultados y porcentaje pregunta 4.

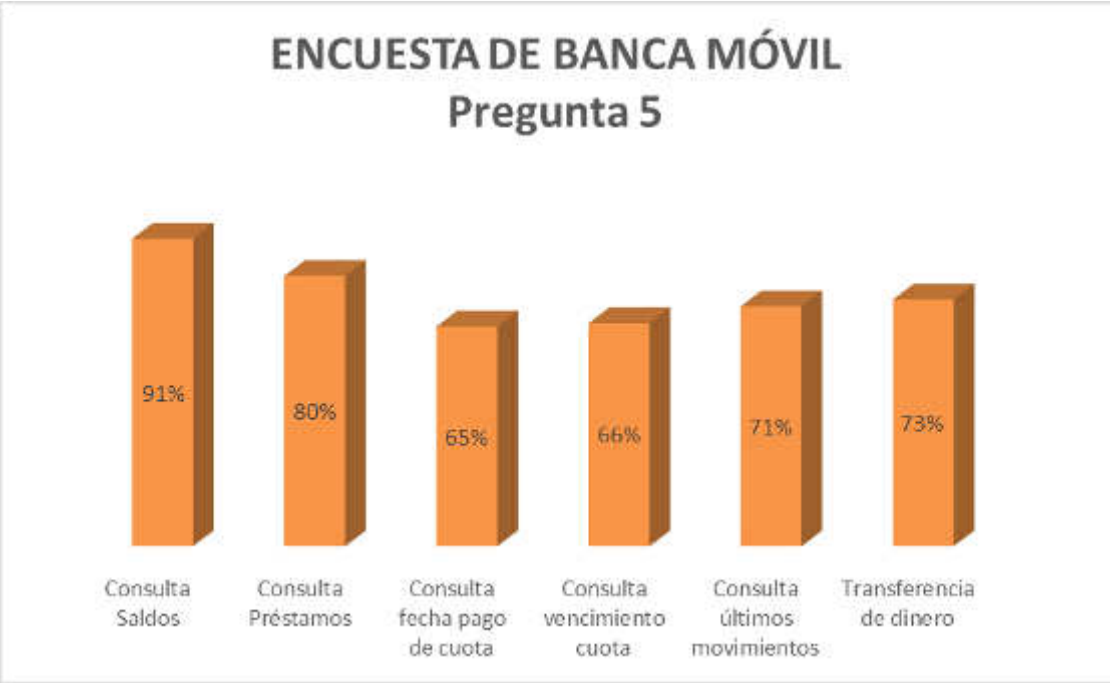


Figura # 3.5 Gráfico de resultados y porcentaje pregunta 5.

ANÁLISIS PREGUNTA #5.

Los diferentes tipos de consultas expuestos en la encuesta, son de gran aceptación por los socios debido a la importancia de respuesta de información, concluyendo que todas estas consultas son de mayor necesidad para una persona perteneciente a la cooperativa de ahorro y crédito San Francisco.

3.2.3 ENTREVISTA.

Para el estudio de innovación de mercado y factibilidad de tecnología que puede lograr y abarcar una aplicación de banca móvil como SFPUCE móvil, se realizó la siguiente entrevista con varias preguntas al Ing. Diego Torres Jefe de Sistemas dentro del departamento de Tecnología en la Institución Financiera San Francisco Ltda.

¿La Institución cuenta actualmente con una aplicación de consulta de información bancaria de un socio a través de telefonía móvil?

Actualmente no posee, ya que se realiza consultas muy limitadas a través de la página web de la institución. Pero en cuanto a consultas por vía celular pues no se lo ha realizado.

¿Qué ventajas brinda el Kiosco Informático de la Institución?

Proporciona una vía rápida y segura de consultas de saldos y de préstamos a los socios, mejora la atención al cliente, al evitar aglomeraciones en las áreas de servicio al cliente.

Permite familiarizar a los socios con el uso de la tecnología, al interactuar informáticamente con la institución.

¿Qué desventaja brinda el Kiosco Informático de la Institución?

La aglomeración que se produce al momento de la consulta en horas pico, ya que se comparte en el hall de servicio y atención al cliente.

El cliente está limitado al horario de consulta ya que el kiosco informático está disponible solo en días y en horarios de atención.

El entorno en el cual está el kiosco informático no garantiza la privacidad y seguridad de ingreso y verificación de datos al momento de la consulta.

¿Se ha recibido quejas acerca del servicio del Kiosco informático de la Institución?

Si, sobre todo porque hay socios que se olvidan las contraseñas, o hacen mal uso de ellas, y se desactivan los usuarios o cosas por el estilo, y hay cierta confusión en el envió y recibo de datos personales bancarios.

¿El plan estratégico de innovación de mercado Institucional ha pensado en algún momento en implementar una tecnología de consulta Móvil?

Si, de hecho está establecido en el plan de innovación de mercado de este año 2012, entrar con el tema de interacción de comunicación de los usuarios por vía celular.
Esto incluiría consultas y transacciones con todos los clientes de la Institución.

¿Actualmente la Institución Financiera San Francisco cuenta con alguna aplicación de Banca Móvil?

No, actualmente no contamos con ninguna aplicación de banca móvil pero se ha decidido implantar en este año, innovaciones de este sentido.

¿Cuál es la misión institucional en cuanto a satisfacer en atención al Cliente?

La misión fundamental es ayudar al socio en establecer los canales de comunicación necesarios para satisfacer sus necesidades de servicios financieros, información relevante y mecanismos para ser escuchados.

¿Le agradaría implementar una aplicación de banca móvil de consulta en la institución para mejorar la atención al cliente y comodidad de los usuarios?

Claro, en vista de que las tendencias de la banca moderna están orientadas a los canales celulares y de internet, es una alternativa válida y necesaria el proporcionar esos canales modernos de interacción a nuestros socios.

¿La infraestructura tecnológica de la Institución es adecuada para trabajar de una manera favorable con una aplicación de Banca Móvil?

Si es favorable en cuanto contamos con arquitecturas que favorecen la interacción con diversas interfaces, ya sea a través de web services, monitores transaccionales o cualquier otro mecanismo, la plataforma se halla probada y funcional interacciones tecnológicas como por ejemplo cajeros automáticos, consultas web pagos de servicios, etc.

¿Cuál cree UD. que serían las nuevas ventajas de trabajar en una empresa con tecnología móvil?

La principal ventaja es que la Cooperativa lograría tener una mejor cercanía con los socios, al estar presente en las tecnologías y dispositivos que utiliza a diario, logrando convertirse así en el mejor aliado de los socios.

3.2.3.1 Análisis de la Entrevista.

Al haber intercambiado ideas con mencionado jefe de sistemas y haber tenido claro las respuestas muy puntuales citadas en cada pregunta, podemos analizar en primer lugar que el Kiosco informático brinda un buen servicio a los socios pero de manera muy limitada, ya que atiende a un cliente a la vez y en horarios estrictos de oficina, lo que al contrario de una aplicación en un dispositivo móvil no restringiría un límite de horario ni un límite de atención unitaria por cliente evitando la incomodidad del socio en espera.

Este nuevo servicio es factible en esta Institución ya que cuenta con la adecuada infraestructura tecnológica para realizar las respectivas conexiones en una aplicación de Banca Móvil.

3.3 funcionales y no funcionales de la aplicación.

3.3.1 FUNCIONES.

SEGURIDAD.

F1 Validación de Usuarios

CONSULTAS.

F2 Consulta de Saldos

F3 Consulta de Préstamos

F4 Consulta de fecha pago de cuota y monto

F5 Consulta fecha de vencimiento de cuota y monto

F6 Consulta de últimos movimientos

TRANSACCIONES.

F7 Transferencia de Dinero entre cuentas pertenecientes a la Institución.

3.3.2 METAS.

Metas Institucionales.

- Innovar el negocio y la inexistencia de este tipo de software en la Institución Financiera San Francisco.
- La cooperativa quiere contar con nuevos canales de comunicación para los socios. Sistema de innovación claramente aprobado.
- Cumplir con las mismas obligaciones que la banca por que se ha convertido en una competencia directa.

Metas Tecnológicas.

- Nueva comodidad de servicio para los socios.
- Rapidez y facilidad de atención.
- Servicio de consulta bancaria las 24 horas del día.
- Seguridad y confianza en el acceso a la información.
- Confiabilidad en la aplicación.
- Aplicación clara y de fácil uso.
- Evitar varias filas de espera.
- Mejor atención al cliente.

Atributos del sistema.

- Seguridad en la consulta.
- Integridad de Información.

3.4 RESUMEN GENERAL.

Una aplicación de banca móvil es necesaria para todo socio por su versatilidad de consulta bancaria, y además por la atención e interés presentado por la gran parte de los encuestados que comprueban la factibilidad de realizar una aplicación como SFPUCE móvil para comodidad en el servicio al cliente.

Se necesita automatizar consultas confidenciales bancarias de una cuenta perteneciente a un socio de la Cooperativa de ahorro y crédito San Francisco Ltda., a través de una aplicación de banca móvil denominada SFPUCE móvil en un dispositivo de comunicaciones IPhone.

Mencionada aplicación ofrecerá al usurio opciones como, consultas de saldo en cuenta de ahorros, consultas de préstamos vigentes, consulta de fecha referente a última cuota y monto de la misma, consulta de vencimiento de cuota (próxima fecha) y el monto de la misma, consultar sus 5 últimos movimientos en cuenta de ahorros, e incluso la opción de poder administrar la transacción de dinero entre cuentas de la misma Institución, desde cualquier parte en la que el socio se encuentre.

DIAGRAMAS UML.

3.5 CASOS DE USO.

3.5.1 Nombre: Autentificación de Usuario.

- **Actores:** Usuario.
- **Propósito:** Ingreso del usuario al sistema.
- **Resumen:** El usuario ingresa su código de usuario y contraseña para ingresar al sistema.
- **Referencias:** F1.
- **Curso de los eventos:**

ACTORES	SISTEMA
1. El usuario inicia la aplicación.	
	2. Mostrar formulario de validación de Datos.
3. Ingresa código de usuario y contraseña.	
	4. Valida y otorga la autorización con un mensaje de bienvenida con el nombre del usuario, llevándolo automáticamente al menú de consultas.
5. El usuario ingresa al sistema.	

- **Cursos alternos:**

a) Línea 4.

Cuenta de ahorros de un cliente inactiva.

La entrada al sistema es denegada al usuario, por una de las siguientes razones:

*Caja de texto usuario y/o caja de texto contraseña vacío.

*Datos incorrectos, el cliente debe repetir el código de usuario y contraseña.

*Código de usuario inactivo.

*Inexistencia de internet en el teléfono móvil IPhone.

*Problemas en el servidor de base de datos o el servidor que aloja el web service.

El usuario debe pedir ayuda al administrador del sistema por lo que el sistema emitirá un mensaje de error de ingreso.

Autentificación de Usuario.

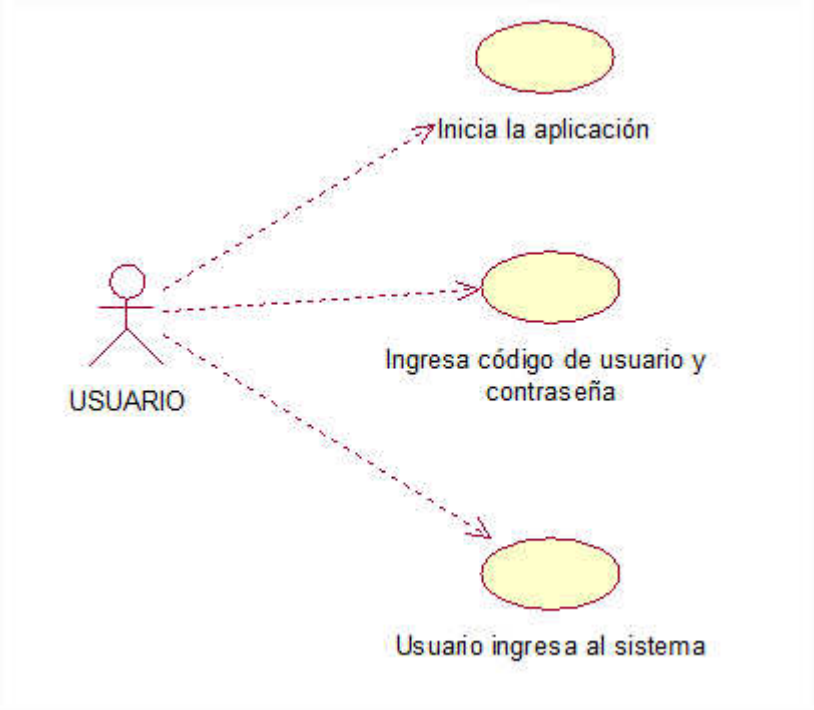


Figura #3.6 Diagrama de caso de uso

Autentificación de usuario.

3.5.2 Nombre: Consulta de Saldos.

- **Actores:** Usuario.
- **Propósito:** Consultar el monto existente en la cuenta de ahorros del usuario.
- **Resumen:** El usuario verifica su monto disponible en su cuenta de ahorros perteneciente a la cooperativa San Francisco.
- **Referencias:** F1, F2.
- **Curso de los eventos:**

ACTORES	SISTEMA
1. El usuario ingresa al sistema.	
	2. Muestra un menú con todas las opciones disponibles de consultas.
3. El usuario selecciona en el menú de pantalla la opción "SALDO".	
	4. Muestra el saldo disponible en la cuenta ahorro del usuario.
3. El usuario obtiene su información confidencial bancaria acerca de saldos de cuenta en ahorros y regresa al menú principal.	

- **Cursos alternos:**

a) Línea 4.

Inexistencia de internet en el teléfono móvil IPhone.

Problemas en el servidor de base de datos o el servidor que aloja el web service.

Consulta de Saldos.

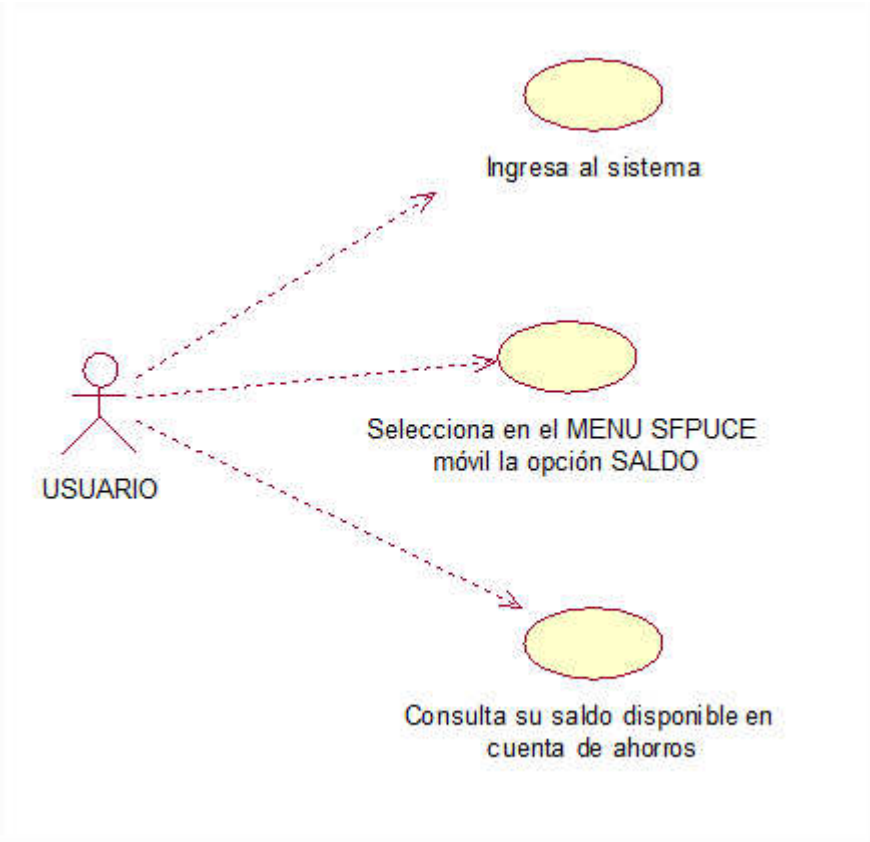


Figura # 3.7 Diagrama de caso de uso

Consulta de saldos en ahorros.

3.5.3 Nombre: Consulta de préstamos.

- **Actores:** Usuario.
- **Propósito:** Consultar si existen préstamos vigentes asociados al número de cuenta de un cliente.
- **Resumen:** El usuario verifica si existen préstamos vigentes aliados a su número de cuenta y el saldo de los mismos.
- **Referencias:** F1, F3.
- **Curso de los eventos:**

ACTORES	SISTEMA
1. El usuario ingresa al sistema.	
	2. Muestra un menú con todas las opciones disponibles de consulta.
3. El usuario selecciona en el menú de pantalla la opción "PRESTAMOS".	
	4. Muestra todos los préstamos activos aliados al código de usuario del cliente.
5. El usuario selecciona el préstamo vigente que desea consultar.	
	6. Muestra el saldo actual del préstamo seleccionado.
7. El usuario obtiene su información confidencial bancaria acerca de préstamos	

activos y saldo de los mismos y regresa al menú principal.	

- **Cursos alternos:**

a) Línea 4.

Inexistencia de préstamos activos relacionados al cliente.

Inexistencia de internet en el teléfono móvil IPhone.

Problemas en el servidor de base de datos o el servidor que aloja el web service.

Consulta de Préstamos.

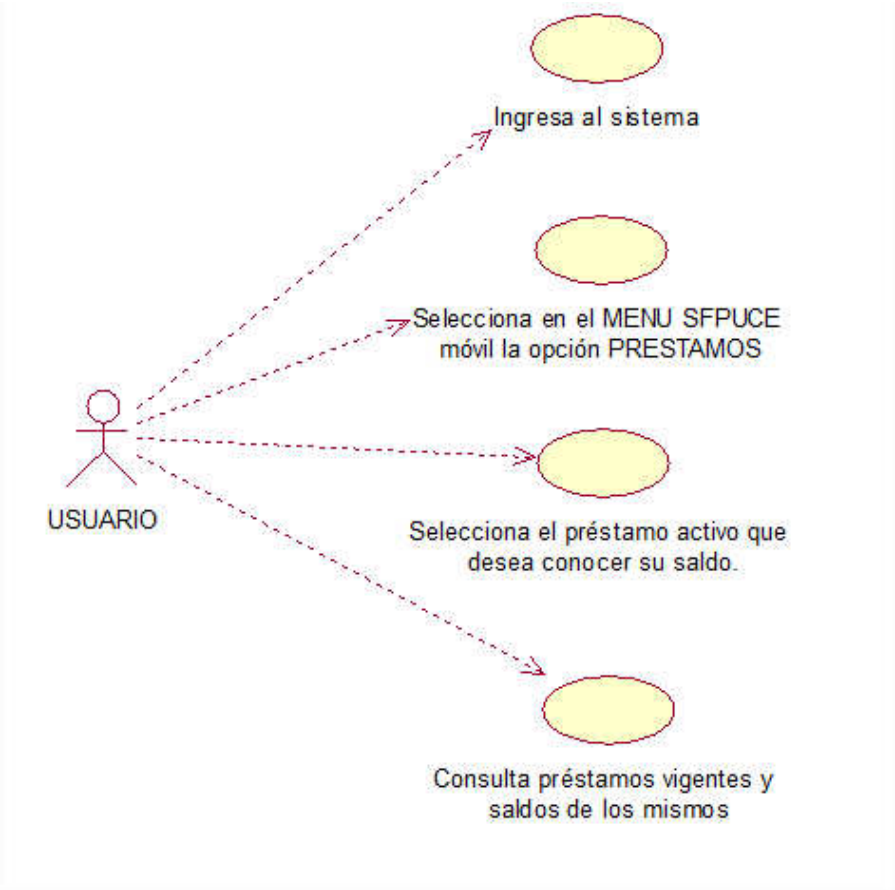


Figura # 3.8 Diagrama de caso de uso

Consulta de préstamos.

3.5.4 Nombre: Consulta fecha de pago de cuota.

- **Actores:** Usuario.
- **Propósito:** Consultar la fecha del último pago de cuota y el monto del mismo.
- **Resumen:** Si el usuario tiene créditos activos, seleccionará cualquiera de ellos para consultar la fecha del último pago de cuota realizado y cuanto tuvo que cancelar del mismo.
- **Referencias:** F1, F4.
- **Curso de los eventos:**

ACTORES	SISTEMA
1. El usuario ingresa al sistema.	
	2. Muestra un menú con todas las opciones disponibles de consulta.
3. El usuario selecciona en el menú de pantalla la opción "FECHA PAGO CUOTA PRESTAMO".	
	4. Muestra todos los préstamos activos aliados al código de usuario del cliente.
5. El usuario selecciona el préstamo en pantalla que requiere información.	
	6. Muestra un informe con la fecha de la última cuota pagada y el valor de la misma.
7. El usuario obtiene su información	

confidencial bancaria acerca de un préstamo vigente con la fecha de la última cuota que ha cancelado y el valor de la misma y regresa al menú principal.	

- **Cursos alternos:**

a) Línea 4.

Inexistencia de préstamos activos relacionados al cliente.

Inexistencia de internet en el teléfono móvil IPhone.

Problemas en el servidor de base de datos o el servidor que aloja el web service.

b) Línea 6.

Préstamo otorgado en el presente mes, el usuario todavía no ha cancelado su primera cuota.

Inexistencia de internet en el teléfono móvil IPhone.

Problemas en el servidor de base de datos o el servidor que aloja el web service.

Consulta fecha de pago de cuota.

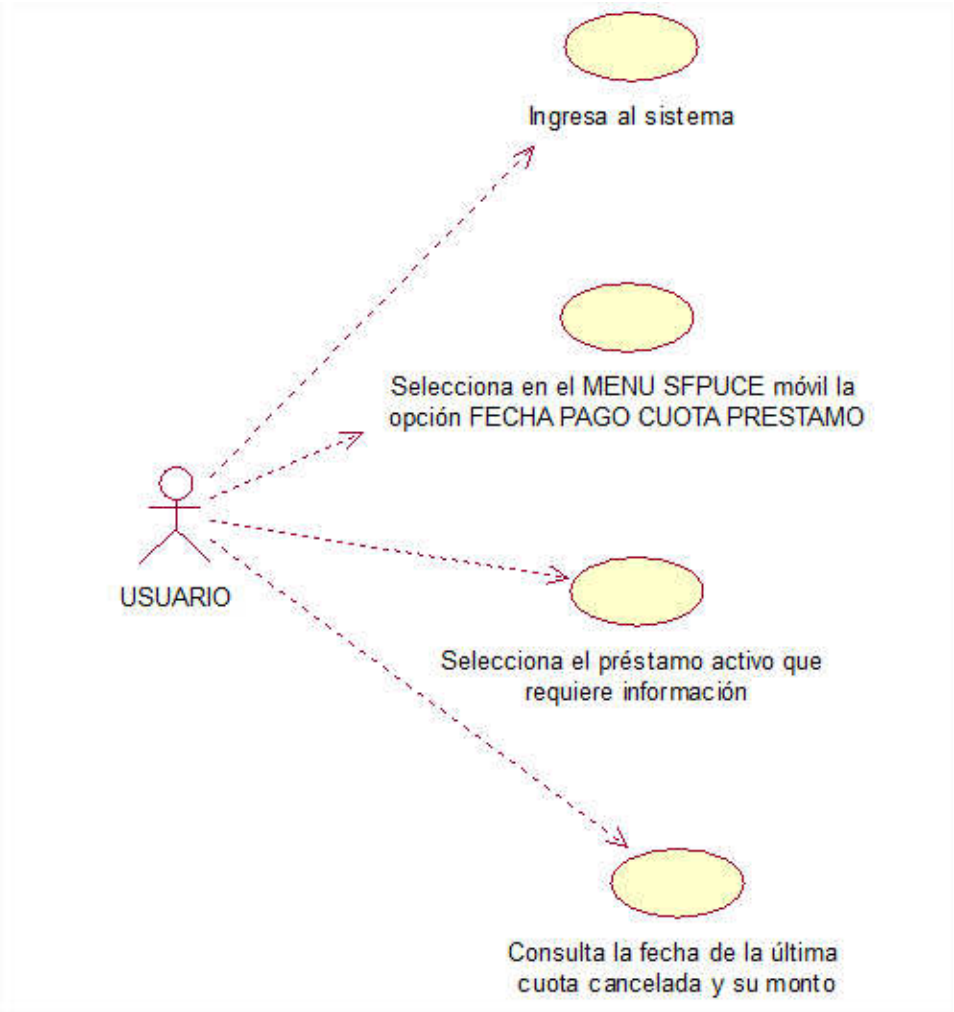


Figura # 3.9 Diagrama de caso de uso

Consulta fecha de pago de cuota.

3.5.5 Nombre: Consulta fecha de vencimiento de cuota.

- **Actores:** Usuario.
- **Propósito:** Consultar la fecha de vencimiento de cuota.
- **Resumen:** Si el usuario tiene préstamos activos aliados a su código de usuario, selecciona el crédito que necesita la información y consulta la fecha del próximo pago de cuota y el valor que tiene que cancelar de la misma.
- **Referencias:** F1, F5.
- **Curso de los eventos:**

ACTORES	SISTEMA
1. El usuario ingresa al sistema.	
	2. Muestra un menú principal con todas las opciones disponibles acerca de consultas.
3. El usuario selecciona en el menú principal de pantalla la opción ¨FECHA VENCIMIENTOS PRESTAMOS¨	
	4. Muestra todos los préstamos activos aliados al código de usuario del cliente.
5. El usuario selecciona un préstamo en pantalla que requiere de información.	
	6. Muestra un informe con la fecha de la próxima cuota y el valor de la misma.

7. El usuario obtiene su información confidencial bancaria acerca de un préstamo vigente con la fecha de la próxima cuota y el valor de cuanto le corresponde cancelar y regresa al menú principal.	

- **Cursos alternos:**

a) Línea 4.

- Inexistencia de préstamos activos relacionados al cliente.
- Inexistencia de internet en el teléfono móvil IPhone.
- Problemas en el servidor de base de datos o el servidor que aloja el web service.

b) Línea 6.

- Inexistencia de internet en el teléfono móvil IPhone.
- Problemas en el servidor de base de datos o el servidor que aloja el web service.

Consulta fecha de vencimientos de cuota.

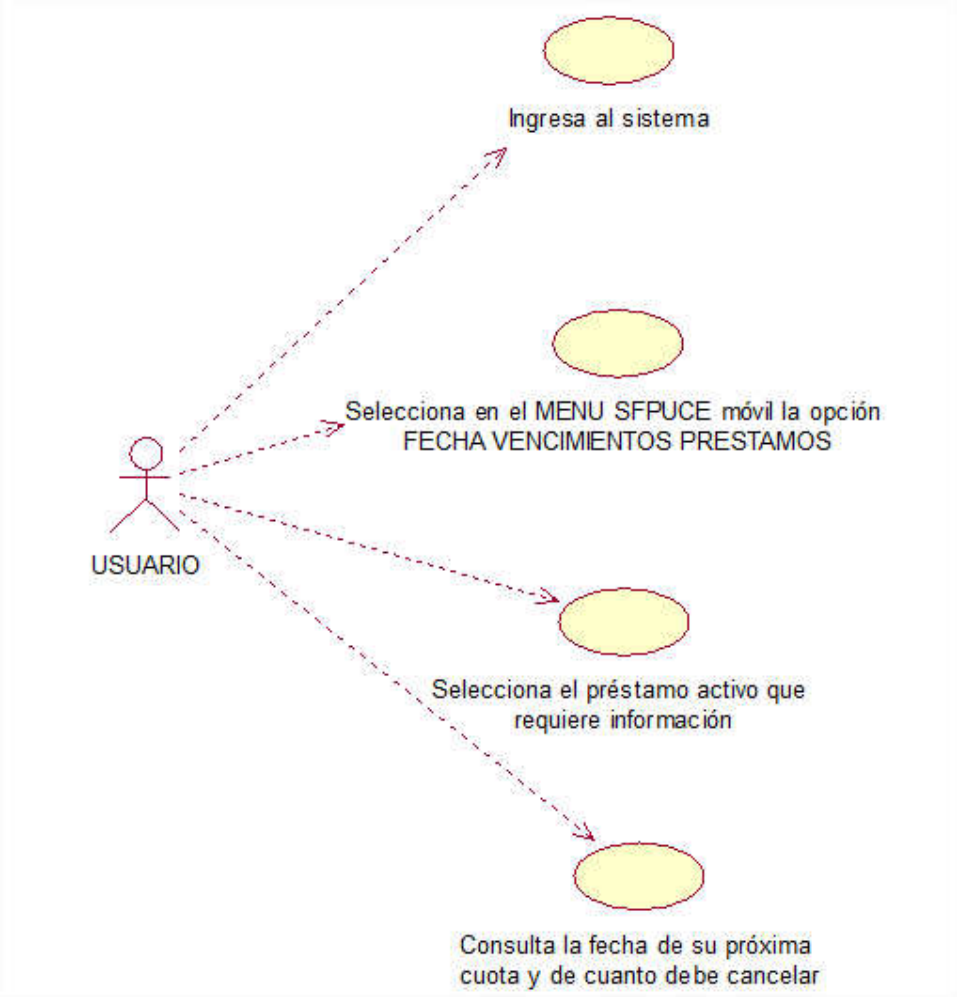


Figura # 3.10 Diagrama de caso de uso
Consulta fecha de vencimientos de cuota.

3.5.6 Nombre: Consulta de últimos movimientos.

- **Actores:** Usuario.
- **Propósito:** Consultar los 5 últimos movimientos realizados en la cuenta de ahorros del usuario.
- **Resumen:** El usuario consulta los últimos 5 movimientos efectuados en su cuenta de ahorros perteneciente a la institución San Francisco.
- **Referencias:** F1, F6.
- **Curso de los eventos:**

ACTORES	SISTEMA
1. El usuario ingresa al sistema.	
	2. Muestra un menú con todas las opciones disponibles de consultas.
3. El usuario selecciona en el menú principal de pantalla la opción "MOVIMIENTOS CUENTA".	
	4. Muestra los 5 últimos movimientos efectuados en la cuenta del usuario.
5. El usuario obtiene su información confidencial bancaria acerca de los últimos 5 movimientos realizados en la cuenta de ahorros y regresa al menú principal.	

- **Cursos alternos:**

a) Línea 4.

Apertura de cuenta sin movimientos (error de operativo de servicio y atención al cliente).

Cuenta de ahorros con menos de 5 movimientos.

Inexistencia de internet en el teléfono móvil IPhone.

Problemas en el servidor de base de datos o el servidor que aloja el web service.

Consulta de últimos movimientos.

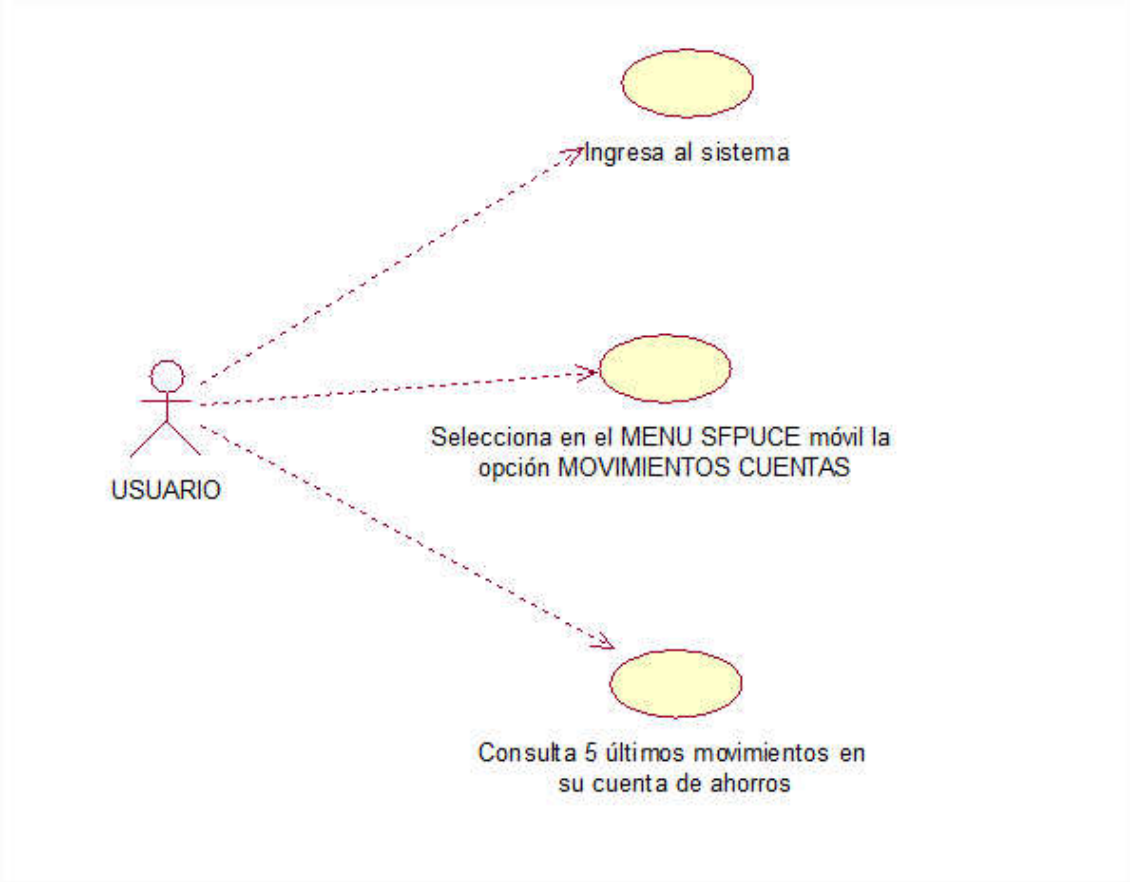


Figura # 3.11 Diagrama de caso de uso
Consulta de últimos movimientos.

3.5.7 Nombre: Transferencia de dinero entre cuentas.

- **Actores:** Usuario.
- **Propósito:** Realizar una transferencia de dinero entre cuentas de ahorro.
- **Resumen:** El usuario realiza una transferencia de dinero hacia otra cuenta perteneciente a la Institución San Francisco.
- **Referencias:** F1, F7.
- **Curso de los eventos:**

ACTORES	SISTEMA
1. El usuario ingresa al sistema.	
	2. Muestra un menú principal con todas las opciones disponibles de consulta.
3. El usuario selecciona en el menú principal la opción "Transferencia de DINERO".	
	4. Muestra un formulario para ingresar la cuenta destino y el valor a transferir.
5. El usuario ingresa el número de cuenta destino y el monto, y presiona el botón de transferir.	
	6. Verifica que este usuario no tenga más de 3 transferencias al día y muestra una ventana emergente de ATENCION, en la

	cual advierte al usuario que el monto ingresado será descontado de su saldo y solicita una confirmación para realizar la transferencia al número de cuenta ingresado.
7. El usuario confirma la transferencia.	**8.** Recibe parámetros de cuenta destino y monto de dinero, y realiza la transacción entre cuentas; verifica que la transacción se haya completado correctamente y envia un correo electrónico al email aliado del usuario indicando el proceso realizado.
10. El usuario realiza una transferencia de dinero de una cuenta a otra perteneciente a la cooperativa de ahorro y crédito San Francisco y regresa al menú principal.	
11. El usuario comprueba la transferencia revisando su correo electrónico personal.	

- **Cursos alternos:**

a) Línea 8.

Usuario ingresa un valor que sobrepasa los 100 dolares.

Usuario sobrepasa límite de 3 transferencias al día.

Inexistencia de saldo en cuenta emisora para efectuar transacción.

Cuenta de destino inactiva.

Cuenta de destino inexistente.

Inexistencia de internet en el teléfono móvil IPhone.

Problemas en el servidor de base de datos o en el servidor que aloja el web service.

b) Línea 11.

Cliente dictó erróneamente su dirección de correo electrónico al operativo de servicio y atención al cliente al momento de elaborar la solicitud de creación de usuario.

Mal digitación de correo electrónico de cliente por parte de operativo y servicio y atención al cliente al momento de elaborar la solicitud de creación de usuario.

Transferencia de dinero entre cuentas.

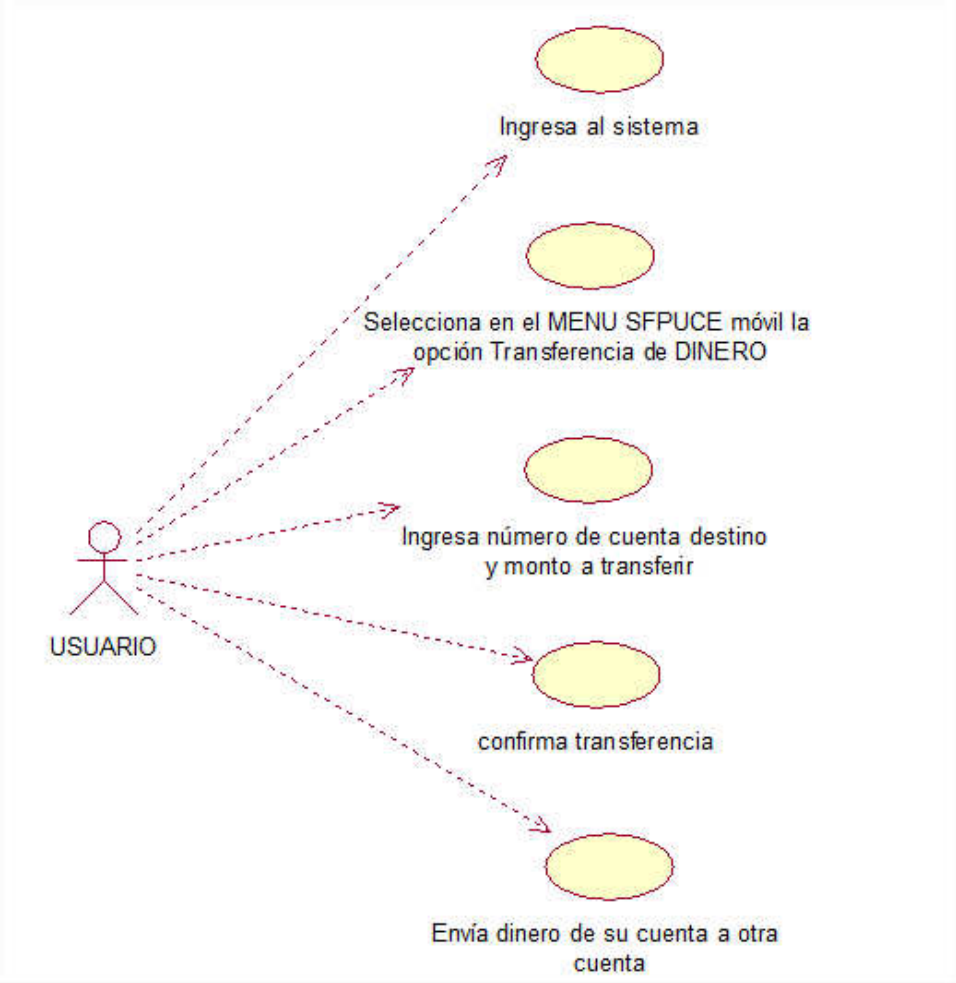


Figura # 3.12 Diagrama de caso de uso
Transferencia de dinero entre cuentas.

3.6 MODELO CONCEPTUAL.

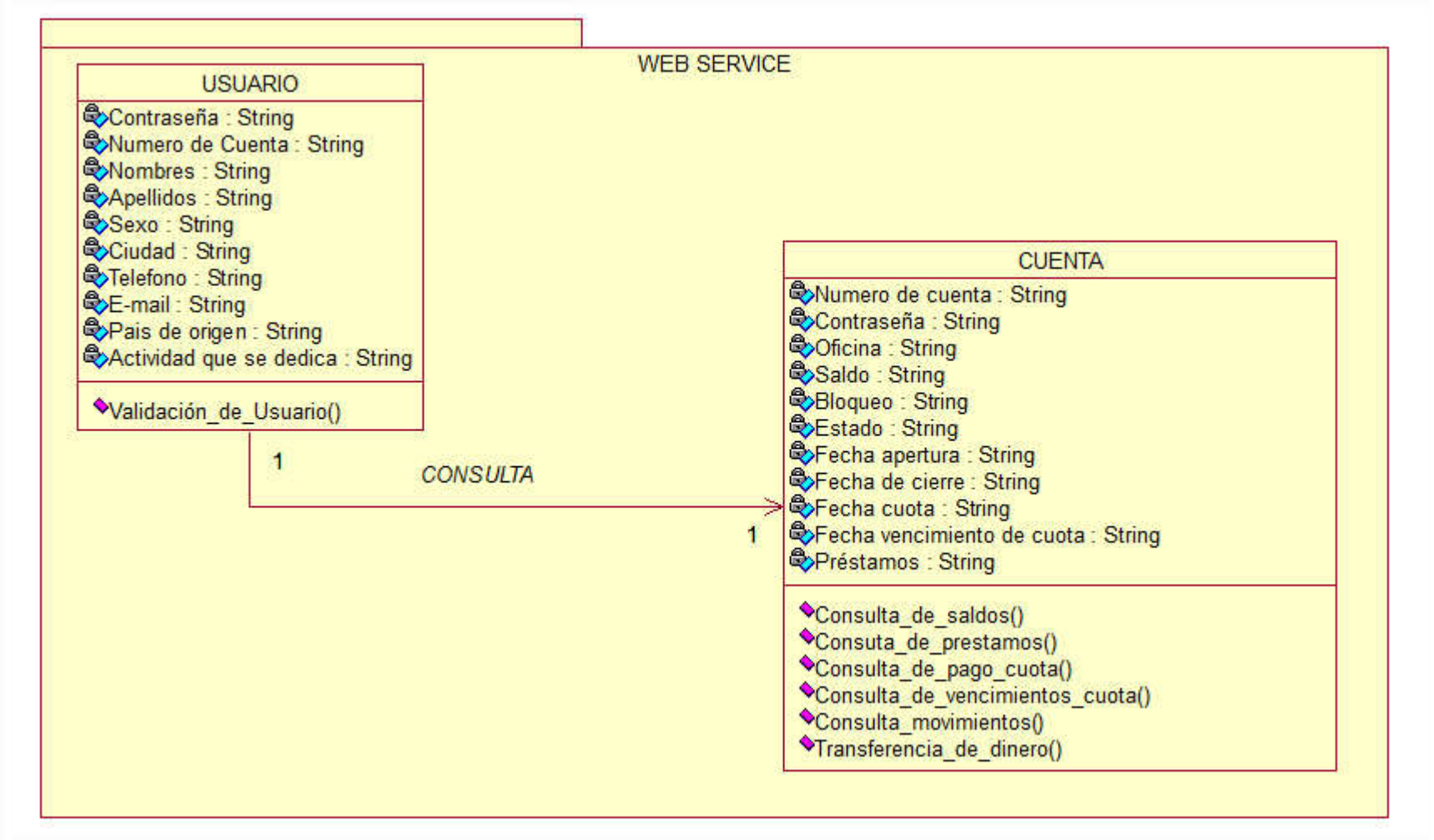


Figura # 3.13 Modelo conceptual.

3.7 DIAGRAMAS DE SECUENCIA.

3.7.1 AUTENTIFICACIÓN DE USUARIO.

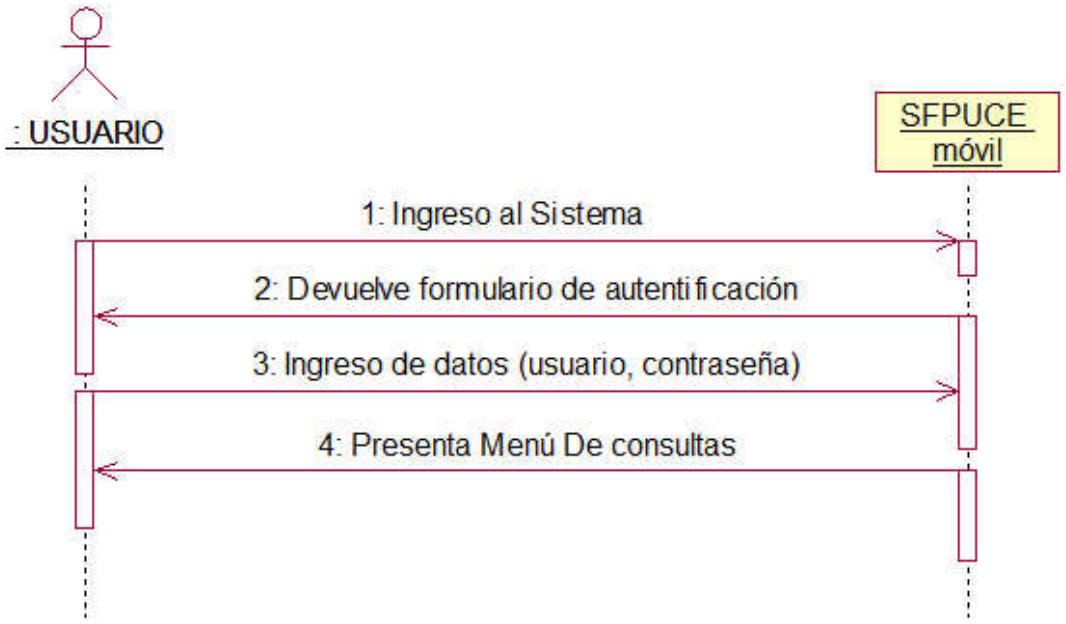


Figura # 3.14 Diagrama de secuencia
Autentificación de usuario.

3.7.2 CONSULTA DE SALDOS.

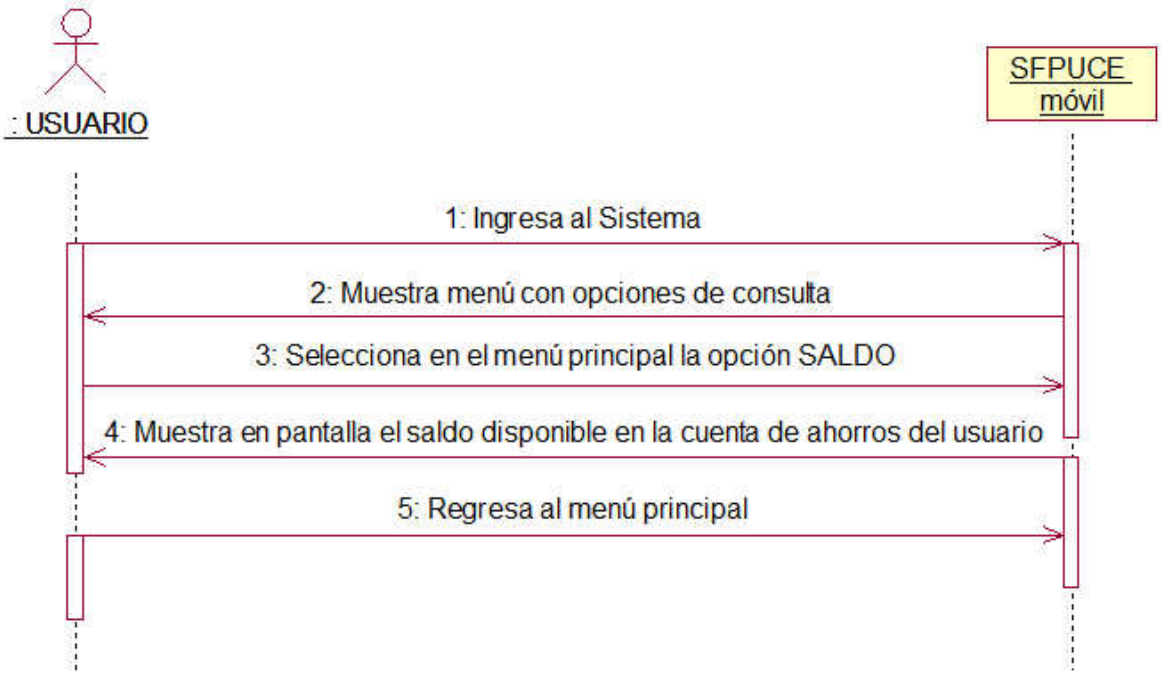


Figura # 3.15 Diagrama de secuencia

Consulta de saldos.

3.7.3 CONSULTA DE PRÉSTAMOS.

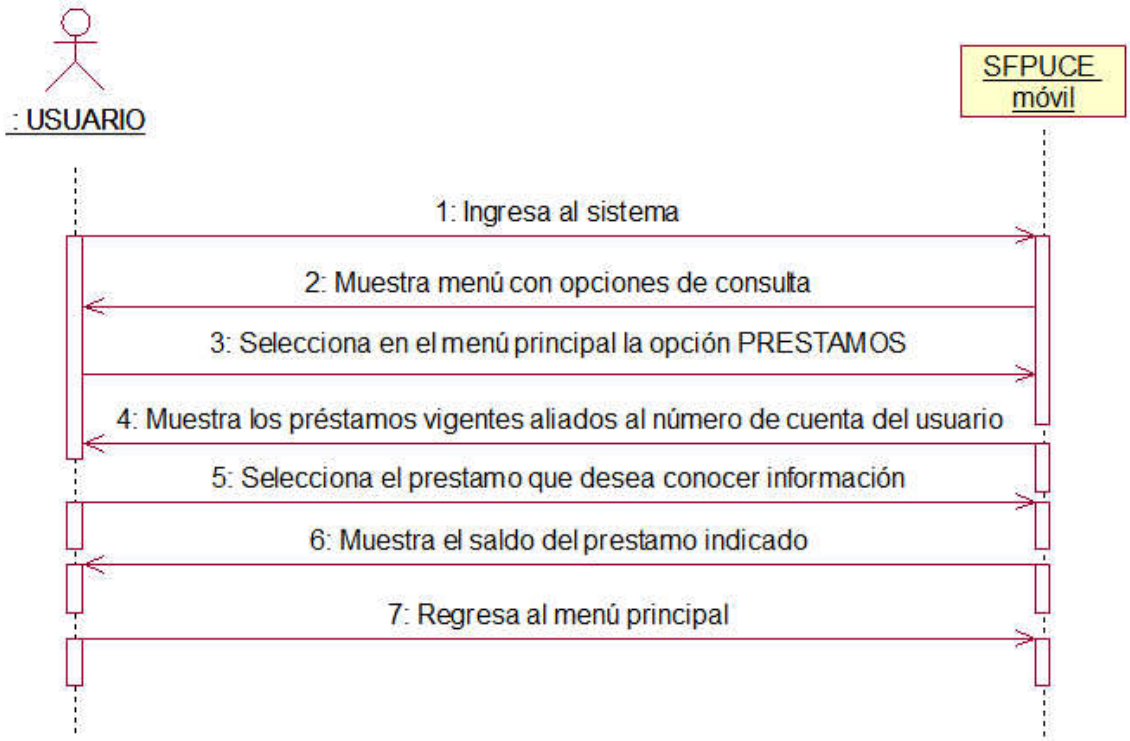


Figura # 3.16 Diagrama de secuencia

Consulta de préstamos

3.7.4 CONSULTA DE FECHA PAGO DE CUOTA.

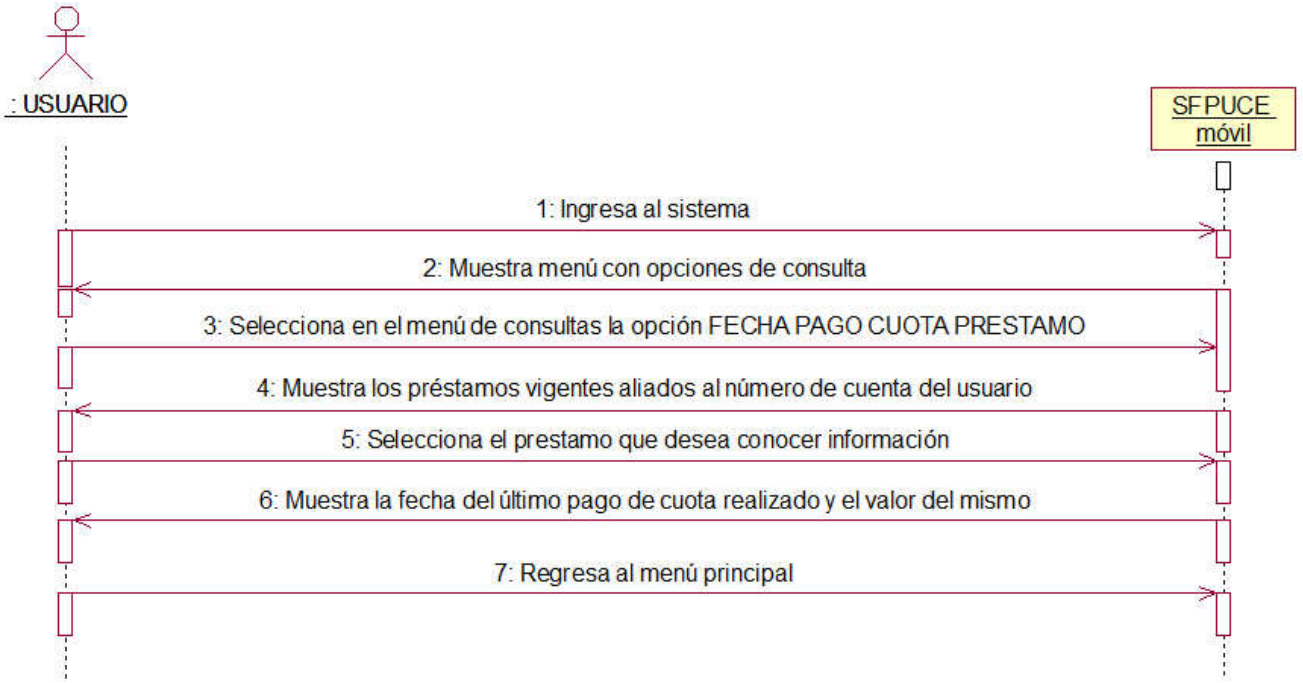


Figura # 3.17 Diagrama de secuencia
Consulta de fecha de pago de cuota.

3.7.5 CONSULTA FECHA DE VENCIMIENTO DE CUOTA.

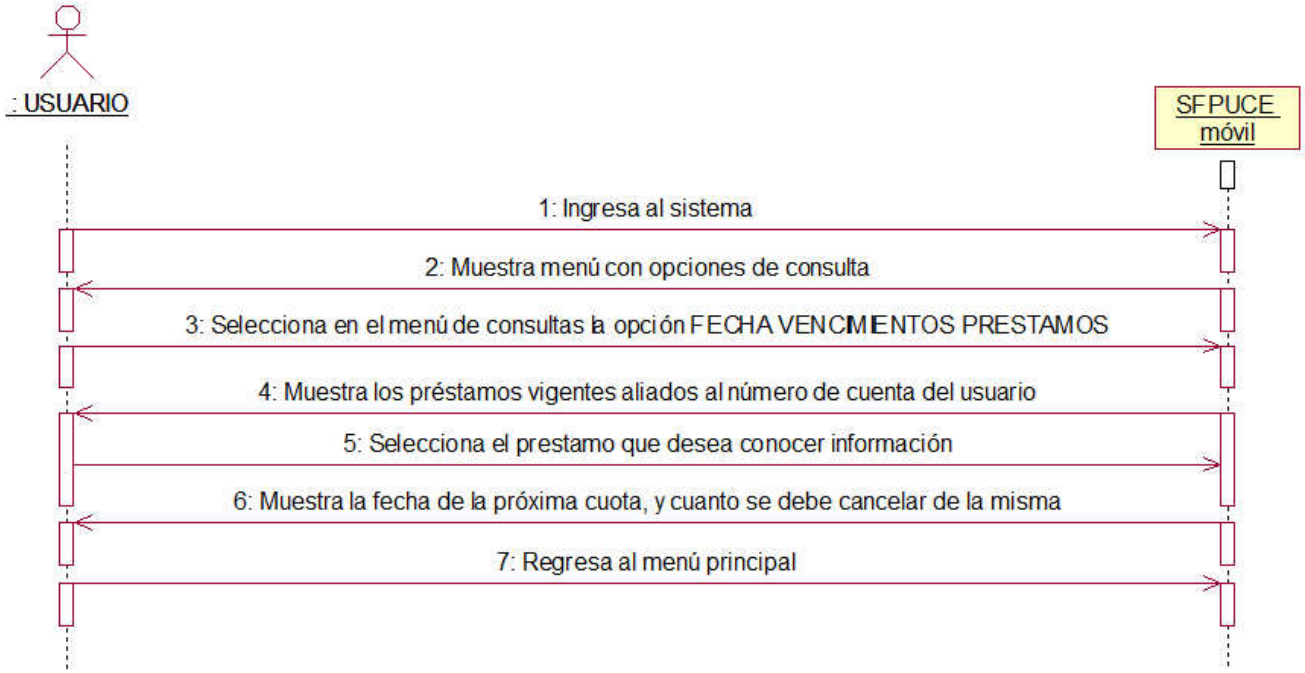


Figura # 3.18 Diagrama de secuencia

Consulta de fecha de vencimientos de cuota.

3.7.6 CONSULTA DE ÚLTIMOS MOVIMIENTOS.

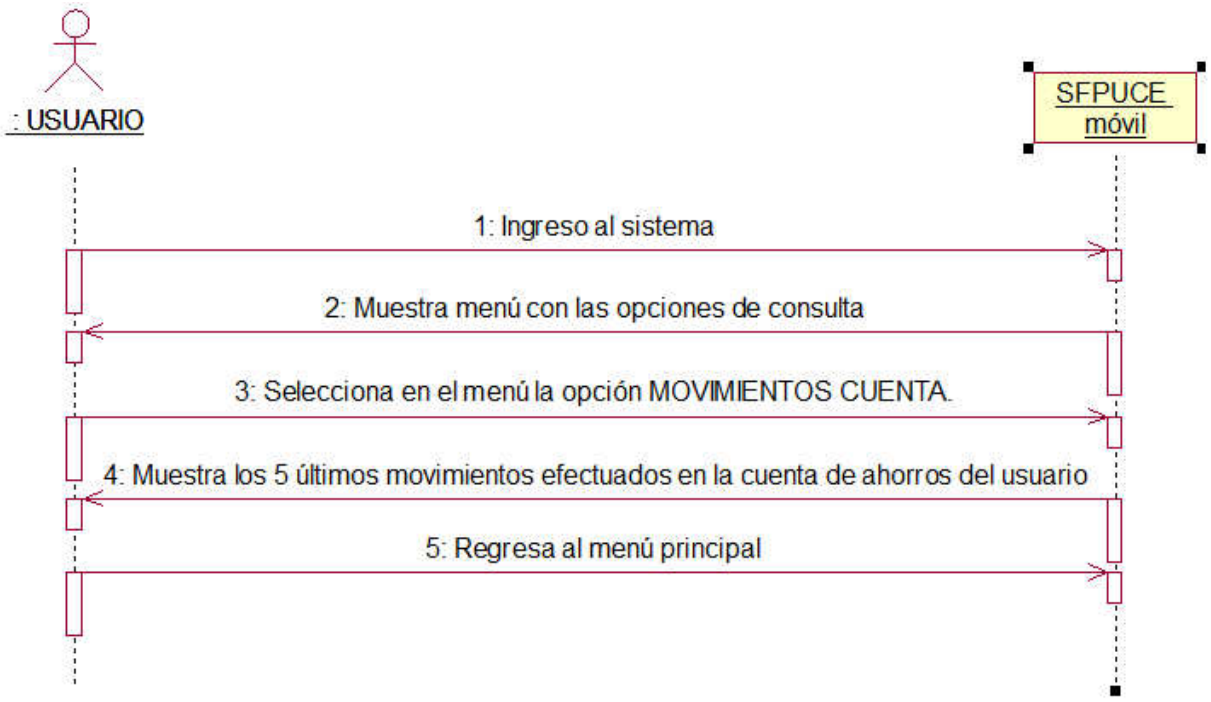


Figura # 3.19 Diagrama de secuencia
Consulta de últimos movimientos.

3.7.7 TRANSFERENCUA DE DINERO.

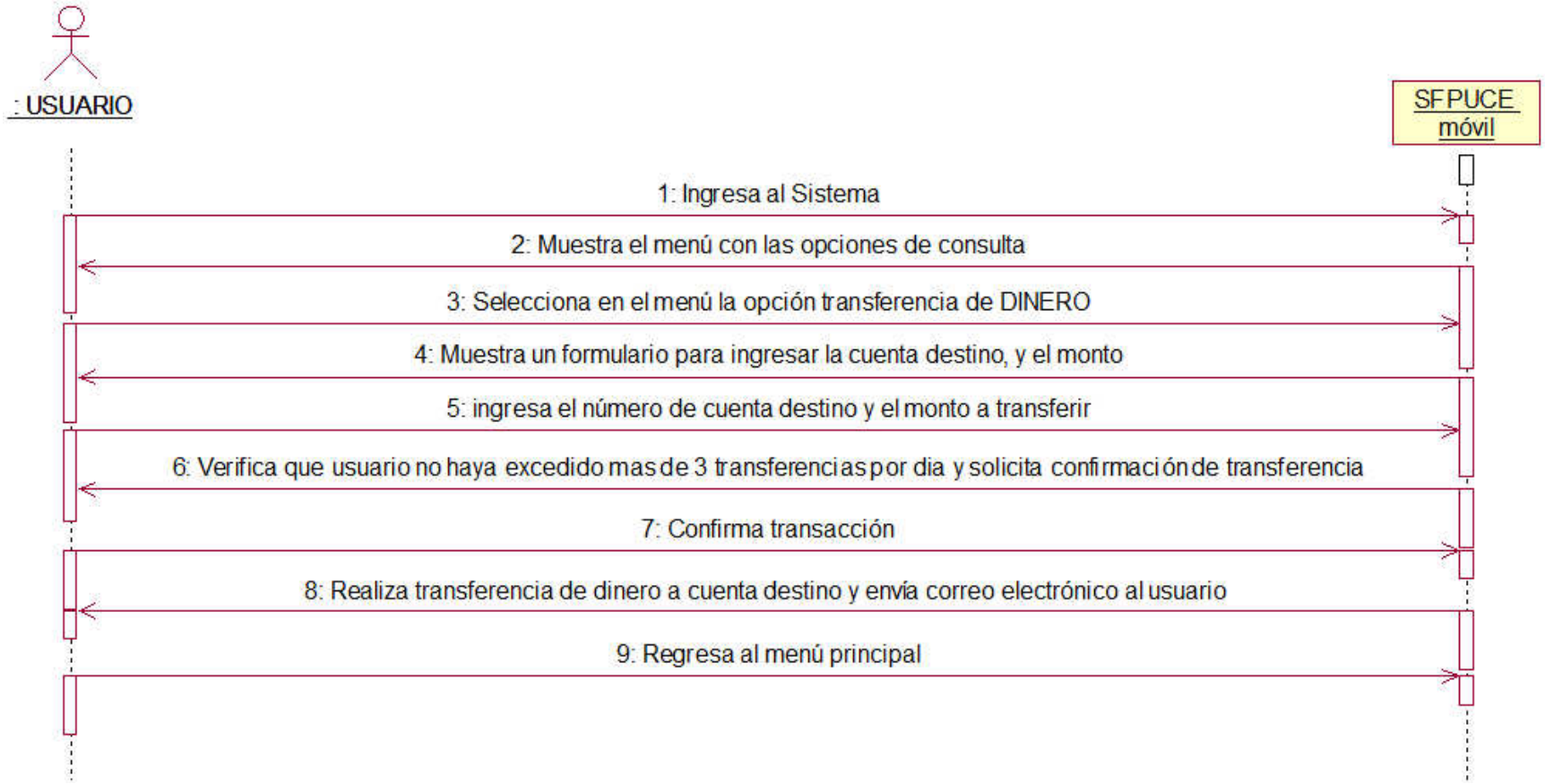


Figura # 3.20 Diagrama de secuencia

Transferencia de dinero entre cuentas.

3.8 DIAGRAMA DE ACTIVIDADES.

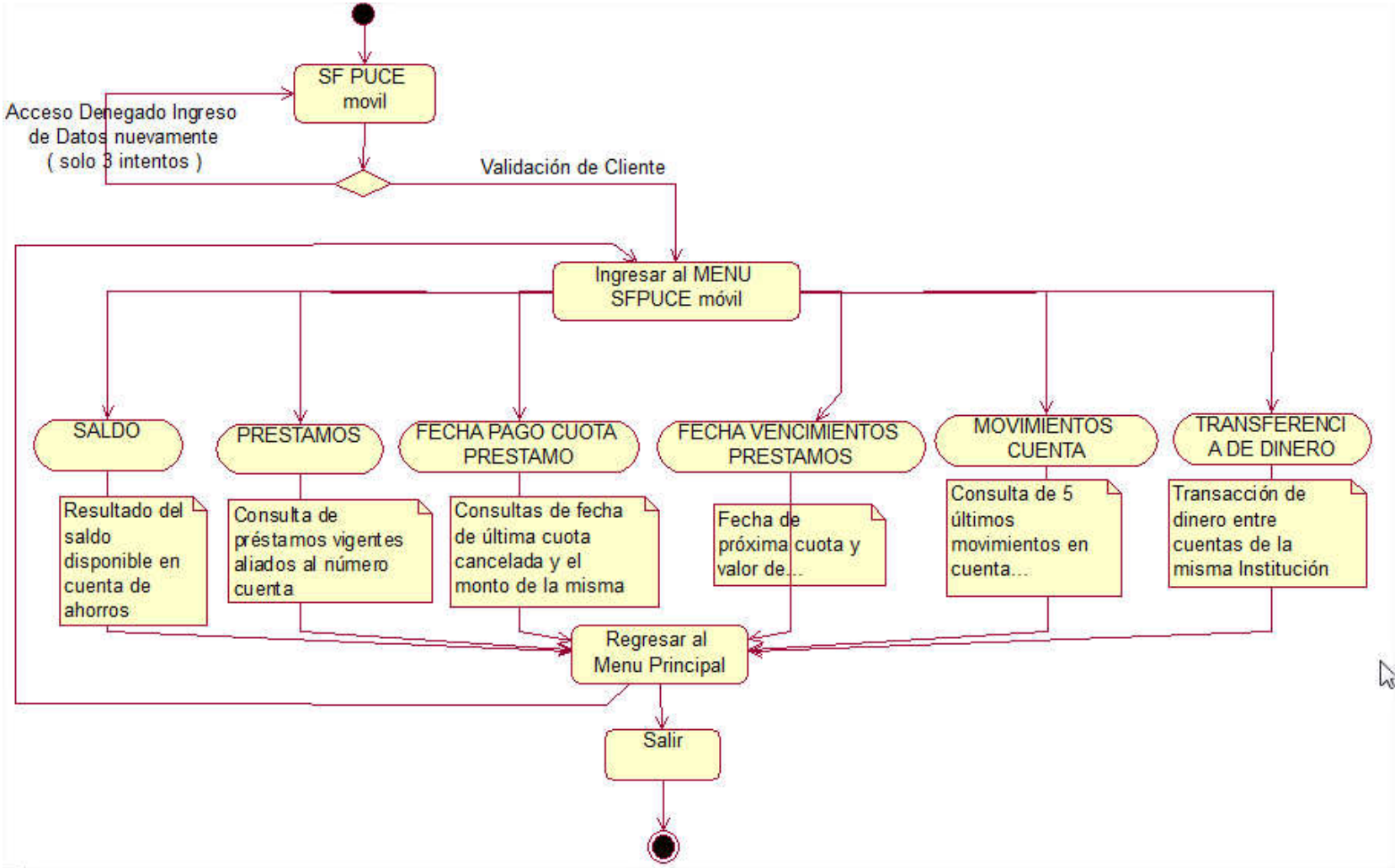


Figura # 3.21 Diagrama de actividades estados de la aplicación.

3.9 DISEÑO.

3.9.1 DISEÑO DE BASE DE DATOS INSTITUCIONAL.

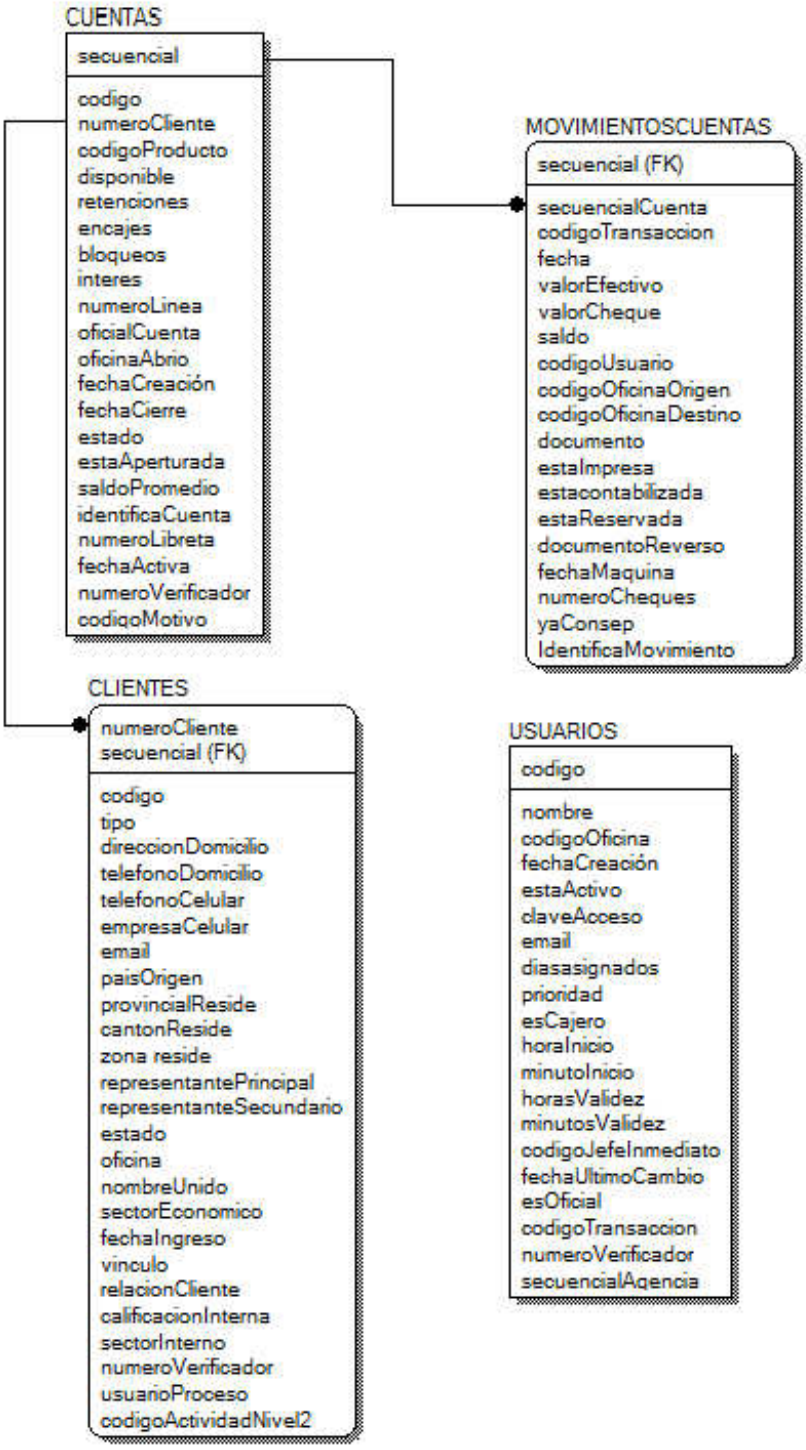


Figura # 3.22 Esquema de base de datos institucional.

3.9.2 DICCIONARIO DE DATOS.

3.9.2.1 Tabla Cuentas.

CUENTAS	
Nombre de la Columna	**TIPO**
Código	Char
Secuencial	Int
NumeroCliente	Int
CodigoProducto	Char
Disponible	Decimal
Retenciones	Decimal
Encajes	Decimal
Bloqueos	Decimal
Interés	Decimal
numeroLinea	Tintyint
oficialcuenta	Char
oficinaAbrio	Tintyint
fechacreacion	Datetime
Fechacierre	Datetime
Estado	Char
estaaperturada	Tintyint
saldopromedio	Decimal
identificacuenta	Varchar
numerolibreta	Tintyint
fechaactiva	Datetime
numeroverificador	Int
codigomotivo	char

Tabla #3.6 Diseño de tablas

Tabla CUENTAS.

3.9.2.2 Tabla MOVIMIENTOS CUENTAS.

MOVIMIENTOS CUENTAS	
Secuencial	**Int**
Secuencialcuenta	Int
Codigotransaccion	Char
Fecha	Datetime
Valorefectivo	Decimal
Valorcheque	Decimal
Saldo	Decimal
Codigousuario	Char
Codigooficinaorigen	Tintyint
Codigooficinadestino	Tintyint
Documento	Varchar
Estaimpresa	Tintyint
Estacontabilizada	Tintyint
Estareversada	Tintyint
Documentoreverso	Varchar
Fechamaquina	datetime

Tabla # 3.7 Diseño de tablas

Tabla MOVIMIENTOS CUENTAS.

3.9.2.3 Tabla USUARIOS.

USUARIOS	
Código	**Char**
Nombre	Varchar
Codigooficina	Tintyint
Fechacreacion	Datetime
Estaactivo	Tintyint
Claveacceso	Datetime
Secuencial	Tintyint
Email	Char
Diasasignados	Int
Prioridad	Varchar
Escajero	Varchar
Horainicio	Tintyint
Minutoinicio	Tintyint
Horasvalidez	Tintyint
Minutosvalides	Tintyint
codigojefeinmediato	Char
fechaultimocambio	Datetime
esoficial	Tintyint
codigotransaccionextranjera	Varchar
numeroverificador	Int
secuenciaagencia	Int

Tabla # 3.8 Diseño de tablas

Tabla USUARIO.

3.9.2.4 Tabla CLIENTES.

CLIENTES	
Código	**Varchar**
Tipo	Char
Direcciondomicilio	Varchar
Telefonodomicilio	Varchar
Telefonocelular	Varchar
Empresacelular	Varchar
Email	Varchar
Paisorigen	Varchar
Provinciareside	Char
Cantonreside	Char
Zonareside	Char
Representanteprincipal	Varchar
Representantesecundario	Varchar
Estado	Char
Oficina	Tintyint
Nombreunido	Varchar
Numerocliente	Int
Sectoreconomico	Char
Fechaingreso	Datetime
Vinculo	Tintyint
relacioncliente	Varchar
Calificación interna	Char
sectorinterno	Varchar
numeroverificador	Int
usuarioproceso	Char
Codigoactividadnivel2	Char

Tabla # 3.9 Diseño de tablas

Tabla CLIENTES.

3.9.3 DISEÑO DE INTERFACES.

3.9.3.1 ICONO DE LA PLICACIÓN.



Figura # 3.23 Icono aplicación SFPUCE móvil.

3.9.3.2 FORMULARIO DE AUTENTIFICACIÓN DE USUARIO.

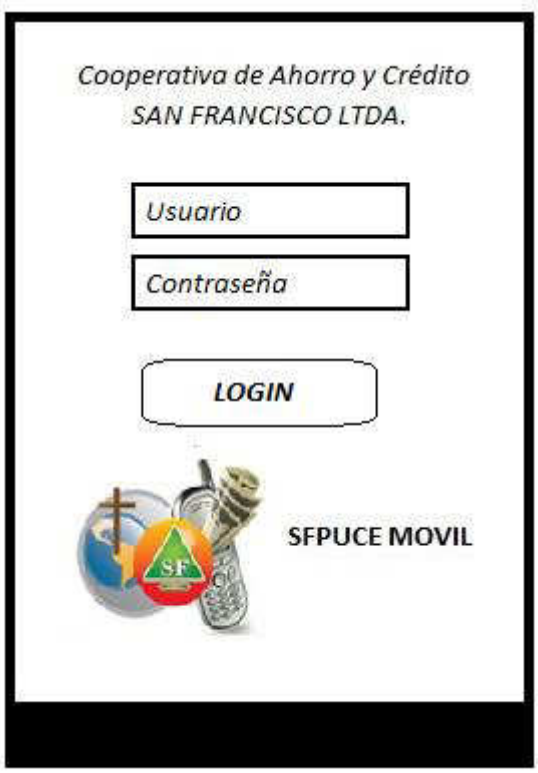


Figura # 3.24 Formulario Autentificación de usuario.

3.9.3.3 FORMULARIO MENU SFPUCE móvil

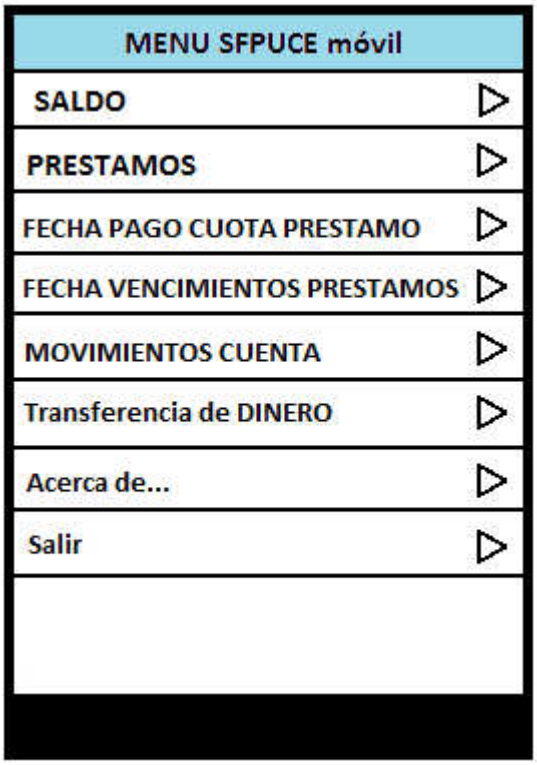


Figura # 3.25 Formulario MENU SFPUCE móvil

3.9.3.4 FORMULARIO CONSULTA DE SALDOS.

Para realizar la consulta de saldos, se trabajará en el siguiente diseño de interfaz:



Figura # 3.26 Formulario Consulta de saldos.

3.9.3.5 FORMULARIO CONSULTA DE PRÉSTAMOS

Para realizar la consulta de préstamos, se trabajará en el siguiente diseño de interfaz:

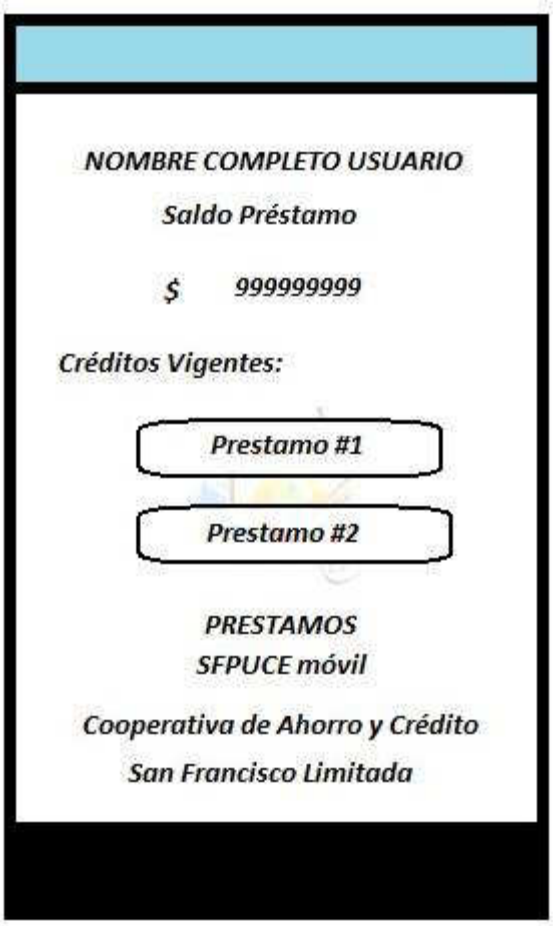


Figura # 3.27 Formulario Consulta de préstamos

Observación: Para la consulta de pago de cuota, consulta de fecha de vencimiento de cuota y consulta de movimientos, se mantendrá el mismo diseño de interfaz mencionado en la figura 3.23

3.9.3.6 FORMULARIO TRANSFERENCIA DE DINERO



Figura # 3.28 Formulario Transferencia de dinero.

3.9.3.7 FORMULARIO ACERCA DE.



Figura # 3.29 Formulario acerca de.

3.9.3.8 FORMULARIO SALIR.



Figura # 3.30 Formulario salir

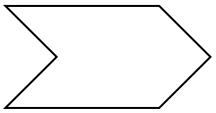
3.10 CONTRATOS.

3.10.1 AUTENTIFICACIÓN DE USUARIO.

1) Nombre: Ingreso de datos Usuario ([Usuario], [Contraseña]).

2) Responsabilidades: Ingresar información confidencial del usuario para acceso a la aplicación

3) Referencias: Caso de Uso Autentificación de Usuario

Diagrama secuencia 

4) Notas:

* La aplicación da la bienvenida del usuario mostrando en pantalla el nombre completo del cliente y llevándolo al menú principal.

5) Excepciones:

* Si la aplicación no da la bienvenida, el usuario digitó sus datos de ingreso incorrectos.

* Si la aplicación no da la bienvenida, el cliente no pertenece a la Institución.

*Si la aplicación no da la bienvenida, la cuenta de ahorros está inactiva.

*Si la aplicación no da la bienvenida, el usuario se encuentra inactivo.

*Si la aplicación no da la bienvenida, la caja de texto usario y/o contraseña se encuentran en blanco.

6) Precondiciones: La aplicación debe estar conectada al sistema por lo que la base de datos institucional tiene en línea los clientes activos de la cooperativa de ahorro y crédito San Francisco para de esta manera llevar a cabo la autentificación.

7) Postcondiciones:

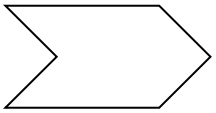
- Se creó una asociación entre usuario y web service institucional.

3.10.2 CONSULTA DE SALDOS.

1) Nombre: Consulta de Saldos [“SALDO”].

2) Responsabilidades: Consultar información confidencial bancaria de un usuario acerca del saldo disponible en su cuenta de ahorros.

3) Referencias: Caso de Uso SALDO

Diagrama secuencia 

4) Notas:

* En el menú principal elegir la opción correcta de SALDO.

5) Excepciones:

* Tener en cuenta si la cuenta de ahorros está activa.

6) Precondiciones:

*Ingreso de datos correctos.

*La aplicación debe tener los datos del usuario correctos.

*La aplicación debe enviar correctamente número de cuenta, usuario y contraseña automáticamente.

*El usuario debe tener permisos del administrador para manejar la aplicación.

7) Postcondiciones:

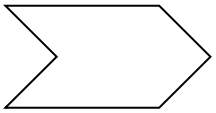
- Se creó una asociación entre usuario y cuenta

3.10.3 CONSULTA DE PRÉSTAMOS.

1) Nombre: Consulta de Préstamos [“PRESTAMOS”].

2) Responsabilidades: Consultar Información confidencial bancaria de un usuario acerca de los préstamos vigentes aliados a su número de cuenta de ahorros.

3) Referencias: Caso de Uso PRÉSTAMOS

Diagrama secuencia 

4) Notas:

* En el menú principal elegir la opción correcta de PRÉSTAMOS.

5) Excepciones:

* Tener en cuenta si la cuenta de ahorros está activa.

6) Precondiciones:

*Ingreso de datos correctos.

*La aplicación debe tener los datos del usuario correctos.

*La aplicación debe enviar correctamente número de cuenta, usuario y contraseña automáticamente.

*El usuario debe tener permisos del administrador para manejar la aplicación.

*Existencia de préstamos vigentes con el número de cuenta de ahorros.

7) Postcondiciones:

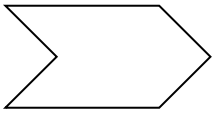
- Se creó una asociación entre usuario y cuenta.

3.10.4 CONSULTA DE FECHA DE PAGO DE CUOTA.

1) Nombre: Consulta de fecha de pago de cuota "FECHA CUOTA PRESTAMO".

2) Responsabilidades: Consultar información confidencial bancaria del usuario acerca de la fecha de la última cuota cancelada y el valor de la misma.

3) Referencias: Caso de Uso Fecha cuota préstamo

Diagrama secuencia 

4) Notas:

* En el menú principal elegir la opción correcta de fecha cuota préstamo.

5) Excepciones:

* Tener en cuenta si la cuenta de ahorros está activa.

*Tener en cuenta si existen préstamos vigentes aliados al número de cuenta del usuario.

*Tener en cuenta si el crédito no fue otorgado en el presente mes.

6) Precondiciones:

*Ingreso de datos correctos.

*La aplicación debe tener los datos del usuario correctos.

*El usuario debe tener permisos del administrador para manejar la aplicación.

*Existencia de préstamos vigentes con el número de Cuenta.

*La aplicación debe enviar correctamente número de cuenta, usuario y contraseña automáticamente.

7) Postcondiciones:

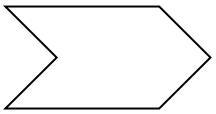
- Se creó una asociación entre usuario y cuenta.

3.10.5 CONSULTA DE FECHA DE VENCIMIENTO DE CUOTA.

1) Nombre: Consulta de fecha de vencimientos de cuota "FECHA VENCIMIENTOS PRESTAMOS".

2) Responsabilidades: Consultar información confidencial bancaria del usuario acerca de la próxima fecha de pago y el monto de la misma.

3) Referencias: Caso de Uso fecha vencimientos préstamos

Diagrama secuencia 

4) Notas:

* En el menú principal elegir la opción correcta de fecha vencimientos préstamos.

5) Excepciones:

* Tener en cuenta si la cuenta de ahorros está activa.

*Tener en cuenta si existe préstamos vigentes aliados al número de cuenta de ahorros del usuario.

6) Precondiciones:

*Ingreso de datos correctos.

*La aplicación debe tener los datos del usuario correctos.

*El usuario debe tener permisos del administrador para manejar la aplicación.

*Existencia de préstamos vigentes con el número de Cuenta.

*La aplicación debe enviar correctamente número de cuenta, usuario y contraseña automáticamente.

7) Postcondiciones:

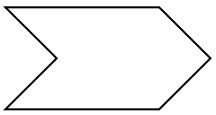
- Se creó una asociación entre usuario y cuenta.

3.10.6 CONSULTA DE ÚLTIMOS MOVIMIENTOS.

1) Nombre: Consulta de últimos movimientos "MOVIMIENTOS CUENTA".

2) Responsabilidades: Consultar información confidencial bancaria del usuario acerca de los 5 últimos movimientos realizados en su cuenta de ahorros aliada al usuario.

3) Referencias: Caso de Uso Movimientos cuenta

Diagrama secuencia 

4) Notas:

* En el menú principal elegir la opción correcta de movimientos cuenta.

5) Excepciones:

* Tener en cuenta si la cuenta de ahorros está activa.

6) Precondiciones:

*Ingreso de datos correctos.

*La aplicación debe tener los datos del usuario correctos.

*El usuario debe tener permisos del administrador para manejar la aplicación.

*La aplicación debe enviar correctamente número de cuenta, usuario y contraseña automáticamente.

7) Postcondiciones:

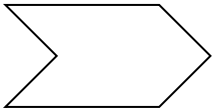
- Se creó una asociación entre usuario y cuenta.

3.10.7 TRANSACCIÓN DE DINERO ENTRE CUENTAS.

1) Nombre: Transacción de dinero entre cuentas "TRANSFERENCIA DE DINERO", [Número de Cuenta destino], [Monto]).

2) Responsabilidades: Realizar una transferencia de dinero de hasta 300 dolares, limitados por 3 movimientos diarios de 100 dólares cada uno entre cuentas pertenecientes a la Cooperativa de Ahorro y Crédito San Francisco Ltda.

3) Referencias: Caso de Uso Transferencia de Dinero

Diagrama secuencia 

4) Notas:

* En el menú principal elegir la opción correcta de Transferencia de Dinero.

*Conocer bien el número de cuenta de destino para evitar transacciones indeseadas.

*Consultar si se tiene saldo disponible en la cuenta antes de realizar una transacción.

*Revisar sus 5 últimos movimientos para verificar que no ha sobrepasado el límite de transferencias por día.

5) Excepciones:

* Tener en cuenta si la cuenta de ahorros está activa.

*Tener en cuenta si la cuenta de ahorros de destino esta activa.

*Saldo Disponible en cuenta de ahorros.

*Cada transferencia no debe exceder de los 100 dólares.

6) Precondiciones:

*Ingreso de número de cuenta destino correcta.

*La aplicación debe tener los datos del usuario correctos.

*El usuario cancela la confirmación de transferencia de dinero entre cuentas.

*El usuario debe tener permisos del administrador para manejar la aplicación.

*La aplicación debe verificar que el usuario no tenga más de 3 movimientos que sean por transferencia.

*La aplicación debe enviar correctamente usuario y número de cuenta.

7) Postcondiciones:

- Se creó una asociación entre usuario, cuenta y cuenta.

3.11 FUNCIONALIDAD.

Para el proceso de compresión de la funcionalidad se determina el desarrollo de la aplicación SF PUCEmóvil con una arquitectura 3 capas:



Figura 3.31 Funcionalidad aplicación 3 capas.

La navegación entre las 3 capas deberá funcionar de la siguiente manera:

- La base de datos es aquella que almacenará toda la transaccionalidad de los usuarios.
- El web service es aquel que permite el tráfico de datos entre la base de datos con el dispositivo móvil, llevada a cabo por una orden del usuario.
- El dispositivo móvil es aquel que permite interactuar mediante el web service con datos de la base de datos al usuario y toda su información.

3.12 ARQUITECTURA.

3.12.1 ARQUITECTURA 3 CAPAS.

3.12.1.1 Capa de Presentación (SFPUCE móvil):

Esta capa es la que ve el usuario, presenta el sistema al usuario, le comunica la información del usuario en un mínimo proceso.

Esta capa se comunica únicamente con la capa de negocio.

Esta capa de presentación también es conocida como interfaz gráfica y tiene la característica de ser amigable, además de presentarse como formularios.

3.12.1.2 Capa de Negocio (Web Service Institucional):

Aquí es donde se reciben las peticiones del usuario y se envían las respuestas tras el proceso.

Se denomina capa de negocio porque es aquel donde se establecen todas las reglas que deben cumplirse.

Esta capa se comunica con la capa de presentación para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos que almacene o recupere datos de él.

3.12.1.3 Capa de Datos (Base de Datos Institucional):

Es donde residen los datos y es la encargada de acceder a los mismos.

Está formada por uno a más gestores de bases de datos que tealizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

3.13 IMPLEMENTACION.

3.13.1 Requerimientos Minimos.

Las herramientas utilizadas son:

Base de Datos.

SQL 2008

Web Service (Capa de Negocio).

C# 2003 (Visual Studio 2003)

Dispositivo Móvil.

Iphone 3Gs

Iphone 4S

Base de Datos.

Se requiere SQL 2008 ya que es la plataforma donde se encuentra desarrollada la base de datos principal del sistema de la Institución, la cual es primordial en este proyecto por la consulta y extracción de datos para la aplicación.

Web Service.

Un web service es importante para gestionar el vínculo entre la base de datos y la aplicación móvil, ya que hace la función de una capa de negocio. Por lo tanto el web service institucional se encuentra desarrollado en la plataforma Visual Studio 2003 en el lenguaje C#. Cabe destacar que para la compatibilidad de web service con la aplicación SFPUCE móvil, se tuvieron que realizar varios cambios en el mismo, tanto como para la transmisión de datos como para la seguridad de la información.

Dispositivo Móvil.

Como se trata de una aplicación de telefónia de banca móvil, se requiere el dispositivo móvil Iphone para la respectiva compilación.

3.14 HERRAMIENTAS.

Para la instalación de estas herramientas ver **ANEXO B.**

3.14.1 XCODE.

Para el desarrollo de la aplicación se utilizó xcode ya que es una serie de herramientas de desarrollo en Mac OS X.

Este es desarrollado por Apple e incluye las herramientas necesarias para programar en el ambiente de Mac OS o iOS (iPhone, iPod Touch, iPad). ALGÚNos lenguajes de programación las cuales están disponibles en Xcode son C, C++, Objective-C, Objective-C++, Java, AppleScript, Python y Ruby, en adición a diferentes modelos de programación como lo son Java, Cocoa y Carbon.

Xcode es completamente funcional para el desarrollo de aplicaciones Mac e incluye un editor de código de clase mundial, un depurador gráfico, e integrado en Objective-C, C y C++ compiladores. Xcode posee un profundo conocimiento de los marcos que el poder del Cacao Mac OS X, e incluso es capaz de identificar los errores mediante el análisis del código que escriba sin ejecutar la aplicación.

La experiencia de desarrollo intuitiva hace que sea fácil de obtener su primera aplicación en funcionamiento, sin embargo, Xcode es tan poderoso que propio Mac OS X está construido utilizando el conjunto de herramientas de exactamente la misma.

Se utilizó xcode ya que ofrece una interfaz de usuario atractiva y potente, además es capaz de organizar y editar los archivos fuente, ver la documentación, crear el producto, limpiar el código y optimizar el rendimiento del programa.

En cuanto a desarrollo xcode se trata de una plataforma integrada con enormes posibilidades de personalización que ofrece prestaciones con las que se crea un entorno de trabajo eficiente y fácil de usar.

Se suministra gratuitamente junto con Mac OS X.

Xcode trabaja conjuntamente con Interface Builder, una herencia de NeXT, una herramienta gráfica para la creación de interfaces de usuario.

Xcode incluye la colección de compiladores del proyecto GNU (GCC), y puede compilar código C, C++, Objective-C, Objective-C++, Java y AppleScript mediante una amplia gama de modelos de programación, incluyendo, pero no limitado a Cocoa, Carbón y Java.

3.14.2 INTERFACE BUILDER.

Se utilizó Interface Builder ya que es una aplicación de desarrollo de software para Mac de Apple, para el sistema operativo OS X. Es parte de Xcode, y forma parte del conjunto de herramientas del desarrollador de Apple Developer Connection.

Se implemento interface builder en el proyecto ya que permite crear interfaces de aplicaciones mediante una interfaz gráfica de usuario. La interfaz resultante se almacena en un archivo Xib.

Además a partir de Xcode 4, Interface Builder ya no existe como una aplicación independiente, y su funcionalidad se integra directamente en Xcode.

3.15 PROGRAMACION

3.15.1 Web Service Institucional

3.15.1.1 Programación de Web Service para conexión con base de datos Institucional.

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
    <appSettings>
            <add key="CadenaConexion"
value="server=;database=;uid=;pwd="/>
    </appSettings>
  <system.web>
      <compilation
        defaultLanguage="c#"
        debug="true"
    />
      <customErrors
    mode="RemoteOnly"
    />
  æ  <authentication mode="Windows" />
  æ  <authorization>
 æ      <allow users="*" />
 æ  </authorization>
æ  <trace
 æ      enabled="false"
 æ      requestLimit="10"
 æ      pageOutput="false"
 æ      traceMode="SortByTime"
            localOnly="true"
 æ  />
 æ  <sessionState
            mode="InProc"
            stateConnectionString="tcpip=127.0.0.1:42424"
     sqlConnectionString="data source=127.0.0.1;user id=password="
            cookieless="false"
            timeout="20"
    />
      <globalization
            requestEncoding="utf-8"
            responseEncoding="utf-8"
   />
    <webServices>
        <protocols>
            <add name="HttpGet"/>
            <add name="HttpPost"/>
        </protocols>
    </webServices>

 </sæstem.web>

</configuration>
```

3.15.1.2 Programación de web service institucional para la aplicación SFPUCE móvil.

```
using System;
using System.Text;
using System.IO;
using System.Configuration;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Diagnostics;
using System.Web;
using System.Web.Services;
using System.Web.Security;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
using System.Security.Principal;
using System.Xml;
using System.Web.Mail;

using RecursosHumanos;
using General;
using CoopUtil;
using Captaciones;
using Seguridad;
using Clientes;
using Contabilidad;
using Colocaciones;
using Parametros;
using Cajas;
using Contabilizaciones;

namespace WebSerSwitchOrm
{
        /// <summary>
        /// Summary description for Service1.
        /// </summary>
        public class Service1 : System.Web.Services.WebService
        {
                private decimal saldo;
                private decimal disponible;
                private int     retorno;
                private string apellido;
                private string nombre;
                private string codigo;
                private      string tipoCodigo;
                private string salida;
                private string apellido1;
                private string nombre1;
                private string codigo1;
                private      string tipoCodigo1;
```

```
        private string apellidoConsulta;
        private string nombreConsulta;

        private string mensaje;

        private ArrayList listaSalida;
        private ArrayList listaSalidaNombres;

        protected Cuenta cuentaPagina;
        protected Caja cajaPagina;

        private string cadena1;
        private string cadena2;
        private string cadena3;
        private string cadena4;
        private string cadena5;

        public Service1()
        {
BaseDatos.FijaCadenaConexion(ConfigurationSettings.AppSettings["CadenaConexion"]);
            //CODEGEN: This call is required by the ASP.NET Web Services Designer
                InitializeComponent();
        }

        #region Component Designer generated code

        //Required by the Web Services Designer
        private IContainer components = null;

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
        }

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        protected override void Dispose( bool disposing )
        {
                if(disposing && components != null)
                {
                        components.Dispose();
                }
                base.Dispose(disposing);
        }

        #endregion
```

3.15.1.3 Programación de web service institucional para autentificación de usuario de SFPUCEmóvil.

```
[WebMethod (Description="Solicita Usuario y contraseña.")]

public  string AccesoSistema(string codigoUsuario, string clave)
{
        try
        {
                this.retorno = 0;

                string unCodigoUsuario = codigoUsuario.ToUpper();
                string unaClave = clave.ToUpper();

                Usuario usuario =
Usuario.GetRegistro(unCodigoUsuario);

                if ( usuario == null )
                {
                        this.retorno = 0;
                        return this.retorno.ToString() ;
                }

                string unaClaveAcceso =
FormsAuthentication.HashPasswordForStoringInConfigFile(unaClave,"SHA1") ;

                if ( !usuario.validaUsuario(unCodigoUsuario,
unaClaveAcceso))
                {
                        this.retorno = -1;
                        return this.retorno.ToString();
                }

        UsuarioEmpleado
unUE=UsuarioEmpleado.GetRegistro(usuario);

                if (unUE==null)
                {
                        return "-2";
                }
                if (usuario.EstaActivo==false)
                {
                        return "2";
                }

                cadena4="CORRECTO";
                cadena5=usuario.Nombre;
                return
cadena4+"\t"+cadena5+"\t"+unUE.EmpleadoAsociado.ClienteAsociado.NumeroCliente.ToString
()+"\t"+unUE.UsuarioAsociado.Email;
        }
        catch
        {
```

```
                                        this.retorno = 99;
                                        cadena1="T1";
                                        cadena2="ERROR";
                                        cadena3="10";
                                        cadena4="FALLO USUARIO/CLAVE INCORRECTA";
                        return cadena1+"\t"+cadena2+"\t"+cadena3+"\t"+cadena4+"\t"+cadena5;
                                }
                        }

                        [WebMethod (Description="Solicita Código (Cédula,pasaporte o
RUC),usuario y contraseña, devuelve la cuenta.")]

public  string ConsultaCuenta(string codigoIdentificacion, string codigoUsuario, string clave)
                        {
                                try
                                {
                                        string cade1;
                                        string cade2="";
                                        this.saldo = 0;
                                        this.disponible = 0;
                                        this.retorno = 0;

                                        string unCodigoUsuario = codigoUsuario.ToUpper();
                                        string unaClave = clave.ToUpper();

                                        Usuario usuario =
Usuario.GetRegistro(unCodigoUsuario);
                                        if ( usuario == null )
                                        {
                                                this.retorno = 10;
                                                cadena1="T1";
                                                cadena2=unCodigoUsuario;
                                                cadena3="10";
                                                cadena4="FALLO";
                        return cadena1+"\t"+cadena2+"\t"+"10"+"\t"+cadena3+"\t"+cadena4+"\t"+cadena5;
                                        }

                                        string unaClaveAcceso =
FormsAuthentication.HashPasswordForStoringInConfigFile(unaClave,"SHA1") ;

                                        if ( !usuario.validaUsuario(unCodigoUsuario,
unaClaveAcceso))
                                        {
                                                this.retorno = 20;
                                                cadena1="T1";
                                                cadena2=unCodigoUsuario;
                                                cadena3="20";
                                                cadena4="FALLO USUARIO/CLAVE
INCORRECTA";
                        return cadena1+"\t"+cadena2+"\t"+"10"+"\t"+cadena3+"\t"+cadena4+"\t"+cadena5;
                                        }

                                        Cliente  unCliente
=Cliente.GetRegistro(codigoIdentificacion);
                                        if ( unCliente == null )
```

```
                                        {
                                        return "T1"+"\t"+"10"+"\t"+"40"+"\t"+"USUARIO
INCORRECTO";
                                        }

        ParametroColocacion parametroColocacion = ParametroColocacion.GetRegistro();
Cuenta cuenta = Cuenta.GetRegistro(unCliente, parametroColocacion.ProductoAhorro,true);
                                        Cuenta cuenta1=
Cuenta.GetRegistro(unCliente,parametroColocacion.ProductoCertificado,true);
                                        if ( cuenta == null )
                                        {
                                return "T1"+"\t"+"10"+"\t"+"40"+"\t"+"USUARIO INCORRECTO";
                                        }
                                        else
                                        {

        cade1=cuenta.ClienteCuenta.NumeroCliente+"\t"+cuenta.Saldo.ToString() ;
                                        }

        cade2=unCliente.Tipo.ToString().Substring(0,1)+"\t"+unCliente.Codigo+"\t"+cuenta1.P
roductoCuenta.Nombre+"\t"+cuenta1.ClienteCuenta.NumeroCliente+"\t"+cuenta1.Saldo.ToStrin
g();
                                        this.retorno = 0;
                                        this.saldo = cuenta.Saldo  ;
                        this.disponible = cuenta.DisponibleTotal -
cuenta.ProductoCuenta.SaldoMinimo ;
                                        return cade1; //+"\t"+cade2;

                                }
                                catch
                                {
                                        return "T1"+"\t"+"10"+"\t"+"90"+"\t"
                                }
                        }
                        [WebMethod (Description="Solicita primer Nombre, primer apellido,usuario
y contraseña, devuelve código (cédula,RUC o pasaporte) y nombres del cliente .")]

                        public  string ConsultaApellidos(string nombreCon,string apellidoCon,string
codigoUsuario, string clave)
                        {
                                string cadenaFinal=" ";
                                try
                                {
                                        string unCodigoUsuario = codigoUsuario.ToUpper();
                                        string unaClave = clave.ToUpper();

                                        string unApellido= apellidoCon.ToUpper();
                                        string unNombre=nombreCon.ToUpper();

                                        Usuario usuario =
Usuario.GetRegistro(unCodigoUsuario);
                                        if ( usuario == null )
                                        {
                                return "T1"+"\t"+"3"+"\t"+"10"+"\t"+"USUARIO INCORRECTO";
                                        }
```

```
                                        string unaClaveAcceso =
FormsAuthentication.HashPasswordForStoringInConfigFile(unaClave,"SHA1");

                                        if ( !usuario.validaUsuario(unCodigoUsuario,
unaClaveAcceso) )
                                        {
                        return "T1"+"\t"+"3"+"\t"+"20"+"\t"+"USUARIO/CLAVE INCORRECTO";
                                        }
                                        ArrayList listaCliente = new ArrayList();
                        listaCliente=Cliente.GetRegistrosApellidoNombre(unApellido,unNombre);
                                        if ( listaCliente == null )
                                        {

                                return "T1"+"\t"+"3"+"\t"+"40"+"\t"+"CLIENTE NO EXISTE";

                                        }
                                        string []cadenaNombres=new string[10];
                                        int i=0;
                                        string terminal="T1";
                                        string proceso="3";
                                        string codError="0";
                                        string estado="Correcto";
                                        string separacion="\t";
                                        foreach (Cliente unCliente in listaCliente)
                                        {

            cadenaNombres[i]=unCliente.NombreUnido+separacion+unCliente.Codigo+separacio
n+unCliente.Tipo.ToString().Substring(0,1)+separacion;
                                                i++;

                                        }

                                        foreach (string unaCade in cadenaNombres)
                                        {
                                                cadenaFinal=cadenaFinal+unaCade;
                                        }

                                        return
terminal+separacion+proceso+separacion+codError+separacion+estado+separacion+cadenaFi
nal.Trim();
                                }
                                catch
                                {
                                        return "T1"+"\t"+"3"+"\t"+"10"+"\t"+"FALLO"+"\t";
                                }
                        }

                        [WebMethod (Description="Solicita cédula, RUC o pasaporte, usuario y
contraseña, devuelve nombres y código.")]
                        public  string ConsultaCodigo(string codigo,string codigoUsuario, string
clave)
                        {
                                try
                                {
                                        string unCodigoUsuario = codigoUsuario.ToUpper();
```

```
                                        string unaClave = clave.ToUpper();
                                        string unCodigo= codigo.ToUpper();

                                        Usuario usuario =
Usuario.GetRegistro(unCodigoUsuario);
                                        if ( usuario == null )
                                        {
                                                this.retorno = 10;
                                return "T1"+"\t"+"22"+"\t"+"10"+"\t"+"USUARIO INCORRECTO";
                                        }
                                        string unaClaveAcceso =
FormsAuthentication.HashPasswordForStoringInConfigFile(unaClave,"SHA1");
                                        if ( !usuario.validaUsuario(unCodigoUsuario,
unaClaveAcceso) )
                                        {
                                                this.retorno = 20;
                                return "T1"+"\t"+"22"+"\t"+"20"+"\t"+"USUARIO/CLAVE
INCORRECTO";
                                        }

                                        Cliente  unCliente = Cliente.GetRegistro(codigo);
                                        if ( unCliente == null )
                                        {

                                                this.retorno = 40;
                                return "T1"+"\t"+"22"+"\t"+"40"+"\t"+"CLIENTE NO EXISTE"+"\t";

                                        }

                                        this.retorno = 0;
                                        this.apellido = unCliente.NombreUnido;
                                        this.codigo=unCliente.Codigo;

        this.tipoCodigo=unCliente.Tipo.ToString().Substring(0,1);
                                        this.apellido1 = "N/A";
                                        this.codigo1= "N/A";
                                        this.tipoCodigo1="N/A";
return "T1"+"\t"+"22"+"\t"+"0"+"\t"+"OK"+"\t"+this.apellido+"\t"+this.codigo+"\t"+this.tipoCodigo;

                                }
                                catch
                                {
                                        return "T1"+"\t"+"22"+"\t"+"99"+"\t"+"FALLO"+"\t";
                                }
                        }
```

3.15.1.4 Programación de web service institucional para consulta de préstamos vigentes y su saldo en SFPUCEmóvil.

```
                [WebMethod]
public  string ConsultaPrestamo(string numeroCuenta, string codigoUsuario, string clave)
                {
                        string cadenaFinal=" ";
                        try
                        {
                                string cade1;
                                string cade2="";
                                this.saldo = 0;
                                this.disponible = 0;
                                string unCodigoUsuario = codigoUsuario.ToUpper();
                                string unaClave = clave.ToUpper();
                                ArrayList listaPrestamos=new ArrayList();
                                string []cadenaPrestamos=new string[5];
                                int i=0;


                                Usuario usuario =
Usuario.GetRegistro(unCodigoUsuario);
                                if ( usuario == null )
                                {
                                        this.retorno = 10;
                                        cadena1="T1";
                                        cadena2=unCodigoUsuario;
                                        cadena3="10";
                                        cadena4="FALLO";
                                        return
cadena3+"\t"+cadena4+"\t"+""+"\t"+""+"\t"+"";
                                }

                                string unaClaveAcceso =
FormsAuthentication.HashPasswordForStoringInConfigFile(unaClave,"SHA1") ;

                                if ( !usuario.validaUsuario(unCodigoUsuario,
unaClaveAcceso))
                                {
                                        this.retorno = 20;
                                        cadena1="T1";
                                        cadena2=unCodigoUsuario;
                                        cadena3="20";
                                        cadena4="FALLO USUARIO/CLAVE
INCORRECTA";
                                        return
cadena3+"\t"+cadena4+"\t"+""+"\t"+""+"\t"+"";
                                }

        Cliente  unCliente =Cliente.GetRegistro(Convert.ToInt32(numeroCuenta));

                                if ( unCliente == null )
                                {
```

```
                    return "40"+"\t"+"CLIENTE NO EXISTE"+"\t"+""+"\t"+""+"\t"+"";
                                }

        ParametroColocacion parametroColocacion = ParametroColocacion.GetRegistro();
        TipoPrestamo unTipoPrestamo= TipoPrestamo.GetRegistro("100");
Cuenta cuenta = Cuenta.GetRegistro(unCliente, parametroColocacion.ProductoAhorro,true);
Prestamo unPrestamo = Prestamo.GetRegistro(unCliente.NumeroCliente,unTipoPrestamo);

        listaPrestamos=Prestamo.GetRegistros(unCliente.NumeroCliente,false);

                                if ( cuenta == null )
                                {
                            return "10"+"\t"+"CTA NO EXISTE"+"\t"+""+"\t"+""+"\t"+"";
                                }
                                else
                                {
                                        foreach (Prestamo unPrestamo1 in
listaPrestamos)
                                        {

        cadenaPrestamos[i]=unPrestamo1.NumeroPagare.ToString()+"\t"+unPrestamo1.Sald
oActual.ToString();
                                                        i++;


                                        }
                                        foreach (string unaCade in
cadenaPrestamos)
                                        {

        cadenaFinal=cadenaFinal+"\t"+unaCade;
                                        }

fechaVencimiento=unPrestamo.FechaVencimientoCuota(numCuota);

        cade1=unPrestamo.NumeroPagare+"\t"+unPrestamo.SaldoActual.ToString()+"\t"+un
Prestamo.FechaUltimoPago.ToShortDateString()+"\t"+fechaVencimiento.ToShortDateString();
                                }

cuenta.ProductoCuenta.SaldoMinimo;
                                return cadenaFinal.Trim();

                            }
                            catch
                            {
                                return "T1"+"\t"+"10"+"\t"+"90"+"\t";
                            }
                    }
```

3.15.1.5 Programación de web service institucional para consulta saldo cuenta de ahorros de SFPUCEmóvil.

```
[WebMethod]
public  string ConsultaCuentaSal(string numeroCuenta, string codigoUsuario, string clave)
{
try
{
string cade1;
string cade2="";
this.saldo = 0;
this.disponible = 0;
string unCodigoUsuario = codigoUsuario.ToUpper();
string unaClave = clave.ToUpper();

Usuario usuario =
Usuario.GetRegistro(unCodigoUsuario);
if ( usuario == null )
{
this.retorno = 10;
cadena1="T1";
cadena2=unCodigoUsuario;
cadena3="10";
cadena4="FALLO";
return
cadena3+"\t"+cadena4+"\t"+""+"\t"+""+"\t"+"";
}

string unaClaveAcceso =
FormsAuthentication.HashPasswordForStoringInConfigFile(unaClave,"SHA1") ;

if ( !usuario.validaUsuario(unCodigoUsuario,
unaClaveAcceso))
{
this.retorno = 20;
cadena1="T1";
cadena2=unCodigoUsuario;
cadena3="20";
cadena4="FALLO USUARIO/CLAVE
INCORRECTA";
return
cadena3+"\t"+cadena4+"\t"+""+"\t"+""+"\t"+"";
}

Cliente  unCliente =Cliente.GetRegistro(Convert.ToInt32(numeroCuenta));

if ( unCliente == null )
{
return "40"+"\t"+"CLIENTE NO
EXISTE"+"\t"+""+"\t"+""+"\t"+"";
}
```

```
        ParametroColocacion parametroColocacion = ParametroColocacion.GetRegistro();
Cuenta cuenta = Cuenta.GetRegistro(unCliente, parametroColocacion.ProductoAhorro,true);

                                        if ( cuenta == null )
                                        {
                                return "10"+"\t"+"CTA NO EXISTE"+"\t"+""+"\t"+""+"\t"+"";
                                        }
                                        else
                                        {

        cade1="0"+"\t"+"OK"+"\t"+cuenta.ProductoCuenta.Nombre+"\t"+cuenta.Saldo.ToStrin
g()+"\t"+ Convert.ToString(cuenta.Retenciones+cuenta.Disponible+cuenta.Encajes);
                                        }

                                        this.retorno = 0;
                                        this.saldo = cuenta.Saldo  ;
                        this.disponible = cuenta.DisponibleTotal -
cuenta.ProductoCuenta.SaldoMinimo;
                                        return cade1;

                                }
                                catch
                                {
                                        return "T1"+"\t"+"10"+"\t"+"90"+"\t";
                                }
                        }
```

3.15.1.6 Programación de web service institucional para consulta de 5 últimos movimientos de SFPUCEmóvil.

```
[WebMethod]
public  string ConsultaMovimientos(string numeroCuenta, string codigoUsuario, string clave)
                        {
                                string cadenaFinal=" ";
                                try
                                {
                                        this.saldo = 0;
                                        this.disponible = 0;
                                        string unCodigoUsuario = codigoUsuario.ToUpper();
                                        string unaClave = clave.ToUpper();
                                        ArrayList listaMovimientos=new ArrayList();
                                        string []cadenaMovimientos=new string[5];
                                        int i=0;

                                        Usuario usuario =
Usuario.GetRegistro(unCodigoUsuario);
                                        if ( usuario == null )
                                        {
                                                this.retorno = 10;
                                                cadena1="T1";
                                                cadena2=unCodigoUsuario;
                                                cadena3="10";
                                                cadena4="FALLO";
                                                return
cadena3+"\t"+cadena4+"\t"+""+"\t"+""+"\t"+"";
                                        }

                                        string unaClaveAcceso =
FormsAuthentication.HashPasswordForStoringInConfigFile(unaClave,"SHA1") ;

                                        if ( !usuario.validaUsuario(unCodigoUsuario,
unaClaveAcceso))
                                        {
                                                this.retorno = 20;
                                                cadena1="T1";
                                                cadena2=unCodigoUsuario;
                                                cadena3="20";
                                                cadena4="FALLO USUARIO/CLAVE
INCORRECTA";
                                                return
cadena3+"\t"+cadena4+"\t"+""+"\t"+""+"\t"+"";
                                        }

                        Cliente  unCliente =Cliente.GetRegistro(Convert.ToInt32(numeroCuenta));


                                        if ( unCliente == null )
                                        {
```

```
                                        return "40"+"\t"+"CLIENTE NO
EXISTE"+"\t"+""+"\t"+""+"\t"+"";
                                        }

ParametroColocacion parametroColocacion = ParametroColocacion.GetRegistro();
Cuenta cuenta = Cuenta.GetRegistro(unCliente, parametroColocacion.ProductoAhorro,true);

                                        if ( cuenta == null )
                                        {
                                        return "10"+"\t"+"CTA NO
EXISTE"+"\t"+""+"\t"+""+"\t"+"";
                                        }
                                        else
                                        {

        listaMovimientos=MovimientoCuenta.GetUltimosRegistros(cuenta);

                                foreach (MovimientoCuenta unMovimientoCuenta in
listaMovimientos)


                        {
                                                        if (i<5)
                                                        {

        cadenaMovimientos[i]=unMovimientoCuenta.ValorEfectivo.ToString()+"\t"+unMovimie
ntoCuenta.Fecha.ToShortDateString()+"\t"+ unMovimientoCuenta.Transaccion.Siglas;
                                                                i++;
                                                        }
                                                        else
                                                        {
                                                                break;
                                                        }
                                                }

                                        foreach (string unaCade in cadenaMovimientos)
                                                {

        cadenaFinal=cadenaFinal+"\t"+unaCade;
                                                }
                                                return cadenaFinal.Trim();
                                        }
                                }
                                catch(Exception e)
                                {
                                        return "ERROR";
                                }
                        }
```

3.15.1.7 Programación de web service institucional para consulta de fecha de último pago de cuota de préstamo y su valor en SFPUCEmóvil.

```
[WebMethod (Description="Solicita NumPagare,NumCuota ,usuario y contraseña, devuelve los datos.")]
                public  string ConsultaDatosPrestamo(int numeroPagare)
                {

                        try
                        {
                                string cade1;
                                this.saldo = 0;
                                this.disponible = 0;
                                this.retorno = 0;
                                int numPagare=numeroPagare;

                                Prestamo unPrestamo=Prestamo.GetRegistro(numPagare);
                                string datoUltimoPago=unPrestamo.UltimoPago().ToString();
                                string datoValorPagado=unPrestamo.ValorCuota.ToString();
                                decimal valorpp=0.00M;
                                DateTime  fechapp=DateTime.Now;
ArrayList listaProximoPago=unPrestamo.ProximoPago(DateTime.Now,out valorpp,out fechapp);
                                if ( unPrestamo == null )
                                {
                        return "T1"+"\t"+"10"+"\t"+"40"+"\t"+"NUMPAGARE INCORRECTO";
                                }
                                else
                                {

          cade1=datoUltimoPago+"\t"+datoValorPagado+"\t"+listaProximoPago[0]+"\t"+unPrestamo.ValorCuota.ToString();
                                }

                                this.retorno = 0;
                                return cade1;

                        }
                        catch
                        {
                                return "ERROR";
                        }
                }
```

3.15.1.8 Programación de web service institucional para consulta de fecha próximo pago de cuota de préstamo y su valor en SFPUCEmóvil.

```
[WebMethod (Description="Solicita NumPagare,NumCuota ,usuario y contraseña, devuelve los datos.")]
                public  string ConsultaDatosPrestamo(int numeroPagare)
                {

                        try
                        {
                                string cade1;
                                this.saldo = 0;
                                this.disponible = 0;
                                this.retorno = 0;
                                int numPagare=numeroPagare;

                                Prestamo unPrestamo=Prestamo.GetRegistro(numPagare);
                                string datoUltimoPago=unPrestamo.UltimoPago().ToString();
                                string datoValorPagado=unPrestamo.ValorCuota.ToString();
                                decimal valorpp=0.00M;
                                DateTime  fechapp=DateTime.Now;
ArrayList listaProximoPago=unPrestamo.ProximoPago(DateTime.Now,out valorpp,out fechapp);
                                if ( unPrestamo == null )
                                {
                        return "T1"+"\t"+"10"+"\t"+"40"+"\t"+"NUMPAGARE INCORRECTO";
                                }
                                else
                                {

        cade1=datoUltimoPago+"\t"+datoValorPagado+"\t"+listaProximoPago[0]+"\t"+unPrestamo.ValorCuota.ToString();
                                }

                                this.retorno = 0;
                                return cade1;

                        }
                        catch
                        {
                                return "ERROR";
                        }
                }
```

3.15.1.9 Programación de web service institucional para transferencia de dinero entre cuentas de SFPUCEmóvil.

```
[WebMethod]
public  string RealizaTransferencia(int cuentaOrigen, int cuentaDestino,
decimal valor,string codigoUsuario)
{
        object con = BaseDatos.DevuelveConexion();
        object miTran = BaseDatos.DevuelveTransaction(con);
        string cade1=" ";
        try
        {
                this.saldo = 0;
                this.disponible = 0;
                string unCodigoUsuario = codigoUsuario.ToUpper();
                Usuario usuarioCaja=Usuario.GetRegistro("ADMIN");
                //string unaClave = clave.ToUpper();
                string codigoTransaccion="13";
                decimal valorEfectivo=valor;
                Cuenta
miCuenta=Cuenta.GetRegistro(cuentaOrigen,"100");

        Cuenta cuentaContraPartida=Cuenta.GetRegistro(cuentaDestino,"100");
TransaccionCaptacion transaccion = TransaccionCaptacion.GetRegistro(codigoTransaccion);
ParametroColocacion unPC=ParametroColocacion.GetRegistro();
                this.cuentaPagina =
Cuenta.GetRegistro(miCuenta.Codigo);


                MovimientoCuenta elMovimiento=
this.cuentaPagina.ActualizaCaptacionCaptacion(cuentaContraPartida,
        transaccion, valorEfectivo, usuarioCaja, DateTime.Now,
1000.00M ,
                        "Transferencia Prueba",con,miTran);

Contabiliza.ContabilizacionCaptacionCaptacion(cuentaContraPartida, elMovimiento,
DateTime.Now,

                BaseDatos.HaceCommit(miTran);
                cade1="OK EXITO";
                return cade1;
        }
        catch(Exception e)
        {
                return "ERROR "+e.Message ;
        }
        finally
        {
                BaseDatos.HaceClose(con);
        }
}
```

3.15.1.10 Programación de web service institucional para bloqueo de usuario en SFPUCEmóvil.

```
[WebMethod (Description="Desactiva Usuario.")]

public  string DesactivaUsuario(string codigoUsuario)
{
        try
        {
                this.retorno = 0;

                string unCodigoUsuario = codigoUsuario.ToUpper();
                Usuario usuario =
Usuario.GetRegistro(unCodigoUsuario);
                usuario.EstaActivo=false;
                usuario.Guardar();

                cadena4="DESACTIVADO";
                cadena5=usuario.Nombre;
                return cadena4+"\t"+cadena5;

        }
        catch
        {
                this.retorno = 99;
                cadena1="T1";
                cadena2="ERROR AL DESACTIVAR USUARIO";
                cadena5="PUCE";
                return cadena2+"\t"+cadena5;
        }
}
    }
}
```

3.15.2 PROGRAMACIÓN DE LA APLICACIÓN SFPUCE móvil.

3.15.2.1 AUTENTIFICACIÓN DE USUARIO.

```
//
//  AUTENTIFICACIONViewController.h
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 25/11/14.
//  Copyright (c) 2014 PUCE. All rights reserved.
//

#import <UIKit/UIKit.h>

@interface AUTENTIFICACIONViewController : UIViewController <UITextFieldDelegate,
NSXMLParserDelegate>
{

IBOutlet UITextField *codigousuario;
IBOutlet UITextField *clave;
IBOutlet UIActivityIndicatorView *activityIndicator;
IBOutlet UITextView *autenticar;
IBOutlet UIButton *bloqueousuarioboton;
NSMutableData *webData;
NSMutableString *soapResults;
NSURLConnection *conn;
NSXMLParser *xmlParser;
BOOL elementFound;
BOOL control;
BOOL control2;
}

@property (nonatomic, retain) UITextField *codigousuario;
@property (nonatomic, retain) UITextField *clave;
@property (nonatomic, retain) UIActivityIndicatorView *activityIndicator;
@property (nonatomic, retain) IBOutlet UITextView *autenticar;
@property (nonatomic, retain) UIButton *bloqueousuarioboton;

- (IBAction)autenticar:(id)sender;
- (IBAction)bloqueousuarioboton:(id)sender;

@end
```

```
//
//  AUTENTIFICACIONViewController.m
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 25/11/14.
//  Copyright (c) 2014 PUCE. All rights reserved.
//


#import "AUTENTIFICACIONViewController.h"

@interface AUTENTIFICACIONViewController  ()

@end

@implementation AUTENTIFICACIONViewController

@synthesize codigousuario;
@synthesize clave;
@synthesize activityIndicator;
@synthesize autenticar;
@synthesize bloqueousuarioboton;


int bloqueo = 0;


-(void) controlarusuario
{
    if ([codigousuario.text length] >= 11)
    {
        codigousuario.text = [codigousuario.text substringWithRange:NSMakeRange (0,11)];
    }
}


-(void) controlarcontrasena
{
    if ([clave.text length] >= 15)
    {
        clave.text = [clave.text substringWithRange:NSMakeRange (0,14)];
    }
}


- (IBAction)autenticar:(id)sender
{
    int longitudusuario = [codigousuario.text length];
    int longitudcontrasena = [clave.text length];


    NSLog (@"la longitud del usuario es = %d", longitudusuario);
    NSLog (@"la longitud de la contrasena es = %d", longitudcontrasena);
```

```
    if (([codigousuario.text length ] == 0) || ([clave.text length ] == 0))
    {
        NSLog (@"uno de los 2 campos esta vacio");
        NSString *mensaje = @"POR FAVOR INGRESE USUARIO Y CONTRASEÑA";
        NSString *app2 = @"SFPUCE MÓVIL";

    NSString * autentificacionvacio = [NSString stringWithFormat:@"%@\n%@", mensaje , app2];

        UIAlertView *alert = [[UIAlertView alloc]
                              initWithTitle:@"ERROR"
                              message: autentificacionvacio

                              delegate:self
                              cancelButtonTitle:@"OK"
                              otherButtonTitles:nil];
        [alert show];
        [alert release];

    }
    else
    {
        NSString *poststring =
        [NSString stringWithFormat:
@"codigousuario=%@&clave=%@",codigousuario.text,clave.text];
        NSLog(@"%@",poststring);

        NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/AccesoSistema"];

        NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
        NSString *msgLength =
        [NSString stringWithFormat:@"%d", [poststring length]];

  [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-Type"];
        [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
        [req setHTTPMethod:@"POST"];
        [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

        [activityIndicator startAnimating];

        conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];
        if (conn) {
            webData = [[NSMutableData data] retain];
        }
    }
}
-(void) connection:(NSURLConnection *) connection
didReceiveResponse:(NSURLResponse *) response {

        [webData setLength: 0];
}

-(void) connection:(NSURLConnection *) connection
        didReceiveData:(NSData *) data {
```

```
        [webData appendData:data];

}

-(void) connection:(NSURLConnection *) connection
  didFailWithError:(NSError *) error {
        [webData release];
    [connection release];
}

-(void) connectionDidFinishLoading:(NSURLConnection *) connection {

    NSLog(@"DONE. Received Bytes: %d", [webData length]);
        NSString *theXML = [[NSString alloc] initWithBytes: [webData mutableBytes]

                length:[webData length]

          encoding:NSUTF8StringEncoding];
        //---shows the XML---
        NSLog(@"%@",theXML);
        autenticar.text= [NSString stringWithFormat:@"%@",theXML];
        [theXML release];

        [activityIndicator stopAnimating];

if (xmlParser)
{
    [xmlParser release];
}

xmlParser = [[NSXMLParser alloc] initWithData: webData];
[xmlParser setDelegate:self];
[xmlParser setShouldResolveExternalEntities:YES];
[xmlParser parse];

[connection release];
[webData release];

}

-(void) parser:(NSXMLParser *) parser
didStartElement:(NSString *) elementName
  namespaceURI:(NSString *) namespaceURI
 qualifiedName:(NSString *) qName
        attributes:(NSDictionary *) attributeDict {
        if( [elementName isEqualToString:@"string"])
        {
                if (!soapResults)
                {
                        soapResults = [[NSMutableString alloc] init];
                }
                elementFound = YES;
        }
}
```

```
-(void)parser:(NSXMLParser *) parser foundCharacters:(NSString *)string
{
        if (elementFound)
        {
              [soapResults appendString: string];
        }
}

-(void)parser:(NSXMLParser *)parser
didEndElement:(NSString *)elementName
 namespaceURI:(NSString *)namespaceURI
qualifiedName:(NSString *)qName
{

if (control == NO)

{

   if ([elementName isEqualToString:@"string"])
        {
      NSString *micadena = soapResults;
      NSArray* cortar = [micadena componentsSeparatedByString: @"      "];
      NSString* correcto = [cortar objectAtIndex: 0];
      NSLog(@"%@",correcto);

      NSRange detectar;
      detectar = [soapResults rangeOfString: @"CORRECTO"];

      if (detectar.location == NSNotFound)
      {

         if ([soapResults isEqualToString: @"0"])
         {


         NSString *erroruno = @"USUARIO INCORRECTO";
         NSString *app1= @"SFPUCE MOVIL";

         NSString * errorusuario = [NSString stringWithFormat:@"%@\n%@", erroruno, app1];

         NSLog(@"%@",soapResults);
         UIAlertView *alert1 = [[UIAlertView alloc]
                                initWithTitle:@"ACCESO DENEGADO"
                                message: errorusuario
                                delegate:self
                                cancelButtonTitle:@"OK"
                                otherButtonTitles:nil];
         [alert1 show];
         [alert1 release];

         }
         else
         {
            NSString *errordos = @"CONTRASEÑA INCORRECTA";
            NSString *app2= @"SFPUCE MOVIL";
```

```
NSString * errorcontrasena = [NSString stringWithFormat:@"%@\n%@", errordos, app2];

    if ([soapResults isEqualToString: @"-1"])
    {

        if (bloqueo < 2)
        {

        NSLog(@"%@",soapResults);
        UIAlertView *alert1 = [[UIAlertView alloc]
        initWithTitle:@"ACCESO DENEGADO"
        message: errorcontrasena
                        delegate:self
                        cancelButtonTitle:@"OK"
                        otherButtonTitles:nil];
        [alert1 show];
        [alert1 release];

        bloqueo ++;
        NSLog(@"%d",bloqueo);
           webData = nil;
        }
        else
        {
             bloqueo = 0;
           control = YES;
           control2 = YES;
           xmlParser = nil;

           NSString *poststring =
           [NSString stringWithFormat: @"codigoUsuario=%@",codigousuario.text];
           NSLog(@"%@",poststring);
            NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:38/service1.asmx/DesactivaUsuario"];

           NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
           NSString *msgLength =
           [NSString stringWithFormat:@"%d", [poststring length]];
[req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-Type"];
           [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
           [req setHTTPMethod:@"POST"];
           [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

           [activityIndicator startAnimating];

           conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];
           if (conn) {
                           NSLog (@"se efectuop el bloqueo");
           }

           NSLog(@"%@",soapResults);
           UIAlertView *alert1 = [[UIAlertView alloc]
           initWithTitle:@"USUARIO HA SIDO DESACTIVADO"
      message: @"USTED HA INGRESADO ERRONEAMENTE SU CONTRASENA 3 VECES"
           delegate:self
```

```
                cancelButtonTitle:@"OK"
                otherButtonTitles:nil];
                [alert1 show];
                [alert1 release];
            }

        }
        else
        {
            if ([soapResults isEqualToString: @"2"])
            {

 NSString *inactivo1 = @"Por favor comuníquese con atención al cliente y solicite la activación";
                NSString *app3= @"SFPUCE MÓVIL";

                NSString *mensajeinactivo = [NSString stringWithFormat:@"%@\n%@",
inactivo1, app3];

                NSLog(@"%@",soapResults);
                UIAlertView *alerti = [[UIAlertView alloc]
                initWithTitle:@"ALERTA!! USUARIO INACTIVO"
                message: mensajeinactivo
                delegate:self
                cancelButtonTitle:@"OK"
                otherButtonTitles:nil];
                [alerti show];
                [alerti release];
            }

        }
    }
    else

    {
   NSString *micadena2 = soapResults;
   NSArray *cortar2 = [micadena2 componentsSeparatedByString:@"      "];
        NSString * nombre = [cortar2 objectAtIndex: 1];
        NSLog(@"%@",nombre);


   NSString *micadena3 = soapResults;
   NSArray *cortar3 = [micadena3 componentsSeparatedByString:@"      "];
        NSString * cuenta = [cortar3 objectAtIndex: 2];
        NSLog(@"%@",cuenta);

   NSString * imprimirtotal = [NSString stringWithFormat:@"%@\n%@",      correcto, nombre];

   NSString *micadena4 = soapResults;
   NSArray *cortar4 = [micadena4 componentsSeparatedByString:@"      "];
        NSString *correo = [cortar4 objectAtIndex: 3];
        NSLog(@"%@",correo);
        NSString *grabarusuario = codigousuario.text;
        NSLog(@"codigousuario: %@",self.codigousuario.text);
```

```
        NSString *grabarcontrasena = clave.text;
        NSLog(@"clave: %@", self.clave.text);


        NSString *grabarcuenta = cuenta;
        NSLog(@"cuenta: %@",cuenta);


        NSString *grabarcorreo = correo;
        NSLog(@"correo: %@",correo);


NSUserDefaults *variable1 = [NSUserDefaults standardUserDefaults];
    [variable1 setObject:grabarusuario forKey: @"grabarusuario"];
    [variable1 synchronize];

 NSUserDefaults *variable2 = [NSUserDefaults standardUserDefaults];
    [variable2 setObject:grabarcontrasena forKey: @"grabarcontrasena"];
    [variable2 synchronize];

 NSUserDefaults *variable3 = [NSUserDefaults standardUserDefaults];
    [variable3 setObject:grabarcuenta forKey: @"grabarcuenta"];
    [variable3 synchronize];

 NSUserDefaults *variable4 = [NSUserDefaults standardUserDefaults];
    [variable4 setObject:grabarcorreo forKey: @"grabarcorreo"];
    [variable4 synchronize];
        NSLog(@"%@",soapResults);
        UIAlertView *alert = [[UIAlertView alloc]
        initWithTitle:@"ACCESO"
        message: imprimirtotal
        delegate:self
        cancelButtonTitle:@"OK"
        otherButtonTitles:nil];
        [alert show];
        [alert release];

      NSString * storyboardName = @"MainStoryboard";
        UIStoryboard * storyboard = [UIStoryboard storyboardWithName: storyboardName
bundle: nil];
        UIViewController * vc = [storyboard
instantiateViewControllerWithIdentifier:@"NAVIGATION"];
        [self presentViewController: vc animated:YES completion: nil];

     NSString *grabarnombre = nombre;
     NSUserDefaults *variable5 = [NSUserDefaults standardUserDefaults];
        [variable5 setObject:grabarnombre forKey: @"grabarnombre"];
        [variable5 synchronize];

     }

     [soapResults setString:@""];
          elementFound = FALSE;

      }
```

```
}
    else
    {
        [soapResults setString:@""];
                elementFound = FALSE;
        control = NO;
    }
}

-(void)viewDidLoad
{
    [super viewDidLoad];

    [self.tabBarController.tabBar setHidden:YES];

    self.codigousuario.delegate = self;
    self.clave.delegate = self;

    UITapGestureRecognizer * tapGesture = [[UITapGestureRecognizer alloc] initWithTarget:self
action:@selector(hideKeyBoard)];

    [self.view addGestureRecognizer:tapGesture];

[NSTimer scheduledTimerWithTimeInterval:0.1 target:self selector:@selector (controlarusuario)
userInfo:nil repeats:YES];

[NSTimer scheduledTimerWithTimeInterval:0.1 target:self selector:@selector
(controlarcontrasena) userInfo:nil repeats:YES];

    control = NO;
    control2 = NO;
}

-(void)hideKeyBoard
{
    [codigousuario resignFirstResponder];
    [clave resignFirstResponder];
}

-(BOOL)textFieldShouldReturn: (UITextField *)textField

{
    [self.codigousuario resignFirstResponder];
    [self.clave resignFirstResponder];

    return YES;
}
- (void)didReceiveMemoryWarning {
        // Releases the view if it doesn't have a superview.
    [super didReceiveMemoryWarning];

        // Release any cached data, images, etc that aren't in use.
}

- (void)viewDidUnload {
```

```
        // Release any retained subviews of the main view.
        // e.g. self.myOutlet = nil;
}

- (void)dealloc {
        [codigousuario release];
        [clave release];
        [activityIndicator release];
        [xmlParser release];
        [soapResults release];
        [super dealloc];
}

@end
```

3.15.2.2 CONSULTA DE SALDOS.

```
//
//  SALDOS.h
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 30/11/14.
//  Copyright (c) 2014 PUCE. All rights reserved.
//

#import <Foundation/Foundation.h>


@interface SALDOS : UIViewController <NSXMLParserDelegate>


{
    IBOutlet UITextField *numerocuenta;
    IBOutlet UITextField *codigousuario;
    IBOutlet UITextField *clave;
    IBOutlet UIActivityIndicatorView *activityIndicator;
    IBOutlet UITextView *consultasaldo;
    IBOutlet UILabel *recibirusuario;
    IBOutlet UILabel *recibirsaldo;
    IBOutlet UILabel *etiquetasaldos;

NSTimer *idleTimer;

    __weak IBOutlet UILabel *recibirsaludo;

NSMutableData *webData;
NSMutableString *soapResults;
NSURLConnection *conn;

NSXMLParser *xmlParser;
BOOL elementFound;

}

@property (nonatomic, retain) UITextField *numerocuenta;
@property (nonatomic, retain) UITextField *codigousuario;
@property (nonatomic, retain) UITextField *clave;
@property (nonatomic, retain) UIActivityIndicatorView *activityIndicator;
@property (nonatomic, retain) UILabel *recibirusuario;
@property (nonatomic, retain) UILabel *recibirsaldo;
@property (nonatomic, retain) UILabel *etiquetasaldos;

@end
```

```
//
//  SALDOS.m
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 30/11/14.
//  Copyright (c) 2014 PUCE. All rights reserved.
//


#import "SALDOS.h"

@interface SALDOS  ()

@end

#define SFPUCEmovilTimeSeconds 180.0

@implementation SALDOS

@synthesize numerocuenta;
@synthesize codigousuario;
@synthesize clave;
@synthesize activityIndicator;
@synthesize recibirusuario;
@synthesize recibirsaldo;
@synthesize etiquetasaldos;


- (void)resetIdleTimer {
    if (!idleTimer) {
        idleTimer = [[NSTimer scheduledTimerWithTimeInterval:SFPUCEmovilTimeSeconds
                                                      target:self
                                                    selector:@selector(idleTimerExceeded)
                                                    userInfo:nil
                                                     repeats:NO] retain];
    }
    else {
        if (fabs([idleTimer.fireDate timeIntervalSinceNow]) < SFPUCEmovilTimeSeconds-1.0) {
            [idleTimer setFireDate:[NSDate
dateWithTimeIntervalSinceNow:SFPUCEmovilTimeSeconds]];
        }
    }
}

- (void)idleTimerExceeded {
    [idleTimer release]; idleTimer = nil;
    [self resetIdleTimer];

    NSLog(@"TIEMPO DE INACTIVIDAD CUMPLIDO");

    NSString * storyboardName = @"MainStoryboard";
   UIStoryboard * storyboard = [UIStoryboard storyboardWithName: storyboardName bundle: nil];
    UIViewController * vc = [storyboard
instantiateViewControllerWithIdentifier:@"SESIONEXPIRADA"];
    [self presentViewController: vc animated:YES completion: nil];
```

```
}

- (UIResponder *)nextResponder {
    [self resetIdleTimer];
    return [super nextResponder];
}

-(void) connection:(NSURLConnection *) connection
didReceiveResponse:(NSURLResponse *) response {
        [webData setLength: 0];
}

-(void) connection:(NSURLConnection *) connection
        didReceiveData:(NSData *) data {
        [webData appendData:data];
}

-(void) connection:(NSURLConnection *) connection
  didFailWithError:(NSError *) error {
        [webData release];
    [connection release];
}

-(void) connectionDidFinishLoading:(NSURLConnection *) connection {
        NSLog(@"DONE. Received Bytes: %d", [webData length]);
        NSString *theXML = [[NSString alloc] initWithBytes: [webData mutableBytes]

              length:[webData length]

           encoding:NSUTF8StringEncoding];
        NSLog(@"%@",theXML);
        consultasaldo.text= [NSString stringWithFormat:@"%@",theXML];
        [theXML release];

        [activityIndicator stopAnimating];

        if (xmlParser)
        {
              [xmlParser release];
        }

        xmlParser = [[NSXMLParser alloc] initWithData: webData];
        [xmlParser setDelegate:self];
        [xmlParser setShouldResolveExternalEntities:YES];
        [xmlParser parse];
        [connection release];
        [webData release];

}
-(void) parser:(NSXMLParser *) parser
didStartElement:(NSString *) elementName
  namespaceURI:(NSString *) namespaceURI
 qualifiedName:(NSString *) qName
        attributes:(NSDictionary *) attributeDict {
```

```
        if( [elementName isEqualToString:@"string"])
        {
            if (!soapResults)
            {
                    soapResults = [[NSMutableString alloc] init];
            }
            elementFound = YES;
        }
}

-(void)parser:(NSXMLParser *) parser foundCharacters:(NSString *)string
{
        if (elementFound)
        {
            [soapResults appendString: string];
        }
}

-(void)parser:(NSXMLParser *)parser
didEndElement:(NSString *)elementName
 namespaceURI:(NSString *)namespaceURI
qualifiedName:(NSString *)qName
{

    [etiquetasaldos setHidden:NO];

    NSString *sacarsaldo = soapResults;
    NSArray* cortare = [sacarsaldo componentsSeparatedByString: @"      "];
    NSString* saldo = [cortare objectAtIndex: 3];

    self.recibirsaldo.text = saldo;

if ([elementName isEqualToString:@"string"])
        {
            NSLog(@"%@",soapResults);
    }
}

-(void)viewDidLoad
{
    [super viewDidLoad];
    [etiquetasaldos setHidden:YES];


    NSUserDefaults *grabarnombre = [NSUserDefaults standardUserDefaults];
    NSString *cargarnombre = [grabarnombre objectForKey:@"grabarnombre"];
    NSLog (@"%@",cargarnombre);

    self.recibirusuario.text =  cargarnombre;

       NSUserDefaults *grabarusuario = [NSUserDefaults standardUserDefaults];
    NSString *cargarusuario = [grabarusuario objectForKey:@"grabarusuario"];
```

```
    NSUserDefaults *grabarcontrasena = [NSUserDefaults standardUserDefaults];
    NSString *cargarcontrasena = [grabarcontrasena objectForKey: @"grabarcontrasena"];

    NSUserDefaults *grabarcuenta = [NSUserDefaults standardUserDefaults];
    NSString *cargarcuenta = [grabarcuenta objectForKey: @"grabarcuenta"];

    NSString *poststring =
    [NSString stringWithFormat:
@"numeroCuenta=%@&codigoUsuario=%@&clave=%@",cargarcuenta,cargarusuario,cargarco
ntrasena];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaCuentaSal"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];

 [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    [activityIndicator startAnimating];

    conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];

    if (conn) {
        webData = [[NSMutableData data] retain];
    }
[self resetIdleTimer];
}

- (void)didReceiveMemoryWarning {
        // Releases the view if it doesn't have a superview.
    [super didReceiveMemoryWarning];

        // Release any cached data, images, etc that aren't in use.
}
- (void)viewDidUnload {
        // Release any retained subviews of the main view.
        // e.g. self.myOutlet = nil;
}
- (void)dealloc {
        [numerocuenta release];
        [codigousuario release];
        [clave release];
        [activityIndicator release];
        [xmlParser release];
        [soapResults release];
    [recibirusuario release];
        [super dealloc];
}

@end
```

3.15.2.3 CONSULTA DE PRESTAMOS

```
//
//  PRESTAMOS.h
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 06/03/15.
//  Copyright (c) 2015 PUCE. All rights reserved.
//

#import <UIKit/UIKit.h>

@interface PRESTAMOS : UIViewController <NSXMLParserDelegate, UITextFieldDelegate>
{
    IBOutlet UIActivityIndicatorView *activityIndicator;
    IBOutlet UITextView *consultasaldo;
    IBOutlet UITextView *consultaprestamo;
    IBOutlet UILabel *recibirusuario;
    IBOutlet UILabel *recibirsaldo;
    IBOutlet UILabel *recibirprestamo;

    IBOutlet UIImageView *golo;
    IBOutlet UILabel *linea;
    IBOutlet UILabel *linea2;
    IBOutlet UILabel *creditosvigentes;

    IBOutlet UIButton *boton1;
    IBOutlet UIButton *boton2;
    IBOutlet UIButton *boton3;
    IBOutlet UIButton *boton4;
    IBOutlet UIButton *boton5;


    IBOutlet UILabel *etiquetasaldos;

    __weak IBOutlet UILabel *recibirsaludo;

    //---web service access saldos---
    NSMutableData *webDatasaldo;
    NSMutableString *soapResults;
    NSURLConnection *connsaldo;

    NSXMLParser *xmlParsersaldo;

    BOOL elementFoundsaldo;
    BOOL condicion;

    NSTimer *idleTimerp;

}

@property (nonatomic, retain) UIActivityIndicatorView *activityIndicator;
@property (nonatomic, retain) IBOutlet UITextView *consultasaldo;
@property (nonatomic, retain) IBOutlet UITextView *consultaprestamo;
@property (nonatomic, retain) IBOutlet UIImageView *golo;
```

```
@property (nonatomic, retain) UILabel *recibirusuario;
@property (nonatomic, retain) UILabel *recibirprestamo;
@property (nonatomic, retain) UILabel *recibirsaldo;
@property (nonatomic, retain) UILabel *linea;
@property (nonatomic, retain) UILabel *linea2;
@property (nonatomic, retain) UILabel *etiquetasaldos;
@property (nonatomic, retain) UILabel *creditosvigentes;

- (IBAction)boton1:(id)sender;
- (IBAction)boton2:(id)sender;
- (IBAction)boton3:(id)sender;
- (IBAction)boton4:(id)sender;
- (IBAction)boton5:(id)sender;

@end


//
//  PRESTAMOS.m
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 06/03/15.
//  Copyright (c) 2015 PUCE. All rights reserved.
//

#import "PRESTAMOS.h"

@interface PRESTAMOS ()

@end

#define SFPUCEmovilTimeSecondsp 180.0

@implementation PRESTAMOS

@synthesize activityIndicator;
@synthesize consultasaldo;
@synthesize recibirusuario;
@synthesize recibirsaldo;
@synthesize consultaprestamo;
@synthesize recibirprestamo;
@synthesize golo;
@synthesize linea;
@synthesize linea2;
@synthesize etiquetasaldos;
@synthesize creditosvigentes;


int contador0=0;
int contador1=0;
int contador2=0;
int contador3=0;
int contador4=0;
int contador5=0;
int contador6=0;
```

```
int contador7=0;
int contador8=0;
int contador9=0;


int prestamo1;
int prestamo2;
int prestamo3;
int prestamo4;
int prestamo5;


//control inactividad
- (void)resetIdleTimer {
    if (!idleTimerp) {
        idleTimerp = [[NSTimer scheduledTimerWithTimeInterval:SFPUCEmovilTimeSecondsp
                                                       target:self
                                                     selector:@selector(idleTimerExceeded)
                                                     userInfo:nil
                                                      repeats:NO] retain];
    }
    else {
        if (fabs([idleTimerp.fireDate timeIntervalSinceNow]) < SFPUCEmovilTimeSecondsp-1.0) {
            [idleTimerp setFireDate:[NSDate
dateWithTimeIntervalSinceNow:SFPUCEmovilTimeSecondsp]];
        }
    }
}

- (void)idleTimerExceeded {
    [idleTimerp release]; idleTimerp = nil;
    [self resetIdleTimer];

    NSLog(@"TIEMPO DE INACTIVIDAD CUMPLIDO");

    NSString * storyboardName = @"MainStoryboard";
   UIStoryboard * storyboard = [UIStoryboard storyboardWithName: storyboardName bundle: nil];
   UIViewController * vc = [storyboard
instantiateViewControllerWithIdentifier:@"SESIONEXPIRADA"];
    [self presentViewController: vc animated:YES completion: nil];

}
- (UIResponder *)nextResponder {
    [self resetIdleTimer];
    return [super nextResponder];
}

-(void) connection:(NSURLConnection *) connection
didReceiveResponse:(NSURLResponse *) response {
        [webDatasaldo setLength: 0];

}
-(void) connection:(NSURLConnection *) connection
        didReceiveData:(NSData *) data {
        [webDatasaldo appendData:data];
```

```
    }

-(void) connection:(NSURLConnection *) connection
  didFailWithError:(NSError *) error {
        [webDatasaldo release];
    [connection release];
}

-(void) connectionDidFinishLoading:(NSURLConnection *) connection {
        NSLog(@"DONE. Received Bytes: %d", [webDatasaldo length]);

        NSString *theXMLsaldo = [[NSString alloc] initWithBytes: [webDatasaldo mutableBytes]

              length:[webDatasaldo length]

           encoding:NSUTF8StringEncoding];

        NSLog(@"%@",theXMLsaldo);

        consultaprestamo.text= [NSString stringWithFormat:@"%@",theXMLsaldo];
    consultasaldo.text= [NSString stringWithFormat:@"%@",theXMLsaldo];


        [theXMLsaldo release];

        [activityIndicator stopAnimating];

        if (xmlParsersaldo)
        {
              [xmlParsersaldo release];
        }

        xmlParsersaldo = [[NSXMLParser alloc] initWithData: webDatasaldo];
        [xmlParsersaldo setDelegate:self];
        [xmlParsersaldo setShouldResolveExternalEntities:YES];
        [xmlParsersaldo parse];

        [connection release];
        [webDatasaldo release];

}

-(void) parser:(NSXMLParser *) parser
didStartElement:(NSString *) elementName
  namespaceURI:(NSString *) namespaceURI
 qualifiedName:(NSString *) qName
        attributes:(NSDictionary *) attributeDict {

        if( [elementName isEqualToString:@"string"])
        {
              if (!soapResults)
              {
                        soapResults = [[NSMutableString alloc] init];
              }
              elementFoundsaldo = YES;
```

```
        }
    }

-(void)parser:(NSXMLParser *) parser foundCharacters:(NSString *)string
{
        if (elementFoundsaldo)
        {
            [soapResults appendString: string];
        }
    }

-(void)parser:(NSXMLParser *)parser
didEndElement:(NSString *)elementName
 namespaceURI:(NSString *)namespaceURI
qualifiedName:(NSString *)qName
{

    int len = [soapResults length];

    NSLog (@"la longitud es = %d", len);

        if ([soapResults length ] == 0)

        {

        [creditosvigentes setHidden:YES];

        NSUserDefaults *grabarnombre = [NSUserDefaults standardUserDefaults];
        NSString *cargarnombre = [grabarnombre objectForKey:@"grabarnombre"];

         NSString *atencion1 = [NSString stringWithFormat:@"Estimado/a %@ usted no tiene
créditos vigentes",cargarnombre];

        UIAlertView *notieneprestamos1 = [[UIAlertView alloc]
                                          initWithTitle:@"ATENCION"
                                          message:atencion1
                                          delegate:self
                                          cancelButtonTitle:@"OK"
                                          otherButtonTitles:nil];
            [notieneprestamos1 show];
            [notieneprestamos1 release];

    }

    else
    {
        [creditosvigentes setHidden:NO];

    NSArray *array = [soapResults componentsSeparatedByString: @"      "];

    for (int i = 0; i<array.count; i++)

    {
        NSLog(@"%@", [array objectAtIndex:i]);
```

```
    if (i == 0)
    {
        contador0 = 0;
    }
    else
    {
       if (i == 2)
        {
            contador2 = 2;
        }
        else
        {
            if (i==4)
            {
                contador4 = 4;
            }
            else
            {
                if (i==6)
                {
                    contador6 = 6;
                }
                else
                    if (i==8)
                    {
                        contador8 =8;
                    }
            }

        }

    }

}

if (contador0 == 0)
{
    [boton1 setHidden:NO];
    NSString *variable1 = [array objectAtIndex: 0];

    NSString *pprestamo = [NSString stringWithFormat:@"# préstamo 1: %@",variable1];
    NSLog(@"%@",pprestamo);
    [boton1 setTitle:pprestamo forState:UIControlStateNormal];

}
else
{

}

   if (contador2 == 2)

{
    [boton2 setHidden:NO];
    NSString *variable2 = [array objectAtIndex: 2];
```

```
    NSString *sprestamo = [NSString stringWithFormat:@"# préstamo 2: %@",variable2];
    NSLog(@"%@",sprestamo);

    [boton2 setTitle:sprestamo forState:UIControlStateNormal];
}

else
{

}

if (contador4 == 4)

{
    [boton3 setHidden:NO];
    NSString *variable3 = [array objectAtIndex: 4];

    NSString *tprestamo = [NSString stringWithFormat:@"# préstamo 3: %@",variable3];
    NSLog(@"%@",tprestamo);
    [boton3 setTitle:tprestamo forState:UIControlStateNormal];

}
else
{

}

if (contador6 == 6)

{
    [boton4 setHidden:NO];
    NSString *variable4 = [array objectAtIndex: 6];

    NSString *cprestamo = [NSString stringWithFormat:@"# préstamo 4: %@",variable4];
    NSLog(@"%@",cprestamo);

    [boton4 setTitle:cprestamo forState:UIControlStateNormal];

}
else
{

}

if (contador8 == 8)

{
    [boton5 setHidden:NO];
    NSString *variable5 = [array objectAtIndex: 8];

    NSString *qprestamo = [NSString stringWithFormat:@"# préstamo 5: %@",variable5];
    NSLog(@"%@",qprestamo);

    [boton5 setTitle:qprestamo forState:UIControlStateNormal];
```

```
    }
    else
    {
    }
    }

}

- (IBAction)boton1:(id)sender

{
    NSString *resultado1 = soapResults;
    NSArray* cortare = [resultado1 componentsSeparatedByString: @"      "];
    NSString* prestamo1 = [cortare objectAtIndex: 1];

    self.recibirprestamo.text = prestamo1;
    [etiquetasaldos setHidden:NO];

}

- (IBAction)boton2:(id)sender

{

    NSString *resultado2 = soapResults;
    NSArray* cortare = [resultado2 componentsSeparatedByString: @"      "];
    NSString* prestamo2 = [cortare objectAtIndex: 3];

    self.recibirprestamo.text = prestamo2;
    [etiquetasaldos setHidden:NO];
}

- (IBAction)boton3:(id)sender

{

    NSString *resultado3 = soapResults;
    NSArray* cortare = [resultado3 componentsSeparatedByString: @"      "];
    NSString* prestamo3 = [cortare objectAtIndex: 5];

    self.recibirprestamo.text = prestamo3;
    [etiquetasaldos setHidden:NO];
}

- (IBAction)boton4:(id)sender

{
    NSString *resultado4 = soapResults;
    NSArray* cortare = [resultado4 componentsSeparatedByString: @"      "];
    NSString* prestamo4 = [cortare objectAtIndex: 7];
    self.recibirprestamo.text = prestamo4;
    [etiquetasaldos setHidden:NO];
}
```

```
- (IBAction)boton5:(id)sender
{
    NSString *resultado5 = soapResults;
    NSArray* cortare = [resultado5 componentsSeparatedByString: @"      "];
    NSString* prestamo5 = [cortare objectAtIndex: 9];

    self.recibirprestamo.text = prestamo5;
    [etiquetasaldos setHidden:NO];
}

-(void)viewDidLoad
{
    [super viewDidLoad];
    [boton1 setHidden:YES];
    [boton2 setHidden:YES];
    [boton3 setHidden:YES];
    [boton4 setHidden:YES];
    [boton5 setHidden:YES];
    [creditosvigentes setHidden:YES];

    [etiquetasaldos setHidden:YES];


    NSUserDefaults *grabarnombre = [NSUserDefaults standardUserDefaults];
    NSString *cargarnombre = [grabarnombre objectForKey:@"grabarnombre"];
    NSLog (@"%@",cargarnombre);

    self.recibirusuario.text =  cargarnombre;

    condicion = NO;

    [self resetIdleTimer];
   condicion = YES;

    NSUserDefaults *grabarcuenta = [NSUserDefaults standardUserDefaults];
    NSString *cargarcuenta = [grabarcuenta objectForKey:@"grabarcuenta"];

    NSUserDefaults *grabarusuario = [NSUserDefaults standardUserDefaults];
    NSString *cargarusuario = [grabarusuario objectForKey:@"grabarusuario"];


    NSUserDefaults *grabarcontrasena = [NSUserDefaults standardUserDefaults];
    NSString *cargarcontrasena = [grabarcontrasena objectForKey: @"grabarcontrasena"];

    NSString *poststring =
    [NSString stringWithFormat:
@"numeroCuenta=%@&codigoUsuario=%@&clave=%@",cargarcuenta,cargarusuario,cargarco
ntrasena];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaPrestamo"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];
```

```
 [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    [activityIndicator startAnimating];

    connsaldo = [[NSURLConnection alloc] initWithRequest:req delegate:self];
    if (connsaldo) {
        webDatasaldo = [[NSMutableData data] retain];
    }
    golo.alpha = 0.3;
    linea.alpha =0.3;
    linea2.alpha =0.3;
}
- (void)didReceiveMemoryWarning {
        // Releases the view if it doesn't have a superview.
    [super didReceiveMemoryWarning];
        // Release any cached data, images, etc that aren't in use.
}
- (void)viewDidUnload {
        // Release any retained subviews of the main view.
        // e.g. self.myOutlet = nil;
}
- (void)dealloc {
        [activityIndicator release];
        [xmlParsersaldo release];
        [soapResults release];
        [super dealloc];
}

@end
```

3.15.2.4 CONSULTA DE FECHA DE ULTIMO PAGO DE CUOTA

```
//
//  FECHAPAGOSCUOTA.h
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 14/03/15.
//  Copyright (c) 2015 PUCE. All rights reserved.
//

#import <UIKit/UIKit.h>

@interface FECHAPAGOSCUOTA : UIViewController <NSXMLParserDelegate,
UITextFieldDelegate>

{
IBOutlet UILabel *recibirusuario;

IBOutlet UIActivityIndicatorView *activityIndicator;

    IBOutlet UITextView *consultaprestamo;
    IBOutlet UITextView *consultafecha;

    IBOutlet UILabel *recibirfecha;
    IBOutlet UILabel *frase1;
    IBOutlet UILabel *frase2;
    IBOutlet UILabel *frase3;
    IBOutlet UILabel *frase4;
    IBOutlet UILabel *recibirsaldocuota;
    IBOutlet UILabel *frase5;
    IBOutlet UILabel *creditosvigentespc;


    IBOutlet UIImageView *logo;

    IBOutlet UIButton *beten1;
    IBOutlet UIButton *beten2;
    IBOutlet UIButton *beten3;
    IBOutlet UIButton *beten4;
    IBOutlet UIButton *beten5;

    BOOL separador;

    NSMutableData *webData;
    NSMutableString *soapResults;
    NSURLConnection *conn;

    NSXMLParser *xmlParser;

     NSTimer *idleTimerpc;


    BOOL elementFoundsaldo;

}
```

```
@property (nonatomic, retain) UILabel *recibirusuario;
@property (nonatomic, retain) UILabel *recibirfecha;
@property (nonatomic, retain) UILabel *recibirsaldocuota;
@property (nonatomic, retain) UIImageView *logo;
@property (nonatomic, retain) UILabel *frase1;
@property (nonatomic, retain) UILabel *frase2;
@property (nonatomic, retain) UILabel *frase3;
@property (nonatomic, retain) UILabel *frase4;
@property (nonatomic, retain) UILabel *frase5;
@property (nonatomic, retain) UILabel *creditosvigentespc;


@property (nonatomic, retain) IBOutlet UITextView *consultaprestamo;
@property (nonatomic, retain) IBOutlet UITextView *consultafecha;

@property (nonatomic, retain) UIActivityIndicatorView *activityIndicator;


- (IBAction)beten1:(id)sender;
- (IBAction)beten2:(id)sender;
- (IBAction)beten3:(id)sender;
- (IBAction)beten4:(id)sender;
- (IBAction)beten5:(id)sender;


@end


//
//  FECHAPAGOSCUOTA.m
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 14/03/15.
//  Copyright (c) 2015 PUCE. All rights reserved.
//

#import "FECHAPAGOSCUOTA.h"

@interface FECHAPAGOSCUOTA ()

@end

#define SFPUCEmovilTimeSecondspc 180.0

@implementation FECHAPAGOSCUOTA

@synthesize recibirusuario;
@synthesize recibirfecha;
@synthesize activityIndicator;
@synthesize logo;
@synthesize frase1;
@synthesize frase2;
@synthesize frase3;
@synthesize frase4;
@synthesize recibirsaldocuota;
```

```
@synthesize frase5;
@synthesize consultaprestamo;
@synthesize consultafecha;
@synthesize creditosvigentespc;


int
secuencial0=0,secuencial1=0,secuencial2=0,secuencial3=0,secuencial4=0,secuencial5=0,secu
encial6=0,secuencial7=0,secuencial8=0,secuencial9=0;


- (void)resetIdleTimer {
    if (!idleTimerpc) {
        idleTimerpc = [[NSTimer scheduledTimerWithTimeInterval:SFPUCEmovilTimeSecondspc
                                                        target:self
                                                      selector:@selector(idleTimerExceeded)
                                                      userInfo:nil
                                                       repeats:NO] retain];
    }
    else {
        if (fabs([idleTimerpc.fireDate timeIntervalSinceNow]) < SFPUCEmovilTimeSecondspc-1.0) {
            [idleTimerpc setFireDate:[NSDate
dateWithTimeIntervalSinceNow:SFPUCEmovilTimeSecondspc]];
        }
    }
}

- (void)idleTimerExceeded {
    [idleTimerpc release]; idleTimerpc = nil;
    [self resetIdleTimer];

    NSLog(@"TIEMPO DE INACTIVIDAD CUMPLIDO");

    NSString * storyboardName = @"MainStoryboard";
    UIStoryboard * storyboard = [UIStoryboard storyboardWithName: storyboardName bundle:
nil];
    UIViewController * vc = [storyboard
instantiateViewControllerWithIdentifier:@"SESIONEXPIRADA"];
    [self presentViewController: vc animated:YES completion: nil];

}

- (UIResponder *)nextResponder {
    [self resetIdleTimer];
    return [super nextResponder];
}

-(void) connection:(NSURLConnection *) connection
didReceiveResponse:(NSURLResponse *) response {
        [webData setLength: 0];

}

-(void) connection:(NSURLConnection *) connection
        didReceiveData:(NSData *) data {
```

```
        [webData appendData:data];

}

-(void) connection:(NSURLConnection *) connection
  didFailWithError:(NSError *) error {
        [webData release];
    [connection release];
}

-(void) connectionDidFinishLoading:(NSURLConnection *) connection {
        NSLog(@"DONE CONEXION PRESTAMOS. Received Bytes: %d", [webData length]);

        NSString *theXML = [[NSString alloc] initWithBytes: [webData mutableBytes]
                                                    length:[webData length]
                                                  encoding:NSUTF8StringEncoding];

        NSLog(@"%@",theXML);

        consultaprestamo.text= [NSString stringWithFormat:@"%@",theXML];

        [theXML release];

        [activityIndicator stopAnimating];

        if (xmlParser)
        {
              [xmlParser release];
        }

        xmlParser = [[NSXMLParser alloc] initWithData: webData];
        [xmlParser setDelegate:self];
        [xmlParser setShouldResolveExternalEntities:YES];
        [xmlParser parse];

        [connection release];
        [webData release];
   }

-(void) parser:(NSXMLParser *) parser
didStartElement:(NSString *) elementName
  namespaceURI:(NSString *) namespaceURI
 qualifiedName:(NSString *) qName
        attributes:(NSDictionary *) attributeDict {

        if( [elementName isEqualToString:@"string"])
        {

       if (!soapResults)
              {
                     soapResults = [[NSMutableString alloc] init];
              }
              elementFoundsaldo = YES;

}
```

```
}

-(void)parser:(NSXMLParser *) parser foundCharacters:(NSString *)string
   {
      if (elementFoundsaldo)
      {
         [soapResults appendString: string];
      }
   }

-(void)parser:(NSXMLParser *)parser
didEndElement:(NSString *)elementName
 namespaceURI:(NSString *)namespaceURI
qualifiedName:(NSString *)qName
{

   if (separador == YES)
   {

   int len = [soapResults length];

   NSLog (@"la longitud es = %d", len);

      if ([soapResults length ] == 0)

   {
      [creditosvigentespc setHidden:YES];

      NSUserDefaults *grabarnombre = [NSUserDefaults standardUserDefaults];
      NSString *cargarnombre = [grabarnombre objectForKey:@"grabarnombre"];

    NSString *atencion2 = [NSString stringWithFormat:@"Estimado/a %@ usted no tiene
créditos vigentes",cargarnombre];

      UIAlertView *notieneprestamos2 = [[UIAlertView alloc]
                                 initWithTitle:@"ATENCION"

                                 message:atencion2
                                 delegate:self
                                 cancelButtonTitle:@"OK"
                                 otherButtonTitles:nil];
            [notieneprestamos2 show];
            [notieneprestamos2 release];
         }
      else
   {
      [creditosvigentespc setHidden:NO];

      NSArray *arraypagos = [soapResults componentsSeparatedByString: @"        "];

      for (int e = 0; e<arraypagos.count; e++)
         {
         NSLog(@"%@", [arraypagos objectAtIndex:e]);
         if (e == 0)
         {
```

```
    secuencial0 = 0;

    NSString *reseltede1 = soapResults;
    NSArray* cortare = [reseltede1 componentsSeparatedByString: @"          "];
    NSString* texto1 = [cortare objectAtIndex: 0];

    NSString *grabarpagare1 = texto1;
    NSLog(@"pagare1: %@",texto1);

    NSUserDefaults *pasar1 = [NSUserDefaults standardUserDefaults];
    [pasar1 setObject:grabarpagare1 forKey: @"grabarpagare1"];
    [pasar1 synchronize];

}

else
{

    if (e == 2)
    {
        secuencial2 = 2;

        NSString *reseltede2 = soapResults;
        NSArray* cortare2 = [reseltede2 componentsSeparatedByString: @"     "];
        NSString* texto2 = [cortare2 objectAtIndex: 2];

        NSString *grabarpagare2 = texto2;
        NSLog(@"pagare2: %@",texto2);

        NSUserDefaults *pasar2 = [NSUserDefaults standardUserDefaults];
        [pasar2 setObject:grabarpagare2 forKey: @"grabarpagare2"];
        [pasar2 synchronize];

    }

    else
    {

        if (e==4)
        {
            secuencial4 = 4;

            NSString *reseltede3 = soapResults;
            NSArray* cortare3 = [reseltede3 componentsSeparatedByString: @" "];
            NSString* texto3 = [cortare3 objectAtIndex: 4];

            NSString *grabarpagare3 = texto3;
            NSLog(@"pagare3: %@",texto3);

            NSUserDefaults *pasar3 = [NSUserDefaults standardUserDefaults];
            [pasar3 setObject:grabarpagare3 forKey: @"grabarpagare3"];
            [pasar3 synchronize];

        }
        else
```

```
            {
                if (e==6)
                {
                    secuencial6 = 6;

                    NSString *reseltede4 = soapResults;
                    NSArray* cortare4 = [reseltede4 componentsSeparatedByString: @"            "];
                    NSString* texto4 = [cortare4 objectAtIndex: 6];

                    NSString *grabarpagare4 = texto4;
                    NSLog(@"pagare4: %@",texto4);

                    NSUserDefaults *pasar4 = [NSUserDefaults standardUserDefaults];
                    [pasar4 setObject:grabarpagare4 forKey: @"grabarpagare4"];
                    [pasar4 synchronize];

                }
                else
                    if (e==8)
                    {
                        secuencial8 =8;

                        NSString *reseltede5 = soapResults;
                        NSArray* cortare5 = [reseltede5 componentsSeparatedByString: @"        "];
                        NSString* texto5 = [cortare5 objectAtIndex: 8];

                        NSString *grabarpagare5 = texto5;
                        NSLog(@"pagare5: %@",texto5);

                        NSUserDefaults *pasar5 = [NSUserDefaults standardUserDefaults];
                        [pasar5 setObject:grabarpagare5 forKey: @"grabarpagare5"];
                        [pasar5 synchronize];
                    }
            }
        }
    }
}


if (secuencial0 == 0)

{
    [beten1 setHidden:NO];
    NSString *posicion1 = [arraypagos objectAtIndex: 0];

    NSString *pcprestamo = [NSString stringWithFormat:@"# préstamo 1: %@",posicion1];
    NSLog(@"%@",pcprestamo);
    [beten1 setTitle:pcprestamo forState:UIControlStateNormal];

}



else
```

```
{

}

if (secuencial2 == 2)

{
    [beten2 setHidden:NO];
    NSString *posicion2 = [arraypagos objectAtIndex: 2];

    NSString *scprestamo = [NSString stringWithFormat:@"# préstamo 2: %@",posicion2];
    NSLog(@"%@",scprestamo);

    [beten2 setTitle:scprestamo forState:UIControlStateNormal];

}

else
{

}

if (secuencial4 == 4)

{
    [beten3 setHidden:NO];
    NSString *posicion4 = [arraypagos objectAtIndex: 4];

    NSString *tcprestamo = [NSString stringWithFormat:@"# préstamo 3: %@",posicion4];
    NSLog(@"%@",tcprestamo);

    [beten3 setTitle:tcprestamo forState:UIControlStateNormal];
}
else
{

}

if (secuencial6 == 6)

{
    [beten4 setHidden:NO];
    NSString *posicion6 = [arraypagos objectAtIndex: 6];

    NSString *ccprestamo = [NSString stringWithFormat:@"# préstamo 4: %@",posicion6];
    NSLog(@"%@",ccprestamo);


    [beten4 setTitle:ccprestamo forState:UIControlStateNormal];

}
else
{

}
```

```
        if (secuencial8 == 8)

        {
            [beten5 setHidden:NO];
            NSString *posicion8 = [arraypagos objectAtIndex: 8];

            NSString *qcprestamo = [NSString stringWithFormat:@"# préstamo 5: %@",posicion8];
            NSLog(@"%@",qcprestamo);


            [beten5 setTitle:qcprestamo forState:UIControlStateNormal];

        }
        else
        {
        }
    }
}

    else
    {

        NSArray* respuesta1 = [soapResults componentsSeparatedByString: @"        "];
        NSString* fechapago = [respuesta1 objectAtIndex: 0];

        NSLog(@"%@",fechapago);

        self.recibirfecha.text = fechapago;

        NSArray* respuesta2 = [soapResults componentsSeparatedByString: @"        "];
        NSString* montocuota = [respuesta2 objectAtIndex: 1];

        NSLog(@"%@",montocuota);

        self.recibirsaldocuota.text = montocuota;

    }
}


- (IBAction)beten1:(id)sender

{
    separador = NO;


    NSUserDefaults *grabarpagare1 = [NSUserDefaults standardUserDefaults];
    NSString *cargarpagare1 = [grabarpagare1 objectForKey: @"grabarpagare1"];

    NSString *poststring =
    [NSString stringWithFormat: @"numeroPagare=%@",cargarpagare1];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaDatosPrestamo"];
```

```
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];


    [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    [activityIndicator startAnimating];

    conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];
    if (conn) {
        webData = [[NSMutableData data] retain];
    }

    soapResults = nil;

    [frase3 setHidden:NO];

        [frase4 setHidden:NO];
        [frase5 setHidden:NO];
}


- (IBAction)beten2:(id)sender

{

        separador = NO;


    NSUserDefaults *grabarpagare2 = [NSUserDefaults standardUserDefaults];
    NSString *cargarpagare2 = [grabarpagare2 objectForKey: @"grabarpagare2"];

    NSString *poststring =
    [NSString stringWithFormat: @"numeroPagare=%@",cargarpagare2];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaDatosPrestamo"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];


        [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
        [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
        [req setHTTPMethod:@"POST"];
        [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

        [activityIndicator startAnimating];
```

```
        conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];
        if (conn) {
            webData = [[NSMutableData data] retain];
        }

        soapResults = nil;

        [frase3 setHidden:NO];
        [frase4 setHidden:NO];
        [frase5 setHidden:NO];

}


- (IBAction)beten3:(id)sender

{
        separador = NO;


    NSUserDefaults *grabarpagare3 = [NSUserDefaults standardUserDefaults];
    NSString *cargarpagare3 = [grabarpagare3 objectForKey: @"grabarpagare3"];


    NSString *poststring =
    [NSString stringWithFormat: @"numeroPagare=%@",cargarpagare3];
    NSLog(@"%@",poststring);


        NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaDatosPrestamo"];
        NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
        NSString *msgLength =
        [NSString stringWithFormat:@"%d", [poststring length]];
 [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-Type"];
        [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
        [req setHTTPMethod:@"POST"];
        [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

        [activityIndicator startAnimating];

        conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];
        if (conn) {
            webData = [[NSMutableData data] retain];
        }
        [frase3 setHidden:NO];
        soapResults = nil;
        [frase4 setHidden:YES];

}



- (IBAction)beten4:(id)sender
```

```
{
        separador = NO;


    NSUserDefaults *grabarpagare4 = [NSUserDefaults standardUserDefaults];
    NSString *cargarpagare4 = [grabarpagare4 objectForKey: @"grabarpagare4"];
    NSString *poststring =
    [NSString stringWithFormat: @"numeroPagare=%@",cargarpagare4];
    NSLog(@"%@",poststring);

        NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaDatosPrestamo"];
        NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
        NSString *msgLength =
        [NSString stringWithFormat:@"%d", [poststring length]];

        [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
        [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
        [req setHTTPMethod:@"POST"];
        [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

        [activityIndicator startAnimating];

        conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];
        if (conn) {
            webData = [[NSMutableData data] retain];
        }

        soapResults = nil;

        [frase3 setHidden:NO];
        [frase4 setHidden:NO];
        [frase5 setHidden:NO];
}

- (IBAction)beten5:(id)sender

{

        separador = NO;

       NSUserDefaults *grabarpagare5 = [NSUserDefaults standardUserDefaults];
    NSString *cargarpagare5 = [grabarpagare5 objectForKey: @"grabarpagare5"];

    NSString *poststring =
    [NSString stringWithFormat: @"numeroPagare=%@",cargarpagare5];
    NSLog(@"%@",poststring);


        NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaDatosPrestamo"];
        NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
        NSString *msgLength =
```

```
        [NSString stringWithFormat:@"%d", [poststring length]];

        [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
        [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
        [req setHTTPMethod:@"POST"];
        [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

        [activityIndicator startAnimating];

        conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];
        if (conn) {
            webData = [[NSMutableData data] retain];
        }

        soapResults = nil;

        [frase3 setHidden:NO];
        [frase4 setHidden:NO];
        [frase5 setHidden:NO];
    }

- (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
{
    self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil];
    if (self) {
        // Custom initialization
    }
    return self;
}

- (void)viewDidLoad
{
    [super viewDidLoad];
        // Do any additional setup after loading the view.

      [beten1 setHidden:YES];
    [beten2 setHidden:YES];
    [beten3 setHidden:YES];
    [beten4 setHidden:YES];
    [beten5 setHidden:YES];
    [frase3 setHidden:YES];
    [frase4 setHidden:YES];
    [frase5 setHidden:YES];
    [creditosvigentespc setHidden:YES];

       NSUserDefaults *grabarnombre = [NSUserDefaults standardUserDefaults];
    NSString *cargarnombre = [grabarnombre objectForKey:@"grabarnombre"];
    NSLog (@"%@",cargarnombre);

    self.recibirusuario.text =  cargarnombre;
    [self resetIdleTimer];

    separador = YES;
```

```
    NSUserDefaults *grabarusuario = [NSUserDefaults standardUserDefaults];
    NSString *cargarusuario = [grabarusuario objectForKey:@"grabarusuario"];

       NSUserDefaults *grabarcontrasena = [NSUserDefaults standardUserDefaults];
    NSString *cargarcontrasena = [grabarcontrasena objectForKey: @"grabarcontrasena"];

    NSUserDefaults *grabarcuenta = [NSUserDefaults standardUserDefaults];
    NSString *cargarcuenta = [grabarcuenta objectForKey:@"grabarcuenta"];


    NSString *poststring =
    [NSString stringWithFormat:
@"numeroCuenta=%@&codigoUsuario=%@&clave=%@",cargarcuenta,cargarusuario,cargarco
ntrasena];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaPrestamo"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];

    [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    [activityIndicator startAnimating];

    conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];
    if (conn) {
        webData = [[NSMutableData data] retain];
    }
    logo.alpha = 0.3;
    frase1.alpha = 0.3;
    frase2.alpha =0.3;
}
- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}

- (void)dealloc {
        [activityIndicator release];
        [xmlParser release];
        [soapResults release];

[super dealloc];
}

@end
```

3.15.2.5 CONSULTA FECHA DE VENCIMIENTO DE CUOTA

```
//
//  FECHAVENCIMIENTOSCUOTA.h
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 14/03/15.
//  Copyright (c) 2015 PUCE. All rights reserved.
//

#import <UIKit/UIKit.h>

@interface FECHAVENCIMIENTOSCUOTA : UIViewController <NSXMLParserDelegate,
UITextFieldDelegate>

{

IBOutlet UILabel *recibirusuariov;

IBOutlet UIActivityIndicatorView *activityIndicatorv;

IBOutlet UITextView *consultaprestamov;
IBOutlet UITextView *consultafechav;

IBOutlet UILabel *recibirfechav;
IBOutlet UILabel *frase1v;
IBOutlet UILabel *frase2v;
IBOutlet UILabel *frase3v;
IBOutlet UILabel *frase4v;
IBOutlet UILabel *recibirsaldocuotav;
IBOutlet UILabel *frase5v;
    IBOutlet UILabel *creditosvigentesvc;

IBOutlet UIImageView *logov;

IBOutlet UIButton *bitin1;
IBOutlet UIButton *bitin2;
IBOutlet UIButton *bitin3;
IBOutlet UIButton *bitin4;
IBOutlet UIButton *bitin5;

BOOL separadorv;

NSMutableData *webDatav;
NSMutableString *soapResultsv;
NSURLConnection *connv;

NSXMLParser *xmlParserv;

BOOL elementFoundv;

     NSTimer *idleTimervc;

}
```

```
@property (nonatomic, retain) UILabel *recibirusuariov;
@property (nonatomic, retain) UILabel *recibirfechav;
@property (nonatomic, retain) UILabel *recibirsaldocuotav;
@property (nonatomic, retain) UIImageView *logov;
@property (nonatomic, retain) UILabel *frase1v;
@property (nonatomic, retain) UILabel *frase2v;
@property (nonatomic, retain) UILabel *frase3v;
@property (nonatomic, retain) UILabel *frase4v;
@property (nonatomic, retain) UILabel *frase5v;
@property (nonatomic, retain) UILabel *creditosvigentesvc;
@property (nonatomic, retain) UITextView *consultaprestamov;

@property (nonatomic, retain) IBOutlet UITextView *consultafechav;

@property (nonatomic, retain) UIActivityIndicatorView *activityIndicatorv;


- (IBAction)bitin1:(id)sender;
- (IBAction)bitin2:(id)sender;
- (IBAction)bitin3:(id)sender;
- (IBAction)bitin4:(id)sender;
- (IBAction)bitin5:(id)sender;

@end


//
//  FECHAVENCIMIENTOSCUOTA.m
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 14/03/15.
//  Copyright (c) 2015 PUCE. All rights reserved.
//

#import "FECHAVENCIMIENTOSCUOTA.h"

@interface FECHAVENCIMIENTOSCUOTA ()

@end

#define SFPUCEmovilTimeSecondsvc 180.0

@implementation FECHAVENCIMIENTOSCUOTA

@synthesize recibirusuariov;
@synthesize recibirfechav;
@synthesize activityIndicatorv;
@synthesize logov;
@synthesize frase1v;
@synthesize frase2v;
@synthesize frase3v;
@synthesize frase4v;
@synthesize recibirsaldocuotav;
@synthesize frase5v;
@synthesize consultafechav;
```

```
@synthesize creditosvigentesvc;
@synthesize consultaprestamov;


int
asignador0=0,asignador1=0,asignador2=0,asignador3=0,asignador4=0,asignador5=0,asignador
6=0,asignador7=0,asignador8=0,asignador9=0;

- (void)resetIdleTimer {
    if (!idleTimervc) {
        idleTimervc = [[NSTimer scheduledTimerWithTimeInterval:SFPUCEmovilTimeSecondsvc
                                                        target:self
                                                      selector:@selector(idleTimerExceeded)
                                                      userInfo:nil
                                                       repeats:NO] retain];
    }
    else {
        if (fabs([idleTimervc.fireDate timeIntervalSinceNow]) < SFPUCEmovilTimeSecondsvc-1.0) {
            [idleTimervc setFireDate:[NSDate
dateWithTimeIntervalSinceNow:SFPUCEmovilTimeSecondsvc]];
        }
    }
}

- (void)idleTimerExceeded {
    [idleTimervc release]; idleTimervc = nil;
    [self resetIdleTimer];

    NSLog(@"TIEMPO DE INACTIVIDAD CUMPLIDO");

    NSString * storyboardName = @"MainStoryboard";
    UIStoryboard * storyboard = [UIStoryboard storyboardWithName: storyboardName bundle:
nil];
    UIViewController * vc = [storyboard
instantiateViewControllerWithIdentifier:@"SESIONEXPIRADA"];
    [self presentViewController: vc animated:YES completion: nil];

}

- (UIResponder *)nextResponder {
    [self resetIdleTimer];
    return [super nextResponder];
}

-(void) connection:(NSURLConnection *) connection
didReceiveResponse:(NSURLResponse *) response {
        [webDatav setLength: 0];

}

-(void) connection:(NSURLConnection *) connection
didReceiveData:(NSData *) data {
        [webDatav appendData:data];

}
```

```
-(void) connection:(NSURLConnection *) connection
didFailWithError:(NSError *) error {
        [webDatav release];
    [connection release];
}


-(void) connectionDidFinishLoading:(NSURLConnection *) connection {
        NSLog(@"DONE CONEXION PRESTAMOS. Received Bytes: %d", [webDatav length]);

        NSString *theXML = [[NSString alloc] initWithBytes: [webDatav mutableBytes]
                                        length:[webDatav length]
                                      encoding:NSUTF8StringEncoding];


        //---shows the XML---
        NSLog(@"%@",theXML);
        consultaprestamov.text= [NSString stringWithFormat:@"%@",theXML];
        [theXML release];
        [activityIndicatorv stopAnimating];

        if (xmlParserv)
        {
              [xmlParserv release];
        }

        xmlParserv = [[NSXMLParser alloc] initWithData: webDatav];
        [xmlParserv setDelegate:self];
        [xmlParserv setShouldResolveExternalEntities:YES];
        [xmlParserv parse];

        [connection release];
        [webDatav release];

}

-(void) parser:(NSXMLParser *) parser
didStartElement:(NSString *) elementName
namespaceURI:(NSString *) namespaceURI
qualifiedName:(NSString *) qName
attributes:(NSDictionary *) attributeDict {


        if( [elementName isEqualToString:@"string"])
        {

       if (!soapResultsv)
              {
                     soapResultsv = [[NSMutableString alloc] init];
              }
              elementFoundv = YES;
    }
}
```

```
-(void)parser:(NSXMLParser *) parser foundCharacters:(NSString *)string
{
    if (elementFoundv)
    {
        [soapResultsv appendString: string];
    }
}

-(void)parser:(NSXMLParser *)parser
didEndElement:(NSString *)elementName
namespaceURI:(NSString *)namespaceURI
qualifiedName:(NSString *)qName
{

    if (separadorv == YES)
    {

        int len = [soapResultsv length];

        NSLog (@"la longitud es = %d", len);


        if ([soapResultsv length ] == 0)

        {
            NSUserDefaults *grabarnombre = [NSUserDefaults standardUserDefaults];
            NSString *cargarnombre = [grabarnombre objectForKey:@"grabarnombre"];

            NSString *atencion3 = [NSString stringWithFormat:@"Estimado/a %@ usted no tiene
créditos vigentes",cargarnombre];

            UIAlertView *notieneprestamos3 = [[UIAlertView alloc]
                                              initWithTitle:@"ATENCION"

                                              message:atencion3
                                              delegate:self
                                              cancelButtonTitle:@"OK"
                                              otherButtonTitles:nil];
            [notieneprestamos3 show];
            [notieneprestamos3 release];
        }

        else
        {
            [creditosvigentesvc setHidden:NO];

            NSArray *arraypagos = [soapResultsv componentsSeparatedByString: @"    "];

            for (int e = 0; e<arraypagos.count; e++)

            {
                NSLog(@"%@", [arraypagos objectAtIndex:e]);

                if (e == 0)
                {
```

```
    asignador0 = 0;

    NSString *risiltidi1 = soapResultsv;
    NSArray* cortarev = [risiltidi1 componentsSeparatedByString: @"          "];
    NSString* pag1 = [cortarev objectAtIndex: 0];

    NSString *guardarpagare1 = pag1;
    NSLog(@"pagare1: %@",pag1);

    NSUserDefaults *poner1 = [NSUserDefaults standardUserDefaults];
    [poner1 setObject:guardarpagare1 forKey: @"guardarpagare1"];
    [poner1 synchronize];

}

else
{

    if (e == 2)
    {
        asignador2 = 2;

        NSString *risiltidi2 = soapResultsv;
        NSArray* cortare2v = [risiltidi2 componentsSeparatedByString: @"    "];
        NSString* pag2 = [cortare2v objectAtIndex: 2];


        NSString *guardarpagare2 = pag2;
        NSLog(@"pagare2: %@",pag2);

        NSUserDefaults *poner2 = [NSUserDefaults standardUserDefaults];
        [poner2 setObject:guardarpagare2 forKey: @"guardarpagare2"];
        [poner2 synchronize];

    }

    else
    {

        if (e==4)
        {
            asignador4 = 4;

            NSString *risiltidi3 = soapResultsv;
            NSArray* cortare3v = [risiltidi3 componentsSeparatedByString: @""];
            NSString* pag3 = [cortare3v objectAtIndex: 4];

            NSString *guardarpagare3 = pag3;
            NSLog(@"pagare3: %@",pag3);

            NSUserDefaults *poner3 = [NSUserDefaults standardUserDefaults];
            [poner3 setObject:guardarpagare3 forKey: @"guardarpagare3"];
            [poner3 synchronize];

        }
```

```
            else
            {
                if (e==6)
                {
                    asignador6 = 6;

                    NSString *risiltidi4 = soapResultsv;
                    NSArray* cortare4v = [risiltidi4 componentsSeparatedByString: @"           "];
                    NSString* pag4 = [cortare4v objectAtIndex: 6];

                    NSString *guardarpagare4 = pag4;
                    NSLog(@"pagare4: %@",pag4);

                    NSUserDefaults *poner4 = [NSUserDefaults standardUserDefaults];
                    [poner4 setObject:guardarpagare4 forKey: @"guardarpagare4"];
                    [poner4 synchronize];

                }
                else
                    if (e==8)
                    {
                        asignador8 =8;

                        NSString *risiltidi5 = soapResultsv;
                     NSArray* cortare5v = [risiltidi5 componentsSeparatedByString: @"           "];
                        NSString* pag5 = [cortare5v objectAtIndex: 8];
                        NSString *guardarpagare5 = pag5;
                        NSLog(@"pagare5: %@",pag5);
                        NSUserDefaults *poner5 = [NSUserDefaults standardUserDefaults];
                        [poner5 setObject:guardarpagare5 forKey: @"guardarpagare5"];
                        [poner5 synchronize];

                    }
            }

        }

    }

}
      if (asignador0 == 0)

{
    [bitin1 setHidden:NO];
    NSString *posicion1 = [arraypagos objectAtIndex: 0];

 NSString *pvprestamo = [NSString stringWithFormat:@"# préstamo 1: %@",posicion1];
    NSLog(@"%@",pvprestamo);

    [bitin1 setTitle:pvprestamo forState:UIControlStateNormal];
}

else
{
```

```
}

if (asignador2 == 2)

{
    [bitin2 setHidden:NO];
    NSString *posicion2 = [arraypagos objectAtIndex: 2];

  NSString *svprestamo = [NSString stringWithFormat:@"# préstamo 2: %@",posicion2];
    NSLog(@"%@",svprestamo);

    [bitin2 setTitle:svprestamo forState:UIControlStateNormal];
}

else
{

}

if (asignador4 == 4)

{
    [bitin3 setHidden:NO];
    NSString *posicion4 = [arraypagos objectAtIndex: 4];


    NSString *tvprestamo = [NSString stringWithFormat:@"# préstamo 3: %@",posicion4];
    NSLog(@"%@",tvprestamo);

    [bitin3 setTitle:tvprestamo forState:UIControlStateNormal];

}
else
{

}

if (asignador6 == 6)

{
    [bitin4 setHidden:NO];
    NSString *posicion6 = [arraypagos objectAtIndex: 6];

 NSString *cvprestamo = [NSString stringWithFormat:@"# préstamo 4: %@",posicion6];
    NSLog(@"%@",cvprestamo);
    [bitin4 setTitle:cvprestamo forState:UIControlStateNormal];
            }
else
{

}

if (asignador8 == 8)

{
```

```
            [bitin5 setHidden:NO];
            NSString *posicion8 = [arraypagos objectAtIndex: 8];

           NSString *qvprestamo = [NSString stringWithFormat:@"# préstamo 4: %@",posicion8];
            NSLog(@"%@",qvprestamo);

            [bitin5 setTitle:qvprestamo forState:UIControlStateNormal];
                    }
          else
          {
          }
      }
   }
      else
   {

      NSArray* respuesta1 = [soapResultsv componentsSeparatedByString: @"       "];
      NSString* fechapago = [respuesta1 objectAtIndex: 2];

      NSLog(@"%@",fechapago);

      self.recibirfechav.text = fechapago;

      NSArray* respuesta2 = [soapResultsv componentsSeparatedByString: @"       "];
      NSString* montocuota = [respuesta2 objectAtIndex: 3];

      NSLog(@"%@",montocuota);

      self.recibirsaldocuotav.text = montocuota;

   }
}

- (IBAction)bitin1:(id)sender

{
      separadorv = NO;

   NSUserDefaults *guardarpagare1 = [NSUserDefaults standardUserDefaults];
   NSString *salvarpagare1 = [guardarpagare1 objectForKey: @"guardarpagare1"];

   NSString *poststring =
   [NSString stringWithFormat: @"numeroPagare=%@",salvarpagare1];
   NSLog(@"%@",poststring);

   NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaDatosPrestamo"];
   NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
   NSString *msgLength =
   [NSString stringWithFormat:@"%d", [poststring length]];

   [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
   [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
   [req setHTTPMethod:@"POST"];
```

```
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    [activityIndicatorv startAnimating];

    connv = [[NSURLConnection alloc] initWithRequest:req delegate:self];
    if (connv) {
        webDatav = [[NSMutableData data] retain];
    }

    soapResultsv = nil;

    [frase3v setHidden:NO];
    [frase4v setHidden:NO];
    [frase5v setHidden:NO];


}

- (IBAction)bitin2:(id)sender

{
     separadorv = NO;

    NSUserDefaults *guardarpagare2 = [NSUserDefaults standardUserDefaults];
    NSString *salvarpagare2 = [guardarpagare2 objectForKey: @"guardarpagare2"];


    NSString *poststring =
    [NSString stringWithFormat: @"numeroPagare=%@",salvarpagare2];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaDatosPrestamo"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];


    [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    [activityIndicatorv startAnimating];

    connv = [[NSURLConnection alloc] initWithRequest:req delegate:self];
    if (connv) {
        webDatav = [[NSMutableData data] retain];
    }

    soapResultsv = nil;

    [frase3v setHidden:NO];
    [frase4v setHidden:NO];
```

```
    [frase5v setHidden:NO];

}


- (IBAction)bitin3:(id)sender

{

    separadorv = NO;

    NSUserDefaults *guardarpagare3 = [NSUserDefaults standardUserDefaults];
    NSString *salvarpagare3 = [guardarpagare3 objectForKey: @"guardarpagare3"];

        NSString *poststring =
    [NSString stringWithFormat: @"numeroPagare=%@",salvarpagare3];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaDatosPrestamo"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];

        [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    [activityIndicatorv startAnimating];

    connv = [[NSURLConnection alloc] initWithRequest:req delegate:self];
    if (connv) {
        webDatav = [[NSMutableData data] retain];
    }
    [frase3v setHidden:NO];
    soapResultsv = nil;

    [frase4v setHidden:YES];


}

- (IBAction)bitin4:(id)sender

{

    separadorv = NO;

    NSUserDefaults *guardarpagare4 = [NSUserDefaults standardUserDefaults];
    NSString *salvarpagare4 = [guardarpagare4 objectForKey: @"guardarpagare4"];


    NSString *poststring =
```

```
    [NSString stringWithFormat: @"numeroPagare=%@",salvarpagare4];
    NSLog(@"%@",poststring);


    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaDatosPrestamo"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];


    [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    [activityIndicatorv startAnimating];

    connv = [[NSURLConnection alloc] initWithRequest:req delegate:self];
    if (connv) {
        webDatav = [[NSMutableData data] retain];
    }

    soapResultsv = nil;

    [frase3v setHidden:NO];
    [frase4v setHidden:NO];
    [frase5v setHidden:NO];
}

- (IBAction)bitin5:(id)sender

{
    separadorv = NO;

    NSUserDefaults *guardarpagare5 = [NSUserDefaults standardUserDefaults];
    NSString *salvarpagare5 = [guardarpagare5 objectForKey: @"guardarpagare5"];


    NSString *poststring =
    [NSString stringWithFormat: @"numeroPagare=%@",salvarpagare5];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaDatosPrestamo"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];

        [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];
```

```
    [activityIndicatorv startAnimating];

    connv = [[NSURLConnection alloc] initWithRequest:req delegate:self];
    if (connv) {
        webDatav = [[NSMutableData data] retain];
    }

    soapResultsv = nil;

    [frase3v setHidden:NO];
    [frase4v setHidden:NO];
    [frase5v setHidden:NO];

}
- (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
{
    self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil];
    if (self) {
        // Custom initialization
    }
    return self;
}

- (void)viewDidLoad
{
    [super viewDidLoad];
         // Do any additional setup after loading the view.

        [bitin1 setHidden:YES];
    [bitin2 setHidden:YES];
    [bitin3 setHidden:YES];
    [bitin4 setHidden:YES];
    [bitin5 setHidden:YES];

    [frase3v setHidden:YES];
    [frase4v setHidden:YES];
    [frase5v setHidden:YES];
    [creditosvigentesvc setHidden:YES];

    NSUserDefaults *grabarnombre = [NSUserDefaults standardUserDefaults];
    NSString *cargarnombre = [grabarnombre objectForKey:@"grabarnombre"];
    NSLog (@"%@",cargarnombre);

    self.recibirusuariov.text =  cargarnombre;


     [self resetIdleTimer];

    separadorv = YES;


    NSUserDefaults *grabarusuario = [NSUserDefaults standardUserDefaults];
    NSString *cargarusuario = [grabarusuario objectForKey:@"grabarusuario"];
```

```
    NSUserDefaults *grabarcontrasena = [NSUserDefaults standardUserDefaults];
    NSString *cargarcontrasena = [grabarcontrasena objectForKey: @"grabarcontrasena"];

        NSUserDefaults *grabarcuenta = [NSUserDefaults standardUserDefaults];
    NSString *cargarcuenta = [grabarcuenta objectForKey:@"grabarcuenta"];

    NSString *poststring =
    [NSString stringWithFormat:
@"numeroCuenta=%@&codigoUsuario=%@&clave=%@",cargarcuenta,cargarusuario,cargarco
ntrasena];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaPrestamo"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];

   [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    [activityIndicatorv startAnimating];

    connv = [[NSURLConnection alloc] initWithRequest:req delegate:self];
    if (connv) {
        webDatav = [[NSMutableData data] retain];
    }

    logov.alpha = 0.3;
    frase1v.alpha = 0.3;
    frase2v.alpha =0.3;
}
- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}

- (void)dealloc {

        [activityIndicatorv release];
        [xmlParserv release];
        [soapResultsv release];
        [super dealloc];
}

@end
```

3.15.2.6 CONSULTA ÚLTIMOS MOVIMIENTOS.

```
//
//  MOVIMIENTOS.h
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 19/03/15.
//  Copyright (c) 2015 PUCE. All rights reserved.
//

#import <UIKit/UIKit.h>

@interface MOVIMIENTOS : UIViewController <NSXMLParserDelegate>


{
    IBOutlet UILabel *recibirusuariom;
    IBOutlet UITextView *consultarmovimiento;

    IBOutlet UIActivityIndicatorView *activityIndicatorm;

    NSMutableData *webDatam;
    NSMutableString *soapResultsm;
    NSURLConnection *connm;

    NSXMLParser *xmlParserm;
    BOOL elementFoundm;

    IBOutlet UITextView *impresionmontos;
    IBOutlet UITextView *impresionfechas;
    IBOutlet UITextView *impresiondescripciones;


    NSTimer *idleTimerm;
    IBOutlet UITextField *ayudadescripcion;
}

@property (nonatomic, retain) IBOutlet UILabel *recibirusuariom;
@property (nonatomic, retain) UIActivityIndicatorView *activityIndicatorm;
@property (nonatomic, retain) IBOutlet UITextView *consultarmovimiento;
@property (nonatomic, retain) IBOutlet UITextView *impresionmontos;
@property (nonatomic, retain) IBOutlet UITextView *impresionfechas;
@property (nonatomic, retain) IBOutlet UITextView *impresiondescripciones;
@property (nonatomic, retain) IBOutlet UITextField *ayudadescripcion;


- (IBAction)ayudadescripcion:(id)sender;

@end
```

```
//
//  MOVIMIENTOS.m
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 19/03/15.
//  Copyright (c) 2015 PUCE. All rights reserved.
//

#import "MOVIMIENTOS.h"

@interface MOVIMIENTOS ()

@end

#define SFPUCEmovilTimeSecondsm 180.0

@implementation MOVIMIENTOS

@synthesize consultarmovimiento;
@synthesize recibirusuariom;
@synthesize activityIndicatorm;
@synthesize impresionmontos;
@synthesize impresionfechas;
@synthesize impresiondescripciones;
@synthesize ayudadescripcion;


NSString *monto1,*monto2,*monto3,*monto4,*monto5;

NSString *fecha1,*fecha2,*fecha3,*fecha4,*fecha5;

NSString *tipo1,*tipo2,*tipo3,*tipo4,*tipo5;


NSString *movimiento1,*movimiento2,*movimiento3,*movimiento4,*movimiento5;


int VARIABLE1=0,VARIABLE2=0,VARIABLE3=0,VARIABLE4=0,VARIABLE5=0;
- (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
{
    self = [super initWithNibName:nibNameOrNil bundle:nibBundleOrNil];
    if (self) {
        // Custom initialization
    }
    return self;
}

- (void)resetIdleTimer {
    if (!idleTimerm) {
        idleTimerm = [[NSTimer scheduledTimerWithTimeInterval:SFPUCEmovilTimeSecondsm
                                                       target:self
                                                     selector:@selector(idleTimerExceeded)
                                                     userInfo:nil
                                                      repeats:NO] retain];
    }
```

```
    else {
        if (fabs([idleTimerm.fireDate timeIntervalSinceNow]) < SFPUCEmovilTimeSecondsm-1.0) {
            [idleTimerm setFireDate:[NSDate
dateWithTimeIntervalSinceNow:SFPUCEmovilTimeSecondsm]];
        }
    }
}

- (void)idleTimerExceeded {
    [idleTimerm release]; idleTimerm = nil;
    [self resetIdleTimer];

    NSLog(@"TIEMPO DE INACTIVIDAD CUMPLIDO");


    NSString * storyboardName = @"MainStoryboard";
    UIStoryboard * storyboard = [UIStoryboard storyboardWithName: storyboardName bundle:
nil];
    UIViewController * vc = [storyboard
instantiateViewControllerWithIdentifier:@"SESIONEXPIRADA"];
    [self presentViewController: vc animated:YES completion: nil];
}

- (UIResponder *)nextResponder {
    [self resetIdleTimer];
    return [super nextResponder];
}

-(void) connection:(NSURLConnection *) connection
didReceiveResponse:(NSURLResponse *) response {
        [webDatam setLength: 0];
}

-(void) connection:(NSURLConnection *) connection
        didReceiveData:(NSData *) data {
        [webDatam appendData:data];
}

-(void) connection:(NSURLConnection *) connection
  didFailWithError:(NSError *) error {
        [webDatam release];
    [connection release];
}

-(void) connectionDidFinishLoading:(NSURLConnection *) connection {
        NSLog(@"DONE. Received Bytes: %d", [webDatam length]);
        NSString *theXML = [[NSString alloc] initWithBytes: [webDatam mutableBytes]

                length:[webDatam length]

          encoding:NSUTF8StringEncoding];
        NSLog(@"%@",theXML);
        consultarmovimiento.text= [NSString stringWithFormat:@"%@",theXML];
        [theXML release];
        [activityIndicatorm stopAnimating];
```

```
        if (xmlParserm)
        {
            [xmlParserm release];
        }

        xmlParserm = [[NSXMLParser alloc] initWithData: webDatam];
        [xmlParserm setDelegate:self];
        [xmlParserm setShouldResolveExternalEntities:YES];
        [xmlParserm parse];

        [connection release];
        [webDatam release];

}

-(void) parser:(NSXMLParser *) parser
didStartElement:(NSString *) elementName
  namespaceURI:(NSString *) namespaceURI
 qualifiedName:(NSString *) qName
        attributes:(NSDictionary *) attributeDict {


        if( [elementName isEqualToString:@"string"])
        {
            if (!soapResultsm)
            {
                    soapResultsm = [[NSMutableString alloc] init];
            }
            elementFoundm = YES;
        }
}
-(void)parser:(NSXMLParser *) parser foundCharacters:(NSString *)string
{
        if (elementFoundm)
        {
            [soapResultsm appendString: string];
        }
}
-(void)parser:(NSXMLParser *)parser
didEndElement:(NSString *)elementName
 namespaceURI:(NSString *)namespaceURI
qualifiedName:(NSString *)qName
{

    NSArray *arraym = [soapResultsm componentsSeparatedByString: @""];


       for (int i = 0; i<arraym.count; i++)

       {
          NSLog(@"%@", [arraym objectAtIndex:i]);

          if (i == 0)
          {
             VARIABLE1 = 0;
```

```
        }
                else
        {

            if (i == 3)
            {
                VARIABLE2 = 3;
            }

            else
            {

                if (i==6)
                {
                    VARIABLE3 = 6;
                    }
                else
                {
                    if (i==9)
                    {
                        VARIABLE4 = 9;
                    }
                    else
                        if (i==12)
                        {
                            VARIABLE5 =12;
                        }
                }

            }

        }

    }


if (VARIABLE1 == 0)

    {

        monto1 = [arraym objectAtIndex:0];
        NSLog(@"%@",monto1);

        fecha1 = [arraym objectAtIndex:1];
        NSLog(@"%@",fecha1);

        tipo1 = [arraym objectAtIndex:2];
        NSLog(@"%@",tipo1);
    }

    else
    {

        monto1 = @"-";
```

```
    NSLog(@"%@",monto1);

    fecha1 = @"-";
    NSLog(@"%@",fecha1);

    tipo1 = @"-";
    NSLog(@"%@",tipo1);
}


if (VARIABLE2 == 3)

{

    monto2 = [arraym objectAtIndex:3];
    NSLog(@"%@",monto2);

    fecha2 = [arraym objectAtIndex:4];
    NSLog(@"%@",fecha2);

    tipo2 = [arraym objectAtIndex:5];
    NSLog(@"%@",tipo2);
}

else
{
    monto2 = @"-";
    NSLog(@"%@",monto2);

    fecha2 = @"-";
    NSLog(@"%@",fecha2);

    tipo2 = @"-";
    NSLog(@"%@",tipo2);

}


if (VARIABLE3 == 6)

{

    monto3 = [arraym objectAtIndex:6];
    NSLog(@"%@",monto3);

    fecha3 = [arraym objectAtIndex:7];
    NSLog(@"%@",fecha3);

    tipo3 = [arraym objectAtIndex:8];
    NSLog(@"%@",tipo3);

}
else
{
    monto3 = @"-";
```

```
    NSLog(@"%@",monto3);

    fecha3 = @"-";
    NSLog(@"%@",fecha3);

    tipo3 = @"-";
    NSLog(@"%@",tipo3);
}


if (VARIABLE4 == 9)

{

    monto4 = [arraym objectAtIndex:9];
    NSLog(@"%@",monto4);

    fecha4 = [arraym objectAtIndex:10];
    NSLog(@"%@",fecha4);

    tipo4 = [arraym objectAtIndex:11];
    NSLog(@"%@",tipo4);
}
else
{
    monto4 = @"-";
    NSLog(@"%@",monto4);

    fecha4 = @"-";
    NSLog(@"%@",fecha4);

    tipo4 = @"-";
    NSLog(@"%@",tipo4);
}


if (VARIABLE5 == 12)

{

    monto5 = [arraym objectAtIndex:12];
    NSLog(@"%@",monto5);

    fecha5 = [arraym objectAtIndex:13];
    NSLog(@"%@",fecha5);

    tipo5 = [arraym objectAtIndex:14];
    NSLog(@"%@",tipo5);
}
else
{
    monto5 = @"-";
    NSLog(@"%@",monto5);

    fecha5 = @"-";
```

```
        NSLog(@"%@",fecha5);

        tipo5 = @"-";
        NSLog(@"%@",tipo5);

    }


    NSString *montos = [NSString
stringWithFormat:@"Valor\n%@\n%@\n%@\n%@\n%@",monto1,monto2,monto3,monto4,mont
o5];


    NSString *fechas = [NSString
stringWithFormat:@"Fecha\n%@\n%@\n%@\n%@\n%@",fecha1,fecha2,fecha3,fecha4,fecha5]
;


    NSString *descripciones = [NSString
stringWithFormat:@"Detalle\n%@\n%@\n%@\n%@\n%@",tipo1,tipo2,tipo3,tipo4,tipo5];


    self.impresionmontos.text = montos;
    self.impresionfechas.text = fechas;
    self.impresiondescripciones.text = descripciones;

    }


-(IBAction)ayudadescripcion:(id)sender

{

    NSString *des1 = @"Inicio ND = nota débito";
    NSString *des2 = @"Inicio NC = nota crédito";
    NSString *des3 = @"DEP = depósito";
    NSString *des4 = @"RET = retiro";
    NSString *des5 = @"NDT = nota de débito por transferencia entre cuentas";

    NSString *ayudita = [NSString
stringWithFormat:@"%@\n%@\n%@\n%@\n%@",des1,des2,des3,des4,des5];

    UIAlertView *alert = [[UIAlertView alloc]
                          initWithTitle:@"ABREVIATURAS"
                          message: ayudita
                          delegate:self
                          cancelButtonTitle:@"Gracias"
                          otherButtonTitles:nil];
    [alert show];
    [alert release];

}

- (void)viewDidLoad
{
```

```
    [super viewDidLoad];

    NSUserDefaults *grabarnombre = [NSUserDefaults standardUserDefaults];
    NSString *cargarnombre = [grabarnombre objectForKey:@"grabarnombre"];
    NSLog (@"%@",cargarnombre);

    self.recibirusuariom.text =  cargarnombre;


    NSUserDefaults *grabarusuario = [NSUserDefaults standardUserDefaults];
    NSString *cargarusuario = [grabarusuario objectForKey:@"grabarusuario"];


    NSUserDefaults *grabarcontrasena = [NSUserDefaults standardUserDefaults];
    NSString *cargarcontrasena = [grabarcontrasena objectForKey: @"grabarcontrasena"];


    NSUserDefaults *grabarcuenta = [NSUserDefaults standardUserDefaults];
    NSString *cargarcuenta = [grabarcuenta objectForKey: @"grabarcuenta"];


    NSString *poststring =
    [NSString stringWithFormat:
@"numeroCuenta=%@&codigoUsuario=%@&clave=%@",cargarcuenta,cargarusuario,cargarco
ntrasena];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaMovimientos"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];


 [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    [activityIndicatorm startAnimating];

    connm = [[NSURLConnection alloc] initWithRequest:req delegate:self];


    if (connm) {
        webDatam = [[NSMutableData data] retain];
    }

     [self resetIdleTimer];

}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
```

```
    // Dispose of any resources that can be recreated.
}

- (void)dealloc {
    [activityIndicatorm release];
        [xmlParserm release];
        [soapResultsm release];
    [recibirusuariom release];
    [impresionmontos release];
    [impresionfechas release];
    [impresiondescripciones release];

        [super dealloc];
}


@end
```

3.15.2.7 TRANSFERENCIA DE DINERO.

```
//
//  TRANSFERENCIADINERO.h
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 22/03/15.
//  Copyright (c) 2015 PUCE. All rights reserved.
//

#import <UIKit/UIKit.h>
#import "SKPSMTPMessage.h"
#import "NSData+Base64Additions.h"


@interface TRANSFERENCIADINERO : UIViewController <SKPSMTPMessageDelegate,
UITextFieldDelegate, NSXMLParserDelegate>
{

    IBOutlet UILabel *recibirusuariot;
    IBOutlet UITextField *cuentadestino;
    IBOutlet UITextField *valordestino;
    IBOutlet UIActivityIndicatorView *activityIndicatort;
    IBOutlet UITextView *transferencia;
    IBOutlet UIButton *transfer;


    NSMutableData *webData;
    NSMutableString *soapResults;
    NSURLConnection *conn;

    NSXMLParser *xmlParser;
    BOOL elementFound;

    NSTimer *idleTimertra;

    BOOL controldiario;

}

@property (nonatomic, retain) UILabel *recibirusuariot;
@property (nonatomic, retain) UITextField *cuentadestino;
@property (nonatomic, retain) UITextField *valordestino;
@property (nonatomic, retain) UIActivityIndicatorView *activityIndicatort;
@property (nonatomic, retain) UITextView *transferencia;
@property (nonatomic, retain) UIButton *transfer;


- (IBAction)transferencia:(id)sender;


@end
```

```
//
//  TRANSFERENCIADINERO.m
//  SFPUCE_MOVIL
//
//  Created by Edwin Munoz on 22/03/15.
//  Copyright (c) 2015 PUCE. All rights reserved.
//

#import "TRANSFERENCIADINERO.h"

@interface TRANSFERENCIADINERO ()

@end

#define SFPUCEmovilTimeSecondstra 180.0

@implementation TRANSFERENCIADINERO

@synthesize recibirusuariot;
@synthesize cuentadestino;
@synthesize valordestino;
@synthesize activityIndicatort;
@synthesize transferencia;
@synthesize transfer;

NSString *monto1t,*monto2t,*monto3t,*monto4t,*monto5t;

NSString *fecha1t,*fecha2t,*fecha3t,*fecha4t,*fecha5t;

NSString *tipo1t,*tipo2t,*tipo3t,*tipo4t,*tipo5t;


NSString *movimiento1t,*movimiento2t,*movimiento3t,*movimiento4t,*movimiento5t;


int agente =0;

int
VARIABLE1T=0,VARIABLE2T=0,VARIABLE3T=0,VARIABLE4T=0,VARIABLE5T=0,intento1=0,i
ntento2=0,intento3=0,intento4=0,intento5=0;


- (void)resetIdleTimer {
    if (!idleTimertra) {
        idleTimertra = [[NSTimer scheduledTimerWithTimeInterval:SFPUCEmovilTimeSecondstra
                                                         target:self
                                                       selector:@selector(idleTimerExceeded)
                                                       userInfo:nil
                                                        repeats:NO] retain];
    }
    else {
        if (fabs([idleTimertra.fireDate timeIntervalSinceNow]) < SFPUCEmovilTimeSecondstra-1.0)
{
            [idleTimertra setFireDate:[NSDate
dateWithTimeIntervalSinceNow:SFPUCEmovilTimeSecondstra]];
```

```
        }
    }
}

- (void)idleTimerExceeded {
    [idleTimertra release]; idleTimertra = nil;
    [self resetIdleTimer];

    NSLog(@"TIEMPO DE INACTIVIDAD CUMPLIDO");


    NSString * storyboardName = @"MainStoryboard";
    UIStoryboard * storyboard = [UIStoryboard storyboardWithName: storyboardName bundle:
nil];
    UIViewController * vc = [storyboard
instantiateViewControllerWithIdentifier:@"SESIONEXPIRADA"];
    [self presentViewController: vc animated:YES completion: nil];

}

- (UIResponder *)nextResponder {
    [self resetIdleTimer];
    return [super nextResponder];
}


-(void) controlarcuentadestino
{
    if ([cuentadestino.text length] >= 6)
    {
        cuentadestino.text = [cuentadestino.text substringWithRange:NSMakeRange (0,6)];

    }
}

-(void) controlarvalordestino
{
    if ([valordestino.text length] >= 3)
    {
        valordestino.text = [valordestino.text substringWithRange:NSMakeRange (0,3)];

    }
}

-(void) connection:(NSURLConnection *) connection
didReceiveResponse:(NSURLResponse *) response {
        [webData setLength: 0];
}

-(void) connection:(NSURLConnection *) connection
        didReceiveData:(NSData *) data {
        [webData appendData:data];
}

-(void) connection:(NSURLConnection *) connection
```

```
  didFailWithError:(NSError *) error {
        [webData release];
    [connection release];
}

-(void) connectionDidFinishLoading:(NSURLConnection *) connection {
        NSLog(@"DONE. Received Bytes: %d", [webData length]);
        NSString *theXML = [[NSString alloc] initWithBytes: [webData mutableBytes]

              length:[webData length]

           encoding:NSUTF8StringEncoding];
        NSLog(@"%@",theXML);
        transferencia.text= [NSString stringWithFormat:@"%@",theXML];
        [theXML release];

        if (xmlParser)
        {
              [xmlParser release];
        }

        xmlParser = [[NSXMLParser alloc] initWithData: webData];
        [xmlParser setDelegate:self];
        [xmlParser setShouldResolveExternalEntities:YES];
        [xmlParser parse];

        [connection release];
        [webData release];

}

-(void) parser:(NSXMLParser *) parser
didStartElement:(NSString *) elementName
  namespaceURI:(NSString *) namespaceURI
 qualifiedName:(NSString *) qName
        attributes:(NSDictionary *) attributeDict {


        if( [elementName isEqualToString:@"string"])
        {
              if (!soapResults)
              {
                        soapResults = [[NSMutableString alloc] init];
              }
              elementFound = YES;
        }
}
-(void)parser:(NSXMLParser *) parser foundCharacters:(NSString *)string
{
        if (elementFound)
        {
              [soapResults appendString: string];
        }
}
```

```
-(void)parser:(NSXMLParser *)parser
didEndElement:(NSString *)elementName
 namespaceURI:(NSString *)namespaceURI
qualifiedName:(NSString *)qName
{

    NSLog(@"%@",soapResults);


    if (controldiario == YES)

    {
 NSArray *arraymt = [soapResults componentsSeparatedByString: @"      "];


        for (int i = 0; i<arraymt.count; i++)

        {
            NSLog(@"%@", [arraymt objectAtIndex:i]);

            if (i == 0)
            {
                VARIABLE1T = 0;
            }
            else
            {
                if (i == 3)
                {
                    VARIABLE2T = 3;
                }
                else
                {
                    if (i==6)
                    {
                        VARIABLE3T = 6;
                    }
                    else
                    {
                        if (i==9)
                        {
                            VARIABLE4T = 9;
                        }
                        else
                            if (i==12)
                            {
                                VARIABLE5T =12;
                            }
                    }

                }

            }

        }
```

```
NSDateFormatter *format = [[NSDateFormatter alloc] init];
format.dateFormat = @"dd/MM/yyyy";
NSLog(@"%@", [format stringFromDate:[NSDate new]]);

NSString *fechahoy = [format stringFromDate:[NSDate date]];
NSLog(@"hoy :%@",fechahoy);


if (VARIABLE1T == 0)

{
    fecha1t = [arraymt objectAtIndex:1];
    NSLog(@"%@",fecha1t);
    tipo1t = [arraymt objectAtIndex:2];
    NSLog(@"%@",tipo1t);


if (([fecha1t isEqual: fechahoy]) && ([tipo1t isEqual: @"NDT"]))

    {

    intento1 = 1;

    }

 else
 {
    NSLog(@"no hubo movimientos en posicion 1");

 }

 }

 else
 {
    NSLog(@"no tiene transferencias en este movimiento 1");

 }


 if (VARIABLE2T == 3)

 {

    fecha2t = [arraymt objectAtIndex:4];
    NSLog(@"%@",fecha2t);

    tipo2t = [arraymt objectAtIndex:5];
    NSLog(@"%@",tipo2t);


if (([fecha2t isEqual: fechahoy]) && ([tipo2t isEqual: @"NDT"]))
    {
       intento2 = 1;
    }
```

```
    else
    {
        NSLog(@"no hubo movimientos en posicion 2");
    }

}

else
{
    NSLog(@"no tiene transferencias en este movimiento 2");
}

if (VARIABLE3T == 6)

{

    fecha3t = [arraymt objectAtIndex:7];
    NSLog(@"%@",fecha3t);

    tipo3t = [arraymt objectAtIndex:8];
    NSLog(@"%@",tipo3t);


if (([fecha3t isEqual: fechahoy]) && ([tipo3t isEqual: @"NDT"]))

    {
        intento3 = 1;
    }

    else
    {
        NSLog(@"no hubo movimientos en posicion 3");
    }
}

else
{
    NSLog(@"no tiene transferencias en este movimiento 3");
}

if (VARIABLE4T == 9)

{

    fecha4t = [arraymt objectAtIndex:10];
    NSLog(@"%@",fecha4t);

    tipo4t = [arraymt objectAtIndex:11];
    NSLog(@"%@",tipo4t);

if (([fecha4t isEqual: fechahoy]) && ([tipo4t isEqual: @"NDT"]))

    {
        intento4 = 1;
```

```
            }

            else
            {
                NSLog(@"no hubo movimientos en posicion 4");
                            }

        }

        else
        {
            NSLog(@"no tiene transferencias en este movimiento 4");
             }


        if (VARIABLE5T == 12)

        {

            fecha5t = [arraymt objectAtIndex:13];
            NSLog(@"%@",fecha5t);

            tipo5t = [arraymt objectAtIndex:14];
            NSLog(@"%@",tipo5t);


       if (([fecha5t isEqual: fechahoy]) && ([tipo5t isEqual: @"NDT"]))

            {
                intento5 = 1;
            }

            else
            {
                NSLog(@"no hubo movimientos en posicion 5");
            }

        }
else
        {
            NSLog(@"no tiene transferencias en este movimiento 5");

        }

agente = intento1 + intento2 + intento3 + intento4 + intento5;
        controldiario = NO;
         soapResults = nil;
}

else
{
if ([soapResults isEqualToString: @"OK EXITO"])

    {
```

```
        NSLog(@"Start Sending");
NSUserDefaults *grabarcorreo = [NSUserDefaults standardUserDefaults];
        NSString *cargarcorreo = [grabarcorreo objectForKey:@"grabarcorreo"];
        NSLog(@"%@",cargarcorreo);

         SKPSMTPMessage *emailMessage = [[SKPSMTPMessage alloc] init];
        emailMessage.fromEmail = @"sfpucemovil@coac-sanfra.com";
        emailMessage.toEmail = cargarcorreo;
        emailMessage.relayHost = @"smtp.1and1.com";
                emailMessage.requiresAuth = YES;
        emailMessage.login = @"sfpucemovil@coac-sanfra.com";
        emailMessage.pass = @"Sfpucemovil";
        emailMessage.subject =@"NOTIFICACION DE TRANSFERENCIA DE DINERO MOVIL";
        emailMessage.wantsSecure = YES;
        emailMessage.delegate = self; //
         NSUserDefaults *grabarnombre = [NSUserDefaults standardUserDefaults];
        NSString *cargarnombre = [grabarnombre objectForKey:@"grabarnombre"];
        NSLog (@"%@",cargarnombre);

        NSString *correo1 = [NSString stringWithFormat:@"Estimado %@",cargarnombre];
        NSString *correo2 = [NSString stringWithFormat:@" usted ha realizado una transferencia
de dinero a la cuenta numero %@",cuentadestino.text];
        NSString *correo3 = [NSString stringWithFormat:@" por un valor de %@
dolares.",valordestino.text];

        NSString *correo4 = [NSString stringWithFormat:@" Gracias por utilizar SFPUCEMóvil"];

        NSString *correo5 = [NSString stringWithFormat:@"COOPERATIVA SAN FRANCISCO
LTDA."];


        NSString *correo6 = [NSString stringWithFormat:@"Si usted NO realizó esta transferencia
por favor comuníquese de inmediato con servicio y atención al cliente."];


        NSString *messageBody = [NSString
stringWithFormat:@"%@%@%@\n\n\n%@\n\n\n%@\n%@",correo1,correo2,correo3,correo6,c
orreo4,correo5];
        NSLog(@"%@",messageBody);


NSDictionary *plainMsg = [NSDictionary

dictionaryWithObjectsAndKeys:@"text/plain",kSKPSMTPPartContentTypeKey,

messageBody,kSKPSMTPPartMessageKey,@"8bit",kSKPSMTPPartContentTransferEncodingK
ey,nil];
        emailMessage.parts = [NSArray arrayWithObjects:plainMsg,nil];

        [emailMessage send];
    }

    else
    {
            [activityIndicatort stopAnimating];
```

```
            UIAlertView *alertt1 = [[UIAlertView alloc]
            initWithTitle:@"ERROR TRANSFERENCIA"
            message: @"Por favor comuniquese con atención al cliente de la Institución para
revisión de estado de transferencia"
                                delegate:self
                                cancelButtonTitle:@"ok"
                                otherButtonTitles:nil];
            [alertt1 show];
            [alertt1 release];

        }
    }
  }

- (IBAction)transferencia:(id)sender;
{

    int destinocontrol = [valordestino.text intValue];

if (destinocontrol <= 100)
{
    if (agente < 3)
    {

        NSString *advertenciaconfirmacion = [NSString stringWithFormat:@"EL VALOR
INGRESADO SERA DESCONTADO DE SU SALDO. CONFIRME SI REALMENTE DESEA
REALIZAR LA TRANSFERENCIA A LA CUENTA Nro %@",cuentadestino.text];

    UIAlertView *alert2 = [[UIAlertView alloc]
                        initWithTitle:@"ATENCIÓN"
                        message: advertenciaconfirmacion
                        delegate:self
                        cancelButtonTitle:@"no"
                        otherButtonTitles:@"si",nil];
    [alert2 show];
    [alert2 release];

    }
    else
    {
        UIAlertView *alertmaximo = [[UIAlertView alloc]
        initWithTitle:@"ACCESO DENEGADO"
        message: @"USTED HA ALCANZADO EL NUMERO DE TRANSFERENCIAS MAXIMAS
POR DIA"
                            delegate:self
                            cancelButtonTitle:@"OK"
                            otherButtonTitles:nil];
        [alertmaximo show];
        [alertmaximo release];
    }

}
     else

     {
```

```
        UIAlertView *alert3 = [[UIAlertView alloc]
        initWithTitle:@"ALERTA"
        message: @"Por políticas de seguridad institucional cada transferencia debe tener un
máximo 100 dólares. Recuerde que solo puede realizar 3 transferencias por día"
                        delegate:self
                        cancelButtonTitle:@"Entendido"
                        otherButtonTitles:nil];

        [alert3 show];
        [alert3 release];

    }
}

-(void) alertView: (UIAlertView *)alertView didDismissWithButtonIndex:(NSInteger)buttonIndex
{
    if (buttonIndex == 1)
    {
        NSLog (@"autorizó");

        [transfer setHidden:YES];


        NSUserDefaults *grabarusuario = [NSUserDefaults standardUserDefaults];
        NSString *cargarusuario = [grabarusuario objectForKey:@"grabarusuario"];


        NSUserDefaults *grabarcuenta = [NSUserDefaults standardUserDefaults];
        NSString *cargarcuenta = [grabarcuenta objectForKey: @"grabarcuenta"];

        NSString *poststring =
        [NSString stringWithFormat:
@"cuentaOrigen=%@&cuentaDestino=%@&valor=%@&codigoUsuario=%@",cargarcuenta,cue
ntadestino.text,valordestino.text,cargarusuario];
        NSLog(@"%@",poststring);

        NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/RealizaTransferencia"];
        NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
        NSString *msgLength =
        [NSString stringWithFormat:@"%d", [poststring length]];


        [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
        [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
        [req setHTTPMethod:@"POST"];
        [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

        [activityIndicatort startAnimating];

        conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];


        if (conn) {
```

```
            webData = [[NSMutableData data] retain];
        }
        [valordestino resignFirstResponder];
        [cuentadestino resignFirstResponder];
}
}

-(void)messageSent:(SKPSMTPMessage *)message{

    [activityIndicatort stopAnimating];
    NSLog(@"MENSAJE ENVIADO");
    UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"TRANSFERENCIA REALIZADA CON
EXITO." message:nil delegate:nil cancelButtonTitle:@"Ok" otherButtonTitles: nil];
    [alert show];
}
-(void)messageFailed:(SKPSMTPMessage *)message error:(NSError *)error{
UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Error!" message:[error
localizedDescription] delegate:nil cancelButtonTitle:@"Ok" otherButtonTitles: nil];
    [alert show];
    NSLog(@"delegate - error(%d): %@", [error code], [error localizedDescription]);

    [transfer setHidden:NO];
}


- (void)viewDidLoad
{
    [super viewDidLoad];

      [transfer setHidden:NO];

    NSUserDefaults *grabarnombre = [NSUserDefaults standardUserDefaults];
    NSString *cargarnombre = [grabarnombre objectForKey:@"grabarnombre"];
    NSLog (@"%@",cargarnombre);

    self.recibirusuariot.text =  cargarnombre;

    self.cuentadestino.delegate = self;
    self.valordestino.delegate = self;


    UITapGestureRecognizer * tapGesture = [[UITapGestureRecognizer alloc] initWithTarget:self
action:@selector(hideKeyBoard)];

    [self.view addGestureRecognizer:tapGesture];


    [NSTimer scheduledTimerWithTimeInterval:0.1 target:self selector:@selector
(controlarcuentadestino) userInfo:nil repeats:YES];

[NSTimer scheduledTimerWithTimeInterval:0.1 target:self selector:@selector
(controlarvalordestino) userInfo:nil repeats:YES];


NSUserDefaults *grabarusuario = [NSUserDefaults standardUserDefaults];
```

```
    NSString *cargarusuario = [grabarusuario objectForKey:@"grabarusuario"];


    NSUserDefaults *grabarcontrasena = [NSUserDefaults standardUserDefaults];
    NSString *cargarcontrasena = [grabarcontrasena objectForKey: @"grabarcontrasena"];


    NSUserDefaults *grabarcuenta = [NSUserDefaults standardUserDefaults];
    NSString *cargarcuenta = [grabarcuenta objectForKey: @"grabarcuenta"];


    NSString *poststring =
    [NSString stringWithFormat:
@"numeroCuenta=%@&codigoUsuario=%@&clave=%@",cargarcuenta,cargarusuario,cargarco
ntrasena];
    NSLog(@"%@",poststring);

    NSURL *url = [NSURL
URLWithString:@"http://201.219.15.138:81/service1.asmx/ConsultaMovimientos"];
    NSMutableURLRequest *req = [NSMutableURLRequest requestWithURL:url];
    NSString *msgLength =
    [NSString stringWithFormat:@"%d", [poststring length]];


    [req addValue:@"application/x-www-form-urlencoded" forHTTPHeaderField:@"Content-
Type"];
    [req addValue:msgLength forHTTPHeaderField:@"Content-Length"];
    [req setHTTPMethod:@"POST"];
    [req setHTTPBody: [poststring dataUsingEncoding:NSUTF8StringEncoding]];

    conn = [[NSURLConnection alloc] initWithRequest:req delegate:self];


    if (conn) {
        webData = [[NSMutableData data] retain];
    }

    controldiario = YES;

     [self resetIdleTimer];
}

-(void)hideKeyBoard
{
    [cuentadestino resignFirstResponder];
    [valordestino resignFirstResponder];

}

-(BOOL)textFieldShouldReturn: (UITextField *)textField

{
    [self.cuentadestino resignFirstResponder];
    [self.valordestino resignFirstResponder];
```

```
    return YES;
}


- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}

- (void)dealloc {
        [cuentadestino release];
        [valordestino release];
        [activityIndicatort release];
        [xmlParser release];
        [soapResults release];
    [recibirusuariot release];

        [super dealloc];
}

@end
```

3.15.3 PRUEBAS.

Para realizar las diferentes consultas se probó pantalla por pantalla verificando que los resultados de las consultas se desplieguen en su totalidad y correctamente.
Además de consultas se realizó pruebas multivista, comprobando que el dispositivo pueda mostrar en pantalla todas las vistas de la aplicación intercambiándose según peticiones de usuario.

Se comprueba el envió de peticiones a través de la aplicación SFPUCE móvil, rectificando que sea correcto la gestión de negocios a través del web service y por último verificacando las respuestas de datos por medio de la capa de datos que en este caso sería la base datos.

Utilizando el método caja negra se pudo observar como interactúa la aplicación con el usuario y los buenos resultados generados en el.

3.15.4 RESULTADOS.

3.15.4.1 AUTENTIFICACIÓN DE USUARIO.

Ingreso de datos pantalla autentificación de usuario.



Figura #3.32 Prueba apertura de app SFPUCEmóvil.

Observación: Se implementó una función en la programación de la aplicación para que al momento de ingresar los datos tanto de usuario como contraseña se muestre un teclado alfanumérico, y otra función que cuando ya no se necesite el teclado al presionar en cualquier parte de la pantalla, éste automáticamente se ocultará.

Además se implemento un control en la caja de usuario que el usuario solo puede ingresar hasta 7 caracteres, y en la caja de texto contraseña hasta 14 caracteres, si el usuario excede estos límites la propia app se encargará de borrar los caracteres no permitidos.

Finalmente existe un control de bloqueo de usuario, si es que el usuario ingresa 3 veces seguidas erróneamente su contraseña, la aplicación SFPUCEmóvil automáticamente procederá con el bloqueo de este código de usuario, por lo que deberá acercarse a la institucón y solicitar el desbloqueo. Control implementado por seguridades.

Resultado de datos correctos en pantalla autentificación de usuario.



Figura # 3.33 Prueba ingreso correcto datos en autentificación de usuario.

Resultado de datos incorrectos pantalla autentificación de usuario.



Figura # 3.34 Prueba ingreso incorrecto de datos en autentificación de usuario.

3.15.4.2 CONSULTA DE SALDOS.

Despliegue de resultados en pantalla de consulta de saldos.



Figura # 3.35 Prueba consulta de saldos en cuenta de ahorro

Observación: La aplicación se encuentra automatizada para que el usuario tenga que seleccionar la opción SALDOS, y obtenga su saldo disponible en ahorros sin tener que ingresar ningún dato, debido a que toda esta información se encuentra relacionada a su código de usuario.

Los resultados se mostrarán siempre y cuando el dispositivo móvil (IPHONE) tenga la debida conectividad a internet, caso contrario la

aplicación se quedara en un entorno de carga y espera, para lo cual en cada una de las pantallas se implementó una animación que se inicia solo al momento de la recepción de datos al momento de getionar con el web service (capa de negocio), denominada activity indicator; esto con el fin de guiar al usuario en que momento la aplicación se encuentra en estado de carga de datos o simplemente en espera de una acción de usuario.

3.15.5 VIRTUALIZACIÓN.

*Para la Instalación de estas herramientas ver **ANEXO B***

Para la elaboración de este proyecto es importante la virtualización por lo que se requiere contar con los sistemas operativos de Mac para su respectivo diseño, programación y pruebas.

En nuestro entorno tecnológico de trabajo la mayoría de equipos operativos cuentan con el sistema operativo WINDOWS, tal vez por el manejo de rutina, por su bajo costo etc, pero no se cuenta mucho con el sistema operativo Macintosh.

Por esta manera se obtuvo la idea de virtualizar otro sistema operativo en un equipo ya que conlleva al ahorro de dinero, consumición exigente de recursos y pues cómo entorno de trabajo ya que rige menos equipos.

3.15.5.1 VIRTUAL BOX.

Se seleccionó la herramienta Virtual Box porque recrea todos los componentes de un ordenador, desde la CPU hasta la tarjeta gráfica, pasando por la conexión a Internet, tarjeta de sonido y conexiones USB.

Aunque es software libre, no corre solamente sobre el mismo, también acepta sistemas operativos privativos como Windows NT 4, 2000, XP, Vista, e incluso está siendo desarrollado para OS/2 Warp y Mac OS X. Lógicamente, funciona también en GNU+Linux, y OpenBSD.

VirtualBox te permite probar programas o configuraciones del sistema sin miedo a perder sus datos, ya que la máquina virtual es un entorno controlado que no afecta a su ordenador.

3.15.5.2 MAC OS X SNOW LEOPARD 10.6.6.

Se trabajó con el sistema operativo Mac Os X Snow Lepard 10.6.6 porque es mucho más centrado y deja las nuevas características en un segundo plano, redoblando esfuerzos en conseguir un sistema mucho más ligero y ágil que el anterior.

Está optimizado para procesadores Intel multi-núcleo, con novedades en el panorama multimedia con la inclusión de QuickTime X y con toda esta serie de ventajas que lo mejoran a nivel de rendimiento, se utilizó uno de los sistemas operativos más fiables del mercado, robusto y además con un rendimiento excelente.

CAPÍTULO IV

4.1 CONCLUSIONES.

- Al momento de la creación y desarrollo de un sistema, lo primero que debe implementarse es la imaginación, innovación en comodidad y creatividad, ya que esto conlleva a la atención e interés del cliente y por lo tanto al comercio destacado de este nuevo sistema.

- Banca móvil es una tecnología muy innovadora en la actualidad, a todo tipo de cliente bancario se le hace muy importante y útil contar con una aplicación como SFPUCEmóvil, teniendo en cuenta que un cliente necesita optimizar el tiempo, el cual es más conocido como Dinero.

- Al realizar un proyecto, si se quiere optimizar recursos y conocer nuevas tecnologías, la virtualización es el mejor método, ya que la virtualización aprovecha de la mejor manera a nuestro ordenador y a parte se puede trabajar en varios sistemas operativos a bajo costo.

- Las herramientas necesarias para el desarrollo de esta aplicación fue xcode, interface builder y Isimulator, que crean un completo entorno de desarrollo como plataforma de código, plataforma de diseño y un simulador Iphone para la respectiva compilación que permite observar que comportamiento tiene la aplicación desarrollada en el dispositivo real.

- El objetivo de todo programador es brindar un servicio y comodidad a cualquier usuario, en este caso, una aplicación de consulta de información bancaria a través del dispositivo móvil Iphone, si un programador conoce el lenguaje C, le será fácil adaptarse a Cocoa y Objective C, el lenguaje utilizado para la realización de este proyecto.

4.2 RECOMENDACIONES.

- Si usted desea tener el sistema operativo de Mac Snow Leopard sin necesidad de adquirir el hardware propio de Mac, pues implemente la virtualización, recomendandole se lo haga en el software VirtualBox o en Vmware (Virtual Machine VM), ya que la última versión de Vmware permite virtualizar MacOs.

- Al momento de descargar una versión de MacSnow Leopard y el sdk de desarrollo de aplicaciones Iphone, asegúrese de que sean compatibles tanto la versión de Mac con el SDK descargado, si no lo hace,esto conllevará nuevamente a descargar una versión diferente de sistema operativo y esto le puede generar pérdida de tiempo.

- Al momento de realizar una nueva máquina virtual tenga la precaución de crear un disco duro con características de expansión dinámica, ya que en la instalación de los componentes necesarios para desarrollo de aplicaciones Iphone no tenga inconvenientes de espacio en disco.

- Si usted desea crear una aplicación similar a SFPUCE móvil, con conexiones a web services, tenga las precauciones de llevar a cabo un control de seguridad electrónica.

- Cuando ejecute la aplicación SFPUCE móvil desde su dispositivo móvil iPhone compruebe que su teléfono tenga conexión a internet, si no se cumple esta recomendación la aplicación tendrá problemas.

- Cuide sus datos, la información bancaria es confidencial, no exponga su integridad y su empresa.

4.3 BIBLIOGRAFÍA.

Alegsa. (s.f.). *Diccionario de Informática y Tecnología*. Recuperado el 11 de Noviembre de 2011, de Definición de SQL: http://www.alegsa.com.ar/Dic/sql.php

Altenberg, B., & Clarke , A. (2008). *Start Programming the Mac Using Objective-C.* Recuperado el 20 de Noviembre de 2011, de http://download.cocoalab.com.s3.amazonaws.com/BecomeAnXcoder%28Tiger%29.pdf

Analia. (2008). *El SDK de Apple para el iPhone y el iPod Touch*. Recuperado el 24 de Noviembre de 2011, de ¿Qué es el SDK de Iphone?: http://www.mastermagazine.info/articulo/12729.php

Banco del Pichincha. (s.f.). *Banca Electrónica*. Recuperado el 13 de Octubre de 2011, de Beneficios: http://wwwp1.pichincha.com/web/temas.php?ID=53

Bermudez, C. (2011). *¿Qué es XCode?* Recuperado el 24 de Noviembre de 2011, de http://www.dgtallika.com/2011/01/xcode-definicion-de-hoy/

Buenastareas. (2010). *Tecnología*. Recuperado el 24 de Noviembre de 2011, de ¿Qué es Xcode?: http://www.buenastareas.com/ensayos/Xcode/286898.html

C., B. G. (2004). *DesarrolloWeb*. Recuperado el 14 de Octubre de 2011, de XML Web Services: http://www.desarrolloweb.com/articulos/1545.php

C., B. G. (2004). *SOAP (Simple Object Access Protocol)*. Recuperado el 14 de Octubre de 2011, de ¿Qué es es Protocolo SOAP?: http://www.desarrolloweb.com/articulos/1557.php

Cad. (s.f.). *Que es Internet*. Recuperado el 14 de Octubre de 2011, de http://www.cad.com.mx/que_es_internet.htm

Castellanos, L. (2010). *Seguridad Electrónica*. Recuperado el 15 de Octubre de 2011, de https://tecnologiaaldia.wordpress.com/category/seguridad-informatica/page/2/

Civila. (s.f.). *Que es Internet*. Recuperado el 14 de Octubre de 2011, de http://www.civila.com/desenredada/que-es.html

CNN Expansion. (2011). *Finanzas Personales*. Recuperado el 2011 de Octubre de 14, de Por que utilizar banca móvil: http://www.cnnexpansion.com/mi-dinero/2011/07/27/10-razones-para-probar-la-banca-movil

Coac-Sanfra. (s.f.). *Información de la Cooperativa de Ahorro y Crédito San Francisco*. Recuperado el 13 de Octubre de 2011, de http://www.coac-sanfra.com/

Connell, S. (1997). *"Desarrollo y gestión de proyectos informáticos".* S.A. Mcgraw-Hill / Interamericana de España. Recuperado el 15 de Noviembre de 2011

Conway, J., & Hillegass, A. (2011). *IOS Programming.* Recuperado el 6 de Septiembre de 2011, de http://www.freelibros.org/e-books/ios-programming-the-big-nerd-ranch-guide-2nd-edition-big-nerd-ranch-guides.html

Coopjep. (s.f.). *¿Que es el Cooperativismo?* Recuperado el 13 de Octubre de 2011, de http://www.coopjep.fin.ec/cooperativismo/que-es-cooperativismo

Coopjep. (2008). *Cooperativismo*. Recuperado el 18 de Octubre de 2011, de Que es una cooperativa: http://www.coopjep.fin.ec/cooperativismo/que-es-una-cooperativa

Coopjep. (2010). *Cooperativismo*. Recuperado el 13 de Octubre de 2011, de ¿Qué es una Cooperativa según la ley ecuatoriana?: http://www.coopjep.fin.ec/cooperativismo/que-es-una-cooperativa

Coopjep. (s.f.). *Valores del Cooperativismo*. Recuperado el 13 de Octubre de 2011, de Cooperativismo: http://www.coopjep.fin.ec/cooperativismo/valores-cooperativismo-universal

Davivienda. (s.f.). *Banca Móvil*. Recuperado el 14 de Octubre de 2011, de https://linea.davivienda.com/opencms/opencms/davivienda/puntosAtencion/BancaMovil.html

Englander, R. (2010). *"JAVA and Soap" Introducción a la programación con el protocolo de conexiones SOAP.* Recuperado el 10 de Noviembre de 2011, de http://www.freelibros.org/programacion/java-and-soap.html

Fecovi. (2004). *La cooperativa*. Recuperado el 15 de Octubre de 2011, de Que es una cooperativa: http://www.fecovi.coop/sistema.php?p=lacooperativa

Geofisica. (2010). *¿Que es el Lenguaje Unificado UML?* Recuperado el 15 de Octubre de 2011, de http://mmc.geofisica.unam.mx/LuCAS/Tutoriales/doc-modelado-sistemas-UML/multiple-html/c12.html

Granneman, S. (2010). *"Mac OS X Snow Leopard for Power Users" Mac OS X Snow Leopard para usuarios avanzados.* Recuperado el 7 de Noviembre de 2011, de

http://www.freelibros.org/programacion/2-books-for-mac-os-x-snow-leopard-users.html

Griffin, J. (2002). *"Creación de sitios web con XML y SQL server 2000".* Pearson Education. Recuperado el 14 de Noviembre de 2011

Gumus. (s.f.). *Informática*. Recuperado el 24 de Noviembre de 2011, de ¿Qué es la herramienta Interface Builder?: http://gumus.informativa.us.es/?p=255

Hardsoftgeek. (21 de Septiembre de 2007). *¿Que es VirtualBox?* Recuperado el 24 de Noviembre de 2011, de https://hardsoftgeek.wordpress.com/2007/09/21/%C2%BFque-es-virtualbox-instalacion-desde-repositorios/

Howard, M., Vieg, J., & Leblanc, D. (2007). *"19 puntos críticos sobre seguridad de software: Fallas más comunues".* Mcgraw-Hill Interamericana de España. Recuperado el 14 de Noviembre de 2011

Izura, P. X. (s.f.). *IP Tables*. Recuperado el 19 de Noviembre de 2011, de Manual Práctico: http://www.pello.info/filez/firewall/iptables.html

Jacobson, I., Booch, G., & Rumbau, J. (2000). *"El proceso unificado de proceso de software".* Recuperado el 15 de Noviembre de 2011

Kernighan, B. (1978). *"El lenguaje de programación C".* Recuperado el 15 de Noviembre de 2011

Liga Coop. (2011). *Institucional*. Recuperado el 14 de Octubre de 2011, de Cooperativismo: http://www.liga.coop/institucional?id=34

Maestros del Web. (2009). *Que son bases de datos*. Recuperado el 15 de Octubre de 2011, de http://www.maestrosdelweb.com/principiantes/%C2%BFque-son-las-bases-de-datos/

Moisés, A. C. (26 de Agosto de 2010). *Uso básico de XCode & Interface Builder en MAC OS X usando la librería Cocoa de Objective C.* Recuperado el 16 de Septiembre de 2011, de http://uncomp.uwe.ac.uk/genaro/Papers/Veranos_McIntosh_files/manualOSX.pdf

Mundomacpunta. (s.f.). *Noticias*. Recuperado el 22 de Noviembre de 2011, de ¿Qué es el sistema operativo Mac OS X Snow Leopard:

http://www.mundomacpunta.com/index.php/noticias/index.php?page=shop.product_details&flypage=flypage.tpl&product_id=328&category_id=44&option=com_virtuemart&Itemid=113

Niño, J. E. (2010). *Programación en C.* Recuperado el 24 de Octubre de 2011, de http://es.scribd.com/doc/246059292/Programacion-en-C-Ernesto-Prada#scribd

Personales.Ya. (s.f.). *¿Qué es COCOA?* Recuperado el 24 de Noviembre de 2011, de http://personales.ya.com/macprog/IntroProgMacOSX.pdf

Personales.Ya. (s.f.). *¿Qué es la herramienta Interface Builder?* Recuperado el 24 de Noviembre de 2011, de http://personales.ya.com/macprog/IntroProgMacOSX.pdf

Picuto, F. R. (2007). *"Hacking y seguridad en Internet".* RA-MA. Recuperado el 16 de Noviembre de 2011

Schach, S. R. (2006). *"Ingeniería de Software Clásica y orientado a objetos".* Mcgraw-Hill / Interamericana de Mexico. Recuperado el 16 de Noviembre de 2011

Slideshare. (2010). *Privacidad y Seguridad*. Recuperado el 15 de Octubre de 2011, de Firewall: http://www.slideshare.net/alu4albertde/privacidad-y-seguridad-informtica

Soft-Tecnic. (s.f.). *Virtual Box*. Recuperado el 25 de Noviembre de 2011, de Ventajas y Desventajas de Virtual Box: http://soft-tecnic.net/foro/index.php?topic=193.0

Tabor, R. (2002). *"Servicios Web XML de Microsoft.net".* Pearson Educación, 2002. Recuperado el 16 de Noviembre de 2011

Torres, D. (2010).

Trent, M., & McCormack, D. (2010). *"Beginning MAC OS X Snow Leopard Programming". Programación para principiantes en el sistema operativo Mac OS X Snow Leopard.* Recuperado el 7 de Noviembre de 2011, de http://pdf.th7.cn/down/files/1312/Beginning%20Mac%20OS%20X%20Snow%20Leopard%20Programming.pdf?yundunkey=14dba16142a68c0f9dcb6ab34e9dd34871430325286_422058442

Uribe, J. C. (2010). *Guía de desarrollo de aplicaciones móviles para iPhone / iPad.* Recuperado el 27 de Septiembre de 2011, de

http://www.maestrosdelweb.com/images/2010/12/maestrosdelweb-guia-iphone.pdf

Valdés, D. P. (2007). *¿Qué son las bases de datos?* Recuperado el 9 de Noviembre de 2011, de http://www.maestrosdelweb.com/que-son-las-bases-de-datos/

Virtualbox. (s.f.). *Virtual Box.es*. Recuperado el 13 de Octubre de 2011, de http://virtualbox.es

Wikipedia. (2015). *Banca Electrónica*. Obtenido de Banca Virtual: http://es.wikipedia.org/wiki/Banca_electr%C3%B3nica

Wikipedia. (s.f.). *Iphone*. Recuperado el 25 de Noviembre de 2011, de Qué es el dispositivo de comunicaciones Iphone: http://es.wikipedia.org/wiki/IPhone

Wikipedia. (s.f.). *IPhone*. Recuperado el 25 de Noviembre de 2011, de Conectividad de Internet y Accesibilidad de un IPhone: http://es.wikipedia.org/wiki/IPhone

4.4 GLOSARIO DE TERMINOS.

Aplicación: En informática, una aplicación es un tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajo.

Blade: Un servidor blade es un tipo de computadora para los centros de proceso de datos específicamente diseñada para aprovechar el espacio, reducir el consumo y simplificar su explotación.

Compilación: En informática, la compilación en tiempo de ejecución (también conocida por sus siglas inglesas, JIT, just-in-time), también conocida como traducción dinámica, es una técnica para mejorar el rendimiento de sistemas de programación que compilan a bytecode, consistente en traducir el bytecode a código máquina nativo en tiempo de ejecución.

Cuenta: La cuenta es el elemento básico y central en la contabilidad y en los servicios de pagos. La cuenta contable es la representación valorada en unidades monetarias, de cada uno de los elementos que componen el patrimonio de una empresa (bienes, derechos y obligaciones) y del resultado de la misma (ingresos y gastos), permite el seguimiento de la evolución de los elementos en el tiempo.

Diseño: El diseño se define como el proceso previo de configuración mental, "pre-figuración", en la búsqueda de una solución en cualquier campo. Utilizado habitualmente en el contexto de la industria, ingeniería, arquitectura, comunicación y otras disciplinas creativas.

Fecha: Del latín facta, es una indicación del tiempo en que ocurre o se hace algo.

Formulario: Un formulario es generalmente una hoja de papel en la que disponemos de lugares en blanco para introducir la información, con un campo separado para cada dato. Un formulario es un documento con espacios (campos) en donde se pueden escribir o seleccionar opciones.

ID: Es su cédula de identidad, documento de identidad, identificación.

Implementación: Una implementación o implantación es la realización de una aplicación, o la ejecución de un plan, idea, modelo científico, diseño, especificación, estándar, algoritmo o política.

Interfaz: Es la conexión entre dos ordenadores o máquinas de cualquier tipo dando una comunicación entre distintos niveles.

Login: Es el término que se usa en computación para referirse al ingreso a las cuentas de usuario, a los sistemas o servicios.

Máquina Virtual: Es que emula a una computadora y puede ejecutar programas como si fuese una computadora real. Este software en un principio fue definido como "un duplicado eficiente y aislado de una máquina física". La acepción del término actualmente incluye a máquinas virtuales que no tienen ninguna equivalencia directa con ningún hardware real.

Monto: El valor en unidades de una moneda o la cantidad de una transacción.

Móvil: La telefonía móvil, también llamada telefonía celular, básicamente está formada por dos grandes partes: una red de comunicaciones (o red de telefonía móvil) y los terminales (o teléfonos móviles) que permiten el acceso a dicha red.

Movimiento: Son movimientos bancarios a todas las acciones que se realicen en la cuenta, tales como depósitos, retiros, entre otros.

Password: Una contraseña o password es una serie secreta de caracteres que permite a un usuario tener acceso a un archivo, a un ordenador, o a un programa. En sistemas multiusos, cada usuario debe incorporar su contraseña antes de que el ordenador responda a los comandos.
Las contraseñas ayudan a asegurar que los usuarios desautorizados no tengan acceso al ordenador. Además, los ficheros de datos y los programas pueden también requerir una contraseña.

Préstamo: Un préstamo bancario consiste en pedir dinero a un banco para financiar algún proyecto personal o negocio, o para saldar deudas, dinero que luego deberá pagarse en cuotas mensuales con la correspondiente tasa de interés.

Programación: La programación es el proceso de diseñar, codificar, depurar y mantener el código fuente de programas computacionales. El código fuente es escrito en un lenguaje de programación. El propósito de la programación es crear programas que exhiban un comportamiento deseado.

Saldo: El saldo es el estado de una cuenta corriente, en cuanto al dinero que hay en ella.

SDK: Un SDK es un conjunto de herramientas y programas de desarrollo que permite al programador crear aplicaciones para un determinado paquete de software, estructura de software, plataforma de hardware, sistema de computadora, consulta de videojuego, sistema operativo o similar.

Servidor: En informática, un servidor es una computadora que, formando parte de una red, provee servicios a otras computadoras denominadas clientes**.**

TabBar: Cuando queremos crear una aplicación que contenga diferentes funcionalidades relacionadas con el uso de la aplicación, pero independientes entre si, como mostrar los datos de diferente manera o diferentes tareas, querremos usar una TabBar en nuestra aplicación.

La TabBar nos dará acceso a esas diferentes tareas o vistas, separándolas dentro de nuestra aplicación como módulos independientes, además, sitúa al usuario dentro de la aplicación indicándole lo que esta viendo en ese momento.

Torrent: Torrent es la extensión de archivos utilizados por un protocolo de transferencia del tipo P2P (Peer to Peer).
Este término se utiliza en la virtualización y se trata del archivo del sistema operativo Mac Snow Leopard; como se trata de un archivo grande, el formato torrent le permite descargarlo en un periodo determinado de tiempo, dando la comodidad de realizar pausas respectivas.

Virtualización: Virtualización es la técnica empleada sobre las características físicas de ALGÚNos recursos computacionales, para ocultarlas de otros sistemas, aplicaciones o usuarios que interactúen con ellos. Esto implica hacer que un recurso físico, como un servidor, un sistema operativo o un dispositivo de almacenamiento, aparezca como si fuera varios recursos lógicos a la vez, o que varios recursos físicos, como servidores o dispositivos de almacenamiento, aparezcan como un único recurso lógico.

Wireless: Wireless (inalámbrico o sin cables) es un término usado para describir las telecomunicaciones en las cuales las ondas electromagnéticas (en vez de cables) llevan la señal sobre parte o toda la trayectoria de la comunicación.

4.5 ANEXOS.

4.5.1 MANUALES.

4.5.1.1 MANUAL INSTALACIÓN DE APLICACIÓN (A).

1.- Tener el archivo de instalación SFPUCE móvil.

2.- Tener instalado en un ordenador el software **ITUNES.**

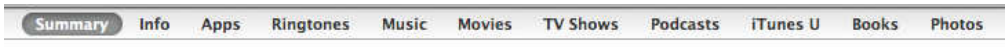
3.- Sincronizar su dispositivo móvil Iphone con el software Itunes de la siguiente manera:

3.1.- Abra iTunes.

3.2.- Utilice el cable USB para conectar el dispositivo iOS al ordenador y selecciónalo en la sección Dispositivos, a la izquierda de la ventana de iTunes.



Es posible que no se muestren algunas pestañas si no se tiene contenidos correspondientes a las mismas en su biblioteca. Por ejemplo, si no se tiene ningún podcast en su biblioteca, la pestaña correspondiente Podcast no se mostrará.



3.3.- Haga clic en Aplicar, en la esquina inferior derecha de la pantalla, para sincronizar.

4.- Listo, una vez que se tenga sincronizado su Iphone a su Itunes transfiere el archivo SFPUCE móvil e inícielo normalmente.

5.- Eso es todo comience a utilizar la aplicación SFPUCE móvil en su dispositivo Iphone para realizar consultas.

4.5.1.2 MANUAL INSTALACIÓN DE APLICACIÓN (B).

1. Abra la aplicación Appstore. Para ello, pulse el ícono que representa la aplicación.



2. Se abrirá la tienda de aplicaciones de Apple. Allí, busca la aplicación con el nombre SFPUCEmovil representada por el siguiente ícono.



3. Ingrese tu ID de Apple y contraseña, y la aplicación se descargará en tu móvil.

4.5.1.3 MANUAL DE USUARIO.

USO DE LA APLICACIÓN SFPUCE móvil.

1.- Ingrese a la aplicación SFPUCE móvil identificada por el ícono siguiente en su teléfono móvil Iphone:



Figura #4.2 Ícono Aplicación SFPUCE móvil

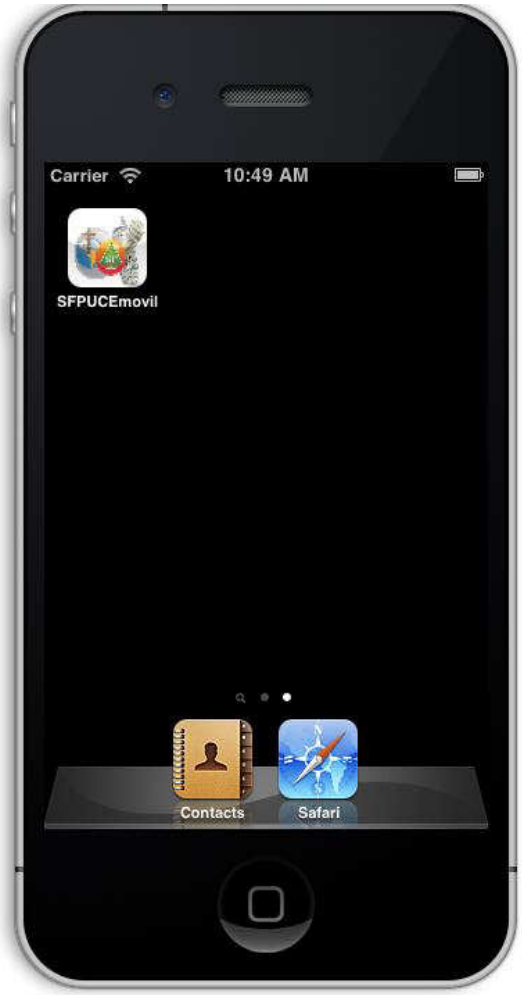


Figura #4.3 Menú de aplicaciones Iphone.

AUTENTIFICACIÓN DE USUARIO.

2.- En la pantalla de autentificación de usuario ingrese su usuario y su contraseña, si no los obtiene contáctese con servicio y atención al cliente de la Institución.

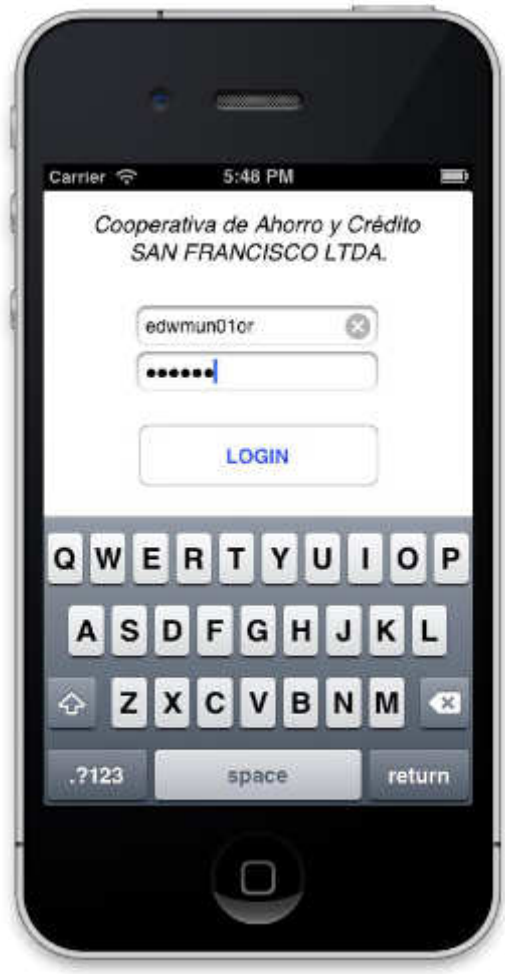


Figura #4.4 Ingreso de usuario y contraseña en pantalla principal de app.

Observación: Si no puede ingresar a la aplicación asegúrese en los siguientes puntos:

- Debe ingresar correctamente su usuario y contraseña.
- Debe tener su cuenta de ahorros activa.
- La conexión a internet de su teléfono debe estar inactiva.
- Su usuario no debe encontrarse bloqueado.
- Ningun campo de texto (usuario y/o contraseña), no debe estar en blanco.

Cuando usted haya ingresado correctamente sus datos puede realizar las diferentes consultas.

CONSULTA DE SALDOS.

3.- Para consultar su saldo en su libreta de ahorros ingrese a la **OPCION SALDO** en el menú principal de la aplicación SFPUCE móvil como se muestra a continuación:



Figura #4.5 Consulta de saldo en cuenta de ahorros en SFPUCEmovil.

La aplicación automáticamente realizará la consulta de saldos, y le mostrará en pantalla lo que usted dispone actualmente en su cuenta de ahorros.

Observación:

- Si la aplicación no le presenta su saldo disponible, por favor asegúrese que su cuenta de ahorros se encuentre activa.

CONSULTA DE PRÉSTAMOS VIGENTES.

5.- Para consultar si su número de cuenta tiene aliado uno o varios préstamos vigentes, además de conocer el saldo de cada uno, ingrese a la **OPCION PRESTAMOS** en el menú principal de la aplicación SFPUCE móvil como se muestra a continuación:



Figura #4.6 Consulta de préstamos y saldo préstamo en SFPUCE móvil.

La aplicación le brindará en botones todos los créditos que se encuentran activos, para que usted presione el préstamo que requiere de información y consulte el saldo del mismo.

CONSULTA DE FECHA DE ULTIMA CUOTA PAGADA Y EL VALOR.

4.- En el caso de que usted tenga créditos activos (vigentes) aliados a su cuenta, puede consultar la fecha de su última cuota pagada, además de conocer el valor que usted canceló de ésta. Para llevar a cabo este proceso ingrese a la **OPCION FECHA PAGO CUOTA PRESTAMO** en el menú principal de la aplicación SFPUCE móvil como se muestra a continuación:



Figura #4.7 Consulta de fecha de última cuota pagada y valor en SFPUCE móvil.

La aplicación le brindará en botones todos los créditos que se encuentran activos, para que usted presione el préstamo que requiere de información y consulte la fecha de la última cuota pagada además de cuanto pagó de la misma.

CONSULTA DE VENCIMIENTOS DE CUOTA (FECHA DE PROXIMA CUOTA A CANCELAR).

5.- En el caso de que usted tenga créditos activos (vigentes) aliados a su cuenta, ud puede consultar la fecha de la próxima cuota que le corresponde, además de conocer el valor que debe cancelar. Para llevar a cabo este proceso ingrese a la **OPCION FECHA VENCIMIENTOS PRESTAMOS** en el menú principal de la aplicación SFPUCE móvil como se muestra a continuación:



Figura #4.8 Consulta fecha de su próxima cuota y valor en SFPUCEmovil.

CONSULTA DE ÚLTIMOS MOVIMIENTOS.

6.- Si usted desea conocer cuáles fueron los 5 últimos movimientos realizados en su cuenta de ahorros debe ingresar a la opción **MOVIMIENTOS CUENTAS** en el menú principal de SFPUCEmóvil como se muestra a continuación:



Figura #4.9 Consulta de 5 últimos movimientos en la aplicación SFPUCE móvil.

La aplicación le ofrece una tabulación detallada de sus 5 últimos movimientos. Si usted no comprende el detalle de cada uno, por favor presione el botón “Ayuda detalle”, el cual le ofrecerá una descripción de cada abreviación.

TRANSFERENCIA DE DINERO ENTRE CUENTAS.

7**.-** SFPUCEmóvil, le permite realizar transferencias de dinero entre cuentas que sean pertenecientes a la Cooperativa de Ahorro y Crédito San Francisco, por lo que si usted desea enviar dinero desde su cuenta a otra, debe ingresar a la opción **transferencia de DINERO** en el menú principal de la aplicación SFPUCEmóvil como se muestra a continuación:



Figura #4.10 Transacción de dinero en la aplicación SFPUCE móvil.

Ingrese su **cuenta destino y monto** (menor y hasta 100 dólares c/u), y a continuación presione el botón TRANSFERIR.

Una vez que usted le indica a la aplicación que realice la transferencia, ésta automáticamente verificará internamente, si usted no ha sobrepasado las 3 trasacciones máximas por día que ésta por motivos de seguridad limita.

Si usted no ha pasado de este límite de movimientos diarios que hayan sido por transferencia, la aplicación inmediatamente le solicitará una confirmación para que usted pueda revisar los datos ingresados y definitivamente confirme el proceso de la siguiente manera:



Figura #4.11 Alerta antes de realizar transferencia en SFPUCE móvil.

Si esta seguro, debe presionar el botón SI, para que la aplicación proceda con la transferencia, recibiendo la siguiente confirmación:



Figura #4.12 Transferencia de dinero satisfactoria en SFPUCE móvil.

Después de haber efectuado correctamente la transferencia, por favor revise su dirección de correo electrónico (dirección registrada con su cuenta), y encontrará un informe enviado por la aplicación en el cual detalla el proceso efectuado por transferencia, señalando los datos correspondientes que formaron parte para la ejecución de proceso (cuenta destino y valor) reforzando la confianza y seguridad.

Ejemplo informe email tras transferencia realizada con éxito:

(En este ejemplo el informe se lo recibió en Outlook.com).



Figura # 4.14 Ejemplo cabecera de email, tras transferencia

Cuerpo del correo:



Figura # 4.15 Ejemplo cuerpo de email, tras transferencia

Adicional si desea otro método para verificar que su transferencia fue realizada con éxito por favor diríjase a la opción **SALDOS** de su aplicación y verifique su nuevo saldo en cuenta de ahorros; ahí podrá notar el debito realizado por este proceso**.**

La última verificación que puede llevar a cabo es ingresar a la opción **MOVIMIENTOS CUENTA** y consulte sus 5 últimos moviemientos. En el caso que la transferencia fue correcta constará un movimiento con la descripción (NDT), la cual significa nota de débito por transferencia.

ACERCA DE.

8.- Si usted desea conocer quien es el creador de la aplicación SFPUCEmóvil y los derechos de autor por favor ingrese a la opción **ACERCA DE** del menú principal como se muestra a continuación y seleccione el botón ¨Información de Desarrollador¨.



Figura #4.16 Acerca de en la aplicación SFPUCE móvil.

SALIR.

8.- Si usted ya obtuvo toda su información confidencial bancaria que mantiene con la Cooperativa de Ahorro y Crédito San Francisco Ltda, y desea salir de aplicación sin dejar rastros de su usuario ingrese a la opción **SALIR** del menú principal como se muestra a continuación y seleccione el botón salir.



Figura #4.17 Salir de la aplicación SFPUCE móvil.

Una vez que usted haya concluido sus consultas y procesos, y haya decidido salir de la aplicación, por favor tome muy en cuenta los consejos y recomendaciones que SFPUCEmóvil le ofrece.

4.5.2 ANEXO A ENCUESTA BANCA MOVIL





ENCUESTA DE BANCA MOVIL PARA LOS SOCIOS DE LA COOPERATIVA SAN FRANCISCO

Objetivo:

Determinar las preferencias de los socios en cuanto a modelo de teléfono celular y servicios financieros de banca móvil requeridos.

ESTIMADO SOCIO:

Indique con una X su respuesta:

1.- ¿CONOCE USTED ACERCA DE LAS VENTAJAS Y SERVICIOS DE BANCA MOVIL?

SI ( ) **NO ( )**

2.- ¿LA COOPERATIVA FINANCIERA SAN FRANCISCO OFRECE ALGÚN SERVICIO DE CONSULTA DE DATOS A TRAVES DE TELEFONIA CELULAR?

SI ( ) **NO ( )**

3.- ¿LE GUSTARIA CONTAR CON UN NUEVO SERVICIO, DENTRO DE LA COOPERATIVA, QUE LE PERMITA CONSULTAR TODA SU RESPECTIVA INFORMACION BANCARIA CONFIDENCIAL A TRAVES DE SU TELEFONO CELULAR?

SI ( ) **NO ( )**

4.- ¿CONTANDO CON ESTE SERVICIO EN QUE TELEFONO CELULAR LE GUSTARIA IMPLEMENTARLO?

BLACKBERRY ( )

IPHONE ( )

OTROS ( ) ……

5.- ¿QUE TIPO DE CONSULTAS LE GUSTARIA REALIZAR A TRAVES DE SU TELEFONO CELULAR?

( ) Consulta de Saldos

( ) Consulta de Préstamos

( ) Fecha de pago de cuota

( ) Fecha de vencimiento de cuota

( ) Consulta de últimos movimientos

( ) Transferencia entre cuentas de la misma institución

GRACIAS POR SU PARTICIPACION ☺

4.5.3 ANEXO B VIRTUALIZACIÓN.

INSTALACIÓN DE VIRTUAL BOX.

Ejecutar el instalador de VirtualBox y dar clic en siguiente (NEXT).



Figura # 4.18 Bienvenida de Instalación de VirtualBox.

Seleccionar la instalación personalizada de Virtual Box que esta por defecto y poner siguiente (NEXT).

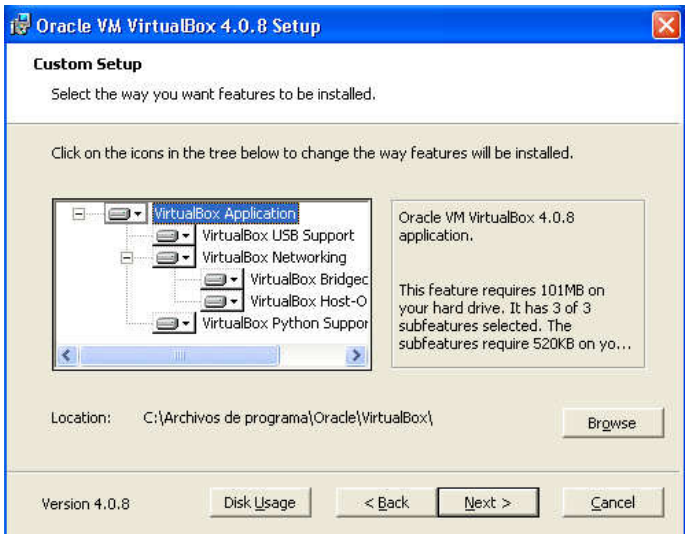


Figura # 4.19 Instalación Personalizada de Virtual Box.

En la siguiente ventana se encuentran seleccionados la creación de accesos directos a necesidad del usuario, si esta de acuerdo marquelos, si no lo esta puede deseleccionarslos, y en seguida por favor clic en siguiente (NEXT).

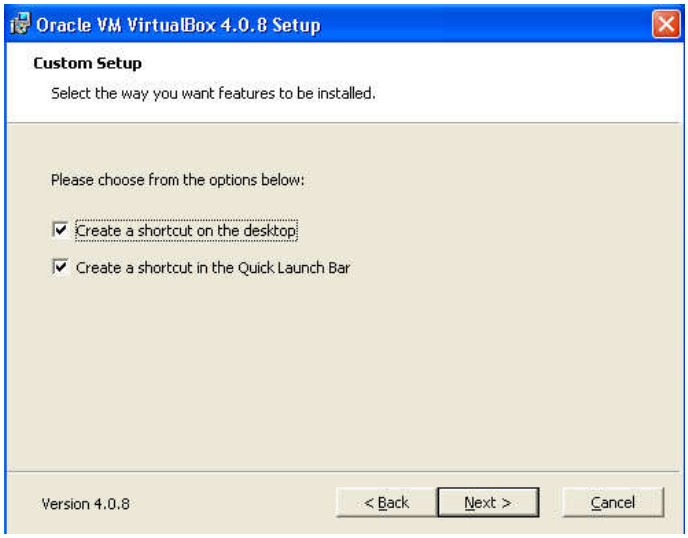


Figura # 4.20 Creación de Accesos Directos de Virtual Box.

Al realizar el siguiente paso para la instalación de VirtualBox, pues el asistente informará la siguiente alerta que informa que se perderá cualquier tipo de conexión de red de la máquina anfitrión, si esta de acuerdo con este paso y perder todo tipo de conexión continúe con la instalación y de un clic en si (YES), caso contrario de un clic en la opción NO para continuar con el trabajo que estaba efectuándose anteriormente en su ordenador.



Figura # 4.21 Mensaje de Alerta en el asistente de Instalación de Virtual Box Pérdida de Conexiones de red de la máquina anfitrión.

Después de haber cumplido con todos los pasos anteriores el asistente de instalación de VirtualBox esta listo para empezar a instalarlo, de acuerdo con esto de un clic en instalar (INSTALL).

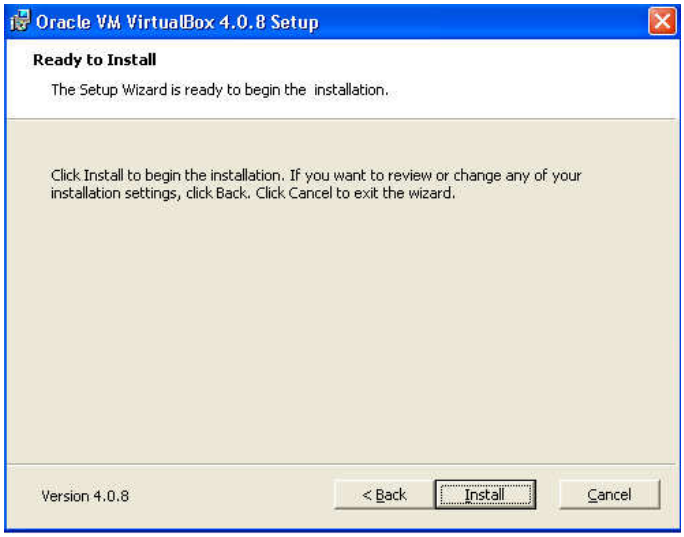


Figura # 4.22 Confirmación del asistente VirtualBox listo para instalarse.

Proceso de Instalación de la Herramienta VirtualBox.

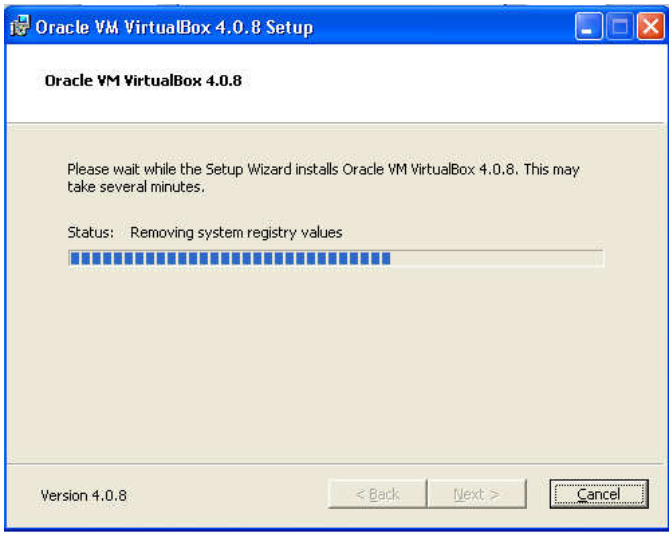


Figura # 4.23 Proceso de Instalación de VirtualBox.

Asistente de instalación de VirtualBox terminado, la herramienta de virtualización esta correctamente instalada y lista para usarse, de un clic en finalizar (FINISH).

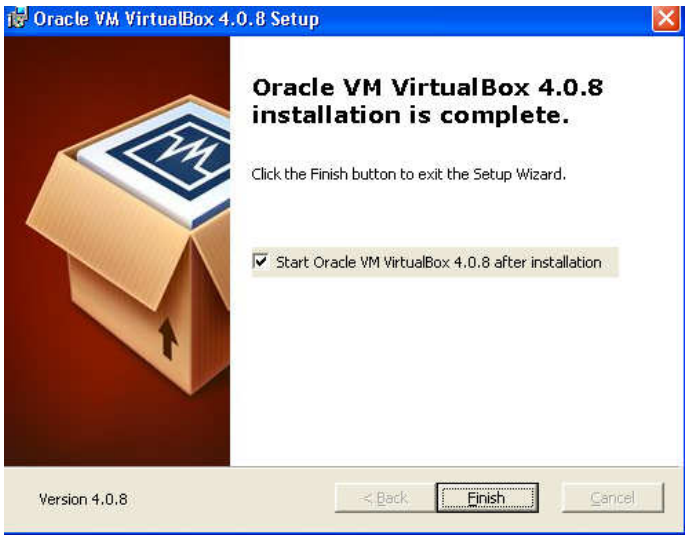


Figura # 4.24 Asistente de Instalación de VirtualBox terminado Herramienta de Virtualización lista para usarse.

INSTALACIÓN DE MAC OS X SNOW LEOPARD 10.6.6 EN MAQUINA VIRTUAL VIRTUALBOX.

Instalación del Sistema Operativo Mac Snow Leopard 10.6.6 en la máquina virtual con el nombre SF MOVIL, carga del archivo.

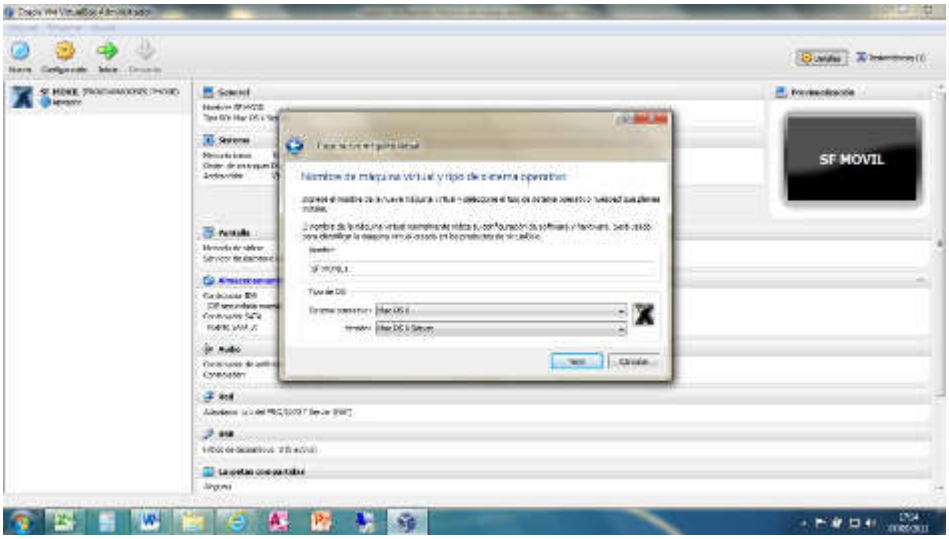


Figura # 4.25 Asistente de Instalación de Mac Snow Leopard.

Inicio de Instalación del sistema operativo Mac Snow Leopard.

El inicio de instalación le puede tomar varios minutos.

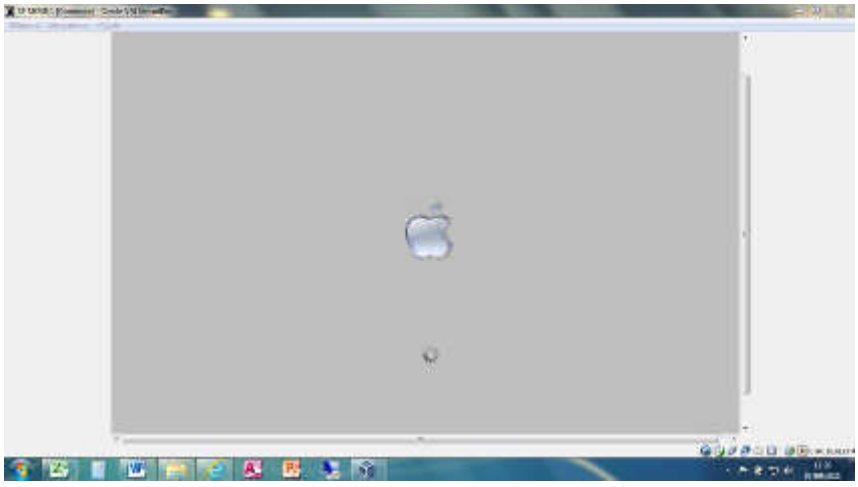


Figura # 4.26 Inicio de Instalación de Mac Snow Leopard.

Inmediatamente después si la instalación se la lleva a cabo correctamente usted tendrá la primera pantalla de bienvenida indicando en que idioma desea trabajar con el sistema operativo, en este caso seleccionamos el idioma Español como idioma principal y damos un clic en la flecha azul que significa siguiente.

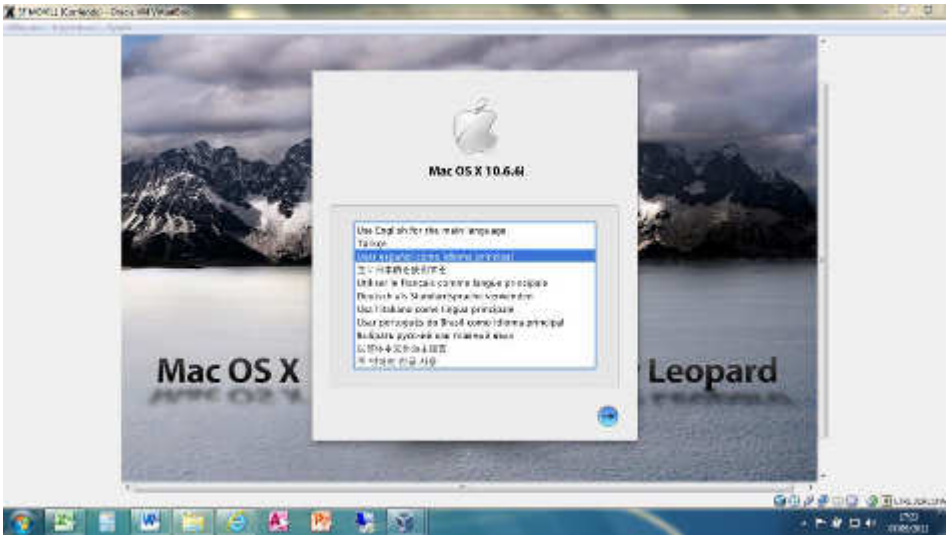


Figura # 4.27 Configuración de Idioma en la Instalación de Mac Snow Leopard.

A continuación tenemos el proceso de preparación de instalación, dejamos que el asistente se ejecute sin interrupciones para no tener errores posteriores. Este paso puede tardar varios minutos.

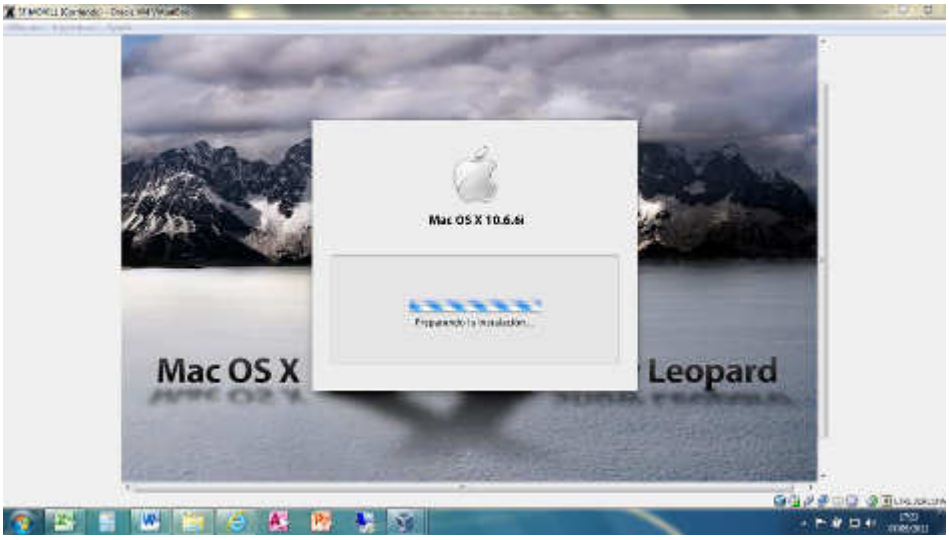


Figura # 4.28 Preparación de Instalación de Mac Snow Leopard.

En esta ventana del asistente nos solicita la aprobación de configuración de la instalación, lo cual es necesaria porque de aquí seguiremos los pasos para establecer y poner en funcionamiento nuestro disco duro del sistema operativo. De esta manera damos clic en la fecha hacia la derecha que significa siguiente.



Figura # 4.29 Inicio de Configuración de Mac Snow Leopard.

En la ventana de bienvenida del SO (sistema operativo), nos requiere términos de aceptación del contrato de licencia, por consiguiente damos un clic en acepto para pasar a la siguiente ventana del asistente.



Figura # 4.30 Términos de contrato de Licencia para Mac Snow Leopard.

Para que nuestra maquina virtual se lleve a cabo de la mejor manera necesitamos configurar un disco duro, en cual en la pantalla VBOX HARDWAREDISK media, en la pestaña borrar, en el campo nombre ingresamos el nombre de cómo se llamara nuestro disco duro, en este caso lo hemos denominado DISCO SF MOVIL 1; para continuar damos clic en el botón borrar y regresamos a la ventana anterior.

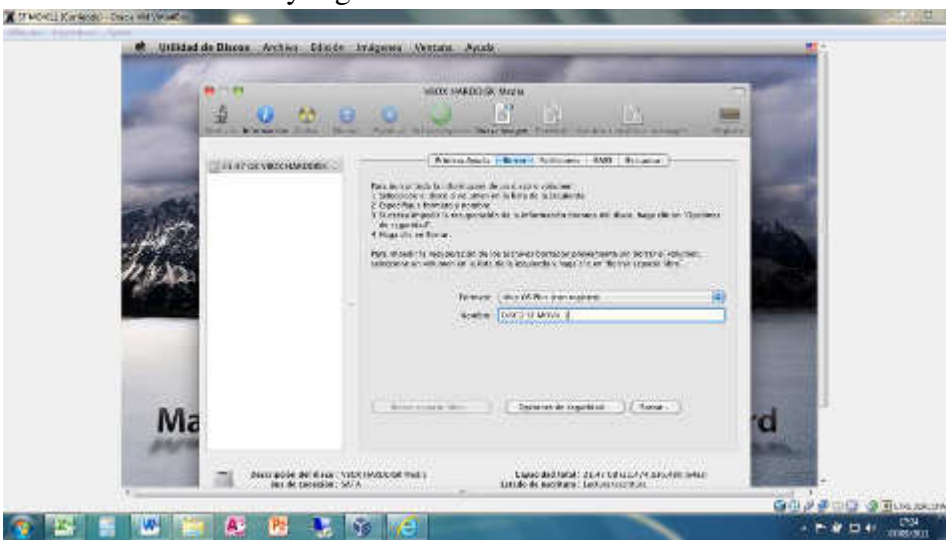


Figura # 4.31 Asistente de Creación de Imagen de Disco de Mac Snow Leopard.

Como podemos observar tenemos listo y disponible la imagen de disco duro en nuestro asistente de configuración, con el cual podemos trabajar para el correcto almacenamiento de datos en nuestra maquina virtual. Seleccionamos el disco duro creado y damos clic en el botón instalar.



Figura # 4.32 Selección de Imagen de Disco para instalación de Mac Snow Leopard.

Inicio de instalación de nuestro sistema operativo en el nuevo disco duro creado en nuestra maquina virtual.

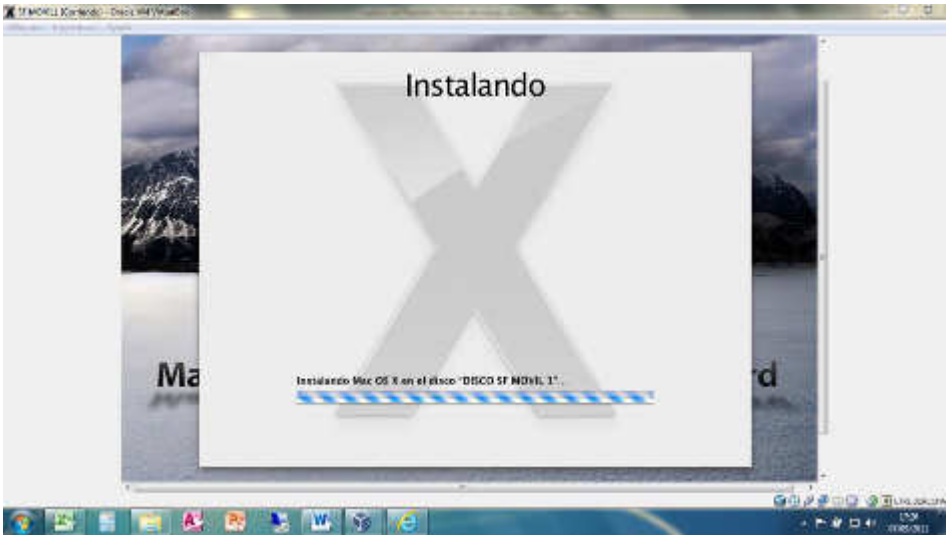


Figura # 4.33 Instalación de Mac Snow Leopard en disco duro.

Seguimiento de Instalación, Este proceso puede tardar varios minutos.

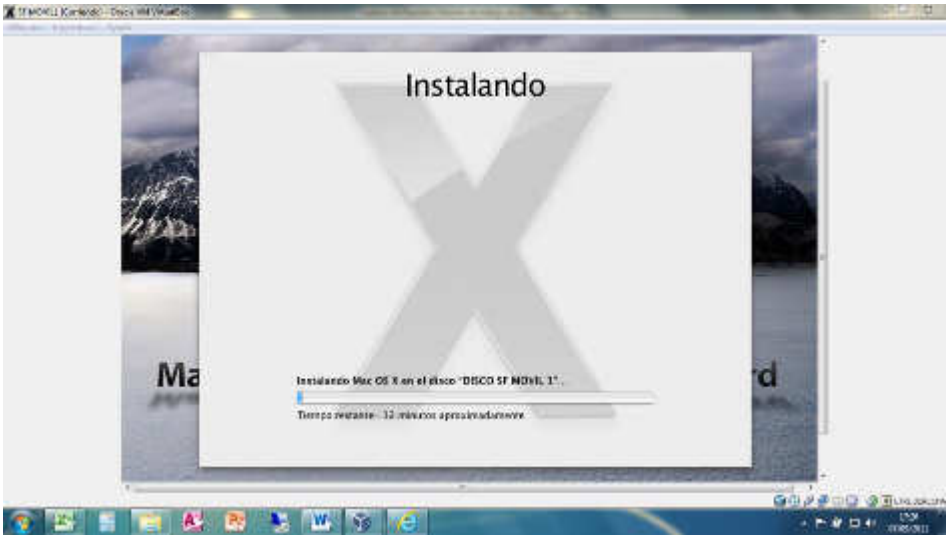


Figura # 4.34 Seguimiento de Instalación de Mac Snow Leopard.

Después de llevarse a cabo toda la instalación, el asistente nos informa que el sistema operativo Mac Snow Leopard se encuentra ya instalado en nuestra maquina virtual de la siguiente manera, a continuación de un clic en continuar.

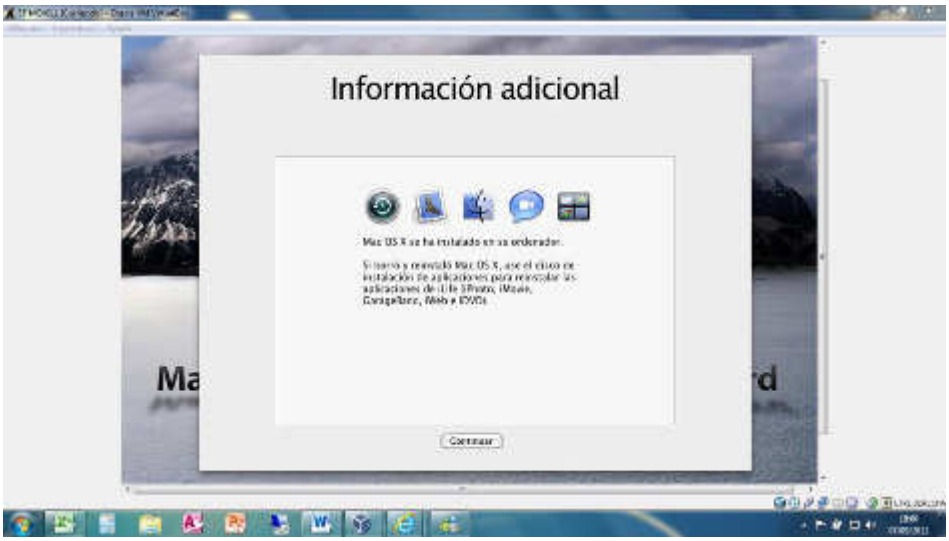


Figura # 4.35 Confirmación de Instalación de Mac Snow Leopard.

Si cumplimos los pasos anteriores a cabalidad y de la mejor manera tendremos una pantalla de confirmación de instalación correcta como la que se presenta a continuación, rectificándonos el disco duro donde se encuentra creado.

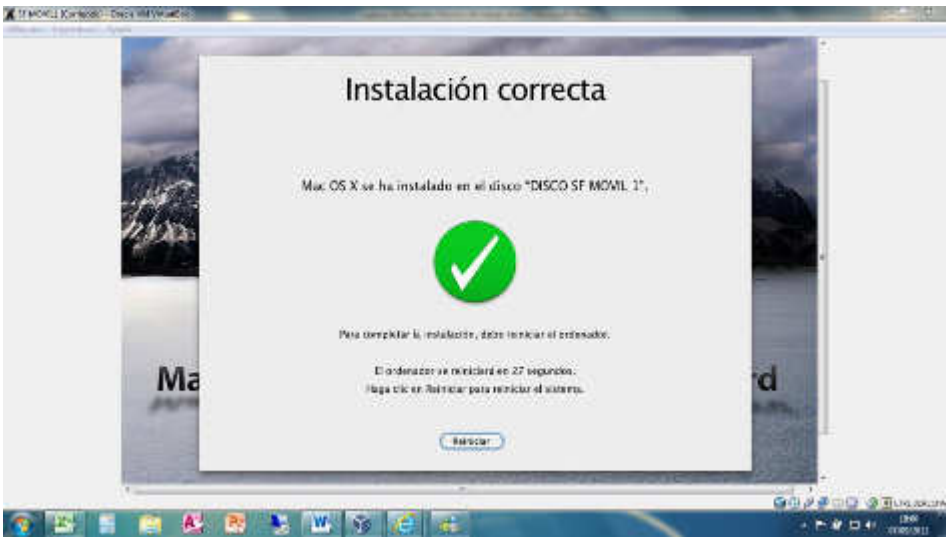


Figura # 4.36 Notificación instalación correcta de Mac Snow Leopard.

Al obtener esta ventana de confirmación, la maquina virtual necesita un reinicio de sistema para iniciar sus operaciones, para seguir con esto damos un clic en el botón reiniciar.

Al momento del Reinicio.

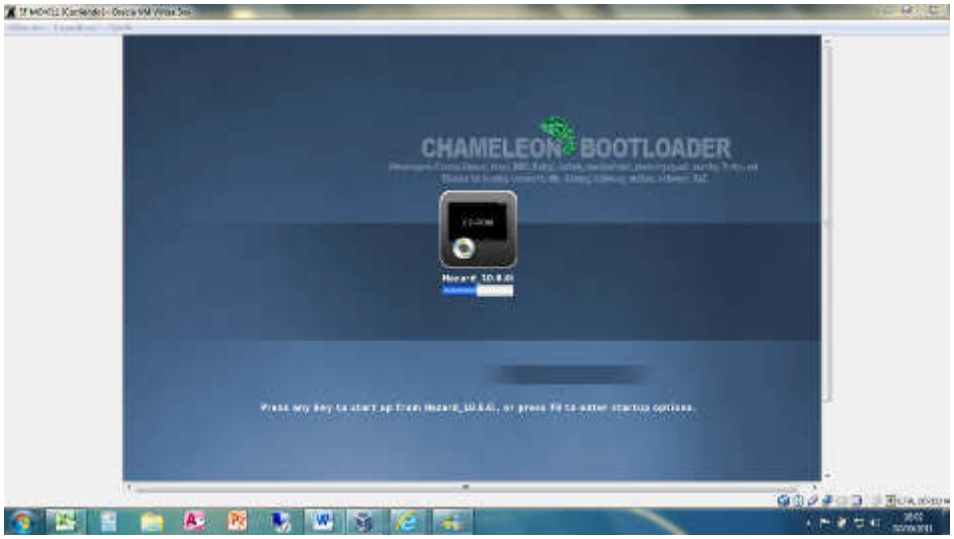


Figura # 4.37 Reinicio después de instalación de Mac Snow Leopard.

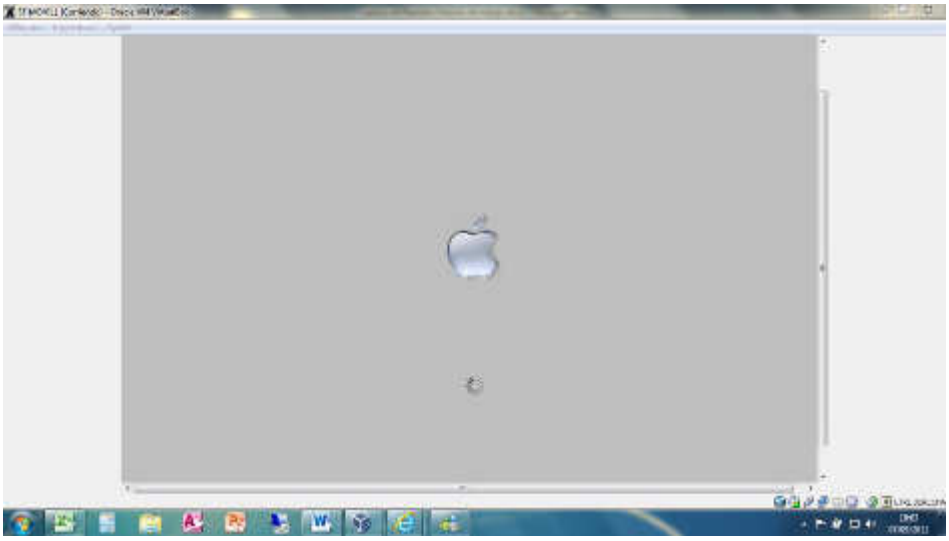


Figura # 4.38 Carga de Sistema Operativo Mac Snow Leopard.

Al arrancar de la manera correcta el sistema, usted se va a encontrar con la disponibilidad de visualizar su disco duro virtual de arranque. Presionar enter.

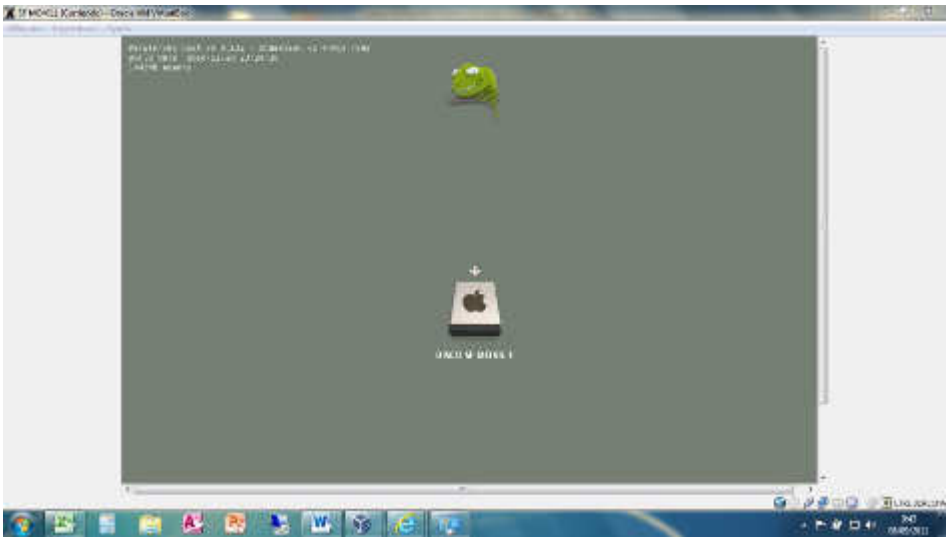


Figura # 4.39 Visualización de Disco Duro Virtual.

Configuración de Sistema Operativo Mac, Selección de Idioma.

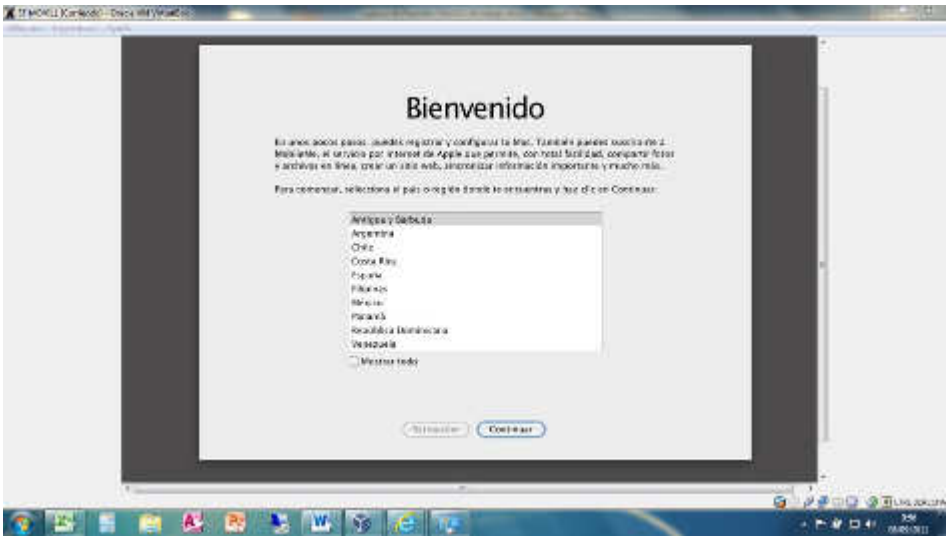


Figura # 4.40 Configuración de Sistema Operativo Selección de Idioma.

Elección de idioma español ISO (Con la investigación se comprobó que es un sistema estandarizado).

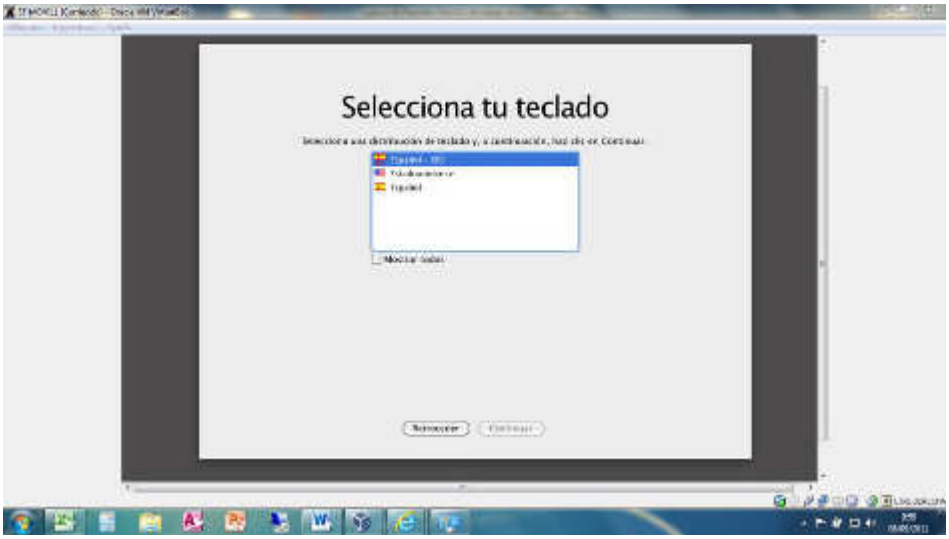


Figura #4.41 Selección de idioma español ISO sistema estandarizado.

Bienvenida de Sistema Operativo Mac OS X.



Figura #4.42 Bienvenida de Sistema Operativo Mac OS x.

Creación de cuenta en sistema operativo Mac Snow Leopard 10.6.6 compatible con SDK Nuevo (Punto importante para trabajo en el sistema).



Figura #4.43 Creación de Cuenta en Sistema Operativo Mac Snow Leopard.

Registro de Datos con cuenta.

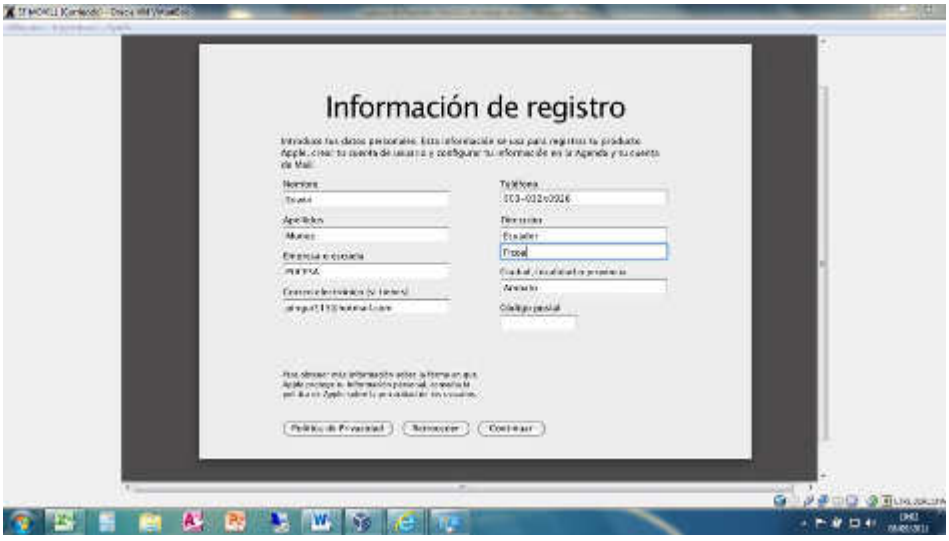


Figura #4.44 Registro de Datos con cuenta.

Preguntas adicionales en inicio de sesión.

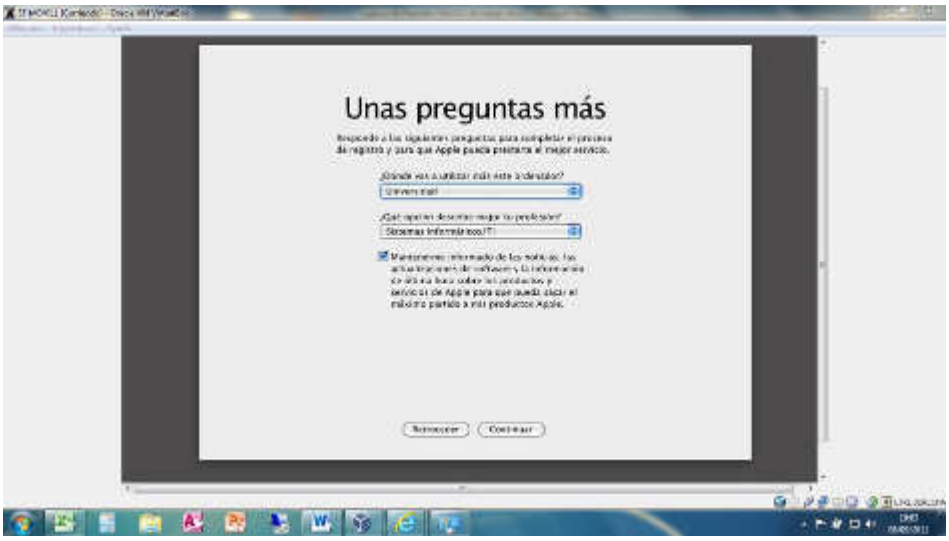


Figura #4.45 Registro de Datos con cuenta en Mac Snow Leopard.

Creación de Cuenta.

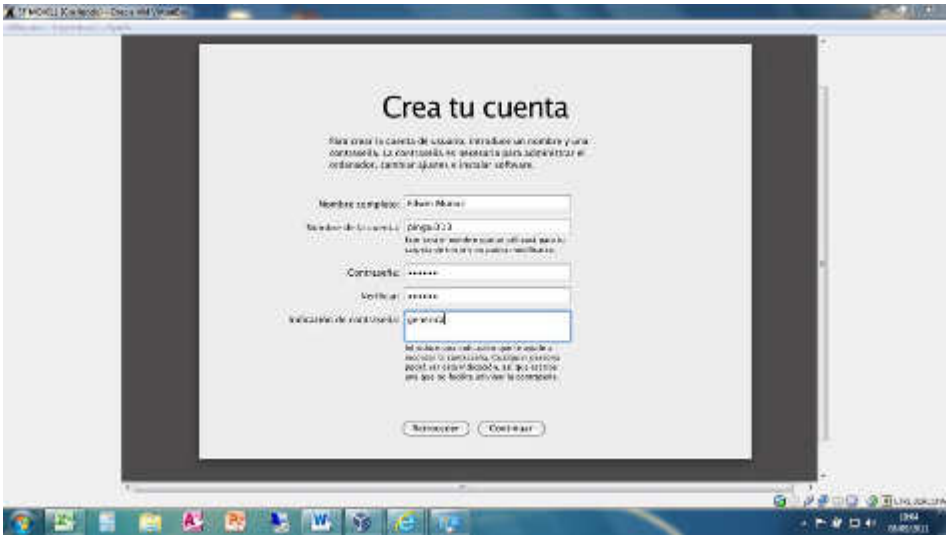


Figura #4.46 Creación de Datos en Mac Snow Lepard.

Culminación de creación de cuenta.



Figura #4.47 Confirmación de registro de Cuenta en Mac Snow Leopard.

Configuración de Teclado de Sistema Operativo Mac Snow Leopard.

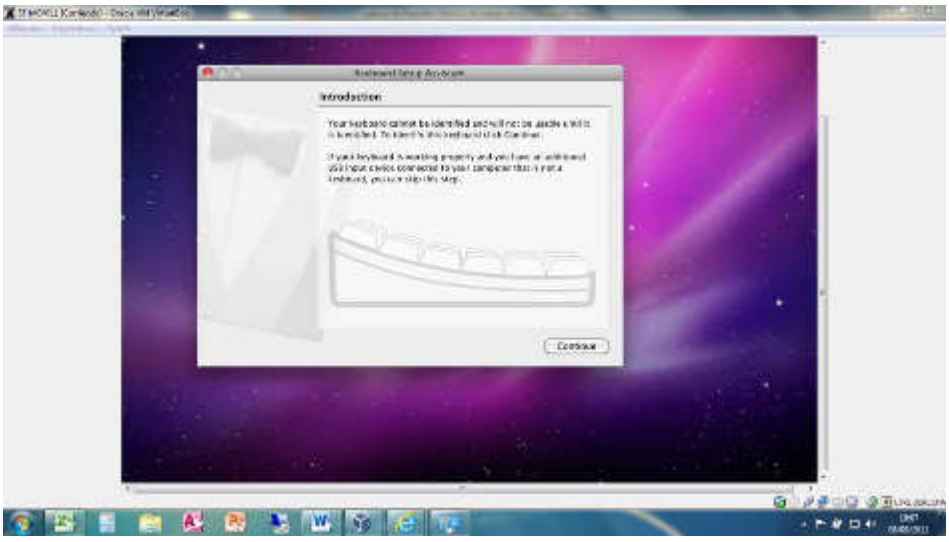


Figura # 4.48 Configuración de Teclado de Sistema Operativo Mac Snow Leopard.

Sistema Mac Snow Leopard 10.6.6 listo para el desarrollo del proyecto.



Figura # 4.49 Entorno listo Mac Snow Leopard 10.6.6.

REGISTRO PARA LA OBTENCION DE HERRAMIENTAS.

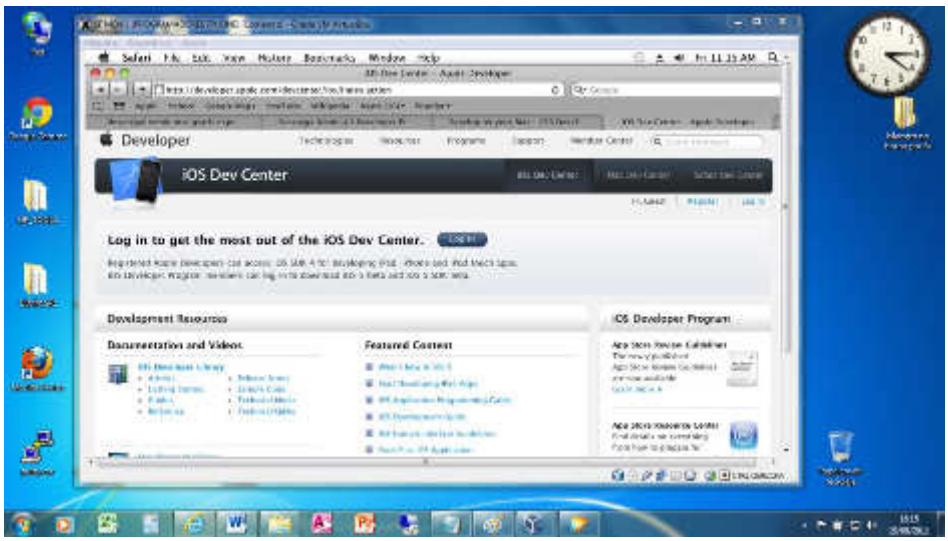


Figura # 4.50 Registro en pagina Ios Dev Center.

Ingreso de datos a la página de Apple Developer.



Figura # 4.51 Registro para descarga de herramientas.

Creación de nuevo ID.

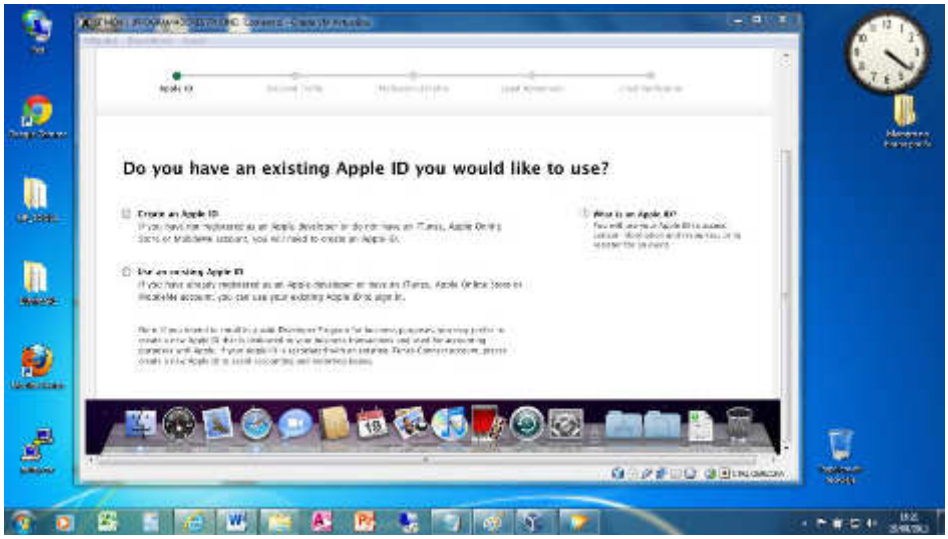


Figura # 4.52 Creación de nuevo ID en página de Aple.

Registro de datos en nuevo ID.

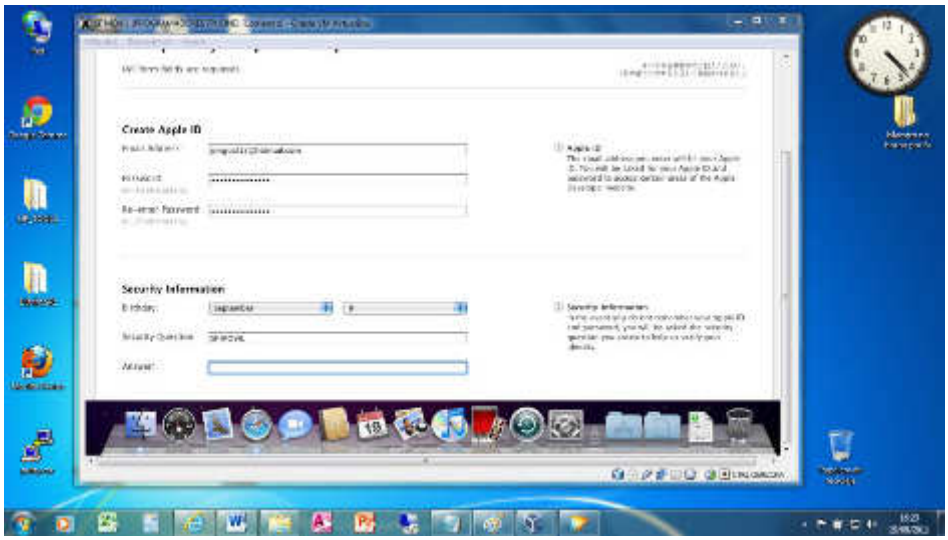


Figura # 4.53 Ingreso de datos en nuevo ID.

Completando prefil profesional.

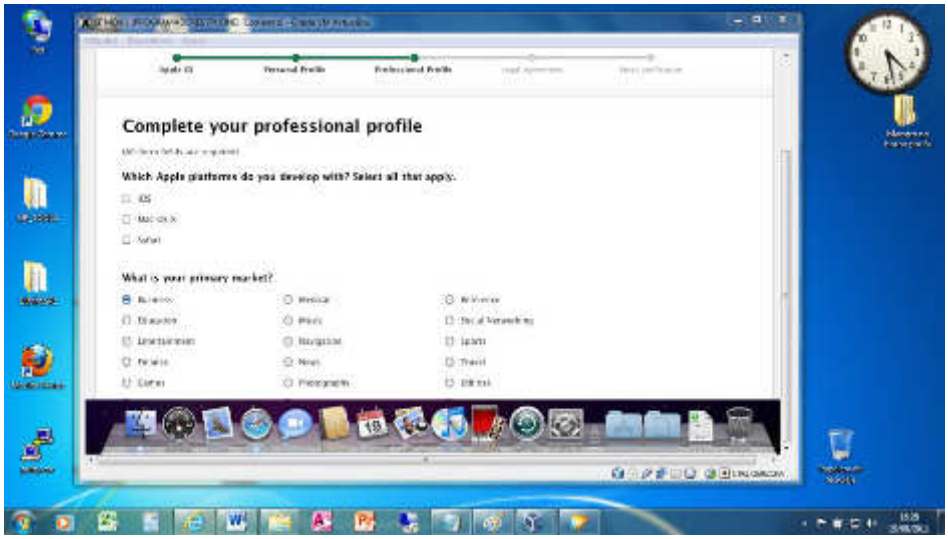


Figura # 4.54 Completando datos en perfin profesional.

Confirmación de Registro.

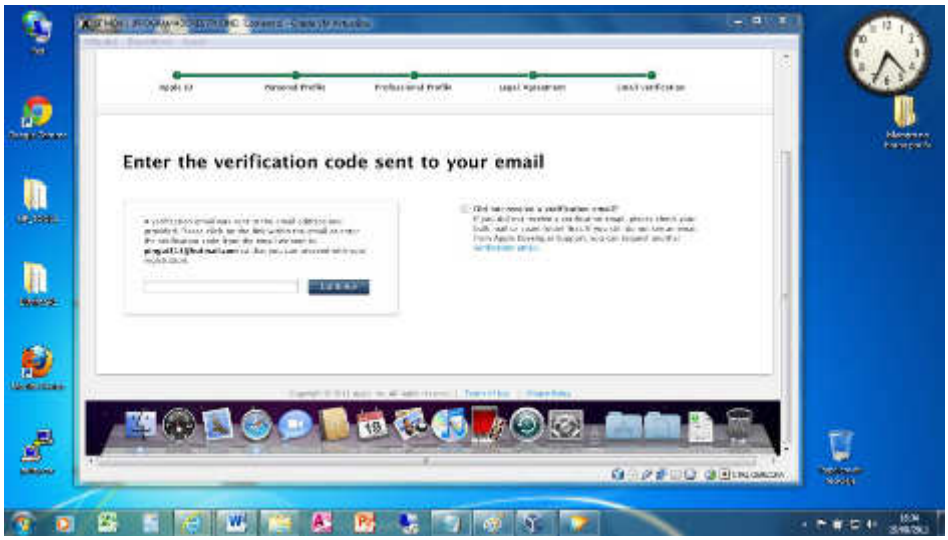


Figura # 4.55 Confirmación de Registro.

Email de confirmación para activación de cuenta.

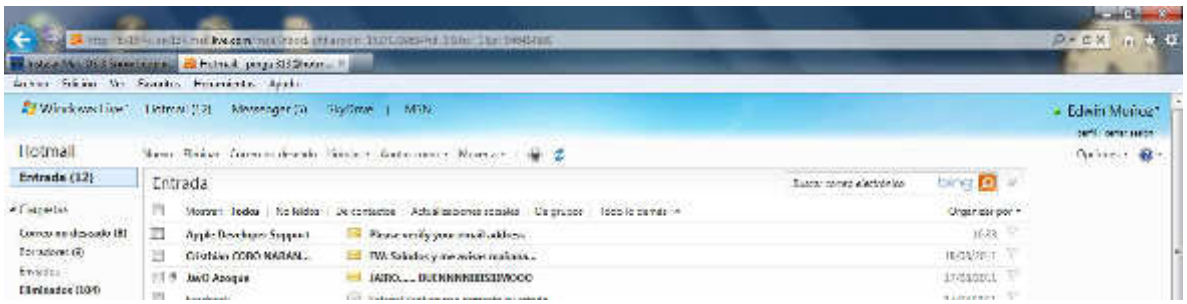


Figura # 4.56 E mail activación.

E MAIL ENVIADO DE MAC.

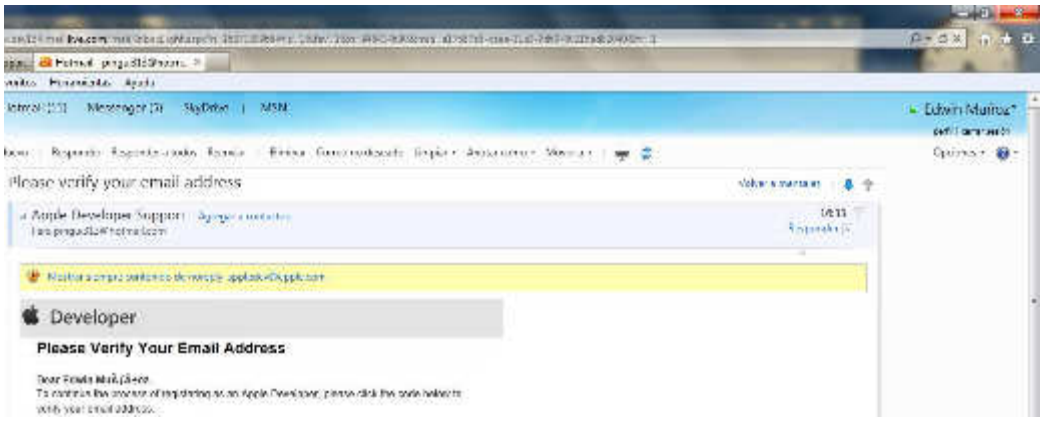


Figura # 4.57 Activación de cuenta.

EDWIN MUÑOZ REGISTRADO EN DESARROLLADORES DE MAC.

La cuenta ha sido creada.

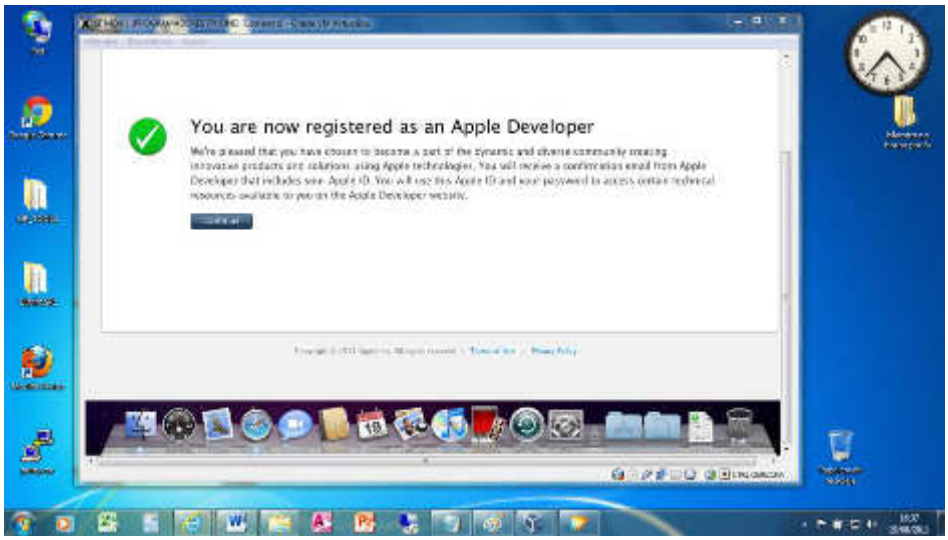


Figura # 4.58 Cuenta creada en Apple Developers.

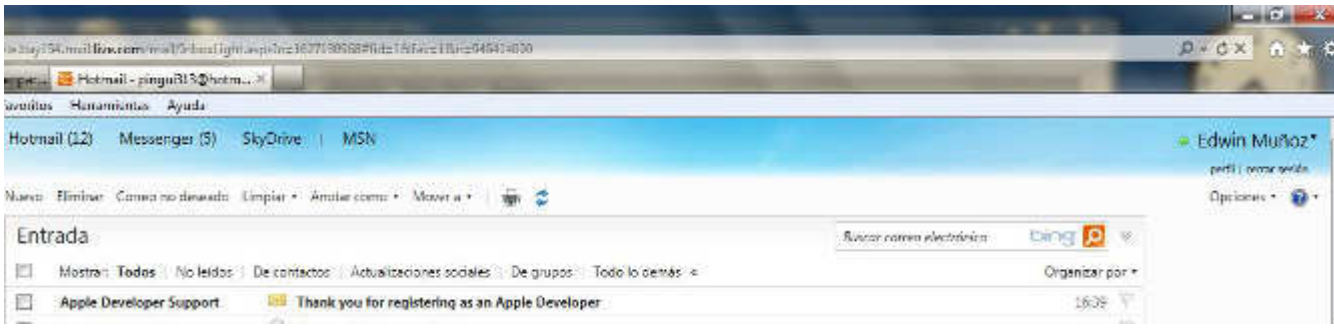


Figura # 4.59 Servicio de Apple.

Una vez creada la cuenta puede descargar las herramientas.



Figura # 4.60 Descarga de Herramientas de desarrollo.

X CODE introducción.

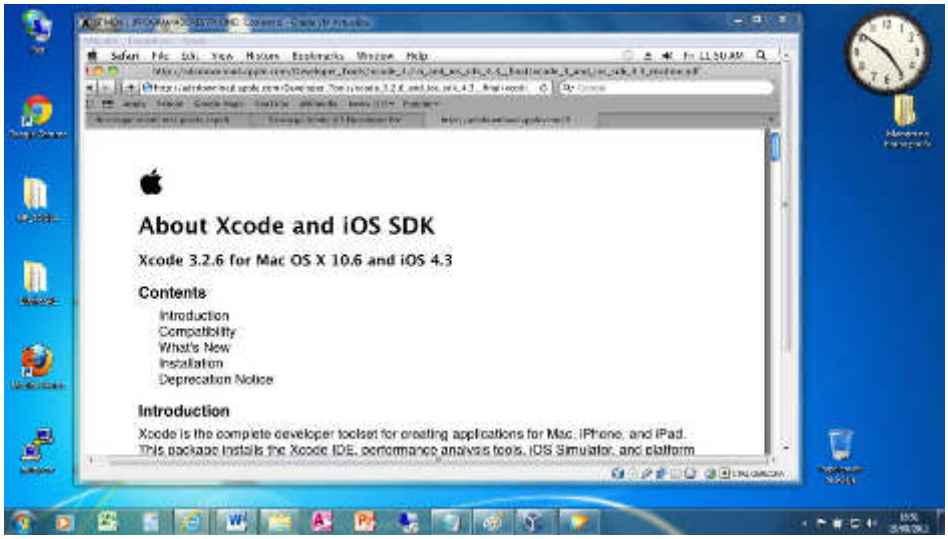


Figura # 4.61 Introduccion de Xcode.

DESCARGA DE XCODE DE LA PROPIA PAGINA DE DESARROLLADORES DE MAC.

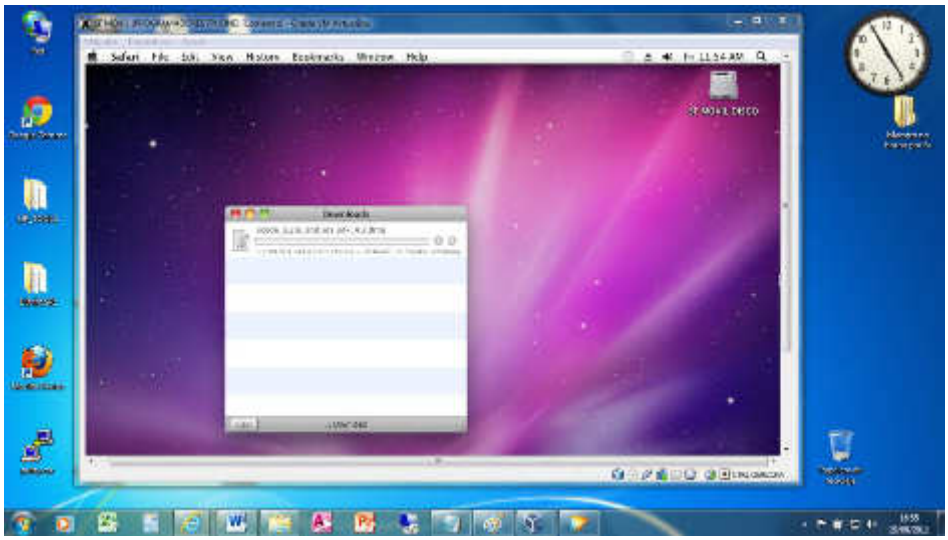


Figura # 4.62 Estado de descarga de Xcode.

ARCHIVOS DE DESCARGA.

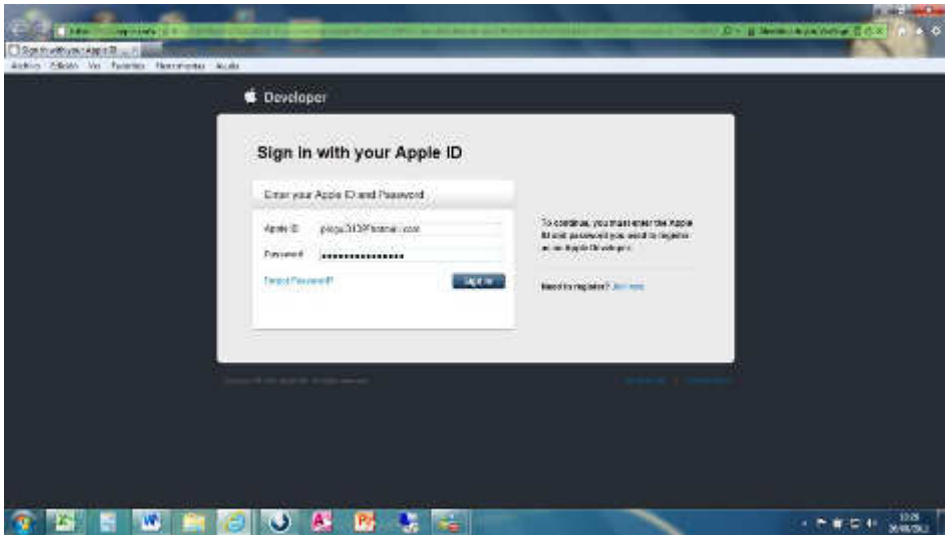


Figura # 4.63 Archivos de descarga.

INGRESO DE SESION PARA DESCARGAR SDK DE XCODE.



Figura # 4.64 Acceso al archivo xcode.

DESCARGA DEL SDK CON EL GESTOR DE DESCARGAS ORBIT.

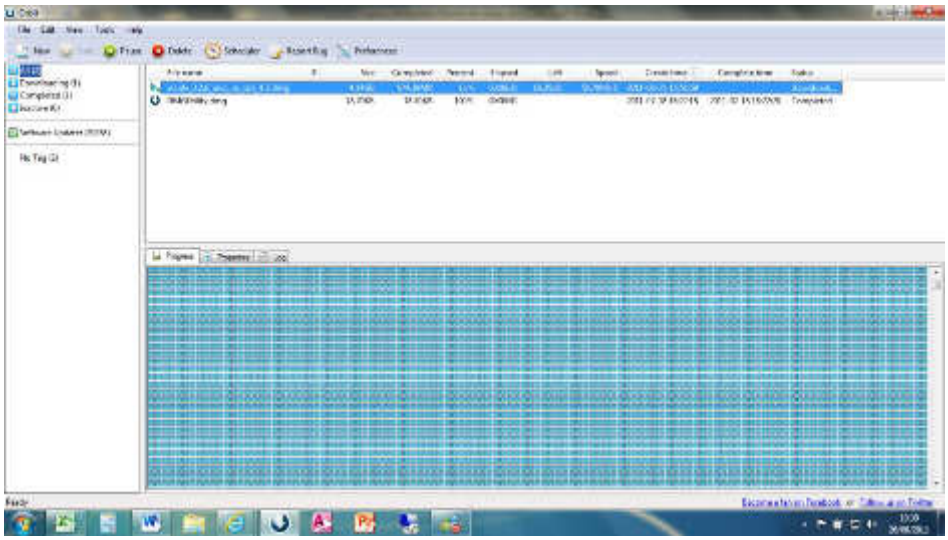


Figura # 4.65 Download Xcode.

INSTALACION DE SDK X CODE EN LA MAQUINA DE VIRTUAL DE SNOW LEOPARD LISTO PARA EMPEZAR A PROGRAMAR.
(LA DESCARGA DEL ARCHIVO SNOW LEOPARD SE TARDO APROXIMADAMENTE 5 DIAS AL IGUAL QUE EL SDK).

INSTALACION DE SDK.

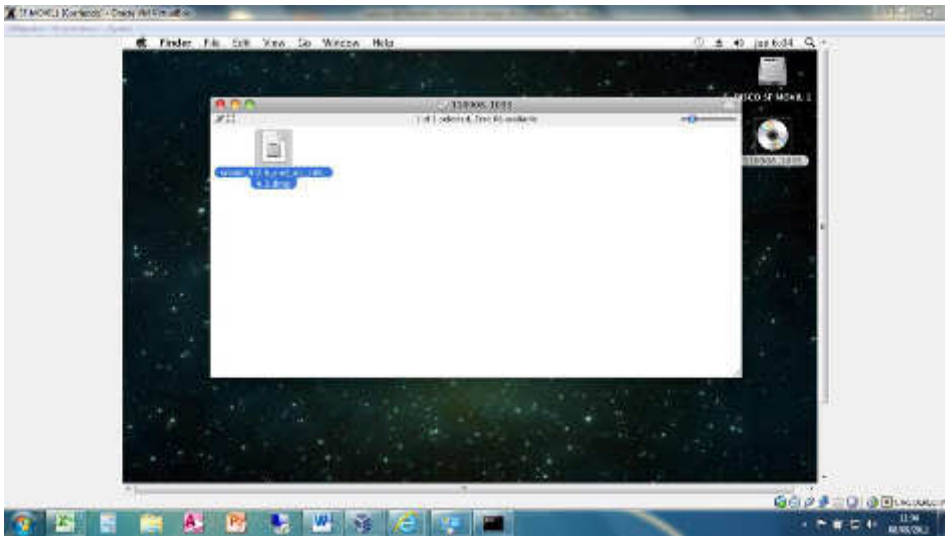


Figura # 4.66 Inicio de instalación SDK de desarrollo de Aplicaciones Iphone.

INSTALADOR LISTO.

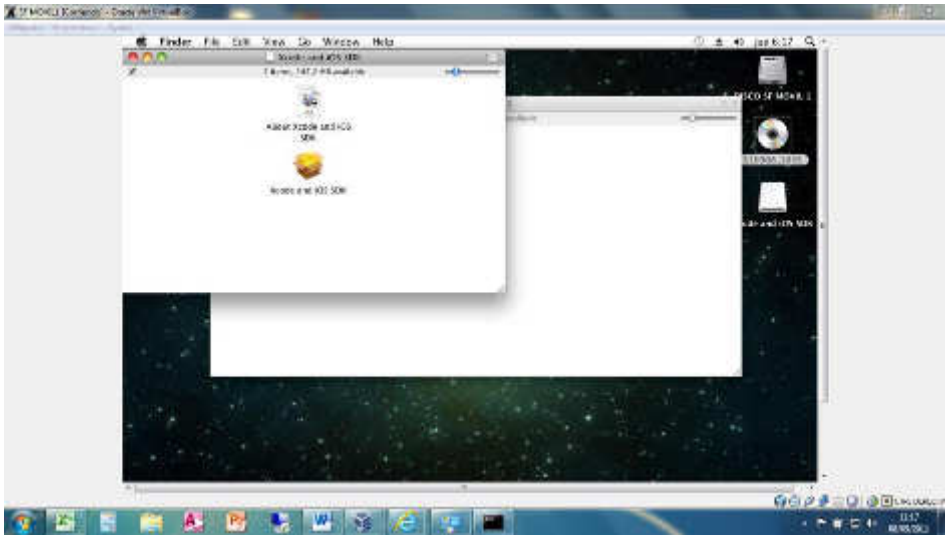


Figura # 4.67 Carga de Instalador SDK.

SEGURIDADES.

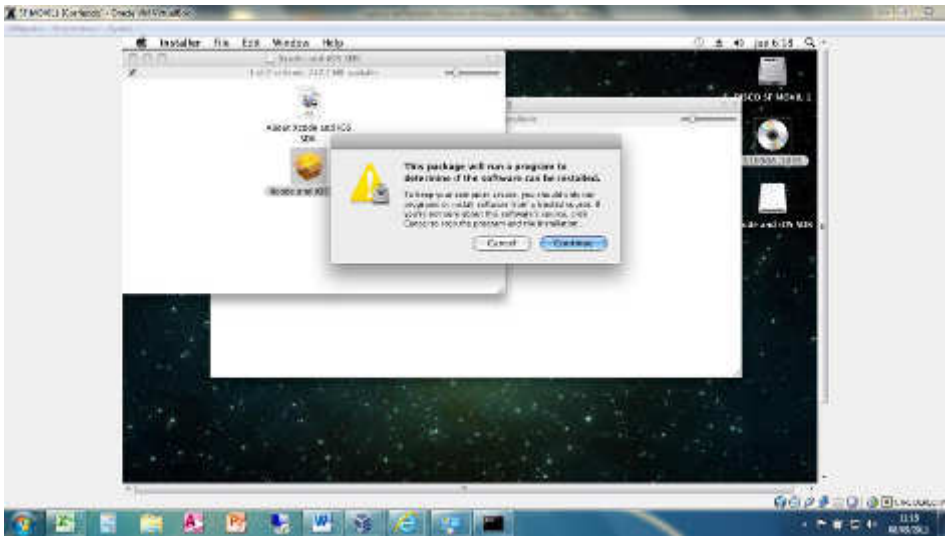


Figura # 4.68 Seguridades de SDK.

ASISTENTE DE INSTALACION.

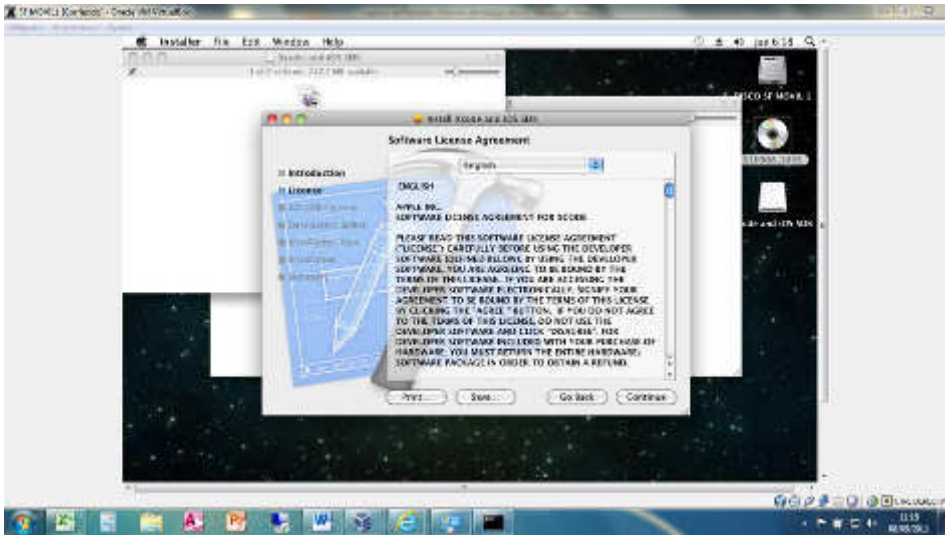


Figura # 4.69 Asistente de Instalación de SDK.

ACUERDOS.

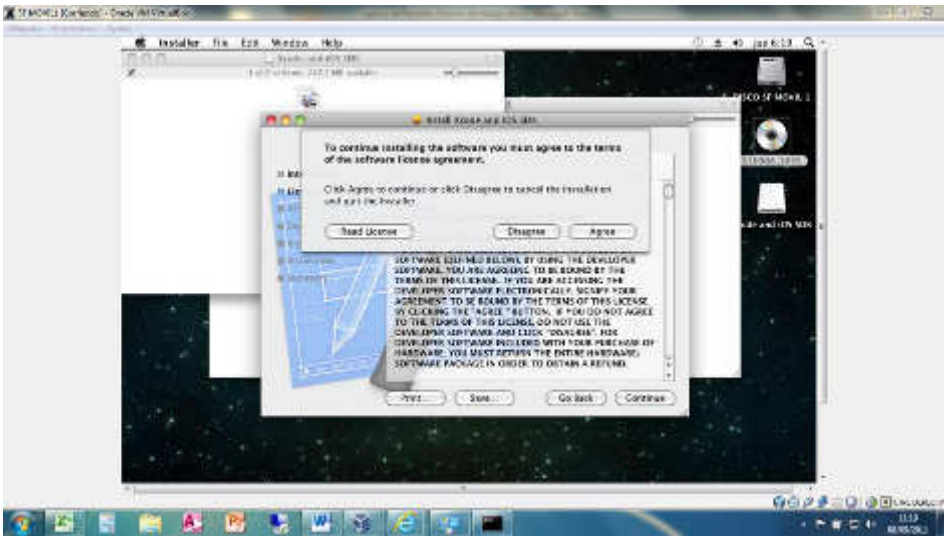


Figura # 4.70 Políticas de acuerdo de SDK.

LICENCIA.

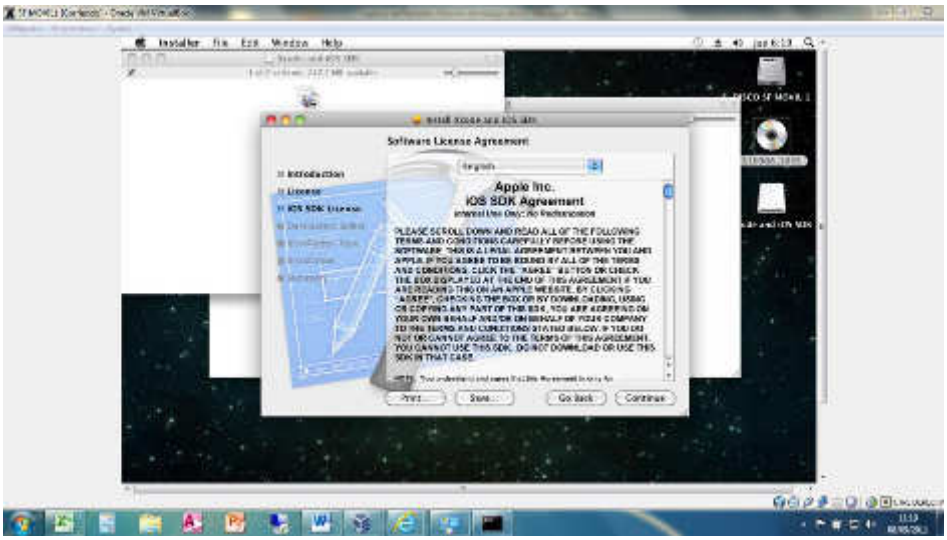


Figura # 4.71 Licencia SDK.

Ubicación de Instalación en nuestra imagen de Disco.

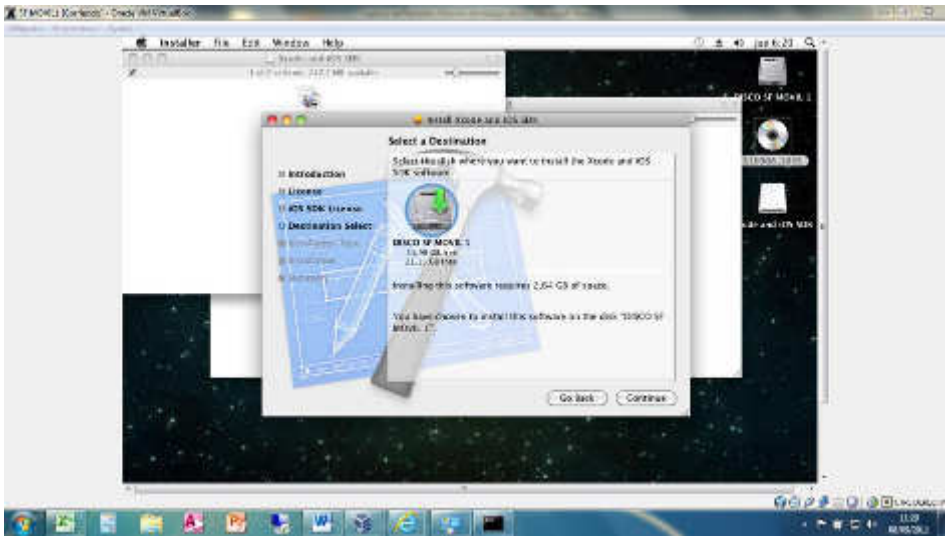


Figura # 4.72 Direccionamiento de SDK en imagen de Disc.

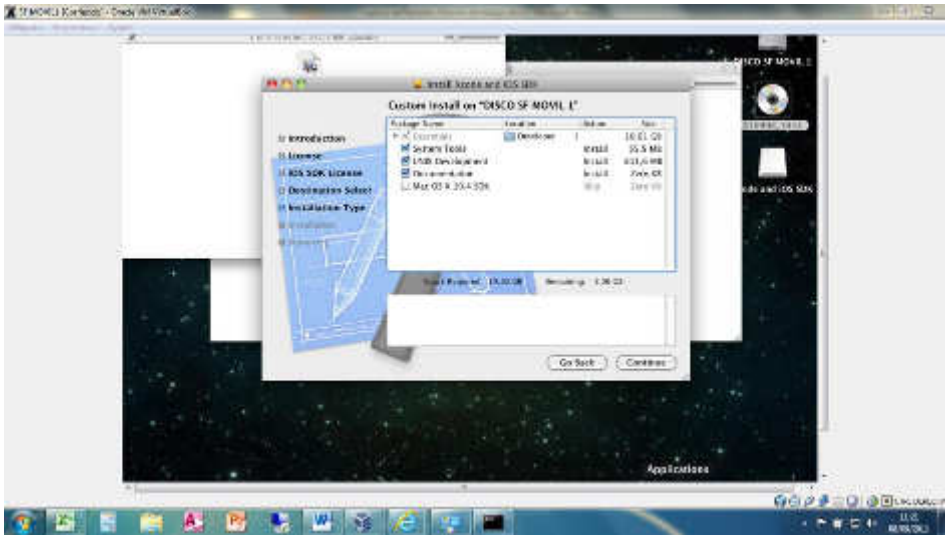


Figura # 4.73 Informe de espacio de SDK en imagen de Disc.

SEGURIDAD DE LA SESION ANTES DE LA INSTALACION.

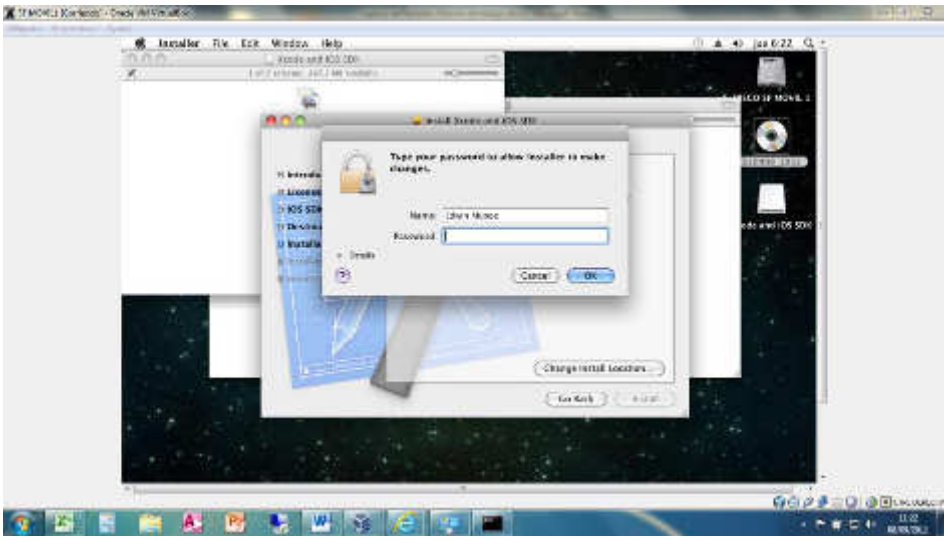


Figura # 4.74 Confirmación de administración en instalación.

PREPARANDO PARA LA INSTALACION.

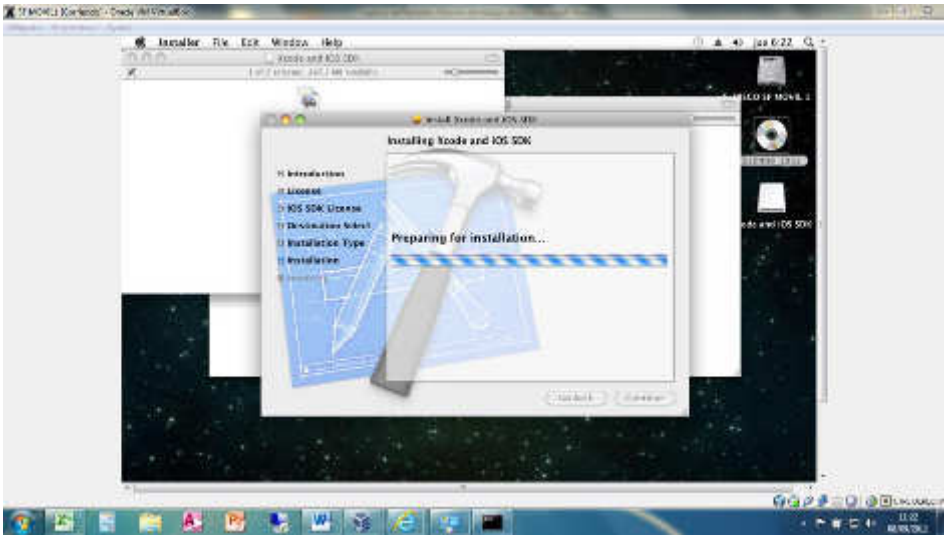


Figura # 4.75 Inicio de Instalación de Xcode.

INSTALACION COMPLETA.

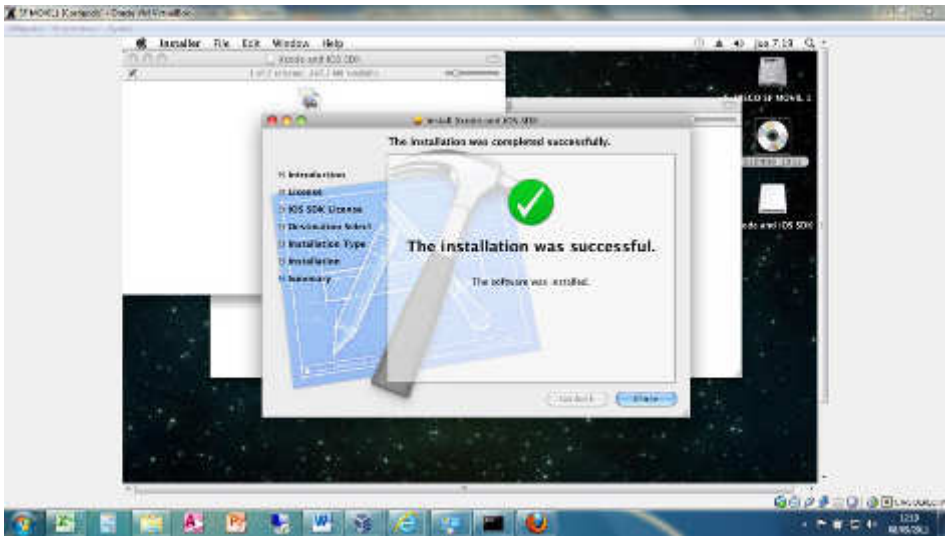


Figura # 4.76 Xcode Instalado.

MAQUINA VIRTUAL CON HERRAMIENTAS LISTA.
DEVELOPER DE APPLE.

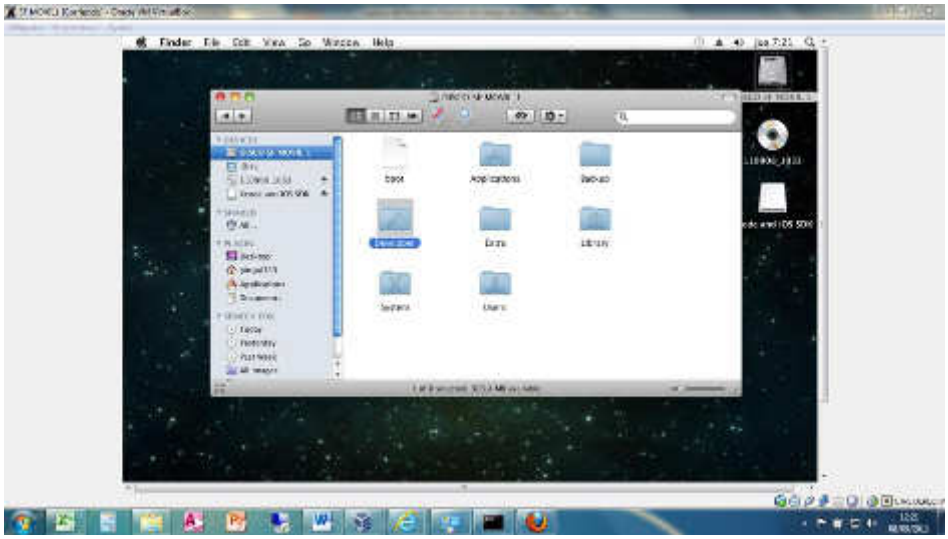
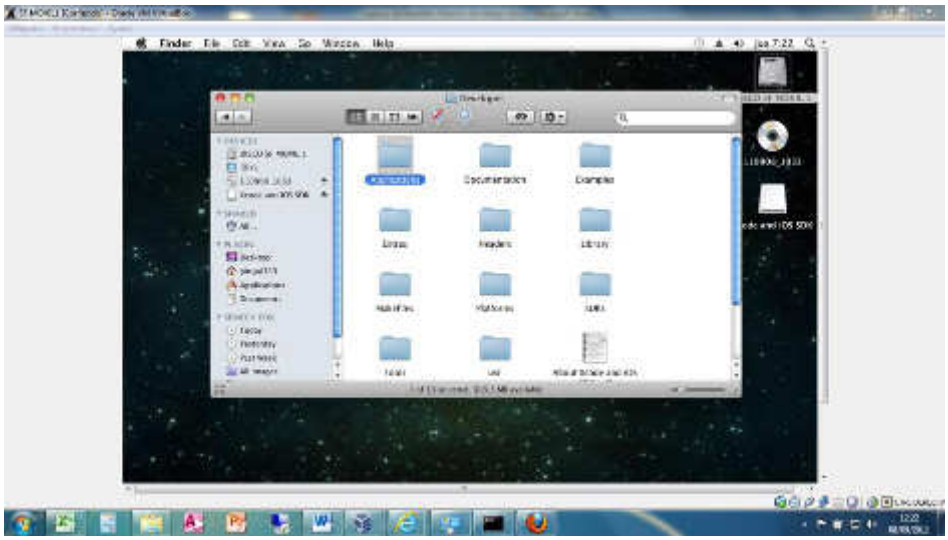
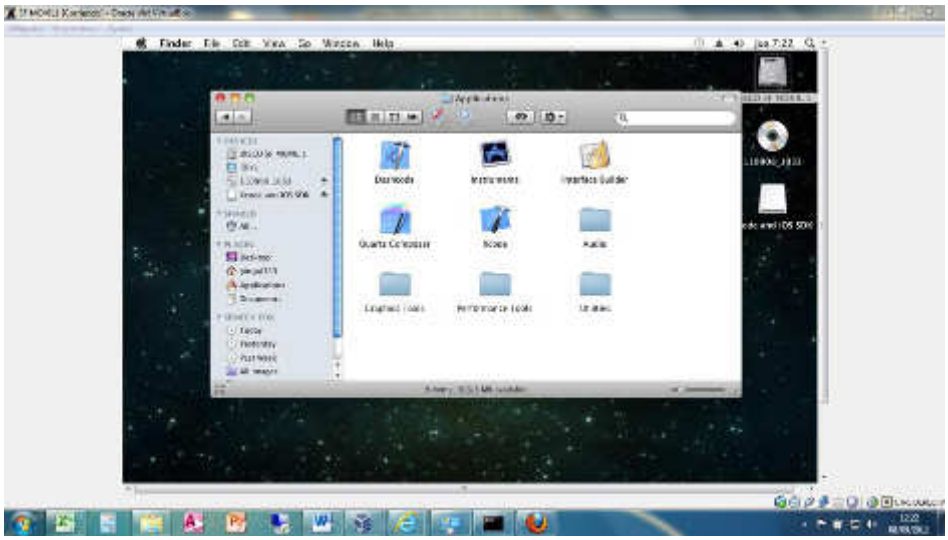


Figura # 4.77 Ruta de Instalación de Xcode.

APLICACIONES.





X CODE.



Figura # 4.78 Bienvenida de Xcode.

INTERFACE BUILDER.

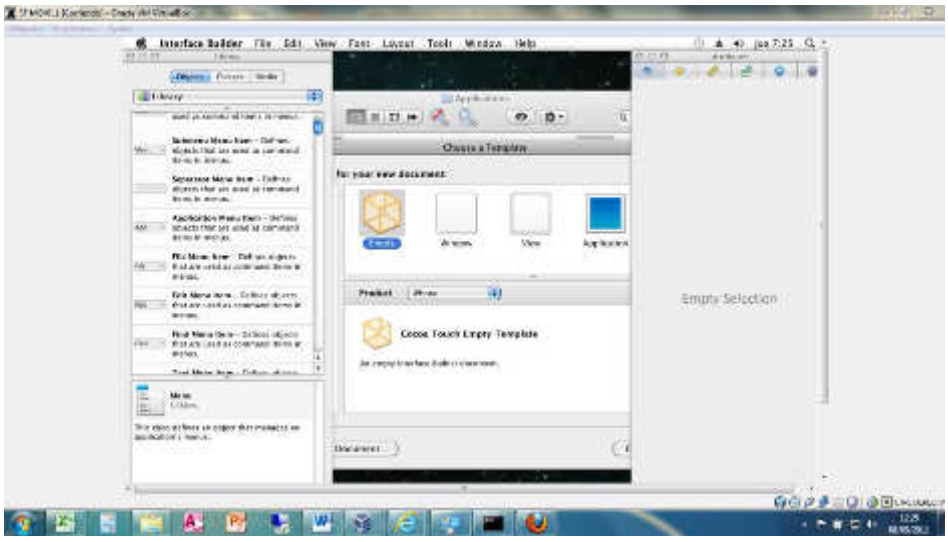


Figura # 4.79 Bienvenida de Interface Builder.

SIMULADOR DE IPHONE.



Figura # 4.80 Bienvenida de Isimulator.