

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO – CHILE
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

**SISTEMA WEB PARA LA EMPRESA ACECON,
CONSIDERANDO LA MOVILIDAD DEL USUARIO**

SEBASTIÁN ENRIQUE CAMPOS DAVILA
FRANCISCO ESAU GONZÁLEZ URRUTIA

INFORME FINAL DEL PROYECTO
PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO DE EJECUCIÓN EN INFORMÁTICA

Diciembre, 2010

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO – CHILE
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

**SISTEMA WEB PARA LA EMPRESA ACECON,
CONSIDERANDO LA MOVILIDAD DEL USUARIO**

SEBASTIÁN ENRIQUE CAMPOS DAVILA
FRANCISCO ESAÚ GONZÁLEZ URRUTIA

Profesor Guía: José Miguel Rubio León
Profesor Correferente: Ricardo Soto De Giorgis

Carrera: **Ingeniería de Ejecución en Informática**

Diciembre, 2010

Dedicatoria

A mi familia, especialmente mi madre, amigos,
compañeros y profesor guía por todo el apoyo brindado,
y a todas las personas que me ayudaron en este largo camino

Sebastián Enrique Campos Dávila

Dedicatoria

A mis Padres quienes me impulsaron a seguir este camino

A mi polola quien fue siempre apoyo en todo momento

A mi hermano y amigos con quienes logramos llegar a este lugar

A mi profesor guía quien proporcionó las herramientas para cumplir mi objetivo

A todos ellos sinceramente muchas gracias

Francisco Esaú González Urrutia

ÍNDICE

Resumen	1
Abstract.....	1
Lista de Figuras	2
Lista de Tablas.....	4
CAPÍTULO 1: INTRODUCCIÓN.....	6
CAPÍTULO 2: DEFINICIÓN DE OBJETIVOS.....	7
2.1 Objetivo general	7
2.2 Objetivos específicos.....	7
CAPÍTULO 3: ESTADO DEL ARTE	8
3.1 Situación actual	8
3.1.1 Identificación de la empresa.....	8
3.1.2 Descripción de la empresa.....	8
3.1.3 Flujo de trabajo de la empresa.....	9
3.2 Problemas identificados.....	11
CAPÍTULO 4: ANÁLISIS DE TECNOLOGÍAS.....	12
4.1 Dispositivos móviles	12
4.1.1 ¿Qué es un dispositivo móvil?.....	12
4.1.2 Tipos de dispositivo móvil	12
4.1.3 Historia de los dispositivos móviles.....	14
4.1.4 Plataforma y lenguajes soportados	17
4.1.5 Herramientas de desarrollo.....	23
4.1.6 Limitaciones sobre el desarrollo.....	24
4.2 Servicios Web.....	26
4.2.1 Arquitectura de los Servicios Web	27
4.2.2 Nuevos Protocolos para Servicios Web.....	30
4.2.3 Ventajas de los servicios web.....	31
4.2.4 Inconvenientes de los servicios Web.....	32
4.2.5 Razones para crear servicios Web	32
4.2.6 Plataformas de Servicios Web.....	33
CAPÍTULO 5: SOLUCIÓN PLANTEADA	40
5.1 Solución general	40
5.2 Solución específica para los problemas identificados	41
5.3 Tecnologías.....	42
5.3.1 Motor de base de datos	42
5.3.2 Lenguaje de programación	42

5.3.3	Javaservertimepages	43
5.3.4	Entorno de desarrollo	44
CAPÍTULO 6: ESTUDIO DE FACTIBILIDAD		45
6.1	Factibilidad operativa	45
6.2	Factibilidad técnica.....	45
6.3	Factibilidad legal	46
6.4	Factibilidad económica.....	47
6.4.1	Costos asociados.....	47
6.4.2	Beneficios para la empresa	49
CAPÍTULO 7: ANALISIS DE RIESGO		50
CAPÍTULO 8: PARADIGMA Y METODOLOGIA DE DESARROLLO		52
8.1	Paradigma de desarrollo	52
8.2	Metodología de desarrollo.....	52
CAPÍTULO 9: HERRAMIENTAS Y ARQUITECTURA.....		54
9.1	Herramientas.....	54
9.2	Arquitectura.....	55
CAPÍTULO 10: PLAN DE TRABAJO.....		57
CAPÍTULO 11: ESPECIFICACIÓN DE REQUERIMIENTOS.....		58
11.1	Introducción.....	58
11.2	Identificación de usuarios participantes	58
11.3	Catálogo de requisitos del sistema.....	58
11.3.1	Objetivos y alcance del sistema.....	59
11.3.2	Definiciones, acrónimos y abreviaturas.....	59
11.3.3	Descripción general	60
11.3.4	Requisitos funcionales.....	60
11.3.5	Suposiciones y dependencias.....	73
11.3.6	Requisitos de usuario y tecnológicos	73
11.3.7	Requisitos de interfaces externas.....	74
11.3.8	Requisitos de rendimiento	74
11.3.9	Requisitos de desarrollo.....	74
CAPÍTULO 12: DIAGRAMA DE CASOS DE USO.....		75
12.1	Caso de uso general	75
12.2	Especificación caso de uso diagrama general.....	76
12.3	Caso de uso gestionar cliente particular - Grafico extendido.....	78
12.3.1	Caso de uso gestionar cliente particular – Grafico narrativo extendido.....	79
12.4	Caso de uso gestionar cliente administración - Grafico extendido	80

12.4.1	Caso de uso gestionar cliente administración – Grafico narrativo extendido	81
12.5	Caso de uso gestionar técnicos – Grafico extendido	83
12.5.1	Caso de uso gestionar técnicos - Grafico narrativo extendido	83
12.6	Caso de uso gestionar vendedores – Grafico extendido	85
12.6.1	Caso de uso gestionar vendedores – Grafico narrativo extendido.....	86
12.7	Caso de uso gestionar representante – Grafico extendido	88
12.7.1	Caso de uso gestionar representante – Grafico narrativo extendido	88
12.8	Caso de uso gestionar obras – Grafico extendido.....	91
12.8.1	Caso de uso gestionar obras – Grafico narrativo extendido	91
12.9	Caso de uso gestionar orden de trabajo móvil – Grafico extendido.....	94
12.9.1	Caso de uso gestionar orden de trabajo móvil - Descargar listado ot –Gráfico narrativo extendido	94
12.9.2	Caso de uso gestionar orden de trabajo móvil - Descarga listado ot pendiente – Gráfico narrativo extendido.....	95
12.10	Caso de uso gestionar presupuesto – Grafico extendido	97
12.10.1	Caso de uso gestionar presupuesto – Gráfico narrativo extendido.....	97
CAPÍTULO 13: DIAGRAMA DE CLASES Y MODELO DE DATOS		100
13.1	Diagrama de clases persistentes	100
13.2	Diagrama de clases de lógica y acceso a datos.....	101
13.3	Modelo entidad – relación	102
13.4	Modelo entidad-relación - refinado	103
13.5	Modelo relacional	104
CAPÍTULO 14: DIAGRAMAS DE SECUENCIA		105
14.1	Diagramas de secuencia: Gestionar cliente administración	105
14.1.1	Diagrama de secuencia: Ingresar cliente administración	105
14.1.2	Diagrama de secuencia: Buscar cliente administración por rut.....	106
14.1.3	Diagrama de secuencia: Buscar cliente administración por nombre/apellido.....	106
14.1.4	Diagrama de secuencia: Modificar cliente administración búsqueda por rut.....	107
14.1.5	Diagrama de secuencia: Modificar cliente administración búsqueda por nombre/apellido	108
14.2	Diagramas de secuencia: gestionar cliente particular.....	109
14.2.1	Diagrama de secuencia: Ingresar cliente particular	109
14.2.2	Diagrama de secuencia: Buscar cliente particular por rut	109
14.2.3	Diagrama de secuencia: Buscar cliente particular por nombre/apellido	110
14.2.4	Diagrama de secuencia: Modificar cliente particular búsqueda por rut	111

14.2.5 Diagrama de secuencia: Modificar cliente particular búsqueda por nombre/apellido	112
14.3 Diagramas de secuencia: Gestionar representante	113
14.3.1 Diagrama de secuencia: Ingresar representante.....	113
14.3.2 Diagrama de secuencia: Buscar representante por rut.....	113
14.3.3 Diagrama de secuencia: Buscar representante por nombre/apellido.....	114
14.3.4 Diagrama de secuencia: Modificar representante búsqueda por rut.....	115
14.3.5 Diagrama de secuencia: Modificar representante búsqueda por nombre/apellido	116
14.4 Diagramas de secuencia: Gestionar vendedor	117
14.4.1 Diagrama de secuencia: Ingresar vendedor	117
14.4.2 Diagrama de secuencia: Buscar vendedor por rut	117
14.4.3 Diagrama de secuencia: Buscar vendedor por nombre/apellido	118
14.4.4 Diagrama de secuencia: Modificar vendedor búsqueda por rut	119
14.4.5 Diagrama de secuencia: Modificar vendedor búsqueda por nombre/apellido	120
14.5 Diagramas de secuencia: Gestionar técnico	121
14.5.1 Diagrama de secuencia: Ingresar técnico	121
14.5.2 Diagrama de secuencia: Buscar técnico por rut.....	121
14.5.3 Diagrama de secuencia: Buscar técnico por nombre/apellido.....	122
14.5.4 Diagrama de secuencia: Modificar técnico búsqueda por rut.....	123
14.5.5 Diagrama de secuencia: Modificar técnico búsqueda por nombre/apellido.....	124
14.6 Diagramas de secuencia: Gestionar obras	125
14.6.1 Diagrama de secuencia: Ingresar obra.....	125
14.6.2 Diagrama de secuencia: Buscar obra por su nombre.....	125
14.6.3 Diagrama de secuencia: Modificar obra búsqueda por su nombre.....	126
14.6.4 Diagrama de secuencia: Dar de baja obra búsqueda por su nombre	127
14.7 Diagramas de secuencia: OT móvil.....	128
14.7.1 Diagrama de secuencia: Descargar orden de trabajo.....	128
14.7.2 Diagrama de secuencia: Descargar orden de trabajo pendiente	129
14.8 Diagramas de secuencias: Gestionar Presupuesto	130
14.8.1 Diagrama de secuencia: Crear Presupuesto.....	130
CAPITULO 15: DIAGRAMA DE COMPONENTES Y DESPLIEGUE	131
15.1 Diagrama de componentes.....	131
15.2 Diagrama de despliegue.....	132
CAPÍTULO 16: DISEÑO DE INTERFACES	133

16.1 Elementos de interfaz	133
16.2 Interfaces	134
16.2.1 Interfaz de inicio.....	134
16.2.2 Interfaz Portada.....	135
16.2.3 Interfaz Agregar Cliente Administrador	136
16.2.4 Interfaz Modificar Cliente Administrador	137
16.2.5 Interfaz Buscar Clientes.....	138
16.2.6 Interfaz Modificar Obra.....	139
16.2.7 Interfaz Agregar Presupuesto	140
16.2.8 Interfaces Aplicación Móvil	141
CAPÍTULO 17: PLAN DE PRUEBAS.....	144
17.1 Plan de pruebas	147
17.1.1 Módulo Técnico.....	147
17.1.2 Módulo Vendedor.....	149
17.1.3 Módulo Obras	151
17.1.4 Módulo Representante	152
17.1.5 Módulo Cliente Particular	154
17.1.6 Módulo Cliente Administración	155
17.1.7 Módulo Presupuesto	156
17.1.8 Módulo Órdenes de Trabajo	158
17.1.9 Dispositivo Móvil.....	159
CONCLUSIONES Y TRABAJO FUTURO	161
REFERENCIAS	162

Resumen

El presente informe de proyecto de título dará a conocer el trabajo realizado para la empresa Acecon, la cual se desempeña en el rubro de la construcción e instalación de artefactos a gas. La principal problemática es la de no contar con un sistema de control de flujo de trabajo, el cual pueda gestionar la información de la empresa.

El informe dedica páginas al análisis realizado sobre distintas tecnologías que permitieron dar solución a la problemática de la empresa como los dispositivos móviles y servicios web. Se especifica además, un plan de pruebas realizado para asegurar la calidad del sistema entregado.

El modelo de desarrollo utilizado corresponde al de Prototipos Evolutivos, de manera de poder hacer entregas sucesivas que incorporen cada vez más funcionalidades al sistema hasta evolucionar a la versión final. El enfoque utilizado corresponde a Orientación a Objetos, dada su amplia difusión y correspondencia con el lenguaje de modelado utilizado UML (Unified Model Language).

Palabras Clave: Dispositivos móviles, servicios web, java, diagramas UML.

Abstract

The present document will show the work done for the company Acecon, which plays in the field of construction and installation of gas appliances. The main problem is to not have a control system workflow, which can manage enterprise information.

The report devotes pages to the analysis of different technologies that allow to solve the problems of the company as mobile devices and Web services. It also specifies a test plan made to ensure the quality of the delivered system.

The development model used is the Evolutionary Prototyping so you can make successive deliveries increasingly incorporate functionality to the system to evolve to the final version. The approach corresponds to Object Orientation, given its wide distribution and correspondence with the used modeling language UML (Unified Model Language).

Keywords: mobile devices, web services, java, UML diagrams.

Lista de Figuras

Figura 3.1. Organigrama Acecon Ltda.	8
Figura 3.2. Flujo de Trabajo Acecon Ltda.	10
Figura 4.1. iPHONE teléfono inteligente multimedia.	13
Figura 4.2. PDA Acer Modelo N° 311.	13
Figura 4.3. Consola PSP (Play Station Portable) de SONY.	14
Figura 4.4. Panel De Control De La Newton.	14
Figura 4.5. Calculadora Casio FX-750p.	15
Figura 4.6. Pilot 5000.	15
Figura 4.7. PocketPC Hp.	16
Figura 4.8. Emulador de Windows Mobile 6.2.	19
Figura 4.9. Sistema Operativo Symbian, Serie 60.	20
Figura 4.10. Interfaz iPHONE.	22
Figura 4.11. Sistema Operativo Android.	23
Figura 4.12. Api Servicios Web en Java.	38
Figura 5.1. Arquitectura Solución Planteada.	40
Figura 9.1. Arquitectura de tres capas.	56
Figura 10.1. Plan de Trabajo.	57
Figura 11.1. Presupuesto Acecon.	68
Figura 11.2. Orden de trabajo Acecon.	70
Figura 12.1. Diagrama General de Casos de Uso.	75
Figura 12.2. Caso de Uso Gestionar Cliente Particular.	78
Figura 12.3. Caso de Uso Gestionar Cliente Administración.	80
Figura 12.4. Caso de Uso Gestionar Técnicos.	83
Figura 12.5. Caso de Uso Gestionar Vendedores.	85
Figura 12.6. Caso de Uso Gestionar Representante.	88
Figura 12.7. Caso de Uso Gestionar Obras.	91
Figura 12.9. Caso de Uso Gestionar Orden de Trabajo Móvil.	94
Figura 12.10. Caso de Uso Gestionar Presupuesto.	97
Figura 13.1. Diagrama de clases persistentes.	100
Figura 13.2. Diagrama de clases de lógica y acceso a datos.	101
Figura 13.3. Modelo Entidad-Relación.	102
Figura 13.4. Modelo Entidad-Relación – Refinado.	103
Figura 13.5. Modelo Relacional.	104
Figura 14.1. Diagrama de secuencia: ingresar cliente administración.	105
Figura 14.2. Diagrama de secuencia: buscar cliente administración por rut.	106
Figura 14.3. Diagrama de secuencia: buscar cliente administración por nombre/apellido.	106
Figura 14.4. Diagrama de secuencia: modificar cliente administración, búsqueda por rut.	107
Figura 14.5. Diagrama de secuencia: modificar cliente administración por nombre/apellido.	108
Figura 14.6. Diagrama de secuencia: Ingresar cliente particular.	109
Figura 14.7. Diagrama de secuencia: buscar cliente particular por rut.	109
Figura 14.8. Diagrama de secuencia: buscar cliente particular por nombre/rut.	110
Figura 14.9. Diagrama de secuencia: modificar cliente particular por rut.	111
Figura 14.10. Diagrama de secuencia: Modificar cliente particular por nombre/apellido.	112
Figura 14.11. Diagrama de secuencia: Ingresar representante.	113
Figura 14.12. Diagrama de secuencia: buscar representante por rut.	113
Figura 14.13. Diagrama de secuencia: buscar representante por nombre/apellido.	114

Figura 14.14. Diagrama de secuencia: Modificar representante por rut.....	115
Figura 14.15. Diagrama de secuencia: Modificar Representante por nombre/apellido.....	116
Figura 14.16. Diagrama de secuencia: Ingresar vendedor.....	117
Figura 14.17. Diagrama de secuencia: buscar vendedor por rut.....	117
Figura 14.18. Diagrama de secuencia: buscar vendedor por nombre/apellido.....	118
Figura 14.19. Diagrama de secuencia: Modificar vendedor por rut.....	119
Figura 14.20. Diagrama de secuencia: Modificar vendedor por nombre/apellido.....	120
Figura 14.21. Diagrama de secuencia: Ingresar técnico.....	121
Figura 14.22. Diagrama de secuencia: buscar técnico por rut.....	121
Figura 14.23. Diagrama de secuencia: buscar técnico por nombre/apellido.....	122
Figura 14.24. Diagrama de secuencia: Modificar técnico por rut.....	123
Figura 14.25. Diagrama de secuencia: Modificar técnico por nombre/apellido.....	124
Figura 14.26. Diagrama de secuencia: ingresar obra.....	125
Figura 14.27. Diagrama de secuencia: buscar obra por nombre.....	125
Figura 14.28. Diagrama de secuencia: modificar obra, búsqueda por nombre.....	126
Figura 14.29. Diagrama de secuencia: dar de baja una obra, búsqueda por nombre.....	127
Figura 14.30. Diagrama de secuencia: Descargar orden de trabajo dispositivo móvil.....	128
Figura 14.31. Diagrama de secuencia: Descargar orden de trabajo pendiente dispositivo móvil.....	129
Figura 14.32. Diagrama de secuencia: Crear Presupuesto.....	130
Figura 15.1. Diagrama de componentes del sistema.....	131
Figura 15.2. Diagrama de despliegue del sistema.....	132
Figura 16.1. Estructura Interfaz.....	133
Figura 16.2. Interfaz de Inicio del Sistema.....	134
Figura 16.3. Interfaz de bienvenida del Sistema.....	135
Figura 16.4. Interfaz Agregar Cliente.....	136
Figura 16.5. Interfaz Modificar Cliente.....	137
Figura 16.6. Interfaz Buscar Cliente.....	138
Figura 16.7. Interfaz Modificar Obra.....	139
Figura 16.8. Interfaz Agregar Presupuesto.....	140
Figura 16.9. Interfaz Móvil Bienvenidos.....	141
Figura 16.10. Interfaz Móvil Login.....	141
Figura 16.11. Interfaz Móvil Menú Principal.....	142
Figura 16.12. Interfaz Móvil Orden de Trabajo.....	142
Figura 16.13. Interfaz Móvil Orden de Trabajo Pendiente.....	143
Figura 17.1. Tipos de Pruebas.....	144

Lista de Tablas

Tabla 4.1. Comparativa Sistemas Operativos Móviles.....	24
Tabla 6.1. Costo Personal.....	47
Tabla 6.2. Costo de hardware.....	47
Tabla 6.3. Costo de software.....	48
Tabla 6.4. Costo Total.....	48
Tabla 7.1. Análisis de Riesgo.....	50
Tabla 7.2. Plan de Mitigación y Contingencia.....	51
Tabla 11.1. Agregar cliente particular.....	61
Tabla 11.2. Agregar cliente administración.....	61
Tabla 11.3. Modificar cliente particular búsqueda rut.....	61
Tabla 11.4. Modificar cliente particular búsqueda nombre y apellido.....	61
Tabla 11.5. Buscar cliente particular búsqueda rut.....	61
Tabla 11.6. Buscar cliente particular búsqueda nombre y apellido.....	62
Tabla 11.7. Agregar representante.....	62
Tabla 11.8. Modificar representante búsqueda por rut.....	62
Tabla 11.9. Modificar representante búsqueda por nombre y apellido.....	63
Tabla 11.10. Buscar representante por rut.....	63
Tabla 11.11. Buscar representante por nombre y apellido.....	63
Tabla 11.12. Agregar técnico.....	64
Modificar Técnico búsqueda por rut.....	64
Tabla 11.13. Modificar técnico búsqueda por rut.....	64
Tabla 11.14. Modificar técnico búsqueda por nombre y apellido.....	64
Tabla 11.15. Buscar técnico por rut.....	64
Tabla 11.16. Buscar técnico por nombre y apellido.....	65
Tabla 11.17. Agregar vendedor.....	65
Tabla 11.18. Modificar vendedor búsqueda por rut.....	65
Tabla 11.19. Modificar Vendedor búsqueda por nombre y apellido.....	66
Tabla 11.20. Buscar vendedor por rut.....	66
Tabla 11.21. Buscar vendedor por nombre y apellido.....	66
Tabla 11.22. Agregar Obra.....	66
Tabla 11.23. Modificar obra búsqueda por nombre.....	67
Tabla 11.24. Modificar obra búsqueda por id.....	67
Tabla 11.25. Buscar obra por nombre.....	67
Tabla 11.26. Buscar obra por id.....	67
Tabla 11.27. Generar presupuesto.....	69
Tabla 11.28. Modificar presupuesto búsqueda por nombre.....	69
Tabla 11.29. Modificar presupuesto búsqueda por id.....	69
Tabla 11.30. Presupuesto búsqueda por nombre.....	69
Tabla 11.31. Presupuesto búsqueda por id.....	70
Tabla 11.32. Generar OT.....	71
Tabla 11.33. Modificar OT.....	71
Tabla 11.34. OT búsqueda por id.....	71
Tabla 11.35. Descargar OT.....	71
Tabla 11.36. Cerrar OT.....	72
Tabla 11.37. Crear Usuario.....	72
Tabla 12.1. Gestionar cuentas de usuario.....	76

Tabla 12.2. Gestionar cliente particular.....	76
Tabla 12.3. Gestionar cliente administrador.....	76
Tabla 12.4. Gestionar representante legal.....	76
Tabla 12.5. Gestionar vendedores.....	77
Tabla 12.6. Gestionar técnicos.....	77
Tabla 12.7. Gestionar obras.....	77
Tabla 12.8. Gestionar presupuesto.....	77
Tabla 12.9. Gestionar orden de trabajo.....	78
Tabla 12.10. Caso de uso gestionar cliente particular.....	80
Tabla 12.11. Caso de uso gestionar cliente administración.....	82
Tabla 12.12. Caso de uso gestionar técnicos.....	85
Tabla 12.13. Caso de uso gestionar vendedores.....	87
Tabla 12.14. Caso de uso gestionar representante.....	90
Tabla 12.15. Caso de uso gestionar obras.....	93
Tabla 12.16. Caso de uso gestionar orden de trabajo móvil.....	95
Tabla 12.17. Caso de uso gestionar orden de trabajo móvil.....	96
Tabla 12.18. Caso de uso gestionar presupuesto.....	99
Tabla 17.1. Plan de pruebas módulo técnico.....	148
Tabla 17.2. Plan de pruebas módulo vendedor.....	150
Tabla 17.3. Plan de pruebas módulo obras.....	152
Tabla 17.4. Plan de pruebas módulo representante.....	154
Tabla 17.5. Plan de pruebas módulo cliente particular.....	155
Tabla 17.6. Plan de pruebas módulo cliente administración.....	156
Tabla 17.7. Plan de pruebas módulo presupuesto.....	157
Tabla 17.8. Plan de pruebas módulo técnico.....	158
Tabla 17.9. Plan de pruebas dispositivo móvil.....	160

CAPÍTULO 1: INTRODUCCIÓN

El objetivo de este informe es en primer lugar, dar a conocer la problemática de una empresa, para posteriormente investigar las tecnologías existentes en la actualidad, que permitan dar una solución recomendable a la compañía.

Esta empresa se desenvuelve en el área de la construcción y presta servicios de normalización de edificios en el campo de emanación de gas, entregando post realización de labores, la posibilidad de obtener sello verde por parte de la superintendencia de electricidad y combustible. Además proporciona soporte técnico a ciertos requerimientos de la empresa Mademsa.

El objetivo principal es desarrollar un sistema web, que considere la movilidad del usuario y así permitir una mejor gestión de la empresa, usando tecnologías actuales como los servicios web y dispositivos móviles.

Para lograr este objetivo en el presente informe se reflejaran las distintas etapas que permiten diseñar un software de calidad acorde con las necesidades de la empresa, en una primera instancia se definirán los objetivos claves así como el funcionamiento actual de la compañía buscando los problemas principales a los cuales dar una solución.

Luego se detallara la investigación realizada, necesaria para dar una solución, seguido se hará un estudio de factibilidad del proyecto y se elegirán las tecnologías necesarias para realizarlo.

En las secciones finales podremos apreciar el análisis, diseño e implementación del sistema, mediante el uso de diagramas UML (Unified Model Language) y otras herramientas. Así como el diseño de interfaces que darán la navegabilidad final a la aplicación.

Finalmente se especifica un plan de pruebas realizado para asegurar la calidad del sistema final generado.

CAPÍTULO 2: DEFINICIÓN DE OBJETIVOS

2.1 Objetivo general

Análisis, desarrollo e implementación de un sistema web para la empresa Acecon que considere la movilidad del usuario como parte del proceso.

2.2 Objetivos específicos

- Analizar la problemática de la empresa para determinar el flujo de trabajo.
- Capturar los requerimientos principales que permitan dar solución a la problemática de la empresa.
- Analizar las tecnologías existentes que dan solución a la problemática de la empresa y que además permitan realizar aplicaciones sobre dispositivos móviles.
- Analizar plataformas de desarrollo para construir servicios Web para entornos móviles.
- Construir y testear una aplicación de control de flujo de trabajo para la empresa, así como un módulo que permita la interacción con un entorno móvil.

CAPÍTULO 3: ESTADO DEL ARTE

3.1 Situación actual

3.1.1 Identificación de la empresa

RAZON SOCIAL	:	ACECON LTDA
RUT	:	76.054.038-2
GIRO	:	CONSTRUCCION E INSTALACIONES
DOMICILIO LEGAL	:	PORVENIR 464, SANTIAGO
WEB	:	WWW.ACECON.CL
E-MAIL	:	INFO@ACECON.CL
TELEFONO	:	(02) 634 69 48 — (02) 635 05 62

3.1.2 Descripción de la empresa

Acecon, inicia sus actividades orientada a brindar seguridad y confiabilidad a sus clientes a través de un trabajo desarrollado por especialistas del área gas, instaladores autorizados SEC y personal con experiencia en construcción. Además ofrecen desarrollar proyectos de viviendas, remodelaciones, construcción de redes de gas y agua, mantenciones e instalaciones. Acecon actualmente está a cargo del servicio técnico proporcionado por la empresa Mademsa la cual hace envío de órdenes de trabajo que la empresa debe realizar diariamente.

Consta de tres departamentos principales a cargo del Departamento de Gerencia: Ventas, Operaciones y Finanzas, no cuenta con ningún sistema informático en la actualidad y maneja toda su información en documentos archivados que son guardados en estantes dentro de la empresa, es por este motivo que la gestión de la información manejada se hace lenta y engorrosa. La figura 3.1 nos muestra el organigrama de la empresa.

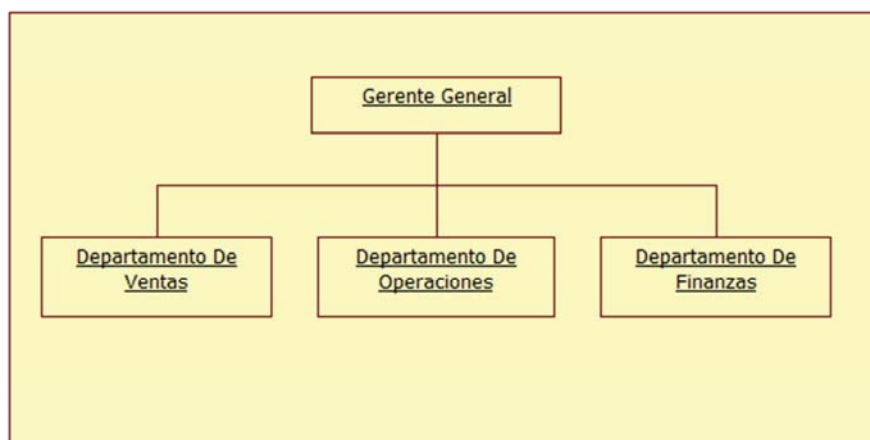


Figura 3.1. Organigrama Acecon Ltda.

Dentro de la gama de servicios ofrecidos, destaca la normalización de edificios referente al área de gas, resolución exenta 489 dictada por la Superintendencia de Electricidad y Combustible (SEC), que actualiza el procedimiento para la inspección periódica de las instalaciones de gas.

Es una empresa nueva en el mercado de la construcción, con menos de un año de funcionamiento y con grandes proyectos expansivos en un futuro, cuenta con un equipo multidisciplinario de profesionales altamente calificados en diversas materias, tales como:

- Instalaciones de redes GAS Natural y Licuado.
- Conversión de todo tipo de artefactos a GAS.
- Detección de fugas de GAS.
- Reparación de artefactos a GAS.
- Reparación de Fugas a GAS.
- Contratos anuales.
- Ampliaciones y Remodelaciones.
- Instalaciones Eléctricas.
- Instalaciones de Agua y Alcantarillado.

3.1.3 Flujo de trabajo de la empresa

Una vez concluido el análisis de la empresa, se pudo determinar el siguiente flujo de trabajo, identificando procesos principales y actores involucrados en cada etapa:

1. **Captura de Clientes:** a través de vendedores de la empresa los cuales reciben comisión por venta una vez concluida la obra, o bien, medios como la publicidad en internet, entre otras.
2. **Captura de Necesidades:** pueden determinarse a través de un informe de inspección entregado por la SEC, el cual detalla todas las observaciones encontradas en el edificio que impiden la obtención del sello verde, o bien, son los mismos clientes que necesitan un servicio otorgado por la empresa y se comunican con esta.
3. **Generación de presupuesto inicial:** una vez capturada estas necesidades, se crea un presupuesto inicial, que permite determinar la envergadura de la obra en un principio y obtener un precio aproximado para el cliente de manera rápida.
4. **Generación orden de visita técnica:** la orden de visita técnica busca corroborar en terreno los trabajos a realizar descritos en el presupuesto inicial, ya que en la mayoría de los casos los presupuestos sufren modificaciones debido al análisis técnico que se hace en el momento y que no se puede determinar en una primera etapa.
5. **Generación de presupuesto final:** una vez realizada la visita técnica, la empresa está en condiciones de generar la evaluación final de los trabajos a realizar en el inmueble, con lo que se procede a generar el presupuesto final para ser entregado al cliente vía correo electrónico o personalmente por personal de la empresa.

6. **Generación orden de trabajo:** si el cliente acepta el presupuesto ofrecido por la empresa, se procede a generar la orden de trabajo que especifica los datos del cliente, trabajos a realizar y personal técnico que debe asistir a la obra.

La figura 3.1, nos muestra el flujo actual de la empresa que comienza desde la captura de clientes hasta el cierre de una orden de trabajo.

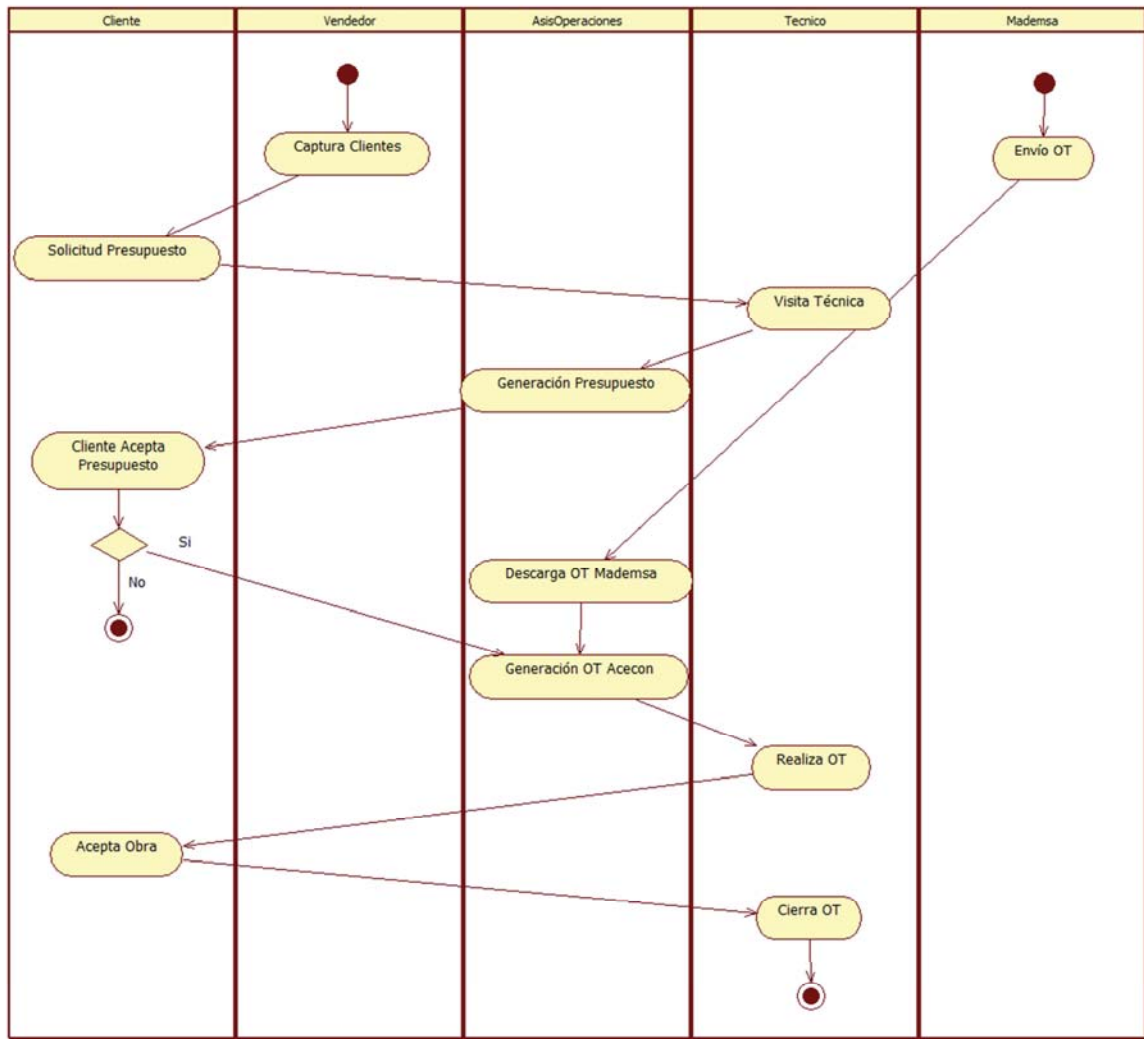


Figura 3.2. Flujo de Trabajo Acecon Ltda.

Una vez concluido el proceso de asignación de porcentajes, los documentos generados en todo el flujo de trabajo son archivados y guardados en dependencias de la empresa.

Con esta actividad concluye el flujo principal de trabajo, las otras actividades recaen en partes específicas del proceso, como consultas por datos de clientes, información sobre presupuestos realizados, cálculo de porcentaje de ventas obtenidas por los vendedores, entre otras tareas.

3.2 Problemas identificados

Durante el análisis realizado a la empresa, se pudo identificar una serie de problemas, que hacen que el flujo de trabajo de la compañía, sea lento y engorroso en muchas oportunidades.

A continuación se detallan los principales inconvenientes que tiene la compañía:

- **Gestión de Clientes:** En estos momentos toda la información reside en planillas que son archivadas y guardadas dentro de la empresa, lo que hace de la búsqueda y manejo de estos datos un proceso lento, engorroso y tedioso.
- **Gestión de Trabajadores:** Si bien el manejo de esta información (datos personales) en la actualidad no representa mayor problema, en un futuro plan de expansión de la empresa el control del personal debe ser primordial para el crecimiento de la compañía.
- **Gestión de Presupuestos:** Todos los presupuestos en la actualidad son generados a partir de una planilla Excel, lo que hace demasiado lento el proceso al tener que digitar toda la información, por otra parte al tener archivada esta información el acceso que se tiene a ella, en el caso de cualquier consulta del cliente deriva en una demora considerable, si se desea tener un servicio de calidad.
- **Manejo de Documentación Archivada:** El tener información importante de la compañía guardada en archivadores, hace primero que todo, que las búsquedas de ésta sea lenta y engorrosa, dando en muchas oportunidades un servicio ineficiente para el cliente en cuanto a tiempos de espera y seguridad de la información.
- **Lentitud al generar Presupuestos:** el proceso de designar un técnico para realizar la visita técnica antes de obtener un presupuesto final, hace lenta para el cliente la obtención del precio final que debe cancelar por los trabajos solicitados.

CAPÍTULO 4: ANÁLISIS DE TECNOLOGÍAS

4.1 Dispositivos móviles

La vida actual no se podría entender sin computadoras. Lo que se conoció como "revolución de la microinformática" a principios de los años 80, con la popularización de los primeros ordenadores domésticos, ha terminado por resultar en la universalización de los ordenadores como herramientas y como electrodomésticos de consumo.

Es tal la presencia de dispositivos electrónicos en nuestras vidas, que a veces no nos damos cuenta de la relación tan estrecha que tenemos con ellos. Tan estrecha como para llevar uno de ellos siempre en el bolso.

Porque los dispositivos móviles no son más que otro miembro más de la familia de los ordenadores. El hermano pequeño, cierto, pero un hermano más a fin de cuentas.

4.1.1 ¿Qué es un dispositivo móvil?

La línea entre lo que es un dispositivo móvil y lo que no lo es puede ser un poco difusa, pero en general, se pueden definir como aquellos micro-ordenadores que son lo suficientemente ligeros como para ser transportados por una persona, y que disponen de la capacidad de batería suficiente como para poder funcionar de forma autónoma.

Normalmente, son versiones limitadas en prestaciones, y por tanto en funcionalidades, de los ordenadores portátiles o de sobremesa. Por cierto, los ordenadores portátiles no se consideran como dispositivos móviles, ya que consumen más batería y suelen ser un poco más pesados de lo que se espera de algo pensado para llevar siempre encima.

4.1.2 Tipos de dispositivo móvil

A grandes rasgos, y dependiendo del tamaño los dispositivos se pueden dividir en tres clases:

1. **Teléfono Móvil**, es un dispositivo inalámbrico electrónico que permite tener acceso a la red de telefonía celular o móvil. Se denomina celular debido a las antenas repetidoras que conforman la red, cada una de las cuales es una célula, si bien existen redes telefónicas móviles satelitales. Su principal característica es su portabilidad, que permite comunicarse desde casi cualquier lugar. Aunque su principal función es la comunicación de voz, como el teléfono convencional, su rápido desarrollo ha incorporado otras funciones como son cámara fotográfica, agenda, acceso a Internet, reproducción de vídeo e incluso GPS y reproductor mp3.



Figura 4.1. iPHONE teléfono inteligente multimedia.

2. **PDA**s, organizadores electrónicos u ordenadores de mano. Su nombre (PDA) significa Personal Digital Assistant (asistente personal digital), un término acuñado en sus primeros años de historia, pero que resume bien su funcionalidad principal, que es servir como organizadores, con agenda, calendario, gestión de contactos, y que posteriormente han ido creciendo, de forma que actualmente sirven tanto como aparatos en los que se puede leer un libro, ver películas, crear documentos, jugar, ver correo electrónico, navegar por Internet, reproducir archivos de audio, etc. La línea que los separa de los teléfonos es cada vez más difusa debido a las funcionalidades que han ido adquiriendo los primeros.



Figura 4.2. PDA Acer Modelo N° 311.

3. **Consolas**, en realidad esta categoría debería llamarse “dispositivos orientados a jugar”, porque son más que simples consolas. Los dos ejemplos actualmente en el mercado son la Sony PlayStation Portable (PSP) y la Nintendo DS, que no sólo sirven para jugar, sino que integran algunas de las funcionalidades típicas de una PDA, como reproducción de archivos multimedia, integración con agenda y calendario, o navegador de Internet.



Figura 4.3. Consola PSP (Play Station Portable) de SONY.

4.1.3 Historia de los dispositivos móviles

El primer dispositivo móvil, para muchos, es la Newton, desarrollada y comercializada por Apple, y que estuvo a la venta entre 1993 y 1998. Era un dispositivo revolucionario para su tiempo, que implementaba un sistema de reconocimiento de escritura y que podía sincronizarse con un ordenador de sobremesa de la misma compañía. Fue tan revolucionaria, y se adelantó tanto a su tiempo, que fue un fracaso comercial, por lo que terminó retirándose del mercado, sin embargo mucha gente considera que la Newton fue muy influyente en el desarrollo de los famosísimos iPhone.

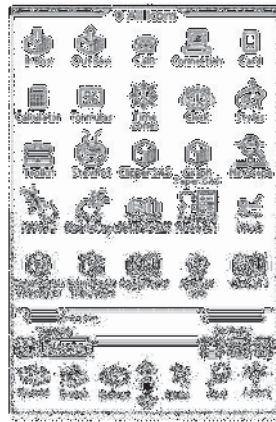


Figura 4.4. Panel De Control De La Newton.

Pero aunque la Newton pueda considerarse como la primera PDA, desde luego no fue el primer dispositivo portátil programable.

Durante los años 80, tanto Casio como Hewlett-Packard desarrollaron y comercializaron varias calculadoras programables, que si bien no tenían la capacidad de sincronizar sus datos con un ordenador de sobremesa, sí tenían capacidades gráficas, y accesorios como impresoras, o tarjetas de memoria extraíbles de 1Kb.

Un ejemplo la Casio FX-750p, que tenía pantalla de una sola línea, con 32 caracteres de 5x7 puntos, teclado QWERTY y teclado numérico, 4 Kb de RAM, dos ranuras para tarjetas de memoria extraíbles de hasta 2 Kb, y puerto serie por el cual se conectaba a una impresora, siendo una de las primeras calculadoras Casio programable en BASIC.

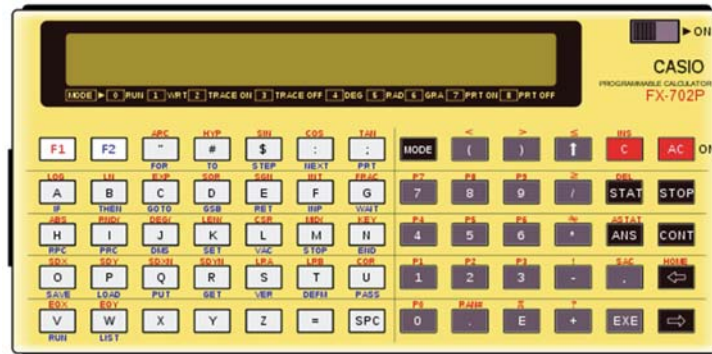


Figura 4.5. Calculadora Casio FX-750p.

Tras la muerte de la Newton, nacieron los dos dispositivos que durante unos años dominaron el mercado: la Pilot (primera generación de PDAs) y el PocketPC.

Los dispositivos de Palm se adelantaron a los de Microsoft en un par de años. En parte por ello, su salida a producción fue un éxito de ventas, llegando incluso a acumular, en 2001, un 60 por ciento del mercado americano. Sin embargo, Palm se resintió del declive global del mercado de las PDAs, comenzando un declive que coincidió prácticamente en el tiempo con la entrada en el mercado del gigante de la informática de consumo: Microsoft.



Figura 4.6. Pilot 5000.

En el año 2000 vio la luz el primer PocketPC, denominado hijo de la Newton, y que al contrario que ésta, sí ha sobrevivido hasta hoy, pese a no ser un éxito comercial de grandes dimensiones, gracias a la continuidad y el apoyo decidido de Microsoft.

Los primeros PocketPC tenían como sistema operativo el llamado Windows CE 3.0. Por su nombre podría parecer que era una versión aligerada del sistema operativo más utilizado en el mundo, pero en realidad no tenía mucho que ver con éste.

La mayor razón para el éxito del PocketPC ha sido su relativa sencillez de manejo y su integración con ordenadores de escritorio basados en Windows. La integración es tal que para realizar una sincronización entre ambos basta con conectar el PocketPC al ordenador con un cable.



Figura 4.7. PocketPC Hp.

Actualmente, los PocketPC y las Palm tienen pantallas de resolución VGA, en prácticamente todos los casos incorporan protocolos de comunicaciones inalámbricos, como Bluetooth o Wifi, o unidades de GPS.

Si la Newton, la Palm y el PocketPC han sido los que han abierto el camino en lo que se refiere a dispositivos móviles, los que realmente han entrado con toda la fuerza posible a ocupar ese mercado han sido los teléfonos móviles.

Actualmente, se calcula que en España hay más terminales móviles (unos 44 millones) que habitantes. En el 37% de los hogares hay un móvil, en el 34% hay dos terminales, y en el 17% hay tres, según datos de Cetelem de finales de 2005.

Pero ¿cuál es el motivo o los motivos para que haya tantos teléfonos? Es difícil de decir con total seguridad, pero se puede resumir principalmente en dos:

En primer lugar, el precio. Se pueden conseguir terminales por cero pesos en ocasiones excepcionales, al sacar un plan telefónico en cualquier compañía, pero en cualquier caso los precios de los teléfonos de gama baja suele ser bastante asequible para cualquier usuario.

Como segundo lugar es que cada vez los móviles pueden hacer más cosas, y eso crea una doble necesidad. Por un lado, es necesario tener un dispositivo que permita hacer ciertas cosas, como grabar vídeo, escuchar mp3, ver los goles de tu equipo favorito en tiempo real, etc. Pero eso también provoca que cada vez se intenten hacer más cosas con el teléfono, lo que vuelve a

hacer necesario un terminal más potente, que a su vez creara la necesidad de contenidos de más calidad.

Sin embargo, los móviles que verdaderamente pueden considerarse equivalentes a las PDAs son los de gama alta o Smartphone.

El término Smartphone es engañoso. Su traducción literal sería “teléfonos inteligentes”, y se utiliza indiscriminadamente para hacer referencia a cualquier teléfono de gama alta, englobando tanto a dispositivos de la Serie 60 o superior de Symbian, como a los que funcionan bajo Windows Mobile o bajo Palm OS. De todas formas, la marca comercial Smartphone es propiedad de Microsoft.

Porque, efectivamente, hay una línea de teléfonos que funcionan bajo Windows Mobile. Por un lado, los PocketPC Phone Edition, que son híbridos de teléfono y PocketPC y por otro los llamados comercialmente Smartphone.

También Palm mantiene productos en el mercado dentro de los teléfonos de gama alta: Un ejemplo, la serie Treo. Los Treo son híbridos de teléfono y PDA, con teclado QWERTY, y funcionan, salvo uno de los modelos (en concreto el Treo 700w, que utiliza PocketPC Phone Edition) bajo PalmOS.

Sin embargo, en los últimos años hemos sido testigos de la irrupción del potentísimo iPhone de Apple. Si Apple fue la que abrió el camino con la Newton, es el que ha vuelto a revolucionar el mercado con el iPhone, en parte porque ha sido el único fabricante que realmente ha conseguido desarrollar un dispositivo que integrara a todos los anteriores existentes por separado: teléfono, reproductor mp3 y PDA, reinventando la forma en la que se interactúa con él.

4.1.4 Plataforma y lenguajes soportados

Cada una de las plataformas tiene sus particularidades, no sólo en cuanto al manejo del dispositivo por el usuario, sino también a la hora de desarrollar aplicaciones para las mismas.

No será lo mismo programar una aplicación para Windows Mobile que para Symbian, entre otras cosas porque no todas las plataformas soportan los mismos lenguajes de programación.

- **WINDOWS MOBILE**

Actualmente llamado Windows Phone en su versión 7.0, es un sistema operativo móvil compacto desarrollado por Microsoft, y diseñado para su uso en teléfonos inteligentes (Smartphones) y otros dispositivos móviles [2].

Se basa en el núcleo del sistema operativo Windows CE y cuenta con un conjunto de aplicaciones básicas utilizando las API de Microsoft Windows, pretendiendo así ser vendido como una versión aligerada de Windows.

El paradigma de funcionamiento es similar, basado en ventanas, aunque éstas se comportan de forma muy diferente a como lo hacen en su hermano mayor de escritorio. Por ejemplo, al cerrar la ventana de un programa éste no se cierra realmente, sino que se sigue ejecutando como si se hubiera minimizado.

El punto fuerte de estos dispositivos es que ofrecen funcionalidades similares a las de sus hermanos mayores, Por ejemplo, se pueden editar documentos de Word, hojas de cálculo de Excel, leer libros en formato PDF, recibir y enviar correo electrónico, manejar una agenda, la libreta de contactos, sincronizar datos con el PC, navegar por internet, utilizar un GPS, en general casi lo mismo que en un PC, pero sin teclado y con la pantalla bastante más pequeña.

La mayoría de las aplicaciones para estos dispositivos se desarrollan en .Net, la plataforma de desarrollo de Microsoft, o directamente en C++, aunque aún quedan, algunas aplicaciones escritas en Embedded C++ o Embedded Basic, que eran dos entornos de desarrollo basados respectivamente en C++ y Basic, pero con muchas limitaciones.

Pero en la actualidad casi todos los desarrollos para Windows Mobile se realizan en C++ o en .Net, basándose en el Compact Framework, que es una versión del .Net Framework que está diseñado para funcionar en Windows CE basado en móviles/dispositivos embebidos tales como PDAs, teléfonos móviles, los controladores de fábrica, set-top boxes, etc.

El principal problema a la hora de desarrollar aplicaciones para estos dispositivos es el precio de las herramientas de desarrollo. Herramientas como Visual Studio, un entorno de desarrollo que soporta varios lenguajes como C#, C++, J#, JScript o ASP .Net son de altísima calidad, pero de un precio que no está al alcance de todos los que se quieran aventurar a escribir una aplicación para PocketPC.

Por otra parte, el hecho de que Microsoft esté detrás de la plataforma, dándole todo su apoyo, se nota en la cantidad y la calidad de la documentación disponible para los desarrolladores.

Tal vez por eso mismo, y pese al precio del entorno de desarrollo, la competencia es muy alta. Hay muchas compañías que producen aplicaciones para Windows Mobile, y que invierten mucho dinero en su desarrollo. No es, por tanto, un mercado en el que sea sencillo introducirse.

En cualquier caso, siempre que se vaya a desarrollar aplicaciones para esta plataforma, deben tenerse en cuenta las limitaciones específicas de la misma, sobre todo a la hora de intentar desarrollar interfaces que necesiten de la menor introducción de textos posible.

Una de las principales limitaciones de este sistema operativo es la memoria, suelen utilizar entre 64 y 256 Mb lo que requiere técnicas de programación eficientes. No hay distinción entre el almacenamiento de programas y memoria de la aplicación. Además, el sistema operativo es el que realiza la gestión de memoria en forma independiente de la aplicación ejecutada.

Otro tema importante referente a este tema es que las aplicaciones ejecutadas no son necesariamente eliminadas cuando el formulario se minimiza. Cabe destacar que el dispositivo de almacenamiento puede ampliar la capacidad de almacenamiento a través de tarjetas, sin embargo hay una reducción del rendimiento cuando se tiene acceso a datos dentro de esta en comparación con la memoria principal.



Figura 4.8. Emulador de Windows Mobile 6.2.

• SYMBIAN

Si los PocketPC están basados en un sistema operativo de Microsoft llamado Windows Mobile, la gran mayoría de los teléfonos móviles funcionan gracias a un sistema operativo llamado Symbian OS. El cual representa un consorcio en el que participan los mayores fabricantes de teléfonos móviles, con Nokia a la cabeza. Sony Ericsson, BenQ (que hace poco absorbió a Siemens), Fujitsu, Lenovo, Motorola, Panasonic, Samsung o Sharp están representados en Symbian, y son por tanto, partícipes en el desarrollo y la expansión del sistema.

De todos estos fabricantes, el que mayor cuota de mercado tiene es Nokia, que con algunas fluctuaciones, viene siendo desde hace años de alrededor del 40%, además es el fabricante que primero apostó por este sistema operativo, y el que más uso hace de él.

Symbian es un sistema operativo escrito en C++, por lo que presenta muy bajo consumo de recursos del dispositivo, a la vez que se ejecuta con gran rapidez. Se presenta en varias versiones, donde Nokia divide sus dispositivos Symbian en cuatro familias, que se llaman respectivamente Serie 40, Serie 60, Serie 80 y actualmente la última Serie 90.

La Serie 40 es la que agrupa a los teléfonos Symbian con pantallas más pequeñas (en general, hasta de 240x320 píxeles), y tiene ciertas limitaciones sobre las otras series, sobre todo en lo referente a la cantidad de recursos del teléfono que puede poner a disposición de las aplicaciones que se ejecuten sobre él.

La Serie 60 es la más extendida, y es donde suelen estar los teléfonos de gama media y alta de Nokia, a los que muchas veces se hace referencia como teléfonos multimedia. Con alguna excepción, suelen tener pantallas más grandes que los de la serie 40, y suelen ser teléfonos específicamente optimizados para ejecutar aplicaciones J2ME.

La Serie 80, es la de los llamados Communicators, esos teléfonos que se abren longitudinalmente y que esconden teclados QWERTY. En realidad son un paso intermedio entre el teléfono y la PDA.

Finalmente la Serie 90, que representa principalmente a los Smartphone los últimos equipos que está sacando Nokia al mercado.

Aparte de la implementación que hace Nokia de Symbian, Sony Ericsson implementa otra familia más de ese sistema, llamado UIQ, que se maneja por teclado o a través de una pantalla táctil. Estos dispositivos suelen implementar sistemas de reconocimiento de escritura, y los más conocidos son, la serie P de Sony Ericsson.

Actualmente, en el mercado de los teléfonos móviles Symbian es el sistema más extendido, aunque no el único. La mayoría de los fabricantes siguen implementando sus propios sistemas, por lo que sigue sin haber una uniformidad o coherencia entre dispositivos.



Figura 4.9. Sistema Operativo Symbian, Serie 60.

Las aplicaciones compatibles con Symbian se desarrollan a partir de lenguajes de programación orientados a objetos como C++, Java (con sus variantes como PJava, J2ME, etc.), Visual Basic para dispositivos móviles, entre otros, incluyendo algunos lenguajes disponibles en versión libre. La extensa variedad de aplicaciones que se pueden desarrollar van desde administradores de archivos hasta visualizadores de películas, guías de ciudades, mapas, diccionarios, emuladores de juegos, por mencionar algunas.

- **iPHONE OS**

El iPHONE, el último en llegar, rompe también con las líneas maestras en cuanto a desarrollo se refiere, marcadas por sus antecesores.

El sistema operativo utilizado es una versión aligerada de Mac OS X, el sistema detrás de los ordenadores de Apple, y por tanto, las herramientas de desarrollo que se deben utilizar son las mismas que para trabajar en escritorio.

El iPHONE y el iPod touch oficialmente no pueden instalar programas hechos por alguien que no sea Apple. Sin embargo, muchas de las aplicaciones hechas por terceros ahora están disponibles, a pesar de la carencia de apoyo de Apple. Tales aplicaciones se exponen a la posibilidad de que sean estropeadas por cualquier actualización del iPHONE OS, aunque Apple haya declarado que no se diseñaran actualizaciones de software para estropear dichas aplicaciones. Cydia es el mejor ejemplo de instaladores de programas de terceros, para poder incluir este instalador en el iPHONE tienes que hacerle el jailbreak, proceso que permite a los usuarios de los dispositivos iPHONE, iPod Touch e iPad ejecutar código no oficial en éstos dispositivos traspasando el mecanismo de distribución oficial de Apple.

El lenguaje que debe utilizarse para desarrollar aplicaciones nativas para iPHONE OS es Objective-C, un superset de C (una especie de C enriquecido), de forma que el desarrollador se apoya en un extenso framework orientado a objetos, escalable y altamente modular, llamado Cocoa.

En realidad, Cocoa es una colección de frameworks, que proporcionan todas las piezas necesarias para construir una aplicación: desde elementos de interfaz hasta gestión de tráfico de red.

Pese a lo modular y extenso de los frameworks, el mayor problema con el que se encuentran los recién llegados a la plataforma es que la curva de aprendizaje de Objective-C es bastante acusada. Especialmente si, además, se viene de un entorno de desarrollo Windows, ya que no sólo hay que cambiar de lenguaje, sino de forma de desarrollo por completo.

Además, para poder desarrollar aplicaciones para iPHONE es necesario estar registrado (previo pago) como desarrollador con Apple, para así poder obtener los certificados digitales necesarios para que las aplicaciones funcionen en los dispositivos. Por si fuera poco, la única forma de poner esas aplicaciones en esos dispositivos, es a través de una tienda de aplicaciones controlada por Apple.



Figura 4.10. Interfaz iPHONE.

• ANDROID

Android es un sistema operativo para teléfonos, basado en el núcleo de Linux, y que, aunque disponible para cualquier fabricante como open-source, actualmente es el motor de los dispositivos comercializados por Google [3].

Esta plataforma permite el desarrollo de aplicaciones por terceros a través del SDK, proporcionada por el mismo Google, y mediante el lenguaje de programación Java. Una alternativa es el uso del NDK (Native Development Kit) de Google para emplear el lenguaje de programación C.

El código fuente de Android está disponible bajo diversas licencias de software libre y código abierto destacando la versión 2 de la licencia Apache. Algunas características de Android:

- Framework de aplicaciones: permite reutilización y reemplazo de componentes.
- Máquina virtual Dalvik: optimizada para dispositivos móviles.
- Navegador integrado: basado en el motor de código abierto WebKit.
- Gráficos optimizados, con una biblioteca de gráficos 2D; gráficos 3D basado en la especificación OpenGL ES 1.0 (aceleración por hardware opcional).
- SQLite para almacenamiento de datos estructurados.
- Soporte para medios con formatos comunes de audio, vídeo e imágenes planas (MPEG4, H.264, MP3, OGG, AAC, AMR, JPG, PNG, GIF)
- Telefonía GSM (dependiente del hardware)
- Bluetooth, EDGE, 3G, y Wi-Fi (dependiente del hardware)
- Cámara, GPS, brújula, y acelerómetro (dependiente del hardware)
- Ambiente rico de desarrollo incluyendo un emulador de dispositivo, herramientas para depurar, perfiles de memoria y rendimiento, y un complemento para el IDE Eclipse.
- Pantalla táctil.



Figura 4.11. Sistema Operativo Android.

4.1.5 Herramientas de desarrollo

Las herramientas de desarrollo a utilizar dependerán única y exclusivamente de la plataforma en que se haya diseñado la aplicación.

Si la aplicación debe funcionar en dispositivos Windows Mobile, se recomienda utilizar el Visual Studio de Microsoft, un entorno de desarrollo bastante caro, pero que es casi sin discusión el mejor de todos los disponibles.

Se Deberá basar la aplicación, en código C++ nativo, necesitando conocimientos técnicos bastante elevados antes de comenzar el desarrollo, o bien utilizar .Net Compact Framework, que es un subconjunto del .Net Framework, que permite programar en C#.

Para dispositivos Symbian, la opción más extendida es, J2ME. Aunque las herramientas de desarrollo no están tan refinadas como las de Windows Mobile, aun así, hay un entorno específico como el NetBeans Mobility Pack, además de plugins y add-ons para los IDEs de java más populares: Eclipse e IDEA.

Los tres entornos, una vez instalados los plugin correspondientes, permiten desarrollar, empaquetar y testear aplicaciones J2ME en los emuladores incorporados. Pero nunca se debe olvidar que los emuladores no son el dispositivo real, sino máquinas mucho más potentes que el dispositivo real. Por tanto, lo que en un emulador funciona bien, puede que en el dispositivo no funcione tan bien.

Por eso, siempre que se desarrolle para un dispositivo móvil, pero sobre todo si se está haciendo en J2ME o Flash Lite, hay que probar los programas en el dispositivo según se están desarrollando. Y cuanto antes se empiece a probar, mejor, de esa forma los posibles problemas aflorarán antes, y serán más fáciles de resolver.

En el caso del iPHONE será necesario un Mac con procesador Intel, ya que no existe SDK para Windows ni para Linux, aprender un lenguaje y una metodología de desarrollo completamente nueva que al día de hoy no está excesivamente documentada.

Para poder incluir nuevas funcionalidades y aplicaciones, es importante que la plataforma admita desarrollo de terceros. A continuación se muestra en la Tabla 4.1 una comparativa que muestra la disponibilidad que tienen los sistemas operativos para acceder tanto como al SDK y Aplicaciones Nativas, entre otras características.

	Android	iPHONE OS	Symbian S60	Windows Mobile 6.5
Disponibilidad Del SDK	Si	Si	Si	Si
Tienda de aplicaciones	Si	Si	Próximamente	Si
Disponibilidad de aplicaciones	Alta	Alta	Mediana	Alta
Aplicaciones nativas	Si	Si	Si	Si
Administración local de aplicaciones	Excelente	Excelente	Buena	Buena

Tabla 4.1. Comparativa Sistemas Operativos Móviles.

4.1.6 Limitaciones sobre el desarrollo

Hay dos limitaciones fundamentales que se debe tener en cuenta a la hora de desarrollar un aplicación para ser ejecutada en un dispositivo móvil, y son las particularidades de hardware y las de la conexión.

Aunque los dispositivos cada vez tienen más memoria y más capacidad de proceso, siguen siendo hermanos pequeños de los ordenadores de escritorio. Por tanto, debe tenerse especial cuidado con no realizar demasiadas animaciones, ni gastar ciclos de proceso en procedimientos que no sean estrictamente necesarios. Las pantallas son relativamente pequeñas y suelen ser miradas en movimiento.

Por ello la entrada de datos debe ser lo más sencilla posible. En el caso de las PDAs, se realiza escribiendo en la pantalla con un lápiz, para que sea el dispositivo el que realice el reconocimiento de la escritura. Aunque ya no hace falta escribir de forma especial, como ocurría con el sistema de reconocimiento de escritura de las primeras Pilot, llamado Graffiti.

Además, no hay que olvidar que los interfaces de los distintos dispositivos son heterogéneos, donde ni el tamaño de las pantallas, ni el tamaño, ni la ubicación de las teclas ayudan a la hora de introducir datos por el usuario. Por eso, siempre hay que intentar desarrollar los interfaces de las aplicaciones de forma que no sean necesarias muchas pulsaciones de teclas, e intentar que, cuando sea necesario pulsar alguna tecla sea alguna de las softkeys, o teclas especiales, preocupándose de organizar la información que se desea presentar.

La premisa es bastante sencilla: dadas las dificultades para la introducción de datos, hay que intentar simplificarla al máximo. Para ello, es muy importante organizar el interfaz de la aplicación de forma que los procesos de entrada de datos sean lo más naturales posibles, y que el usuario nunca pase por pantallas cuya presentación no sea estrictamente necesaria. Pero en realidad esas precauciones no hay que tomarlas sólo a la hora de introducir datos, sino también a la hora de presentar información, y a la hora de estructurar la navegación por la aplicación.

Normalmente es mucho más efectivo presentar la información agrupada de forma jerárquica, para de ese modo discriminar cuanto antes qué es lo que va a necesitar ver el usuario, y no mostrarle nada que no le resulte estrictamente necesario.

En cuanto a las conexiones, lo primero que hay que tener siempre en cuenta es que, al contrario que en las aplicaciones web estándar, no están siempre disponibles. Por tanto, nunca se puede suponer que el dato que es necesario para poder seguir adelante en la ejecución de la aplicación, va a poderse obtener a través de una conexión a Internet.

Además, debe tenerse en cuenta que antes de realizar una conexión a cualquier servicio, el dispositivo va a solicitar confirmación al usuario, y que éste puede denegarla, lo cual es perfectamente posible, dado el precio de las conexiones, que son facturables por peso, por cantidad de bytes enviados y descargados, no por tiempo, por lo que siempre hay que optimizar todo lo posible lo que se vaya a enviar en uno u otro sentido.

Por si no fueran dificultades suficientes, la latencia de la red, es decir, el tiempo de espera entre una petición de datos y el comienzo de la llegada de la respuesta, es mucho mayor que la de cualquier aplicación web normal. Por tanto, hay que ser muy cuidadosos a la hora de informar al usuario sobre lo que está ocurriendo en todo momento, si se está en espera de datos, si ya se están recibiendo, dando siempre, si es posible, estimaciones sobre el tiempo que resta para la terminación de todas las tareas.

Tampoco se puede suponer que la ejecución de la aplicación sea lineal y completa. La función básica de muchos dispositivos es emitir y recibir llamadas telefónicas. Por tanto, nuestra aplicación debe ser capaz de comportarse adecuadamente al recibir una llamada, entre otras cosas porque la función de teléfono tiene asignada la prioridad más alta de todas las del dispositivo, es decir, se va a parar la ejecución de cualquier aplicación cuando entre una llamada nueva. Se debe tener prevista una forma de congelar el estado de la aplicación justo antes de recibir la llamada, para que cuando ésta termine, y tras la confirmación del usuario, se pueda volver a ese estado. Por tanto, si es un juego, todos los elementos del mismo deben permanecer estáticos hasta que la llamada termine y el usuario confirme que desea continuar con la aplicación.

En general, hay que intentar probar las aplicaciones en el dispositivo para el que se vayan a desarrollar cuanto antes. De hecho, hay que probar mucho, y comenzar a probar lo antes posible. Nunca hay que fiarse de emuladores, entre otras cosas, porque se ejecutan en hardware cuya capacidad de procesamiento es superior a la del dispositivo en varios órdenes de magnitud.

Por tanto, al desarrollar para dispositivos móviles, es muy importante comenzar a probar pronto, y probar a menudo, a fin de evitar sorpresas desagradables.

4.2 Servicios Web

"Los web services son componentes software que permiten a los usuarios usar aplicaciones de negocio que comparten datos con otros programas modulares, vía Internet. Son aplicaciones independientes de la plataforma que pueden ser fácilmente publicadas, localizadas e invocadas mediante protocolos web estándar, como XML, SOAP, UDDI o WSDL. El objetivo final es la creación de un directorio online de web services, que pueda ser localizado de un modo sencillo y que tenga una alta fiabilidad"[4].

Un servicio web es un componente de software que se comunica con otras aplicaciones codificando los mensaje en XML y enviando estos mensaje a través de protocolos estándares de Internet tales como el Hypertext Transfer Protocol (HTTP). Intuitivamente un servicio web 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 servicio web, 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 servicios web. 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. Microsoft y otros proveedores líderes promocionan los servicios web como un modelo de programación para la comunicación entre aplicaciones. Estas compañías piensan que 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 servicio web 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. Por ejemplo, varias compañías están hoy en día creando servicios web que actúan como front end para aplicaciones de entrada de órdenes que están residentes internamente en un mainframe. Estas compañías permiten a los sistemas de compras de sus clientes enviar órdenes de compra a través de la Internet. Poner una capa de servicio web sobre las aplicaciones existentes es una solución muy interesante para integrar las aplicaciones desarrolladas por los diferentes departamentos y así reducir los costos de integración de las mismas.

4.2.1 Arquitectura de los Servicios Web

El conjunto de servicios y protocolos para los servicios web es conocido comúnmente como “Web Services Protocol Stack” y básicamente son utilizados para definir, localizar, implementar y hacer que un servicio web interactúe con otro. Este conjunto está conformado esencialmente de cuatro subconjuntos [6]:

- Servicio de transporte
- Mensajería XML
- Descripción del servicio
- Descubrimiento de Servicios

4.2.1.1 Servicio de Transporte

Es el encargado del transporte de los mensajes entre aplicaciones sobre la red. Incluye varios protocolos del nivel de aplicación. A continuación se relata sobre los más utilizados.

- *HTTP (HyperText Transfer Protocol)*: Protocolo del nivel de aplicación más utilizado en la Internet. Es el protocolo que define la sintaxis y la semántica utilizada para la arquitectura web. En el contexto de los servicios web es utilizado para la transferencia de las transacciones XML a través de la red utilizando los mismos principios del HTML.
- *FTP (File Transfer Protocol)*: Es un protocolo de la capa de aplicación encargado de los servicios de transmisión de archivos a través de redes soportadas sobre TCP. En el ámbito de los servicios web el FTP permite realizar modificaciones en equipos remotos evitando el uso de permisos sobre los archivos en la máquina cliente en sistemas operativos diferentes a Windows.
- *SMTP (Simple Mail Transfer Protocol)*: Es un estándar de la capa de aplicación ampliamente utilizado para el envío de mensajes de correo electrónico a través de Internet. Es un estándar de Facto basado en texto, que requiere como cliente software de tipo POP3 o IMAP.
- *BEEP (Block Exensible Exchange Protocol)*: Es un protocolo del nivel de aplicación, también conocido como BXXP, está diseñado para la interacción asíncrona punto a punto sobre una red TCP/IP Fue estandarizado por el IETF y provee un marco para administrar las conexiones punto a punto, autenticación., transporte de mensajes y manejo de errores.
- *JMS (Java Message Service)*: Es una aplicación de interface de programación para JAVA (API) para el envío de mensajes entre dos o más clientes. Soporta dos modelos el modelo punto a punto y el modelo de publicación y suscripción. Una aplicación JMS está compuesta por las siguientes partes:

1. Un proveedor JMS que implementa las interfaces que proveen las características de administración y el control.
2. Clientes JMS que son los componentes escritos en JAVA que producen y consumen los mensajes.
3. Los Mensajes que son los objetos dato entre los clientes JMS.
4. Objetos administradores que son objetos configurados previamente por un administrador del sistema para el uso de los clientes.
5. Clientes Nativos que son programas que utilizan los mensajes de la API de manera similar que la API JMS.

4.2.1.2 Mensajería XML

Es el conjunto encargado de la codificación de los mensajes en XML estándar y pueda así ser interpretado en cualquiera de los nodos de la red. Los componentes más utilizados en este conjunto son los siguientes:

- *REST (Representational State Transfer)*: Fielding da la siguiente definición: “estilo de arquitectura de software para sistemas hipermedias distribuidos tales como la World Wide Web”. En resumen, es un conjunto de principios para el diseño de redes, que es utilizado comúnmente para definir una interfaz de transmisión sobre HTTP de manera análoga a como lo hace SOAP. Aunque REST como tal no es un estándar, posee un conjunto de estándares tales como HTML, URL, XML, GIF, JPG y tipos MIME. Los principios de REST son:
 - Escalabilidad de la interoperabilidad con los componentes.
 - Generalidad de Interfaces.
 - Puesta en funcionamiento independiente.
 - Compatibilidad con componentes intermedios.
- *RPC (Remote Procedure Calls)*: Es una tecnología de software que permite ejecutar una rutina en un equipo o segmento de red de manera remota. Es un paradigma popular para la implementación de sistemas distribuidos bajo arquitecturas cliente servidor.
- *XML-RPC*: Es un protocolo de llamada remota que utiliza XML como lenguaje de codificación y HTTP como mecanismo de transporte. Es un protocolo sencillo ya que solo define algunos tipos de datos y comandos.
- *XML (eXtended Markup Language)*: XML es uno de los lenguajes más utilizados para el intercambio de datos sobre la web. Su desarrollo se remonta en el año 1996 por el grupo de trabajo de la World Wide Web Consortium lanzando su primera versión el 10 de Febrero de 1998. El lenguaje XML está concebido para describir objetos de datos llamados Documentos XML y describir de cierta forma los programas que los procesan. Está restringido bajo la norma ISO 8879 el Estándar Generalized Markup Language. Un documento XML es un objeto de datos que está bien formado, y se dice que lo está cuando tomado en su conjunto coincide con la producción del documento etiquetado, reúne todas las especificaciones de formato definidas y cada una de las

entidades que se llaman directa o indirectamente están también bien definidas. El XML es un lenguaje etiquetado, característica que le permite definir objetos de datos estructurados en partes bien definidas llamadas elementos. Una etiqueta es una señal realizada dentro del documento XML que delimita un segmento definido y con sentido de este documento.

Adjunto al documento XML existe una definición de tipo de documento (DTD) en donde se describe la estructura y la definición de los datos de un documento XML. Una DTD describe normalmente los elementos, que son los segmentos etiquetados, la estructura, que es el orden en el cual van los elementos y el nivel de anidamiento.

- *SOAP (Simple Object Access Protocol)*: SOAP es un protocolo de la capa de aplicación para el intercambio de mensajes basados en XML sobre redes de computadores. Básicamente es una vía de transmisión entre un SOAP Sender y un SOAP Receiver, pero los mensajes SOAP deben interactuar con un conjunto de aplicaciones para que se pueda generar un “dialogo” a través de mensajes SOAP. Un mensaje SOAP es la unidad fundamental de una comunicación entre nodos SOAP. SOAP es básicamente un paradigma de una sola vía pero con la ayuda de las aplicaciones se puede llegar a crear patrones más complejos. SOAP básicamente está constituido por:
 - Un marco que describe el contenido del mensaje e instrucciones de proceso.
 - Un conjunto de reglas para representar los tipos de datos definidos.
 - Convenciones para representar llamadas a procedimientos remotos y respuestas.
 - Algunos lineamientos entre SOAP y HTTP.

4.2.1.3 Descripción del Servicio

El servicio web debe contar con una interfaz pública la cual es descrita por un formato llamado WSDL (Web Services Descripción Languages).

- *WSDL (Web Services Description Language)*: WSDL es un tipo de documento XML que describe lo que hace un servicio web, donde se encuentra y la forma de ser invocado. Este provee información muy importante para los desarrolladores, este lenguaje describe el formato de los mensajes que utiliza y a cuales puede responder. Siempre un documento XML WSDL presenta los siguientes elementos:
 - Tipos: Tipos de datos usados por los mensajes.
 - Mensaje: Que datos son enviados desde un nodo a otro.
 - Tipo de puerto: Define las operaciones que pueden ser llamadas.
 - Operación: Define la configuración de mensajes de entrada, salida y error.
 - Entrada: Mensaje que es enviado hacia el servidor.
 - Salida: Mensaje enviado hacia el cliente.
 - Falta: Error en el envío de un mensaje.
 - Límite: Es la descripción del protocolo que se está utilizando para transportar el mensaje que puede ser HTTP POST, HTTP GET, SOAP y MIME.

- Servicio: Define una colección de puertos (nodos); el puerto especifica una dirección para el límite definiendo así la comunicación para un nodo específico.

4.2.1.4 Descubrimiento de Servicios

UDDI (Universal Description Discovery and Integration): UDDI es un marco independiente de la plataforma para describir servicios, negocios e integrar servicios de negocios. La estructura de UDDI está basada sobre los servicios estándares de la web, lo que quiere decir que UDDI es accesible como otros servicios web. UDDI es un esfuerzo de la industria iniciada en Septiembre de 2000 por Ariva, IBM, Microsoft y otras 33 compañías. Los propietarios de los Servicios Web los publican en el registro UDDI. Una vez publicados se mantienen allí apuntadores a la descripción del Servicio Web y al servicio. UDDI permite a los clientes buscar tal registro, encontrar el servicio deseado y extraer sus detalles. Estos detalles incluyen el punto de invocación así como otras características del servicio y su funcionalidad. La estructura de datos con UDDI está compuesta en cuatro partes:

- **businessEntity:** Describe al proveedor del servicio web. Tiene datos como nombre de compañía, detalle de contacto y otra información del negocio.
- **businessService:** Describe un conjunto lógico de uno o muchos servicios web.
- **bindingTemplate:** Describe un único Servicio Web, describe toda la información técnica para que el cliente pueda interactuar con él.
- **tModel:** Representa especificaciones técnicas, metadatos sobre las especificaciones del documento, el nombre puntero URL, es presentado en forma de un documento WSDL.

4.2.2 Nuevos Protocolos para Servicios Web

Algunas especificaciones adicionales han sido desarrolladas o están empezando a ser desarrolladas con el ánimo de extender las capacidades de los servicios web. De manera habitual estas especificaciones son nombradas como ws-?. Algunas de las más importantes son:

- **WS-Security:**

Es un protocolo de comunicaciones encargado de proveer seguridad a las aplicaciones de Servicios Web. Fue desarrollado originalmente por Microsoft, IBM, Verisign y Forum Systems, ahora el protocolo es llamado WSS. WSS ofrece mejoras en el sistema de mensajería SOAP para proveer calidad en la protección a través de la integridad de mensajes, confidencialidad y autenticación. Define como usar encriptación XML y firma XML en SOAP, es una alternativa diferente a HTTPS.

- **WS-Reliability:**

Es un protocolo basado en SOAP para el intercambio de mensajes con distribución garantizada, sin duplicados y garantizando el orden del mensaje.

- **WS-ReliabilityMessaging:**

Esta especificación describe un protocolo que permite enviar mensajes SOAP de manera confiable entre sistemas distribuidos en presencia de fallas de los sistemas, componentes o aplicaciones. El objetivo de esta especificación es asegurar que el mensaje enviado por el emisor sea recibido por el receptor. La confiabilidad en los Servicios Web es algo difícil de definir, pero se puede realizar un símil de WSRM para XML como JMS para Java.

- **WS-Addressing:**

Esta es una especificación de mecanismos de transporte que permite a los Servicios Web comunicar información direccionada. Tiene una estructura compuesta en principio por dos partes. La primera, es una estructura para comunicar una referencia al nodo final del servicio web, y la segunda, es un conjunto de propiedades de direccionamiento con las cuales se asocia la información direccionada con un mensaje en particular. Las propiedades de direccionamiento son:

- Destinación de mensaje URI
- Origen del nodo final.
- Reenvío de nodo final.
- Falla del nodo final.
- Acción.
- Identificador único del mensaje.
- Relación con mensajes previos.

- **WS-Transaction:**

Es una especificación desarrollada inicialmente por Microsoft, IBM y BEA Systems. Esta describe tipos de coordinación que son usadas con el marco extensible de coordinación descrito en la especificación WS-Coordination.

- **WS-Coordination:**

Es una especificación que describe un marco extensible para proveer protocolos que coordinen las acciones de aplicaciones distribuidas. El marco definido en tal especificación habilita un servicio de aplicación para crear un contexto necesario para propagar una actividad a otros servicios y registrarlos a protocolos de coordinación.

4.2.3 Ventajas de los servicios web

El principal objetivo que se logra, es la interoperabilidad y la integración. Mediante los servicios web, las empresas pueden compartir servicios software con sus clientes y sus socios de negocio. Esto ayudará a las compañías a escalar sus negocios, reduciendo el coste en desarrollo y mantenimiento de software, y sacando los productos al mercado con mayor rapidez. La integración de aplicaciones hará posible obtener la información demandada en

tiempo real, acelerando el proceso de toma de decisiones. La evolución de Internet hacia los servicios web, mejorará los resultados globales de las empresas, reduciendo sus gastos y guiándolas hacia una mejora progresiva de la calidad. La adopción de la tecnología de servicios web por la industria es el primer paso hacia una economía global.

- Aportan interoperabilidad entre aplicaciones de software independiente de sus propiedades o de las plataformas sobre las que se instalen, ya que posee un lenguaje propio de comunicación el cual no depende de los lenguajes de los dispositivos a los cuales conecta.
- Los servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento.
- Al apoyarse en HTTP, los servicios Web pueden aprovecharse de los sistemas de seguridad firewall sin necesidad de cambiar las reglas de filtrado, ya que el puerto 80 (HTTP) en la mayoría de los casos no posee bloqueo ya que es el puerto de navegabilidad de internet.
- Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.
- Permiten la interoperabilidad entre plataformas de distintos fabricantes por medio de protocolos estándar y abiertos. Las especificaciones son gestionadas por una organización abierta, la W3C, por tanto no hay secretismos por intereses particulares de fabricantes concretos y se garantiza la plena interoperabilidad entre aplicaciones.
- Bajos costos de comunicación, Servicio Web usa SOAP sobre el protocolo HTTP para las comunicaciones, así se puede utilizar Internet de bajo costo para la aplicación de Servicios Web. Esta solución es mucho menos costosa en comparación con soluciones como EDI/B2B. frente a SOAP sobre HTTP, Servicios Web también se puede implementar en otros mecanismos de transporte fiable como FTP.

4.2.4 Inconvenientes de los servicios web

- Para realizar transacciones no pueden compararse en su grado de desarrollo con los estándares abiertos de computación distribuida como CORBA (Common Object Request Broker Architecture).
- Su rendimiento es bajo si se compara con otros modelos de computación distribuida, tales como RMI (Remote Method Invocation), CORBA o DCOM (Distributed Component Object Model). Es uno de los inconvenientes derivados de adoptar un formato basado en texto. Y es que entre los objetivos de XML no se encuentra la concisión ni la eficacia de procesamiento.
- Al apoyarse en HTTP, pueden esquivar medidas de seguridad basadas en firewall cuyas reglas tratan de bloquear o auditar la comunicación entre programas a ambos lados de la barrera.

4.2.5 Razones para crear servicios web

La principal razón para usar servicios Web es que se basan en HTTP sobre TCP (Transmission Control Protocol) en el puerto 80. Dado que las organizaciones protegen sus

redes mediante firewalls -que filtran y bloquean gran parte del tráfico de Internet-, cierran casi todos los puertos TCP salvo el 80, que es, precisamente, el que usan los navegadores. Los servicios Web utilizan este puerto, por la simple razón de que no resultan bloqueados.

Otra razón es que, antes de que existiera SOAP, no había buenas interfaces para acceder a las funcionalidades de otros ordenadores en red. Las que había eran ad hoc y poco conocidas, tales como EDI (Electronic Data Interchange), RPC (Remote Procedure Call), u otras APIs.

Una tercera razón por la que los servicios Web son muy prácticos es que pueden aportar gran independencia entre la aplicación que usa el servicio Web y el propio servicio. De esta forma, los cambios a lo largo del tiempo en uno no deben afectar al otro. Esta flexibilidad será cada vez más importante, dado que la tendencia a construir grandes aplicaciones a partir de componentes distribuidos más pequeños es cada día más utilizada.

4.2.6 Plataformas de Servicios web

Tanto J2EE de Sun y Microsoft. NET proporcionan mecanismos de ejecución que aíslan a los desarrolladores de software de las dependencias en particular. Además de un servicio Web XML orientado a la capa de direccionamiento indirecto entre plataformas, lenguajes y arquitecturas de empresa, Sun J2EE y. NET ofrecen un nivel de intermediación a través del Java Runtime Environment (JRE) y el Common Language Runtime (CLR), respectivamente. Es por esto que se pasará a explicar las características de cada uno de ellos para poder tener una mejor postura al momento de elegir entre una u la otra.

Las aplicaciones J2EE están alojadas dentro de un contenedor, que proporciona cualidades de servicio necesario para las aplicaciones empresariales, tales como las transacciones, la seguridad, y servicios de persistencia. La capa de negocio realiza el procesamiento de negocio y la lógica de datos. En aplicaciones a gran escala de J2EE, la lógica de negocio se ha construido utilizando componentes de Enterprise JavaBeans (EJB). Esta capa realiza el procesamiento de negocio y la lógica de datos. Se conecta a bases de datos usando Java Database Connectivity (JDBC) o SQL / o los sistemas existentes utilizando la Java Connector Architecture (JCA). También puede usar tecnologías de servicios web (SOAP, UDDI, WSDL, ebXML) a través de la API Java para XML (JAX las API). Se pueden comunicar con las aplicaciones J2EE a través de tecnologías de servicios web (SOAP, UDDI, WSDL, ebXML). Un servlet, que es una petición / respuesta orientada a objetos de Java, puede aceptar las solicitudes de servicios web. El servlet utiliza las API de JAX para realizar servicios web.

En cambio. NET está alojado dentro de un contenedor, que proporciona cualidades de servicio necesario para las aplicaciones empresariales, tales como las transacciones, la seguridad, y servicios de mensajería. La capa de negocio de la aplicación. NET está construido utilizando. NET componentes administrados. Esta capa realiza el procesamiento de negocio y la lógica de datos. Se conecta a bases de datos con Active Data Objects (ADO.NET) y los sistemas existentes utilizando los servicios proporcionados por Microsoft Host Integration Server 2000, como el Integrador de transacciones COM (COM TI). También se puede

conectar a socios de empresas que utilicen tecnologías de servicios web (SOAP, UDDI, WSDL). Los socios comerciales pueden contactar con la aplicación .NET a través de tecnologías de servicios web (SOAP, UDDI, WSDL, BizTalk).

4.2.6.1 Servicios web en .NET

.NET es la plataforma de Microsoft para servicios Web XML. Es la siguiente generación de software que conecta información, dispositivos y personas de una manera unificada y personalizada. La Plataforma .NET permite la creación y uso de servicios de aplicaciones, procesos y sitios Web basados en XML, que compartan y combinen información y funcionalidad por su diseño, en cualquier plataforma o dispositivo inteligente. La Plataforma .NET incluye una familia de productos integral, construida en estándares de la industria y de Internet, que provee servicios Web XML para cada aspecto del desarrollo, administración y uso. Algunas herramientas de desarrollo de esta aplicación son:

- El modelo de programación .NET Framework, que permite a los desarrolladores crear aplicaciones Web, aplicaciones de cliente inteligente y servicios Web XML que exponen su funcionalidad mediante programación a través de una red utilizando protocolos estándar como SOAP, XML (Lenguaje de marcado extensible) y HTTP.
- Herramientas de desarrollo, como Visual Studio® .NET, que proporcionan un entorno de programación integrado de gran rapidez para la programación con .NET Framework.
- Un conjunto de servidores, incluidos Windows® 2000, SQL Server™
- BizTalk™ Server, que integra, ejecuta y administra aplicaciones y servicios Web XML.
- Software de cliente, como Windows XP, Microsoft Office XP, que permite a los desarrolladores ofrecer una experiencia de usuario convincente e intensa a través de una familia de dispositivos y productos existentes.

Desde un punto de vista lógico, se puede pensar en un Servicio Web como en un componente, o caja negra, que suministra algún servicio útil a los clientes, o consumidores. Un consumidor accede a un Servicio Web usando un protocolo estándar, aceptado, y bien conocido –HTTP – y un formato de datos basado en XML.

Un Servicio Web puede implementarse usando una gran variedad de lenguajes. En la actualidad, C++, Jscript, C#, y VB.NET están soportados, pero probablemente habrá muchos otros en el futuro. En lo que se refiere al consumidor, el lenguaje usado por el Servicio Web, es intrascendente. El punto de vista del consumidor del Servicio Web es el de una interfaz que expone un número de métodos bien definidos. Todo lo que necesita el consumidor es llamar a esos métodos usando los protocolos estándar de Internet, pasando parámetros en formato XML, y recibiendo respuestas también en formato XML.

.NET Framework es el modelo de programación de la plataforma .NET. Administra gran parte de los detalles de infraestructura, permitiendo a los desarrolladores centrarse en escribir el código de la lógica empresarial para sus aplicaciones. .NET Framework incluye el

lenguaje común en tiempo de ejecución (CLR, Common Language Runtime) y bibliotecas de clases.

Uno de los aspectos más interesantes de Microsoft® .NET Framework es la incorporación de los servicios Web XML (Lenguaje de marcado extensible). Los servicios web permiten la comunicación y el intercambio de información entre aplicaciones a través de procesos basados en estándares. La implementación de servicios Web por parte de Microsoft conlleva numerosas ventajas inherentes. El intercambio de información se realiza en formato XML, utilizando el protocolo de transferencia de hipertexto (HTTP) y el protocolo de control de transporte/protocolo Internet (TCP/IP), por lo que los servicios Web pueden utilizar la infraestructura existente e implementarse en diferentes entornos fácilmente. El .NET Framework es un componente del sistema operativo Microsoft Windows® que proporciona el modelo de programación para crear, implementar y utilizar aplicaciones basadas en Web, aplicaciones para dispositivos inteligentes y servicios Web XML.

La infraestructura de los servicios Web ASP.NET proporciona una API simple para dichos servicios basada en la asignación de mensajes SOAP a invocaciones de métodos. Para ello, ofrece un modelo de programación muy simple basado en la asignación de intercambios de mensajes SOAP a invocaciones de métodos individuales. Los clientes de los servicios Web ASP.NET no necesitan saber nada acerca de la plataforma, el modelo de objetos o el lenguaje de programación utilizados para crearlos. Ni siquiera los propios servicios tienen por qué saber nada acerca de los clientes que les envían mensajes. El único requisito es que ambas partes deben estar de acuerdo en cuanto al formato de los mensajes SOAP que se generan y se consumen, establecido en la definición de contrato del servicio Web que se expresa mediante esquemas WSDL y XML (XSD).

.NET Framework simplifica la tarea de exponer código .NET como servicio Web. Una de las razones para tal sencillez es que .NET Framework incluye especificaciones para transformar tipos de datos complejos de .NET en XML (serialización) y para el proceso inverso (deserialización). Algunos de los beneficios de utilizar esta herramienta se describen a continuación:

- **Confiability mejorada**

El .NET Framework toma los principales logros originalmente hechos en Windows 2000 y los lleva a nuevos niveles. Con maneras avanzadas de monitorear el estado de las aplicaciones que se ejecutan. Las aplicaciones creadas para el .NET Framework se mantienen en ejecución por más tiempo que antes.

- **Mejor desempeño**

Gracias en parte a la avanzada compilación y a las técnicas de caché, las aplicaciones de servidor nunca habían sido más rápidas que las creadas para el .NET Framework y con tecnología Microsoft ASP.NET. Los usuarios que han migrado de ASP a ASP.NET comprueban el incremento en la velocidad en un rango de 300 a 500 por ciento.

- **Productividad del desarrollador**

Los desarrolladores de todo tipo encuentran que pueden familiarizarse rápidamente con el .NET Framework. Lo intuitivo de su modelo de programación, la cantidad de código proporcionado en las bibliotecas de clases y la cantidad de trabajo hecho automáticamente por el .NET Framework en áreas como la administración de memoria ha permitido a los desarrolladores del .NET Framework aumentar su productividad a niveles muy altos.

- **Seguridad**

La tecnología de seguridad de acceso al código del .NET Framework fue diseñada para los ambientes actuales de Internet. El .NET Framework puede recolectar evidencia acerca de dónde se origina una aplicación, quién la creó, cuál es su firma digital y cuál es su función. El ambiente de ejecución del .NET Framework puede combinar esa evidencia con políticas de seguridad para decidir si una aplicación puede acceder a determinado recurso. Hasta puede "negociar" con la aplicación, por ejemplo, negando el permiso de escritura a un directorio con protección y permitiendo a la aplicación seleccionar si desea ejecutarse, aunque se le haya negado el permiso.

- **Integración con sistemas existentes**

La tecnología del .NET Framework para la operación con objetos COM genera una envoltura en sus componentes COM existentes y aplicaciones basadas en Windows (como Microsoft Office), permitiendo que se programe para ellos como si hubieran estado diseñados originalmente para el .NET Framework. Las aplicaciones creadas usando el .NET Framework pueden conectarse con sistemas y aplicaciones existentes, sin importar su plataforma base, por medio de servicios Web XML u otros medios específicos de la aplicación. Finalmente, la herramienta Visual Basic Upgrade Tool, disponible con Visual Studio .NET, y la herramienta Java Language Conversion Assistant, ayudan a convertir código existente de Microsoft Visual Basic® 6.0 y Microsoft Visual J++® para que pueda ejecutarse en el .NET Framework.

- **Facilidad de implementación**

El .NET Framework tiene características para hacer más fácil la implementación, ejecución y administración de aplicaciones. El aislamiento de aplicaciones y el control de versiones automático de componentes pueden ayudar a prevenir conflictos de versiones. Las aplicaciones creadas usando el .NET Framework pueden ser implementadas en un cliente o un servidor simplemente copiando el directorio de la aplicación al equipo, sin realizar cambios en el registro. Además, las aplicaciones de Windows pueden ser actualizadas simplemente copiando los componentes necesarios al servidor Web que puede ser accedido por usuarios finales.

- **Soporte nativo para servicios Web XML**

El .NET Framework fue diseñado desde un principio con soporte para servicios Web XML, un modelo para computación distribuida en múltiples ambientes basados en protocolos

estándar como XML, SOAP y HTTP. Los servicios Web XML pueden ser usados para integrar aplicaciones que se ejecutan en diferentes plataformas, o para proporcionar software como un servicio. Con el .NET Framework, una aplicación puede ser transformada en un servicio Web XML con solamente una simple línea de código.

- **Soporte para más de 20 lenguajes de programación**

El .NET Framework tiene soporte para la integración de más de 20 lenguajes de programación de una manera que antes solamente podía imaginarse, permitiendo que los desarrolladores seleccionen el lenguaje de programación adecuado para la tarea que desean realizar. Todos los lenguajes de programación utilizan bibliotecas de clases únicas y extensibles. Los componentes escritos en diferentes lenguajes soportados por el .NET Framework pueden interactuar entre ellos, sin necesidad de utilizar COM.

- **Acceso flexible a datos**

La tecnología del .NET Framework, Microsoft ADO.NET, está diseñada para el estilo Web actual de acceder datos. Usando ADO.NET, los desarrolladores tienen la opción de trabajar con un caché de los datos requeridos basado en XML, en lugar de manipular directamente la base de datos. Esta aproximación al acceso a datos libera conexiones a bases de datos, lo que resulta en una escalabilidad significativamente más grande.

4.2.6.2 Servicios Web en J2EE

Ya se sabe que los servicios Web son aplicaciones basadas en web de uso abierto, basado en XML estándares y protocolos de transporte para el intercambio de datos con los clientes. En el caso de J2EE los servicios Web son desarrollados utilizando la tecnología Java API y las herramientas ofrecidas por un sistema integrado de servicios Web de pila llamado Metro. El Metro de pila consta de JAX-WS, JAXB, y WSIT, le permiten crear e implementar servicios web seguros, fiables y transaccionales, interoperables y clientes. El Metro de pila es parte del Proyecto Planeta y como parte de GlassFish, Java Platform, Enterprise Edition (Java EE), y parcialmente en Java Platform, Standard Edition (Java SE). GlassFish y Java EE también apoyan el legado de JAX-RPC API.

Proyecto GlassFish sustituye a la de Java Web Services Developer Pack, para proporcionar nuevos servicios web entre las distintas versiones de la Sun Java System Application Server, actualmente ya está en su versión 3

Servicios web de Java son una de las principales características nuevas en la plataforma J2EE 1.4, que cuenta con versiones de algunos de los APIs que Sun Microsystems en libertad como parte del paquete de Java Web Services Developer's en 2002. La Figura 4.12, muestra las API de los diversos servicios web (representada por las zonas sombreadas) y cómo se integran.

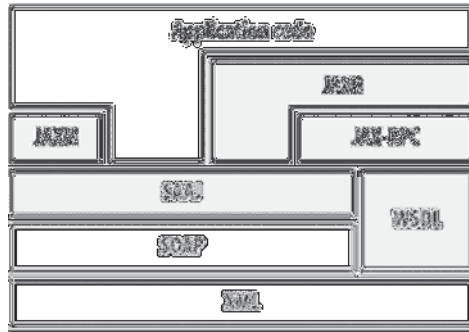


Figura 4.12. Api Servicios Web en Java

Se ha podido apreciar que todos los servicios web dependen de XML, ya que los mensajes intercambiados entre el prestador y el consumidor de servicios están codificados en XML. En la actualidad, estos mensajes se realizan mediante el protocolo SOAP el cual se encuentra basado en XML, Java implementa las siguientes Apis para sus Servicios Web:

- **WSDL**

Como se describió anteriormente, WSDL es un vocabulario XML que se utiliza para describir la interfaz proporcionada por un servicio web. Tanto J2EE 1.4 y JWSDP apoyan el uso de WSDL 1.1 para especificar las interfaces de servicio, pero no proporcionan una API de código de la aplicación de la manipulación directa de WSDL.

- **SAAJ**

SAAJ (SOAP con Attachments API for Java) proporciona una interfaz de programación directa con el protocolo SOAP. Se utilizan para el manejo de protocolo de bajo nivel y se omite la construcción de sus propios mensajes SOAP.

SAAJ provee un modo estándar para enviar documentos XML a través de internet desde la plataforma Java.

- **JAXM**

SAAJ ofrece un servicio básico de mensajería SOAP, pero esta carece de algunas características requeridas por las aplicaciones más avanzadas. El API Java para XML Messaging (JAXM), es una capa en la parte superior de SOAP, proporciona algunas funciones adicionales que para muchos desarrolladores será de gran utilidad, incluyendo la mensajería asíncrona, el apoyo a los perfiles de enrutamiento ebXML y WS-Routing, y una instalación que permite retransmitir mensajes que no se han entregado correctamente la primera vez que fueron enviados.

- ***JAX-RPC o JAX-WS v 2.0***

Para la mayoría de los desarrolladores, el API Java para RPC basado en XML (JAX-RPC) es el servicio web más importante de la API en la plataforma J2EE 1.4 (y en el JWSDP). JAX-RPC proporciona una manera relativamente simple para acceder a los servicios web utilizando un lenguaje de programación Java.

- ***JAXR***

El API Java para XML basado en los registros (JAXR) proporciona una interfaz para los registros UDDI y ebXML. A pesar de que esta API pueda ser utilizada por cualquier aplicación, lo más probable es que sea de utilidad para desarrolladores que deseen crear herramientas que permiten un fácil acceso a los registros o de ofrecer búsquedas personalizadas. JAXR permite tanto la publicación de información de los registros tanto como la recuperación de información de los mismos.

CAPÍTULO 5: SOLUCIÓN PLANTEADA

5.1 Solución general

Diseñar e Implementar un sistema, que permita gestionar el flujo de trabajo de la empresa, desde la captura de clientes, pasando por la generación de presupuestos hasta llegar a la asignación de porcentaje de ventas, insertando una aplicación desarrollada en entorno móvil que ayudará a realizar el proceso de captura de presupuestos así como la ejecución de órdenes de trabajo con mayor rapidez.

La figura 6.1 muestra la interacción entre el sistema principal de la empresa y la aplicación móvil (generación de presupuestos en terreno), donde se implementará un servicio web y una aplicación web que permitirá la comunicación de ambas aplicaciones.

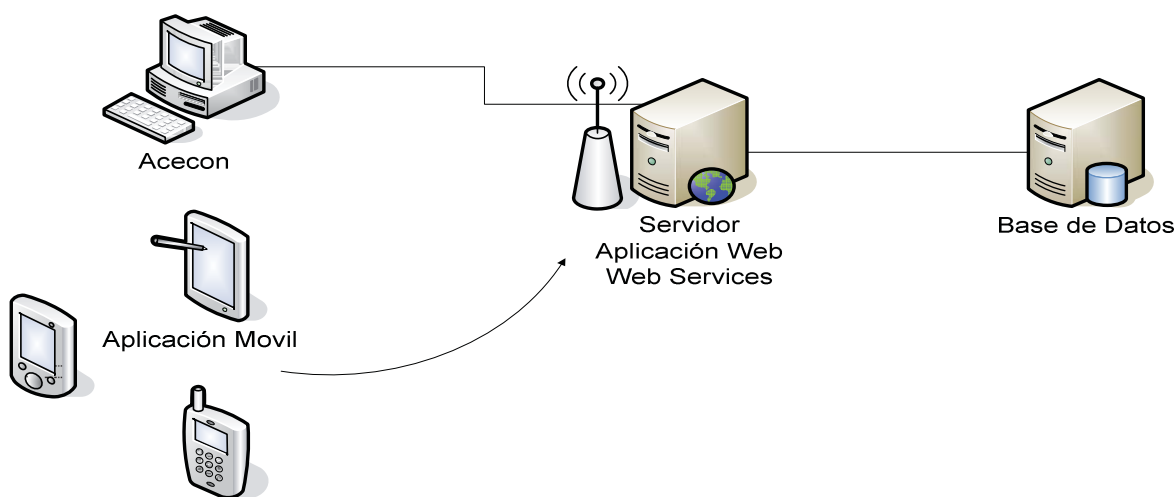


Figura 5.1. Arquitectura Solución Planteada.

5.2 Solución específica para los problemas identificados

- **Gestión de Clientes:** generar un módulo que permita la gestión de clientes, donde el usuario del sistema podrá, agregar, buscar y modificar los datos del cliente. Esto ayudará a dejar de usar las plantillas de texto, permitiendo acceder de manera más rápida, segura y ordenada a la información solicitada.
- **Gestión de Trabajadores:** al igual que en la gestión de clientes, crear un módulo que permita al usuario del sistema tener total control sobre la información de Trabajadores y Vendedores de la empresa.
- **Gestión de presupuestos:** generar un módulo que permita la gestión de presupuestos, donde el usuario del sistema podrá crear nuevos presupuestos, buscar presupuestos ya creados y acceder así a otros módulos, como el de clientes y trabajadores para así asignar información ya ingresada, como datos del vendedor y técnico designado. Además de esto generar un módulo denominado catálogo de tareas, el cual se encargará de administrar todos los trabajos que Acecon puede realizar, con su respectivo precio, esto permitirá al usuario no estar digitando las tareas salvo la primera vez que se ingresan.
- **Asignación de porcentaje a Vendedores:** cada presupuesto se manejará a través de estados dentro del flujo de trabajo: pendiente, aceptado, terminado, etc. Esto permitirá hacer un seguimiento al presupuesto y poder concluir en el mes cuantos presupuestos fueron terminados, para así poder asignar el porcentaje de comisión al vendedor mes a mes.
- **Manejo de mucha documentación archivada:** toda la información recopilada por las aplicaciones (tanto aplicación web como móvil) será derivada a un servidor, para así brindar mayor seguridad en el caso de cualquier anomalía y poder acceder a ella de un modo más expedito y confiable.
- **Asignación de tareas a los trabajadores:** implementar a través de un entorno móvil una aplicación que permita a los trabajadores de la empresa descargar día a día las actividades a realizar mediante la generación de órdenes de trabajo.
- **Lentitud al generar Presupuestos:** generar una aplicación móvil que permita al técnico la creación de presupuestos en terreno, a través, de un dispositivo móvil el cual se encontrará conectado al sistema central mediante la implementación de un servicio web.

5.3 Tecnologías

5.3.1 Motor de base de datos

- **PostgreSQL**

Es un sistema de gestión de base de datos relacional orientada a objetos y libre, publicado bajo la licencia BSD.

Entre sus principales características PostgreSQL, posee una alta concurrencia Mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés) permitiendo que un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

Esta particularidad es primordial al trabajar con entornos móviles, ya que la base de datos va ser consultada constantemente por los dispositivos.

Además PostgreSQL permite trabajar con claves ajenas (foreign keys), disparadores (triggers) y posee una amplia variada de tipos nativos, dando soporte a:

- Números de precisión arbitraria.
- Texto de largo ilimitado.
- Figuras geométricas (con una variedad de funciones asociadas)
- Direcciones IP (IPv4 e IPv6).
- Bloques de direcciones estilo CIDR.
- Direcciones MAC.
- Arrays.

5.3.2 Lenguaje de programación

A lo largo del proyecto y debido a los distintos módulos que se espera implementar, se utilizarán distintos lenguajes de programación que se especificarán a continuación:

- **Java**

Es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90, se basa en la sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria [12].

Una de las principales características de Java es la independencia de la plataforma, permitiendo así que programas escritos en este lenguaje, pueden ejecutarse igualmente en

cualquier tipo de hardware. Este lenguaje de programación será usado para desarrollar la aplicación web en conjunto con JSP.

- **C#**

Es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET.

Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma.NET el cual es similar al de Java aunque incluye mejoras derivadas de otros lenguajes (entre ellos Delphi). Este lenguaje será utilizado para desarrollar la aplicación móvil, junto con un grupo de otros lenguajes que contiene el Compact Framework de .Net.

- **Html**

Siglas de HyperText Markup Language (Lenguaje de Marcado de Hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML [12].

- **JavaScript**

Es un lenguaje de scripting basado en objetos no tipado y liviano, utilizado para acceder a objetos en aplicaciones. Principalmente, se utiliza integrado en un navegador web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas. JavaScript es un dialecto de ECMAScript y se caracteriza por ser un lenguaje basado en prototipos, con entrada dinámica y con funciones de primera clase. JavaScript ha tenido influencia de múltiples lenguajes y se diseñó con una sintaxis similar al lenguaje de programación Java, aunque más fácil de utilizar para personas que no programan [12].

- **Css (Hojas de estilo en cascada)**

CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores [12].

5.3.3 Javaserpáginas

Java Server Pages (JSP) es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

Esta tecnología es un desarrollo de la compañía Sun Microsystems. La Especificación JSP 1.2 fue la primera que se liberó y en la actualidad está disponible la Especificación JSP 2.1.

Las JSP's permiten la utilización de código Java mediante scripts. Además, es posible utilizar algunas acciones JSP predefinidas mediante etiquetas. Estas etiquetas pueden ser enriquecidas mediante la utilización de Bibliotecas de Etiquetas (TagLibs o Tag Libraries) externas e incluso personalizadas [12].

5.3.4 Entorno de desarrollo

- **NetBeans IDE**

Es un entorno de desarrollo integrado para programadores pensado principalmente para escribir, compilar, depurar y ejecutar programas. Está escrito en Java pero puede servir para cualquier otro lenguaje de programación. Es un producto libre y gratuito sin restricciones de uso, bajo la plataforma NetBeans.

NetBeans ofrece servicios comunes a las aplicaciones de escritorio, permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación. Entre las características de la plataforma están:

- Administración de las interfaces de usuario (ej. menús y barras de herramientas).
- Administración de las configuraciones del usuario.
- Administración del almacenamiento (guardando y cargando cualquier tipo de dato).
- Administración de ventanas.
- Framework basado en asistentes (diálogos paso a paso).

- **Microsoft Visual Studio**

Es un entorno de desarrollo integrado para sistemas operativos Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones de escritorio, sitios y sistemas Web, así como servicios web en cualquier entorno que soporte la plataforma .NET. Así se pueden crear aplicaciones que se comuniquen entre estaciones de trabajo, páginas Web y dispositivos móviles.

Para el proyecto, específicamente en la etapa de desarrollo de la aplicación móvil, se ocupará el .NET Compact Framework que está diseñado para ofrecer un rendimiento óptimo bajo las restricciones de los limitados recursos de los dispositivos móviles.

CAPÍTULO 6: ESTUDIO DE FACTIBILIDAD

Después de definir la problemática presente y establecer los principales problemas que ameritan el desarrollo de un nuevo sistema, es pertinente realizar un estudio de factibilidad para determinar la infraestructura tecnológica, ámbito legal y capacidad técnica que implica la implantación del sistema, así como los costos, beneficios y el grado de aceptación que la propuesta genera en la empresa.

Este análisis permitió determinar las posibilidades de diseñar el sistema propuesto y su puesta en marcha, los aspectos tomados en cuenta para este estudio fueron clasificados en cuatro áreas que se describen a continuación:

6.1 Factibilidad operativa

La factibilidad operativa permite predecir, si se pondrá en marcha el sistema propuesto, aprovechando los beneficios que ofrece, a todos los usuarios involucrados con el mismo, ya sea los que interactúan directamente con este, como también aquellos que reciben información producida por el sistema. Por otra parte, el correcto funcionamiento de la aplicación, siempre estará supeditado a la capacidad de los empleados para utilizarlo.

La necesidad y el deseo de utilizar una herramienta que sirva de apoyo al flujo de trabajo expresada por el personal de la empresa, permitió identificar las necesidades de manera más amigable y sencilla. Basándose en las entrevistas y conversaciones sostenidas, y debido a que no se mostró ninguna oposición a usar el sistema, la aplicación que se va a desarrollar es factible operacionalmente.

6.2 Factibilidad técnica

La factibilidad técnica consistió en realizar una evaluación de la tecnología existente en la organización, este estudio estuvo destinado a recolectar información sobre los componentes técnicos que posee la empresa y la posibilidad de hacer uso de los mismos en la implementación del sistema.

El sistema a desarrollar no es un sistema que requiera de grandes maquinas o de tecnologías que no existan y la empresa ya cuenta con computadores con las características necesarias para el uso del sistema. En cuanto a las tecnologías a usar, se encuentran en el mercado múltiples lenguajes de programación, motores de bases de datos, etc. que son suficientes para el desarrollo del sistema.

Otro punto importante es el personal que se requiere para el desarrollo de este sistema informático. Si bien no se cuenta con un gran equipo de trabajo, se tienen los conocimientos, tiempo para investigación, y correcto desarrollo del sistema, además del apoyo que brinda la

universidad en cuanto a software y hardware hacen de que la aplicación se factible técnicamente.

6.3 Factibilidad legal

El objetivo de a factibilidad legal es el poder verificar que en el desarrollo de un sistema no incurra en infracciones, violaciones y otros delitos impidan la puesta en marcha del sistema.

Para el desarrollo del presente proyecto no existen trabas legales que impidan el buen desempeño y funcionamiento del software, puesto que no se incurren en infracciones a las leyes vigentes hoy en día, las cuales se especifican a continuación:

Ley N° 19.223, Relativa a delitos Informáticos que detalla solo cuatro artículos que se describen a continuación:

Artículo 1°

El que maliciosamente destruya o inutilice un sistema de tratamiento de información o sus partes o componentes, o impida, obstaculice o modifique su funcionamiento, sufrirá la pena de presidio menor en su grado medio a máximo.

Si como consecuencia de estas conductas se afectaren los datos contenidos en el sistema, se aplicará la pena señalada en el inciso anterior, en su grado máximo.

Artículo 2°

El que con el ánimo de apoderarse, usar o conocer indebidamente de la información contenida en un sistema de tratamiento de la misma, lo intercepte, interfiera o acceda a él, será castigado con presidio menor en su grado mínimo a medio.

Artículo 3°

El que maliciosamente altere, dañe o destruya los datos contenidos en un sistema de tratamiento de información, será castigado con presidio menor en su grado medio.

Artículo 4°

El que maliciosamente revele o difunda los datos contenidos en un sistema de información, sufrirá la pena de presidio menor en su grado medio. Si quien incurre en estas conductas es el responsable del sistema de información, la pena se aumentará en un grado.

6.4 Factibilidad económica

La Factibilidad Económica es de vital importancia, debido a que la empresa debe poseer los recursos necesarios para la implementación del sistema. El siguiente estudio permite visualizar si se puede solventar y financiar los recursos necesarios del proyecto.

6.4.1 Costos asociados

- **Costo Personal**

Se debe considerar que en el proyecto trabajaran dos personas, por la envergadura del sistema se ha estimado que el tiempo requerido para finalizar la aplicación es de 10 meses.

Según estudio de mercado, el salario promedio de un ingeniero informático recién egresado es de \$600.000 pesos, trabajando 5 días a la semana y 8 horas diarias. Por lo cual es salario por hora es de \$3.750 pesos.

Considerando que el sistema está siendo desarrollado por dos personas, de lunes a sábado trabajando en promedio 3 horas diarias, se estima que el salario mensual para ambos desarrolladores es de \$ 540.000 mil pesos.

Salario	\$ 270.000
Meses	10
Cantidad de desarrolladores	2
Costo Personal	\$ 5.400.000

Tabla 6.1. Costo Personal.

- **Costo Hardware**

Considerando que la empresa cuenta con todos los equipos necesarios para el funcionamiento del sistema, además de las herramientas brindadas por la universidad, el único costo en hardware identificado es el de los dispositivos móviles.

Hardware para la realización del proyecto	\$ 0
Dispositivo Móvil	\$ 230.000
Cantidad de dispositivos	2
Costo de hardware	\$ 460.000

Tabla 6.2. Costo de hardware.

- **Costo Software**

Los costos de los programas utilizados para la realización del proyecto se especifican en la tabla 6.3.

Software	Tipo	Licencia Libre	Costo
Windows XP	Sistema Operativo	Universidad cuenta con licencias	\$ 0
Microsoft Office	Herramientas de apoyo	Universidad cuenta con licencias	\$ 0
NetBeans	Entorno de desarrollo	Licencia libre	\$ 0
Visual Estudio 2008	Entorno de desarrollo	Universidad cuenta con licencias	\$ 0
Dia	Herramienta de modelado	Licencia libre	\$ 0
StarUML	Herramienta de modelado	Licencia libre	\$ 0
PostgreSQL	Software de Gestión de Base de Datos	Licencia Libre	\$ 0

Tabla 6.3. Costo de software.

- **Costo de capacitación**

La capacitación consistirá en dos sesiones de 45 min. Con un costo total de \$100.000 pesos. Esta tiene como objetivo instruir a los usuarios en el uso del sistema, complementando la actividad con el manual de usuario entregado junto con el sistema a la empresa.

- **Otros Costos**

Plan de Internet para el dispositivo móvil \$20.000 pesos aproximados durante un mes.

- **Costo Total**

Los costos finales para la realización del proyecto están dados por:

CT = Costo Personal + Costo de Hardware + Costo de Software + Costo de Capacitación + Internet.

La Tabla 7.4, muestra un resumen final de los costos asociados al proyecto.

Costo Personal	\$ 5.400.000
Costo Hardware	\$ 460.000
Costo Software	\$ 0
Costo Capacitación	\$ 100.000
Internet	\$ 20.000 (1 mes)
Costo Total	\$ 5.980.000

Tabla 6.4. Costo Total.

Todos los cálculos anteriores son estimaciones de los supuestos valores que influyen en el desarrollo del proyecto, en condiciones normales de trabajo el proyecto podría demorarse menos de 10 meses en estar listo.

6.4.2 Beneficios para la empresa

Si bien el sistema implementado en la compañía no produce beneficios económicos directamente para la empresa, aporta un valor agregado importante a esta, ya que logrará reducir tiempos en actividades específicas del flujo de trabajo, además de controlar toda la gestión de la compañía de manera segura.

Estos beneficios, a medida que el tiempo avance significaran reducciones importantes en los costos de la empresa justificando claramente en un futuro la inversión realiza por la compañía para el desarrollo del sistema.

CAPÍTULO 7: ANALISIS DE RIESGO

En el análisis de riesgo buscamos cualquier situación adversa que pueda llegar a afectar al curso normal del proyecto asignándole una probabilidad de ocurrencia y un plan de mitigación asociado, siendo necesario en algunos casos en los que la probabilidad y el grado en el que afecta al proyecto sean demasiado altos la generación de planes de contingencia para minimizar los efectos sobre el mismo.

Para administrar los riesgos asociados al proyecto, se efectuaran las siguientes actividades:

- Identificación de los riesgos, identificar los riesgos de proyecto, negocio y producto.
- Análisis de riesgos, evaluación de la probabilidad y consecuencia de los riesgos.
- Planificación de riesgos, elaboración de planes para minimizar o evitar los efectos del riesgo.
- Monitoreo de riesgos, monitorear los riesgos durante todo el proyecto.

Cada riesgo conlleva un plan de mitigación asociado el cual permite minimizar la ocurrencia de dicho riesgo, sin embargo, los planes de contingencia serán solo realizados para aquellos riesgos que tengan un alto grado de ocurrencia y que además tengan una incidencia en el proyecto que puedan de tal modo afectarlo en forma seria.

Probabilidad de ocurrencia: 1-bajo, 2-medio, 3-alto

Impacto o Severidad del Riesgo: 1-bajo, 2-medio, 3-alto

Exposición al Riesgo: Probabilidad de Ocurrencia x Impacto

Descripción	P	I	E
Bajo nivel de conocimiento por parte de los desarrolladores de las tecnologías a utilizar	3	2	6
El cliente incorpora requerimientos adicionales en el transcurso del proyecto	3	2	6
Incumplimiento de las tareas asignadas durante la planificación del proyecto	2	2	4
El sistema ofrece un pobre rendimiento en calidad y tiempo de respuesta de los servicios web.	2	2	4
El sistema móvil consume más recursos de lo esperado.	2	2	4
El sistema ofrece un nivel bajo de usabilidad.	1	2	2
El sistema desarrollado no cumple con las expectativas del cliente.	1	3	3

Tabla 7.1. Análisis de Riesgo

Descripción	Mitigación	Contingencia
Bajo nivel de conocimiento de las tecnologías por parte de los desarrolladores	Se deberá destinar tiempo extra para aprender la programación en las técnicas a utilizar para el proyecto.	Se contratará personal adicional con experiencia en las técnicas de programación empleadas.
El cliente incorpora requerimientos adicionales en el transcurso del proyecto	Reorganizar la planificación agregando horas hombres al plan de trabajo para contemplar los nuevos requerimientos	Se modifica el alcance del proyecto y se estima el esfuerzo requerido para incluir los nuevos requerimientos.
Incumplimiento de las tareas asignadas durante la planificación del proyecto	Durante el desarrollo del software marcar hitos claros en los cuales presentar los avances	Incrementar la asignación de horas hombres a las tareas del proyecto que corresponden a la línea crítica.
El sistema ofrece un pobre rendimiento en calidad y tiempo de respuesta de los servicios web.	Se debe realizar un análisis detallado de las técnicas de optimización disponibles.	Se vuelve a iterar sobre la fase de construcción y pruebas del sistema para obtener mejores resultados.
El sistema ofrece un nivel bajo de usabilidad.	Mediante validación de prototipos ir logrando la usabilidad esperada	Buscar mejoras en las interfaces mediante pruebas de usabilidad

Tabla 7.2. Plan de Mitigación y Contingencia

CAPÍTULO 8: PARADIGMA Y METODOLOGIA DE DESARROLLO

8.1 Paradigma de desarrollo

Dada la amplia gama de paradigmas existentes se hace difícil la elección de uno en particular para un desarrollo determinado, cada paradigma posee sus ventajas y desventajas que lo hacen único respecto al resto, es por esto que no existe un paradigma ideal, y se hace necesario buscar aquel que sea más conveniente y que cumpla de mejor manera con las necesidades del proyecto.

En nuestro caso este paradigma es el de Desarrollo de Prototipos Evolutivo, algunas características del mismo son:

- Se centra en ayudar al usuario a comprender los requisitos del sistema que desea, esto ayuda en gran medida en casos donde el usuario no tiene una noción acabada de lo que desea.
- Se desarrolla una implementación inicial, exponiéndola a los comentarios del usuario y redefiniéndola a través de las diferentes versiones, con esto tanto usuarios como desarrolladores logran saber si han interpretado de un buen modo las especificaciones y necesidades del usuario.
- Al presentar prototipos ayuda al acercamiento entre cliente y desarrollador, lo que hace que el sistema en sus últimas etapas sea muy cercano a lo que el cliente desea, así como la usabilidad del mismo.
- Las actividades de especificación, desarrollo y validación se llevan a cabo concurrentemente y tienen retroalimentación rápido a lo largo del proceso.
- Es adecuado para sistemas pequeños.

8.2 Metodología de desarrollo

Para el presente proyecto se ha escogido la metodología orientada a objetos dadas las prestaciones que esta nos ofrece, y las ventajas que presenta sobre el modelo de desarrollo estructurado, algunos puntos de vital interés para el desarrollo son:

- Permite una mejor comunicación entre desarrolladores y clientes.
- Abstrae el mundo real, por lo que es fácilmente entendible.
- Orientación a objetos presenta una alta modularidad lo que permite separar el sistema en secciones más chicas que son más fáciles de programar y que más tarde serán unidas al sistema, esto evita si existe un error en algún módulo el tener que revisar todo

el código programado en busca del mismo. A su vez permite cambios en el sistema en el caso en el que el cliente desee agregar nuevos módulos al mismo.

- Fomenta la reutilización de componentes, se pueden buscar componentes ya antes utilizados para hacer así más fácil la tarea de los programadores.
- Dada su modularidad permite la creación de prototipos que pueden ser fácilmente validados con los clientes, esto hace más expedita la programación, llevando el sistema más cercano a lo que el cliente desea.
- Utiliza UML (United Model Language) como lenguaje para documentar y especificar.

CAPÍTULO 9: HERRAMIENTAS Y ARQUITECTURA

9.1 Herramientas

Las herramientas en el desarrollo de un Software permiten dar apoyo a las distintas actividades que uno realiza en cada etapa del proyecto, sin este soporte el desarrollo de un proyecto exitoso no podría llevarse a cabo.

A continuación se detallará de manera breve las herramientas que se utilizaran en el desarrollo del sistema.

- **Microsoft Word**

Poderoso procesador de texto utilizado para la creación de documentos e informes. Se escogió este procesador de texto debido a su larga trayectoria, simplicidad de uso y por las herramientas de apoyo que presenta, siendo la más destacada la corrección de ortografía.

- **Microsoft PowerPoint**

Microsoft PowerPoint ha demostrado a lo largo de su trayectoria ser una de las herramientas más poderosas y sencillas de utilizar para la creación de presentaciones.

Es un programa fácil de entender, con animaciones de texto e imágenes prediseñadas o importadas desde imágenes de la computadora. Además se pueden aplicar distintos diseños de fuente, plantilla y animación. Este tipo de presentaciones suele ser muy llamativo y mucho más práctico que los de Microsoft Word.

- **Microsoft Project**

Es un software de administración de proyectos diseñado, desarrollado y comercializado por Microsoft para asistir a administradores de proyectos en el desarrollo de planes, asignación de recursos a tareas, dar seguimiento al progreso, administrar presupuesto y analizar cargas de trabajo. El uso de esta herramienta está destinado al desarrollo de la carta Gantt del proyecto, que permitirá administrar todas las tareas programadas.

- **Microsoft Visio**

Es un software de dibujo vectorial, adquirido en el año 2000 por Microsoft a través de negociaciones con la compañía Visio.

Las herramientas que lo componen permiten realizar diagramas de oficinas, diagramas de bases de datos, diagramas de flujo de programas, UML, y más, que permiten iniciar al usuario en los lenguajes de programación. Esta herramienta nos permitirá entre otras cosas

modelar bases de datos generar diagramas de UML, como casos de uso, diagramas de secuencia y actividad entre otros.

- **Día**

Es una aplicación informática de propósito general para la creación de diagramas, desarrollada como parte del proyecto GNOME. Está concebido de forma modular, con diferentes paquetes de formas para diferentes necesidades. Se puede utilizar para dibujar diferentes tipos de diagramas. Actualmente se incluyen diagramas entidad-relación, diagramas UML, diagramas de flujo, diagramas de redes, diagramas de circuitos eléctricos, etc. Nuevas formas pueden ser fácilmente agregadas, dibujándolas con un subconjunto de SVG e incluyéndolas en un archivo XML.

- **StarUML**

Es una fuente abierta de UML , está licenciado bajo una versión modificada de la GNU GPL . Después de ser abandonado por algún tiempo, el proyecto tuvo una nueva versión para pasar de Delphi a Java / Eclipse. Sin embargo, la comunidad sigue siendo activa y muchos temas se discuten en los foros. StarUML soporta la mayoría de los tipos de diagramas especificados en UML 2.0. En la actualidad faltan objetos , paquetes , de tiempo y diagramas de interacción de vista general (aunque los dos primeros pueden ser adecuadamente el modelo a través del diagrama de clases de edición). Se utiliza esta herramienta porque es una versión gratuita, de interfaz amigable y fácil uso.

9.2 Arquitectura

En los inicios de la informática, la programación se consideraba un arte y se desarrollaba como tal, debido a la dificultad que entrañaba para la mayoría de las personas, pero con el tiempo se han ido descubriendo y desarrollando formas y guías generales, con base a las cuales se puedan resolver los problemas. A estas, se les ha denominado Arquitectura de Software, porque, semejanza de los planos de un edificio o construcción, estas indican la estructura, funcionamiento e interacción entre las partes del software [1]

Para el desarrollo de este proyecto se ocupara la arquitectura de 3 capas, la cual es una especialización de la arquitectura cliente-servidor donde la carga se divide en tres partes con un reparto claro de funciones: una capa para la presentación, otra para el cálculo y otra para el almacenamiento.

1.- Capa de presentación: presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso. Esta capa se comunica únicamente con la capa de negocio. También es conocida como interfaz gráfica y debe tener la característica de ser entendible y fácil de usar para el usuario.

2.- Capa de negocio: es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio

porque es aquí 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 para almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.

3.- Capa de datos: es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

La ventaja principal de esta arquitectura es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.

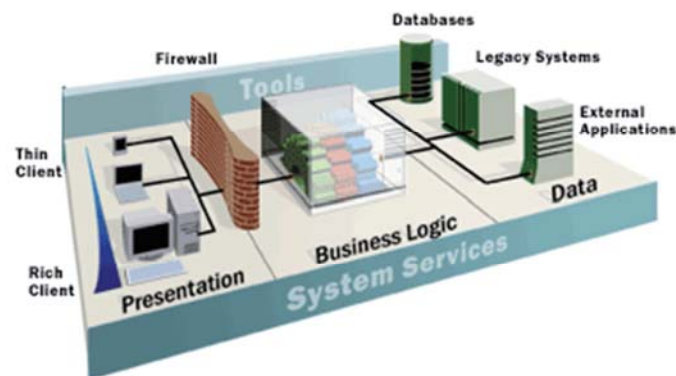


Figura 9.1. Arquitectura de tres capas.

CAPÍTULO 10: PLAN DE TRABAJO

La Figura 10.1, detalla el plan de trabajo a realizado para el desarrollo del sistema de control de flujo de trabajo.

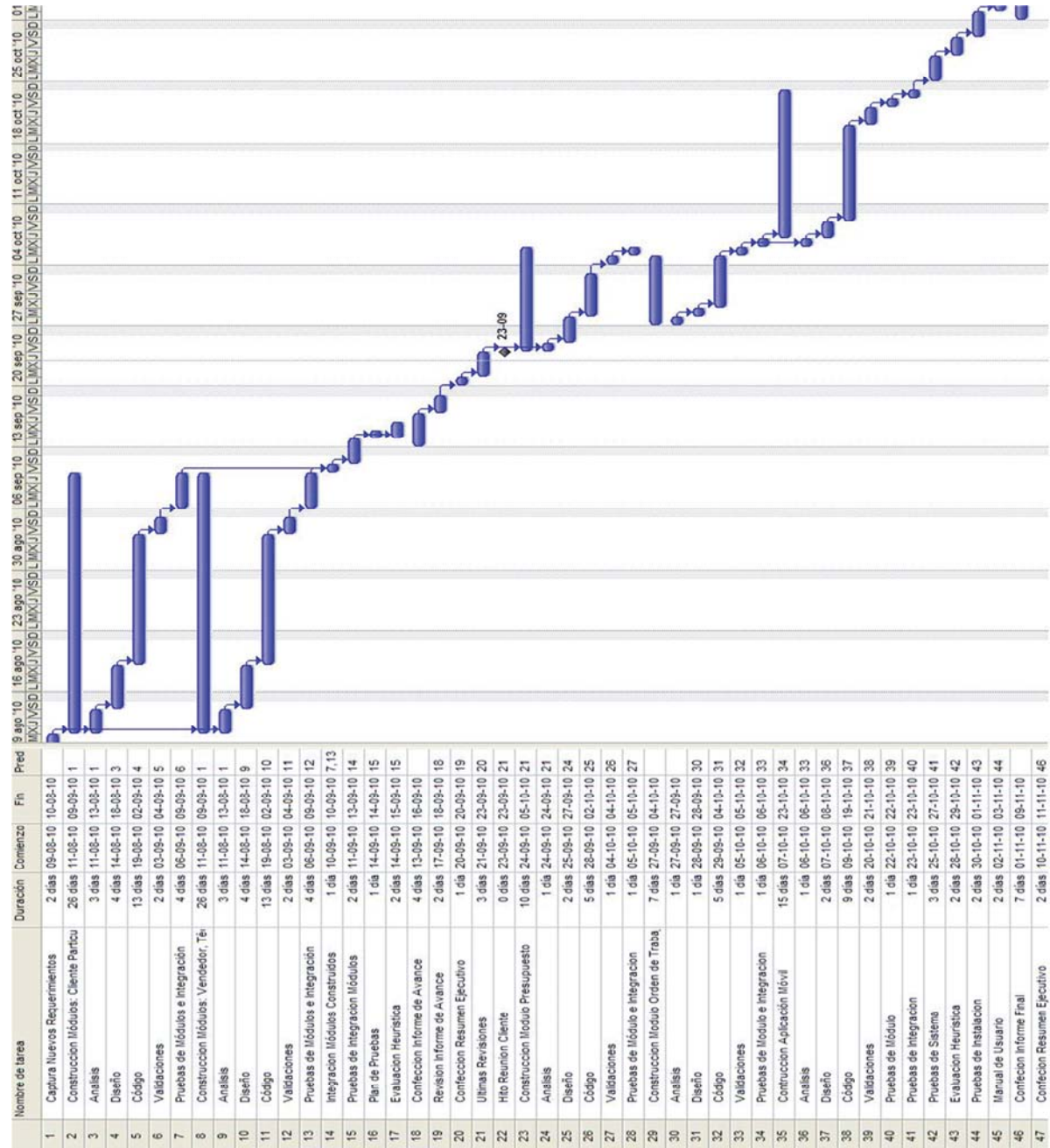


Figura 10.1. Plan de Trabajo.

CAPITULO 11: ESPECIFICACIÓN DE REQUERIMIENTOS

11.1 Introducción

Este documento ha sido desarrollado en base al estándar IEEE/ANSI 830 y tiene como objetivo analizar y documentar las necesidades funcionales y no funcionales que deberán ser soportadas por el sistema a desarrollar. Para ello, se identificarán los requisitos que ha de satisfacer el sistema, junto con el estudio de los problemas de las unidades afectadas y sus necesidades actuales.

11.2 Identificación de usuarios participantes

Los objetivos de esta tarea son identificar a los responsables de cada una de las unidades y a los principales usuarios implicados en el sistema.

En la empresa se identificaron los siguientes usuarios:

- **Administrador:** Responsable de la gestión en la empresa.
- **Asistente de operaciones:** Asistente del gerente de operaciones y del administrador.
- **Vendedores:** Vendedores de la empresa.
- **Técnicos:** Encargados de hacer las visitas técnicas, además de realizar las distintas obras ofrecidas por la empresa.
- **Clientes:** Usuarios que necesitan de los servicios que ofrece la empresa.

11.3 Catálogo de requisitos del sistema

El objetivo de la especificación es definir en forma clara, precisa, completa y verificable todas las funcionalidades y restricciones del sistema que se desea construir. Esta documentación está sujeta a revisiones por el grupo de usuarios que se recogerán por medio de sucesivas versiones del documento, hasta alcanzar su aprobación por parte de la administración y del grupo de usuarios participantes. Una vez aprobado, servirá de base al equipo para la construcción del sistema. Esta especificación se ha realizado de acuerdo al estándar “IEEE Recommended Practice for Software Requirements Specifications (IEEE/ANSI 830-1993)”, y se basa en las entrevistas realizadas a los usuarios participantes y el estudio de la documentación existente en la empresa.

11.3.1 Objetivos y alcance del sistema

El futuro sistema llevará el nombre de Acecon Web cuyos principales objetivos son la gestión de clientes, presupuestos y órdenes de trabajo, además de la creación de una aplicación móvil que permita a los técnicos en terreno cerrar las órdenes de trabajo planificadas diariamente.

Los servicios que ofrece la compañía son específicamente del ámbito de la construcción e instalación de artefactos a gas, tanto en edificios como en casas particulares. Una vez concluidas estas obras el inmueble -si fuese en el ámbito gas- queda en condiciones de adquirir la certificación de sello verde otorgada por la SEC. En la actualidad la empresa además realiza trabajos para la empresa Mademsa, que consisten principalmente en instalaciones de cocinas y conversiones de artefactos a gas. Estas órdenes son descargadas a través de un sistema web que facilita la compañía (Mademsa). En este ámbito la aplicación móvil deberá permitir a los técnicos cerrar las órdenes de trabajo en terreno, además de registrar las observaciones pertinentes.

El sistema a desarrollar debe ser abierto y escalable, y debe permitir la incorporación de nuevos requerimientos a futuro, como la integración de otros módulos de trabajo. El desarrollo se llevará a cabo en dependencias de la Pontificia Universidad Católica de Valparaíso.

11.3.2 Definiciones, acrónimos y abreviaturas

- **Definiciones:**
 - Presupuestos: cálculo anticipado de los ingresos y gastos de una actividad económica (personal, familiar, un negocio, una empresa, una oficina) durante un período.
 - Nombre Comunidad: nombre que recibe un edificio o comunidad en particular.
 - Id: número que identifica a una obra, presupuesto y orden de trabajo al ser creada.

- **Acrónimos:**
 - SEC: Superintendencia de electricidad y combustible.
 - IEEE: Institute of Electrical & Electronics Engineers.
 - IVA: Impuesto al valor agregado.

- **Abreviaturas:**
 - OT: Orden de trabajo.

11.3.3 Descripción general

Esta sección nos presenta una descripción general del sistema con el fin de conocer las funciones que debe soportar, los datos asociados, las restricciones impuestas y cualquier otro factor que pueda influir en la construcción del mismo.

Las funciones que debe realizar el sistema se pueden agrupar de la siguiente manera:

- **Administración de Personas:** debe permitir el registro y la posterior gestión de clientes, representantes, técnicos y vendedores de la empresa.
- **Administración de Obras:** debe permitir gestionar las obras realizadas por la empresa.
- **Administración de Presupuestos:** debe permitir gestionar los presupuestos realizados por la empresa.
- **Administración de OT:** debe permitir gestionar las órdenes de trabajo asignadas a los presupuestos de la empresa.
- **Privilegios de acceso al sistema:** debe permitir el acceso de usuarios al sistema con distintos perfiles definidos.

11.3.4 Requisitos funcionales

a) Gestión de Personas

a.1) Gestión de Clientes

Introducción: El sistema debe permitir introducir información de los clientes, además de buscar y modificar registros de clientes ya existentes. Se debe especificar una sección para clientes particulares y clientes administradores ya que los datos de entrada no son iguales. La búsqueda de los clientes puede ser a través del nombre i/o apellido o el Rut. En el caso de tener coincidencias al realizar una búsqueda por nombre, el sistema deberá arrojar una lista con el nombre y apellidos de los clientes encontrados y su respectivo Rut, para que el usuario pueda seleccionar el cliente correcto.

Agregar Cliente Particular

Entrada	Proceso	Salida
Rut + Nombre + Apellido Paterno + Apellido Materno + Teléfono + Celular + Tipo + Dirección particular + Comuna.	El sistema comprueba si el rut ingresado es válido y registra el nuevo cliente en la base de datos. No pueden existir clientes con el mismo rut y un cliente no puede ser vendedor ni representante ni técnico a la vez.	Mensajes de lo que está ocurriendo.

Tabla 11.1. Agregar cliente particular.

Agregar Cliente Administración

Entrada	Proceso	Salida
Rut + Tipo + Dirección + Comuna + Nombre Comunidad + Representante.	El sistema comprueba si el rut ingresado es válido, se selecciona el representante del edificio y se registra el nuevo cliente en la base de datos. No pueden existir clientes con el mismo rut y un cliente no puede ser vendedor ni representante ni técnico a la vez.	Mensajes de lo que está ocurriendo.

Tabla 11.2. Agregar cliente administración.

Modificar Cliente Particular búsqueda rut

Entrada	Proceso	Salida
Rut + {datos que se van a modificar}	El sistema comprueba si el rut ingresado es válido y si el cliente ingresado existe.	Datos del cliente, dado el rut ingresado.

Tabla 11.3. Modificar cliente particular búsqueda rut.

Modificar Cliente Particular búsqueda nombre y apellido

Entrada	Proceso	Salida
Nombre y/o Apellido + {datos que se van a modificar}	El sistema comprueba si los datos ingresados son válidos y si el cliente ingresado existe.	Una lista con los clientes donde encuentre coincidencias dado los datos ingresados.

Tabla 11.4. Modificar cliente particular búsqueda nombre y apellido.

Buscar Cliente Particular búsqueda rut

Entrada	Proceso	Salida
Rut	El sistema comprueba si el rut ingresado es válido y si el cliente ingresado existe.	Datos del cliente, dado el rut ingresado.

Tabla 11.5. Buscar cliente particular búsqueda rut.

Buscar Cliente Particular búsqueda nombre y apellido

Entrada	Proceso	Salida
Nombre y/o Apellido	El sistema comprueba si los datos ingresados son válidos y si el cliente ingresado existe.	Una lista con los clientes donde encuentre coincidencias dado los datos ingresados.

Tabla 11.6. Buscar cliente particular búsqueda nombre y apellido.

a.2) Gestión de Representantes

Introducción: El sistema debe permitir introducir información de los representantes de edificios (clientes administración), además de buscar y modificar registros de representantes ya existentes. La búsqueda de los representantes puede ser a través del nombre i/o apellido o el rut. En el caso de tener coincidencias al realizar una búsqueda por nombre, el sistema deberá arrojar una lista con el nombre y apellidos de los representantes encontrados y su respectivo rut, para que el usuario pueda seleccionar el representante correcto.

Agregar Representante

Entrada	Proceso	Salida
Rut + Nombre + Apellido Paterno + Apellido Materno + Teléfono + Celular + Observaciones	El sistema comprueba si el rut ingresado es válido y registra al representante en la base de datos. No pueden existir representantes con el mismo rut y este no puede ser vendedor ni cliente ni técnico a la vez.	Mensajes de lo que está ocurriendo.

Tabla 11.7. Agregar representante.

Modificar Representante búsqueda por rut

Entrada	Proceso	Salida
Rut + {datos que se van a modificar}	El sistema comprueba si el rut ingresado es válido y si el representante ingresado existe.	Datos del representante, dado el rut ingresado.

Tabla 11.8. Modificar representante búsqueda por rut.

Modificar Representante búsqueda por nombre y apellido

Entrada	Proceso	Salida
Nombre y/o Apellido + {datos que se van a modificar}	El sistema comprueba si los datos ingresados son válidos y si el representante ingresado existe.	Una lista de los representantes donde encuentre coincidencias dado los datos ingresados.

Tabla 11.9. Modificar representante búsqueda por nombre y apellido.

Buscar Representante por rut

Entrada	Proceso	Salida
Rut	El sistema comprueba si el rut ingresado es válido y si el representante ingresado existe.	Datos del representante, dado el rut ingresado.

Tabla 11.10. Buscar representante por rut.

Buscar Representante por nombre y apellido

Entrada	Proceso	Salida
Nombre y/o Apellido	El sistema comprueba si los datos ingresados son válidos y si el representante ingresado existe.	Una lista con los representantes donde encuentre coincidencias dado los datos ingresados.

Tabla 11.11. Buscar representante por nombre y apellido.

a.3) Gestión de Técnicos

Introducción: El sistema debe permitir introducir información de los técnicos que trabajan en la empresa, además de buscar y modificar registros de técnicos ya existentes. La búsqueda de los técnicos puede ser a través del nombre i/o apellido o el rut. En el caso de tener coincidencias al realizar una búsqueda por nombre, el sistema deberá arrojar una lista con el nombre y apellidos de los técnicos encontrados y su respectivo rut, para que el usuario pueda seleccionar el técnico correcto.

Agregar Técnico

Entrada	Proceso	Salida
Rut + Nombre + Apellido Paterno + Apellido Materno + Teléfono + Celular + Clase + Cargo + N° carnet SEC.	El sistema comprueba si el rut ingresado es válido y registra al técnico en la base de datos. No pueden existir técnicos con el mismo rut y este no puede ser vendedor ni cliente ni representante a la vez.	Mensajes de lo que está ocurriendo.

Tabla 11.12. Agregar técnico

Modificar Técnico búsqueda por rut

Entrada	Proceso	Salida
Rut + {datos que se van a modificar}	El sistema comprueba si el rut ingresado es válido y si el técnico ingresado existe.	Datos del técnico, dado el rut ingresado.

Tabla 11.13. Modificar técnico búsqueda por rut.

Modificar Técnico búsqueda por nombre y apellido

Entrada	Proceso	Salida
Nombre y/o Apellido + {datos que se van a modificar}	El sistema comprueba si los datos ingresados son válidos y si el técnico ingresado existe.	Una lista de los técnicos donde encuentre coincidencias dado los datos ingresados.

Tabla 11.14. Modificar técnico búsqueda por nombre y apellido.

Buscar Técnico por rut

Entrada	Proceso	Salida
Rut	El sistema comprueba si el rut ingresado es válido y si el técnico ingresado existe.	Datos del técnico, dado el rut ingresado.

Tabla 11.15. Buscar técnico por rut.

Buscar Técnico por nombre y apellido

Entrada	Proceso	Salida
Nombre y/o Apellido	El sistema comprueba si los datos ingresados son válidos y si el técnico ingresado existe.	Una lista con los técnicos donde encuentre coincidencias dado los datos ingresados.

Tabla 11.16. Buscar técnico por nombre y apellido.

a.4) Gestión de Vendedores

Introducción: El sistema debe permitir introducir información de los vendedores que trabajan en la empresa, además de buscar y modificar registros de vendedores ya existentes. La búsqueda de los vendedores puede ser a través del nombre i/o apellido o el rut. En el caso de tener coincidencias al realizar una búsqueda por nombre, el sistema deberá arrojar una lista con el nombre y apellidos de los vendedores encontrados y su respectivo rut, para que el usuario pueda seleccionar el vendedor correcto.

Agregar Vendedor

Entrada	Proceso	Salida
Rut + Nombre + Apellido Paterno + Apellido Materno + Teléfono + Celular + Mail	El sistema comprueba si el rut ingresado es válido y registra al vendedor en la base de datos. No pueden existir vendedores con el mismo rut y estos no pueden ser técnicos ni cliente ni representante a la vez.	Mensajes de lo que está ocurriendo.

Tabla 11.17. Agregar vendedor.

Modificar Vendedor búsqueda por rut

Entrada	Proceso	Salida
Rut + {datos que se van a modificar}	El sistema comprueba si el rut ingresado es válido y si el vendedor ingresado existe.	Datos del vendedor, dado el rut ingresado.

Tabla 11.18. Modificar vendedor búsqueda por rut.

Modificar Vendedor búsqueda por nombre y apellido

Entrada	Proceso	Salida
Nombre y/o Apellido + {datos que se van a modificar}	El sistema comprueba si los datos ingresados son válidos y si el vendedor ingresado existe.	Una lista de los vendedores donde encuentre coincidencias dado los datos ingresados.

Tabla 11.19. Modificar Vendedor búsqueda por nombre y apellido.

Buscar Vendedor por rut

Entrada	Proceso	Salida
Rut	El sistema comprueba si el rut ingresado es válido y si el vendedor ingresado existe.	Datos del vendedor, dado el rut ingresado.

Tabla 11.20. Buscar vendedor por rut.

Buscar Vendedor por nombre y apellido

Entrada	Proceso	Salida
Nombre y/o Apellido	El sistema comprueba si los datos ingresados son válidos y si el vendedor ingresado existe.	Una lista con los vendedores donde encuentre coincidencias dado los datos ingresados.

Tabla 11.21. Buscar vendedor por nombre y apellido.

b) Gestión de Obras

Introducción: El sistema debe permitir introducir información de las obras que realiza la empresa, además de buscar y modificar registros de obras ya existentes. La búsqueda de las obras puede ser a través del nombre o id. En el caso de tener coincidencias en el nombre de las obras, el sistema deberá arrojar una lista con el nombre de las obras encontradas y su respectivo id para que el usuario pueda seleccionar la correcta.

Agregar Obra

Entrada	Proceso	Salida
Nombre + Precio + Estado	El sistema comprueba si el nombre de la obra ingresada es válido y la registra en la base de datos. No pueden existir obras con el mismo nombre.	Mensajes de lo que está ocurriendo.

Tabla 11.22. Agregar Obra.

Modificar Obra búsqueda por nombre

Entrada	Proceso	Salida
Nombre + {datos que se van a modificar}	El sistema comprueba si los datos de la obra ingresada son válidos y si el nombre de esta existe.	Datos de la obra o bien una lista con los nombres de las obras donde encuentra coincidencias.

Tabla 11.23. Modificar obra búsqueda por nombre.

Modificar Obra búsqueda por id

Entrada	Proceso	Salida
Id + {datos que se van a modificar}	El sistema comprueba si el id de la obra ingresada es válido y si este ya existe.	Datos de la obra.

Tabla 11.24. Modificar obra búsqueda por id.

Buscar Obra por nombre

Entrada	Proceso	Salida
Nombre	El sistema comprueba si los datos de la obra ingresada son válidos y si el nombre de esta existe.	Datos de la obra o bien una lista con los nombres de las obras donde encuentra coincidencias.

Tabla 11.25. Buscar obra por nombre.

Buscar Obra por id

Entrada	Proceso	Salida
Id	El sistema comprueba si el id de la obra ingresada es válido y si este ya existe.	Datos de la obra.

Tabla 11.26. Buscar obra por id.

c) Gestión de Presupuesto

Introducción: El sistema debe permitir generar presupuestos, detallando el id de la obra, nombre y monto por cada departamento, además de entregar un subtotal y un total general del presupuesto sin contemplar el IVA. Al momento de generar un presupuesto el usuario primero debe buscar al cliente, posteriormente asignar un técnico y un vendedor. Una vez generado el presupuesto este queda en estado pendiente, para que posteriormente una vez aceptado por el cliente este pase a un estado de aceptado. A continuación en la Figura 11.1 se especifica como debiera ser una aproximación al presupuesto generado.


 Presupuesto					
DATOS DE LA EMPRESA			N° DE SERIE:		
EMPRESA: Acecon Ltda.			DATOS DEL CLIENTE		
DIRECCIÓN: Ñuble 1414, Santiago			NOMBRE:		
TELÉFONO: 551 26 62 5516684 554 12 15			DIRECCIÓN:		
E-MAIL: INFO@ACECON.CL			TELÉFONO:		
PAGINA WEB: WWW.ACECON.CL			Cel :		
DATOS DEL TÉCNICO			GARANTÍA: 1 AÑO		
CARGO : GERENTE OPERACIONES			FECHA PRESUPUESTO: :01/2010		
NOMBRE: JOSE BARRAZA P.			EJECUTIVO COMERCIAL:		
DIRECCIÓN: Ñuble N° 1414, Santiago			DURACIÓN DEL TRABAJO: 07 días aprox.		
TELÉFONO:5516684 5541215			FORMA DE PAGO		
CARNÉ sec. N° 9,098,115.3			ADMINISTRACIÓN: 50% anticipo, 50% al Termino.		
DEPTO	CODIGO	TRABAJO A REALIZAR	NETO	IVA	sub-TOTAL
	0	(f/n)	\$ 0	\$ 0	
	0	(f/n)	\$ 0	\$ 0	
	0	(f/n)	\$ 0	\$ 0	\$ 0
	0	(f/n)	\$ 0	\$ 0	
	0	(f/n)	\$ 0	\$ 0	
	0	(f/n)	\$ 0	\$ 0	\$ 0
	0	(f/n)	\$ 0	\$ 0	
			TOTAL	\$ 0	\$ 0

Figura 11.1. Presupuesto Acecon.

Generar Presupuesto

Entrada	Proceso	Salida
Garantía + Duración + Fecha Creación + Obras + Observaciones + Cliente + Vendedor + Técnico.	El usuario busca al cliente, selecciona al vendedor y técnico que corresponda para posteriormente agregar las obras a realizar y la información propia del presupuesto (Fecha Creación, Garantía, Duración, Observaciones).	Mensajes de lo que está ocurriendo.

Tabla 11.27. Generar presupuesto.

Modificar Presupuesto búsqueda por nombre.

Entrada	Proceso	Salida
Nombre + {datos que se van a modificar}	El sistema comprueba si los datos del cliente ingresado son válidos y si el nombre de este ya existe.	Información del presupuesto o bien una lista con los nombres de los clientes que tienen creados presupuestos donde existan coincidencias.

Tabla 11.28. Modificar presupuesto búsqueda por nombre.

Modificar Presupuesto búsqueda por id.

Entrada	Proceso	Salida
Id + {datos que se van a modificar}	El sistema comprueba si el id del presupuesto ingresado es válido y si este ya existe.	Información del presupuesto.

Tabla 11.29. Modificar presupuesto búsqueda por id.

Presupuesto búsqueda por nombre.

Entrada	Proceso	Salida
Nombre	El sistema comprueba si los datos del cliente ingresado son válidos y si el nombre de este ya existe.	Información del presupuesto o bien una lista con los nombres de los clientes que tienen creados presupuestos donde existan coincidencias.

Tabla 11.30. Presupuesto búsqueda por nombre.

Presupuesto búsqueda por id.

Entrada	Proceso	Salida
Id	El sistema comprueba si el id del presupuesto ingresado es válido y si este ya existe.	Información del presupuesto.

Tabla 11.31. Presupuesto búsqueda por id.

d) Gestión de OT

Introducción: El sistema debe permitir generar ordenes de trabajo, detallando la obra a realizar y las observaciones pertinentes. Se debe especificar los datos del cliente y técnico encargado asignado a la obras. Al momento de generar una orden de trabajo el usuario primero debe buscar el presupuesto correspondiente. Una vez generada la orden, esta queda en estado pendiente, para que posteriormente una vez realizada, el técnico en terreno pueda cerrar el trabajo cambiando el estado ha realizado. A continuación en la Figura 11.2 se especifica como debiera ser una aproximación a la orden de trabajo generada.

ACECON		ORDEN DE TRABAJO			Nº 001148		
ACECON LTDA. CONSTRUCCION E INSTALACION RUT: 76.054.038-2 ATENCION CLIENTES OFICINA SANTIAGO ÑUBLE Nº 1414 - STGO TELEFONOS: (2) 551 66 84 (2) 554 1215		INFORMACION DEL CLIENTE					
		NOMBRE: _____ TELEFONO(S) _____ DIRECCION: _____ Nº _____ DEPTO _____ COMUNA _____ EMAIL _____					
		INFORMACION DEL TÉCNICO O RESPONSABLE DE LA ATENCION					
		NOMBRE _____ RUT: _____ CATEGORIA/CLASE _____					
		INFORMACION DEL VENDEDOR					
		NOMBRE: _____ R.U.T.: _____ FECHA DE INICIO DE TRABAJO _____ HORA _____					
		DESCRIPCION DEL TRABAJO			VALORES		
		Nº	DETALLE	VALOR UNI.	NETO	IVA	TOTAL
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
		TOTAL					
OBSERVACIONES							

Figura 11.2. Orden de trabajo Acecon.

Generar OT

Entrada	Proceso	Salida
Cliente + Técnico + Vendedor + Obras + Observaciones + Fecha + Estado	El usuario busca el presupuesto, selecciona al vendedor y técnico que corresponda para posteriormente agregar las obras a realizar y las observaciones pertinentes.	Mensajes de lo que está ocurriendo.

Tabla 11.32. Generar OT.

Modificar OT búsqueda por id.

Entrada	Proceso	Salida
Id + {datos que se van a modificar}	El sistema comprueba si el id de la orden de trabajo ingresado es válido.	Información de la orden de trabajo.

Tabla 11.33. Modificar OT.

OT búsqueda por id.

Entrada	Proceso	Salida
Id	El sistema comprueba si el id de la orden de trabajo ingresado es válido.	Información de la orden de trabajo.

Tabla 11.34. OT búsqueda por id.

Descargar OT

Introducción: El sistema a través de una aplicación móvil debe permitir descargar las órdenes de trabajo asignadas al técnico. Una vez descargadas el usuario podrá cambiar el estado y dar por cerrada la OT, indicando observaciones si es que las hubiera. Para tener acceso a la aplicación el dispositivo móvil debe tener acceso a Internet.

Entrada	Proceso	Salida
Fecha	El sistema comprueba la fecha ingresada y descarga la OT.	Información de la orden de trabajo a realizar.

Tabla 11.35. Descargar OT.

Cerrar OT

Entrada	Proceso	Salida
Estado + Observación	La aplicación móvil registra el cambio de estado y las observaciones enviando al sistema central (aplicación web) la información que es guardada en la base de datos.	Información de lo que está sucediendo.

Tabla 11.36. Cerrar OT.

e) Gestión de privilegios de usuario

Introducción: El sistema debe tener distintos perfiles de acceso, dependiendo de la funcionalidad que se le quiera dar al usuario. Estos perfiles son los siguientes.

- Administrador del sistema: Usuario que tendrá el completo control de la aplicación web, podrá acceder a todos los módulos del sistema y administrar los perfiles de usuario.
- Vendedores: Usuario que podrá acceder al módulo de clientes, presupuestos y ordenes de trabajo. En estos dos últimos solo tendrá acceso de modo lectura, esto quiere decir que no podrá modificar los datos.
- Técnicos: Usuario que podrá acceder a la aplicación móvil para cerrar las ordenes de trabajo en terreno.
- Clientes: Usuario que podrá acceder al módulo de presupuestos de modo lectura, esto quiere decir que no podrá modificar los datos.

Crear Usuario

Entrada	Proceso	Salida
Rut + Pass+ Perfil	El sistema registra al usuario con su respectivo nivel de privilegios.	Información de lo que está sucediendo.

Tabla 11.37. Crear Usuario.

11.3.5 Suposiciones y dependencias

Suposiciones:

Se asume que los requisitos en este documento son estables una vez que sean aprobados por la empresa. Cualquier petición de cambios en la especificación debe ser aprobada por todas las partes intervinientes y será gestionada por el equipo de desarrollo.

Dependencias:

Para acceder al sistema el usuario debe estar registrado y tener acceso a internet, además de poseer un dispositivo móvil en el caso de ser técnicos.

11.3.6 Requisitos de usuario y tecnológicos

Requisitos de usuario:

Los usuarios serán personal de la empresa y clientes. Las interfaces deben ser intuitivas, fáciles de usar y amigables, de manera que con unas breves instrucciones los usuarios sean capaces de ocupar el sistema. Para lograr esto, se pretende realizar una capacitación que constara de dos sesiones que no superaran los 45 minutos. En el caso de los clientes el sistema deberá contar con una ventana de ayuda donde se le explicara al usuario los pasos a seguir para realizar las operaciones que desee.

Requisitos tecnológicos:

En vista de que la aplicación se ejecutara en varios computadores, y teniendo en cuenta un futuro crecimiento de la empresa, se ha optado por un entorno económico y fácil de manejar. La aplicación se ejecutará sobre un esquema cliente/servidor, con los procesos e interfaz de usuario ejecutándose en los clientes y éstos solicitando requerimientos al servidor que cumple su proceso. El sistema podrá ser ejecutado en cualquier computador que tenga acceso a navegadores tradicionales como Internet Explorer, Mozilla y Google Chrome. En el caso de la aplicación móvil, el dispositivo debe tener instalado Windows Mobile 6 y tener acceso a internet.

El sistema en un comienzo será implantado en los servidores de la Universidad, a la espera de que el cliente facilite esta tecnología y la base de datos que se ocupara será PostgreSQL 8.4.

11.3.7 Requisitos de interfaces externas

Interfaces de usuario:

La interfaz de usuario debe ser orientada a ventanas tipo Windows.

Interfaces Hardware:

Ratón, teclado estándar y dispositivo móvil.

11.3.8 Requisitos de rendimiento

El tiempo de respuesta de la aplicación a cada función solicitada por el usuario no debe ser superior a los 5 segundos.

11.3.9 Requisitos de desarrollo

El ciclo de vida será el desarrollo basado en Prototipo Evolutivo, debiendo orientarse hacia el desarrollo de un sistema flexible que permita incorporar de manera sencilla cambios y nuevas funcionalidades.

CAPÍTULO 12: DIAGRAMA DE CASOS DE USO

12.1 Caso de uso general

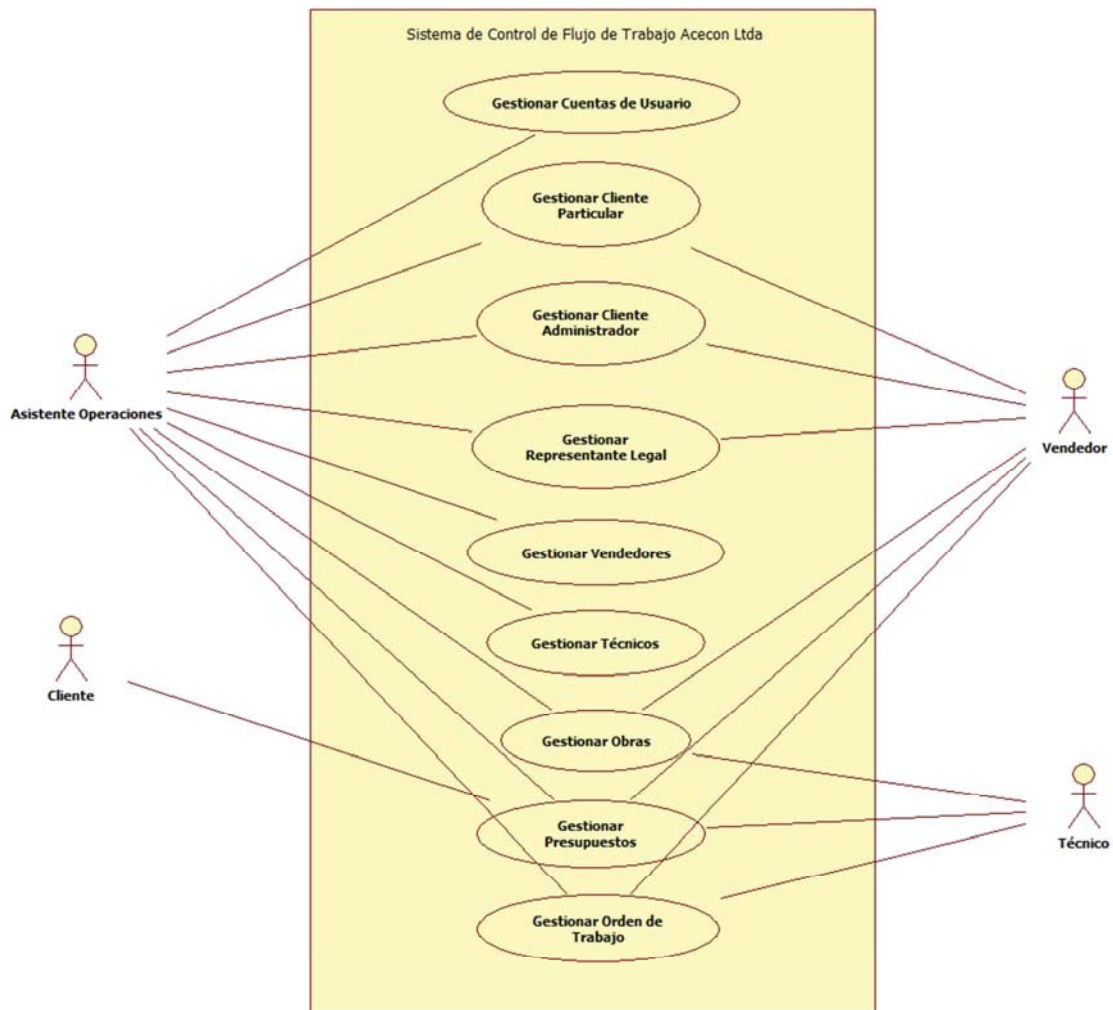


Figura 12.1. Diagrama General de Casos de Uso.

12.2 Especificación caso de uso diagrama general

Caso de Uso:	Gestionar Cuentas de Usuario
Actores:	Principal: Asistente de Operaciones
Tipo:	Principal.
Descripción:	El Asistente de Operaciones tendrá la posibilidad de administrar las cuentas de usuario de quienes acceden al sistema.

Tabla 12.1. Gestionar cuentas de usuario.

Caso de Uso:	Gestionar Cliente Particular
Actores:	Principal: Vendedor, Asistente de Operaciones
Tipo:	Principal.
Descripción:	Los actores antes nombrados podrán, agregar, buscar y modificar los datos del cliente Particular.

Tabla 12.2. Gestionar cliente particular.

Caso de Uso:	Gestionar Cliente Administrador
Actores:	Principal: Vendedor, Asistente de Operaciones
Tipo:	Principal.
Descripción:	Los actores antes nombrados podrán, agregar, buscar y modificar los datos del cliente Administrador.

Tabla 12.3. Gestionar cliente administrador.

Caso de Uso:	Gestionar Representante Legal
Actores:	Principal: Vendedor, Asistente de Operaciones
Tipo:	Principal.
Descripción:	Los actores antes nombrados podrán, agregar, buscar y modificar los datos de los Representantes legales de los edificios con los cuales la empresa trabaja.

Tabla 12.4. Gestionar representante legal.

Caso de Uso:	Gestionar Vendedores
Actores:	Principal: Asistente de Operaciones
Tipo:	Principal.
Descripción:	El Asistente de Operaciones podrá, agregar, buscar y modificar los datos de los vendedores que trabajan o ingresan a la empresa.

Tabla 12.5. Gestionar vendedores.

Caso de Uso:	Gestionar Técnicos
Actores:	Principal: Asistente de Operaciones
Tipo:	Principal.
Descripción:	El Asistente de operaciones podrá, agregar, buscar y modificar los datos de los Técnicos de la empresa.

Tabla 12.6. Gestionar técnicos.

Caso de Uso:	Gestionar obras
Actores:	Principal: Asistente de operaciones, Técnico, Vendedor.
Tipo:	Principal.
Descripción:	El Asistente de Operaciones podrá, agregar, buscar, modificar y dar de baja los distintos trabajos ofrecidos por la empresa, que van directamente a la realización de los presupuestos, teniendo control de los valores y nombres asignados a cada obra. Mientras que el Técnico y el Vendedor podrán solamente acceder a la opción de buscar obras.

Tabla 12.7. Gestionar obras.

Caso de Uso:	Gestionar Presupuesto
Actores	Principal: Asistente de Operaciones, Vendedor, Cliente, Técnico.
Tipo:	Principal.
Descripción:	El Asistente de Operaciones podrá crear nuevos presupuestos, buscar y modificar presupuestos ya creados, acceder a módulos como el de clientes y trabajadores para así asignar información ya ingresada, como los datos del vendedor y del técnico. Además podrá actualizar el IVA para el cálculo de los valores en el presupuesto. El Vendedor, Cliente y Técnico solo podrán ver los presupuestos.

Tabla 12.8. Gestionar presupuesto.

Caso de Uso:	Gestionar Orden de Trabajo
Actores	Principal: Asistente de Operaciones, Vendedor y Técnico.
Tipo:	Principal.
Descripción:	El Asistente de Operaciones podrá crear nuevas órdenes de trabajo mediante un presupuesto ya aceptado anteriormente, en estas órdenes de trabajo se especificarán todas las obras a realizar en terreno, así como los datos del cliente donde deben realizarse, el técnico que debe ejecutarlas, la fecha y hora. Una vez que las órdenes de trabajo hayan sido asignadas, el técnico podrá descargarlas diariamente en su dispositivo móvil para su realización. El Vendedor y Técnico solo podrán buscar órdenes de trabajo ya generadas.

Tabla 12.9. Gestionar orden de trabajo.

12.3 Caso de uso gestionar cliente particular - Grafico extendido

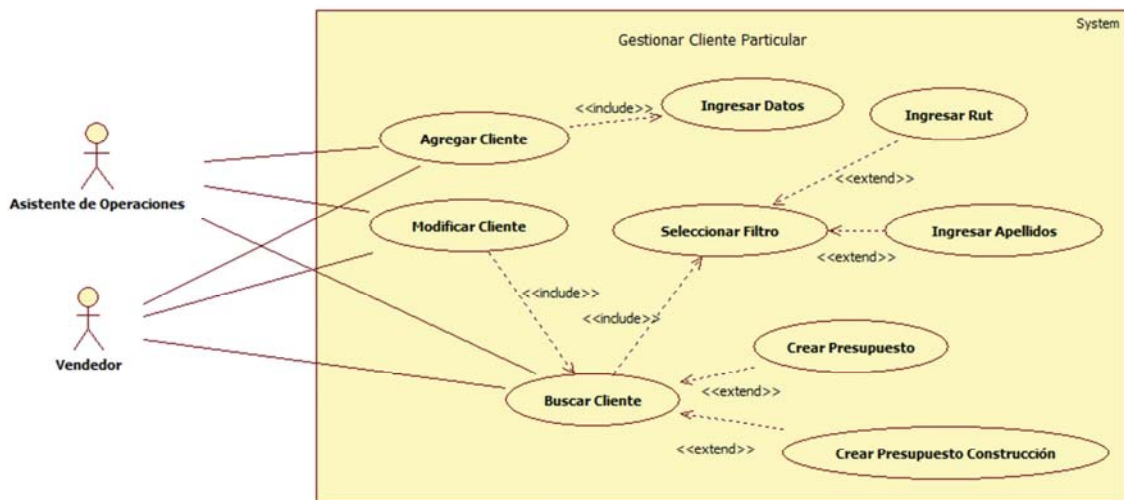


Figura 12.2. Caso de Uso Gestionar Cliente Particular.

12.3.1 Caso de uso gestionar cliente particular – Grafico narrativo extendido

Caso de Uso:	Gestionar cliente Particular.
Actor Principal(s):	Asistente de operaciones, Vendedor.
Actor Secundario (s):	
Propósito:	Gestionar los clientes Particulares de la empresa.
Tipo:	Principal.
Descripción:	El Asistente de operaciones i/o Vendedor podrá realizar distintas acciones sobre los clientes de la empresa.

Curso normal de los eventos:	
ACTORES	RESPUESTA DEL SISTEMA
1. Seleccionar del menú principal la opción “Clientes Particular”.	2. Retorna las opciones: A. Ingresar cliente B. Modificar/Buscar cliente C. Volver
3. Si el usuario selecciona: A. Ingresar cliente B. Modificar/Buscar cliente C. Volver	4. A. Va al paso 5 B. Va al paso 9 C. Va al paso 16
	5. Se solicita ingresar los datos del cliente
6. El usuario ingresa los datos del cliente	7. El sistema verifica que el cliente no exista y que los datos ingresados sean válidos.
	8. Se ingresa el cliente al sistema y se muestra un mensaje de que fue ingresado exitosamente. Salta al paso 14.
	9. El sistema pide ingresar el rut del cliente o bien los apellidos.
10. El usuario ingresa el rut o los apellidos del cliente.	11. El sistema verifica si el rut o los apellidos ingresados son válidos y si el cliente existe, mostrando en pantalla todos los datos del cliente que pueden ser modificados. En el caso de ser la búsqueda por apellidos el sistema despliega una lista con los apellidos y sus respectivos nombres que encuentre.
12. El usuario modifica los datos en pantalla que desee y selecciona el botón modificar. En caso de ser la búsqueda por apellidos selecciona el cliente que se desea modificar.	13. El sistema verifica que los nuevos datos ingresados son válidos y guarda las modificaciones realizadas arrojando un mensaje de éxito. Salta al paso 14

14. Menú Agregar Cliente.	
15. Menú Buscar/Modificar cliente.	
16. Menú Portada.	
Curso alternativo de los eventos:	
	<p>7'. <i>Los datos ingresados no son válidos:</i> El sistema arroja un mensaje indicando de que los datos ingresados no son válidos. Salta al paso 14</p> <p>7'' <i>El cliente ya existe:</i> El sistema arroja un mensaje indicando que el cliente ya fue registrado anteriormente. Salta al paso 24</p>
	<p>11' <i>El rut ingresado no es válido:</i> El sistema envía un mensaje indicando que el rut ingresado no es válido. Salta al paso 15.</p> <p>11'' <i>El cliente no existe:</i> El sistema envía un mensaje indicando que el cliente no existe. Salta al paso 15.</p> <p>11''' <i>La lista de cliente desplegada por el sistema no concuerda con el nombre del cliente buscado.</i> El sistema le da la opción de volver al paso 15.</p>

Tabla 12.10. Caso de uso gestionar cliente particular.

12.4 Caso de uso gestionar cliente administración - Grafico extendido

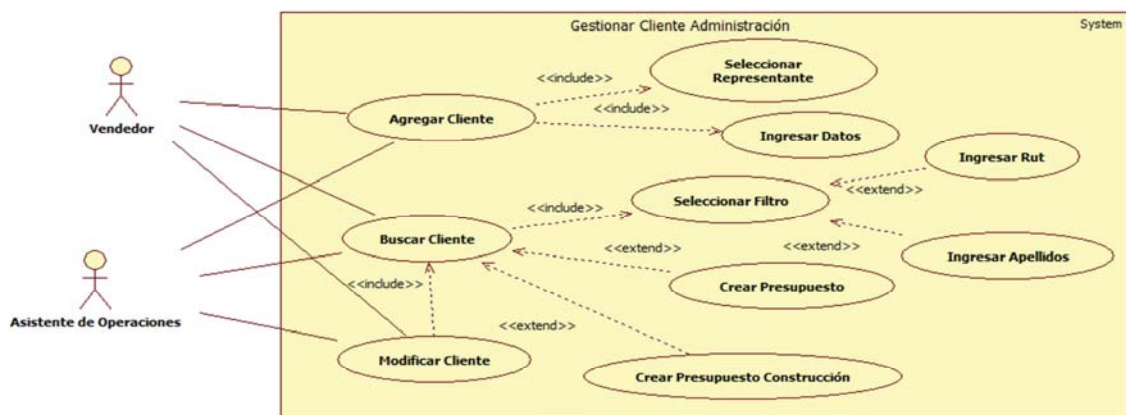


Figura 12.3. Caso de Uso Gestionar Cliente Administración.

12.4.1 Caso de uso gestionar cliente administración – Grafico narrativo extendido

Caso de Uso:	Gestionar cliente Administración.
Actor Principal(s):	Asistente de operaciones, Vendedor.
Actor Secundario (s):	
Propósito:	Gestionar los Clientes Administradores de la empresa.
Tipo:	Principal.
Descripción:	El Asistente de Operaciones y/o Vendedor podrán realizar distintas acciones sobre los clientes de la empresa.

Curso normal de los eventos:	
ACTORES	RESPUESTA DEL SISTEMA
1. Seleccionar del menú principal la opción “Clientes Administración”.	2. Retorna las opciones: A. Ingresar cliente B. Buscar/Modificar cliente C. Volver
3. Si el Usuario selecciona: A. Ingresar cliente B. Modificar/Buscar cliente C. Volver	4. A. Va al paso 5 B. Va al paso 9 C. Va al paso 16
	5. Se solicita ingresar los datos del cliente.
6. El Usuario ingresa los datos del cliente.	7. El sistema verifica que el cliente no exista y que los datos ingresados sean válidos.
	8. Se ingresa el cliente al sistema seleccionando un representante legal previamente ingresado y se muestra un mensaje de que fue ingresado exitosamente. Salta al paso 14.
	9. El sistema pide ingresar el Rut del cliente o bien los apellidos.
10. El usuario ingresa el rut o los apellidos del cliente.	11. El sistema verifica si el rut o los apellidos ingresados son válidos y si el cliente existe, mostrando en pantalla todos los datos del cliente que pueden ser modificados. En el caso de ser la búsqueda por apellidos el sistema despliega una lista con los apellidos y sus respectivos nombres que encuentre.
12. El usuario modifica los datos en pantalla	13. El sistema verifica que los nuevos

que desee y selecciona el botón modificar. En caso de ser la búsqueda por apellidos selecciona el cliente que se desea modificar.	datos ingresados son válidos y guarda las modificaciones realizadas arrojando un mensaje de éxito. Salta al paso 14
14. Menú Agregar Cliente.	
15. Menú Buscar/Modificar Cliente.	
16. Menú Portada.	
Curso alternativo de los eventos:	
	<p><i>7'. Los datos ingresados no son válidos:</i> El sistema arroja un mensaje indicando de que los datos ingresados no son válidos. Salta al paso 14</p> <p><i>7'' El cliente ya existe:</i> El sistema arroja un mensaje indicando que el cliente ya fue registrado anteriormente. Salta al paso 24</p>
	<p><i>11' El Rut ingresado no es válido:</i> El sistema envía un mensaje indicando que el rut ingresado no es válido. Salta al paso 15.</p> <p><i>11'' El cliente no existe:</i> El sistema envía un mensaje indicando que el cliente no existe. Salta al paso 15.</p> <p><i>11''' La lista de cliente desplegada por el sistema no concuerda con el nombre del cliente buscado.</i> El sistema le da la opción de volver al paso 15.</p>

Tabla 12.11. Caso de uso gestionar cliente administración.

12.5 Caso de uso gestionar técnicos – Grafico extendido

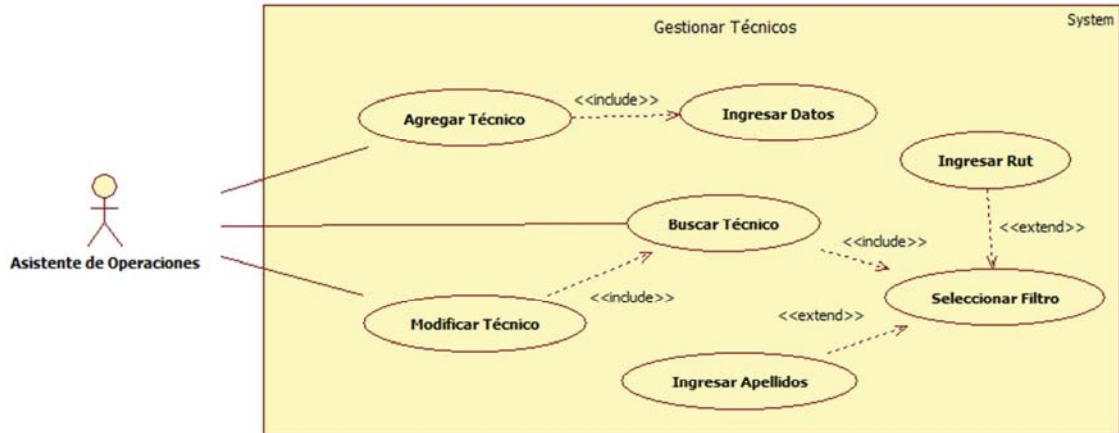


Figura 12.4. Caso de Uso Gestionar Técnicos.

12.5.1 Caso de uso gestionar técnicos - Grafico narrativo extendido

Caso de Uso:	Gestionar técnicos.
Actor Principal(s):	Asistente de operaciones.
Actor Secundario (s):	
Propósito:	Gestionar los técnicos que trabajan en la empresa.
Tipo:	Principal.
Descripción:	El Asistente de operaciones podrá realizar distintas acciones sobre los técnicos que trabajan en la empresa.

Curso normal de los eventos:	
ACTORES	RESPUESTA DEL SISTEMA
1. Seleccionar del menú principal la opción "Técnico".	2. Retorna las opciones: A. Ingresar técnico. B. Modificar técnico. C. Buscar técnico. D. Volver
3. Si el usuario selecciona: A. Ingresar técnicos. B. Modificar técnicos. C. Buscar técnico. D. Volver	4. A. Va al paso 5 B. Va al paso 9 C. Va al paso 14 D. Va al paso 21
	5. Se solicita ingresar los datos del técnico.

6. El usuario ingresa los datos del técnico.	7. El sistema verifica que el técnico no exista y que los datos ingresados sean válidos.
	8. Se ingresa el técnico al sistema y se muestra un mensaje de que fue ingresado exitosamente. Salta al paso 18.
	9. El sistema pide ingresar el rut del técnico o bien los apellidos.
10. El usuario ingresa el rut o los apellidos del técnico.	11. El sistema verifica si el rut o los apellidos ingresados son válidos y si el técnico existe, mostrando en pantalla todos los datos del técnico que pueden ser modificados. En el caso de ser la búsqueda por apellidos el sistema despliega una lista con los apellidos y sus respectivos nombres que encuentre.
12. El usuario modifica los datos en pantalla que desee y selecciona el botón modificar. En caso de ser la búsqueda por apellidos selecciona el técnico que desea modificar.	13. El sistema verifica que los nuevos datos ingresados son válidos y guarda las modificaciones realizadas arrojando un mensaje de éxito. Salta al paso 18
	14. Se pide ingresar el rut del técnico o bien los apellidos.
15. El usuario ingresa el rut del técnico o los apellidos.	16. El sistema verifica que el rut ingresado es válido y que existe, mostrando por pantalla todos los datos del técnico. En caso de ser la búsqueda por apellidos el sistema despliega una lista con los apellidos y sus respectivos nombres que encuentre.
17. Menú técnico.	
18. Menú Agregar técnico.	
19. Menú Modificar técnico.	
20. Menú Buscar técnico.	
21. Menú Portada	
Curso alternativo de los eventos:	
	<p>7'. <i>Los datos ingresados no son válidos:</i> El sistema arroja un mensaje indicando de que los datos ingresados no son válidos. Salta al paso 18</p> <p>7'' <i>El técnico ya existe:</i> El sistema arroja un mensaje indicando que el técnico ya fue registrado anteriormente. Salta al paso 18</p>
	<p>11' <i>El rut ingresado no es válido:</i> El sistema envía un mensaje indicando que</p>

	<p>el rut ingresado no es válido. Salta al paso 19.</p> <p>11'' <i>El técnico no existe:</i> El sistema envía un mensaje indicando que el técnico no existe. Salta al paso 19.</p> <p>11''' <i>La lista de técnicos desplegada por el sistema no concuerda con el nombre del técnico buscado.</i> El sistema le da la opción de volver al paso 19.</p>
	<p>14' <i>El rut ingresado no es válido:</i> El sistema envía un mensaje indicando que el rut ingresado no es válido. Salta al paso 20.</p> <p>14'' <i>El técnico no existe:</i> El sistema envía un mensaje indicando que el técnico no existe. Salta al paso 20.</p> <p>14''' <i>La lista de técnicos desplegada por el sistema no concuerda con el nombre del técnico buscado.</i> El sistema le da la opción de volver al paso 20.</p>

Tabla 12.12. Caso de uso gestionar técnicos.

12.6 Caso de uso gestionar vendedores – Grafico extendido

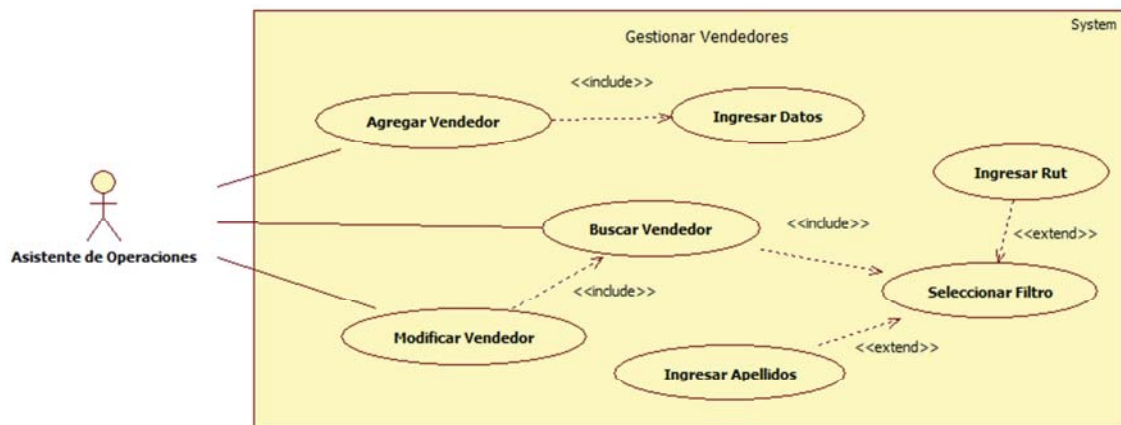


Figura 12.5. Caso de Uso Gestionar Vendedores.

12.6.1 Caso de uso gestionar vendedores – Grafico narrativo extendido

Caso de Uso:	Gestionar Vendedores.
Actor Principal(s):	Asistente de operaciones.
Actor Secundario (s):	
Propósito:	Gestionar los vendedores que trabajan en la empresa.
Tipo:	Principal.
Descripción:	El Asistente de operaciones podrá realizar distintas acciones sobre los vendedores que trabajan en la empresa.

Curso normal de los eventos:	
ACTORES	RESPUESTA DEL SISTEMA
1. El usuario selecciona del menú principal la opción “Vendedores”.	2. Retorna las opciones: A. Ingresar vendedor B. Modificar vendedor C. Buscar vendedor D. Volver
3. Si el usuario selecciona: A. Ingresar vendedor B. Modificar vendedor C. Buscar vendedor D. Volver	4. A. Va al paso 5 B. Va al paso 9 C. Va al paso 20 D. Va al paso 21
	5. Se solicita ingresar los datos del vendedor
6. El usuario ingresa los datos del vendedor	7. El sistema verifica que el vendedor no exista y que los datos ingresados sean válidos.
	8. Se ingresa el vendedor al sistema y se muestra un mensaje de que fue ingresado exitosamente. Salta al paso 23.
	9. El sistema pide ingresar el rut del vendedor o bien los apellidos
10. El usuario ingresa el rut o los apellidos del vendedor.	11. El sistema verifica si el rut o los apellidos ingresados son válidos y si el vendedor existe, mostrando en pantalla todos los datos del vendedor que pueden ser modificados. En el caso de ser la búsqueda por apellidos el sistema despliega una lista con apellidos y sus respectivos nombres que encuentre.
12. El usuario modifica los datos en pantalla que desee y selecciona el botón modificar. En	13. El sistema verifica que los datos ingresados son válidos y guarda las

caso de ser la búsqueda por apellidos selecciona el vendedor que desea modificar.	modificaciones realizadas arrojando un mensaje de éxito. Salta al paso 23
	14. Se pide ingresar el rut del vendedor o bien los apellidos.
15. El usuario ingresa el rut del vendedor o los apellidos.	16. El sistema verifica que el rut ingresado es válido y que existe, mostrando por pantalla todos los datos del vendedor. En el caso de ser la búsqueda por apellidos el sistema despliega una lista con los apellidos y sus respectivos nombres que encuentre.
17. Menú vendedor.	
18. Menú Agregar vendedor.	
19. Menú Modificar vendedor.	
20. Menú Buscar vendedor.	
21 Menú Portada	
Curso alternativo de los eventos:	
	<p>7'. <i>Los datos ingresados no son válidos:</i> El sistema arroja un mensaje indicando de que los datos ingresados no son válidos. Salta al paso 18</p> <p>7'' <i>El vendedor ya existe:</i> El sistema arroja un mensaje indicando que el vendedor ya fue registrado anteriormente. Salta al paso 18</p>
	<p>11' <i>El rut ingresado no es válido:</i> El sistema envía un mensaje indicando que el rut ingresado no es válido. Salta al paso 19.</p> <p>11'' <i>El vendedor no existe:</i> El sistema envía un mensaje indicando que el vendedor no existe. Salta al paso 19.</p> <p>11''' <i>La lista de vendedores desplegada por el sistema no concuerda con el nombre del vendedor buscado.</i> El sistema le da la opción de volver al paso 19.</p>

Tabla 12.13. Caso de uso gestionar vendedores.

12.7 Caso de uso gestionar representante – Grafico extendido

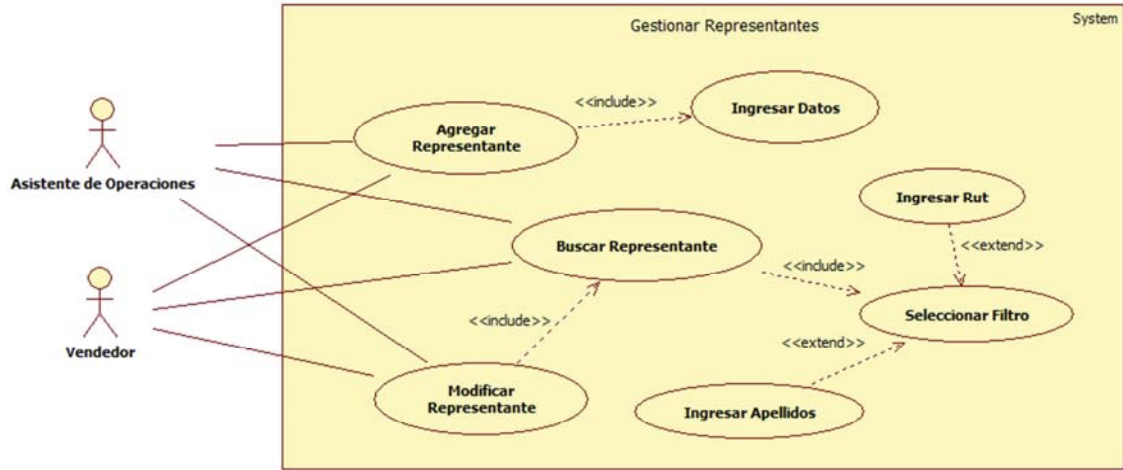


Figura 12.6. Caso de Uso Gestionar Representante.

12.7.1 Caso de uso gestionar representante – Grafico narrativo extendido

Caso de Uso:	Gestionar Representantes.
Actor Principal(s):	Asistente de Operaciones, Vendedor.
Actor Secundario (s):	
Propósito:	Gestionar los Representantes legales de los edificios a los cuales la empresa hace labores.
Tipo:	Principal
Descripción:	El Asistente de Operaciones y/o Vendedor podrán realizar distintas acciones sobre los representantes legales de los edificios o condominios.

Curso normal de los eventos:	
ACTORES	RESPUESTA DEL SISTEMA
1. Seleccionar del menú principal la opción "Representantes".	2. Retorna las opciones: A. Ingresar representante B. Modificar representante C. Buscar representante D. Volver
3. Si el usuario selecciona: A. Ingresar representante B. Modificar representante C. Buscar representante	4. A. Va al paso 5 B. Va al paso 9 C. Va al paso 20

D. Volver	D. Va al paso 21
	5. Se solicita ingresar los datos del representante
6. El usuario ingresa los datos del representante.	7. El sistema verifica que el representante no exista y que los datos ingresados sean válidos.
	8. Se ingresa el representante al sistema y se muestra un mensaje de que fue ingresado exitosamente. Salta al paso 23.
	9. El sistema pide ingresar el rut del representante o bien los apellidos
10. El usuario ingresa el rut o los apellidos del representante.	11. El sistema verifica si el rut o los apellidos ingresados son válidos y si el representante existe, mostrando en pantalla todos los datos del representante que pueden ser modificados. En el caso de ser la búsqueda por apellidos el sistema despliega una lista con apellidos y sus respectivos nombres que encuentre.
12. El usuario modifica los datos en pantalla que desee y selecciona el botón modificar. En caso de ser la búsqueda por apellidos selecciona el representante que desea modificar.	13. El sistema verifica que los datos ingresados son válidos y guarda las modificaciones realizadas arrojando un mensaje de éxito. Salta al paso 23
	14. Se pide ingresar el rut del representante o bien los apellidos.
15. El usuario ingresa el rut del representante o los apellidos.	16. El sistema verifica que el rut ingresado es válido y que existe, mostrando por pantalla todos los datos del representante. En el caso de ser la búsqueda por apellidos el sistema despliega una lista con los apellidos y sus respectivos nombres que encuentre.
17. Menú representante.	
18. Menú Agregar representante.	
19. Menú Modificar representante.	
20. Menú Buscar representante.	
21 Menú Portada	
Curso alternativo de los eventos:	
	7'. Los datos ingresados no son válidos: El sistema arroja un mensaje indicando de que los datos ingresados no son válidos.

	<p>Salta al paso 18</p> <p>7'' <i>El representante ya existe:</i> El sistema arroja un mensaje indicando que el representante ya fue registrado anteriormente. Salta al paso 18.</p>
	<p>11' <i>El rut ingresado no es válido:</i> El sistema envía un mensaje indicando que el rut ingresado no es válido. Salta al paso 19.</p> <p>11'' <i>El representante no existe:</i> El sistema envía un mensaje indicando que el representante no existe. Salta al paso 19.</p> <p>11''' <i>La lista de representantes desplegada por el sistema no concuerda con el nombre del representante buscado.</i> El sistema le da la opción de volver al paso 19.</p>
	<p>16' <i>El rut ingresado no es válido:</i> El sistema envía un mensaje indicando que el rut ingresado no es válido. Salta al paso 20.</p> <p>16'' <i>El representante no existe:</i> El sistema envía un mensaje indicando que el representante no existe. Salta al paso 20.</p> <p>16''' <i>La lista de representantes desplegada por el sistema no concuerda con el nombre del representante buscado.</i> El sistema le da la opción de volver al paso 20.</p>

Tabla 12.14. Caso de uso gestionar representante.

12.8 Caso de uso gestionar obras – Grafico extendido

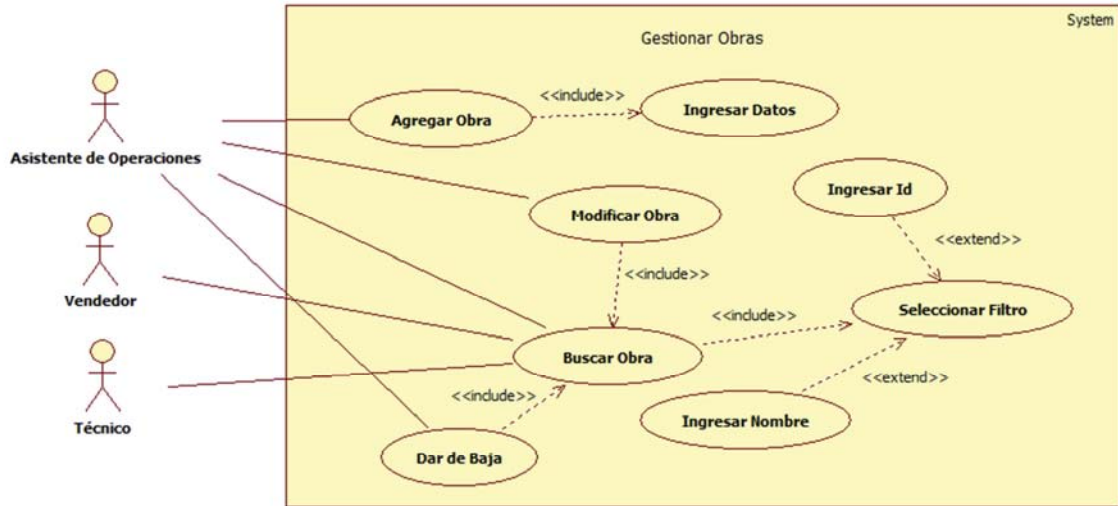


Figura 12.7. Caso de Uso Gestionar Obras.

12.8.1 Caso de uso gestionar obras – Grafico narrativo extendido

Caso de Uso:	Gestionar Obras.
Actor Principal(s):	Asistente de operaciones, Vendedor y Técnico.
Actor Secundario (s):	
Propósito:	Gestionar las obras que realiza la empresa.
Tipo:	Principal.
Descripción:	El Asistente de operaciones podrá realizar distintas operaciones con las obras que realiza la empresa.

Curso normal de los eventos:	
ACTORES	RESPUESTA DEL SISTEMA
1. Seleccionar del menú principal la opción "Obras".	2. Retorna las opciones: A. Ingresar obra. B. Modificar obra. C. Dar de baja. D. Buscar obra.
3. Si el usuario selecciona: A. Ingresar obra. B. Modificar obra. C. Dar de baja. D. Buscar obra.	4. A. Va al paso 5 B. Va al paso 9 C. Va al paso 14 D. Va al paso 20
	5. Se solicita ingresar los datos de la obra.

6. El usuario ingresa los datos de la obra.	7. El sistema verifica los datos de la obra y si el nombre de esta ya existe.
	8. Se ingresa la obra al sistema y se muestra un mensaje de que fue ingresada exitosamente. Salta al paso 23.
	9. El sistema pide ingresar el ID de la obra o bien el nombre.
10. El usuario ingresa el id de la obra o bien el nombre	11. El sistema verifica si los datos ingresados son válidos y si la obra existe, mostrando en pantalla todos los datos de la obra que pueden ser modificados. En el caso de ser la búsqueda por nombres el sistema despliega una lista con las obras que encuentre.
12. El usuario modifica los datos en pantalla que desee y selecciona el botón modificar. En caso de ser la búsqueda por nombre, selecciona la obra que desea modificar.	13. El sistema verifica que los datos ingresados son válidos y guarda las modificaciones realizadas arrojando un mensaje de éxito. Salta al paso 23.
	14. El sistema pide ingresar el id de la obra o bien el nombre.
15. El usuario ingresa el id o el nombre de la obra.	16. El sistema verifica si los datos ingresados son válidos y si la obra existe, En el caso de ser la búsqueda por nombre el sistema despliega una lista con los nombres de las obras que encuentre para que el asistente seleccione la que va ser dada de baja.
	17. El sistema arroja un mensaje de advertencia indicando que se va a ser dada de baja la obra, mostrando por pantalla las opciones de aceptar o cancelar.
18. El asistente oprime el botón aceptar.	19. El sistema cambia el estado de la obra y se muestra por pantalla un mensaje de que la operación realizada fue correcta. Salta al paso 23.
	20. El sistema pide ingresar el id de la obra o bien el nombre.
21. El usuario ingresa el id o el nombre de la obra.	22. El sistema verifica si los datos ingresados son válidos y si la obra existe, En el caso de ser la búsqueda por nombre el sistema despliega una lista con los nombres de las obras que encuentre.
23. Menú obra.	
24. Menú Agregar obra.	
25. Menú Modificar obra.	
26. Menú Eliminar obra.	

27. Menú Buscar obra.	
Curso alternativo de los eventos:	
	<p>7'. <i>Los datos ingresados no son válidos:</i> El sistema arroja un mensaje indicando de que los datos ingresados no son válidos. Salta al paso 24</p> <p>7'' <i>El nombre de la obra ya existe:</i> El sistema arroja un mensaje indicando que el nombre de la obra ya fue registrado anteriormente. Salta al paso 24.</p>
	<p>11' <i>El id ingresado no es válido:</i> El sistema envía un mensaje indicando que el id ingresado no es válido. Salta al paso 25.</p> <p>11'' <i>La obra no existe:</i> El sistema envía un mensaje indicando que la obra no existe. Salta al paso 25.</p> <p>11''' <i>La lista de obras desplegada por el sistema no concuerda con el nombre de la obra buscada.</i> El sistema le da la opción de volver al paso 25.</p>
	<p>16' <i>El id ingresado no es válido:</i> El sistema envía un mensaje indicando que el id ingresado no es válido. Salta al paso 26.</p> <p>16'' <i>La obra no existe:</i> El sistema envía un mensaje indicando que la obra no existe. Salta al paso 26.</p>
	<p>17' <i>El asistente presiona el botón cancelar.</i> Salta al paso 23.</p>
	<p>22'. <i>El id ingresado no es válido:</i> El sistema envía un mensaje indicando que el id ingresado no es válido. Salta al paso 27.</p> <p>22''. <i>La obra no existe:</i> El sistema envía un mensaje indicando que la obra no existe. Salta al paso 27.</p> <p>22''' <i>La lista de obras desplegada por el sistema no concuerda con el nombre de la obra buscada.</i> El sistema le da la opción de volver al paso 27.</p>

Tabla 12.15. Caso de uso gestionar obras.

12.9 Caso de uso gestionar orden de trabajo móvil – Grafico extendido



Figura 12.9. Caso de Uso Gestionar Orden de Trabajo Móvil.

12.9.1 Caso de uso gestionar orden de trabajo móvil - Descargar listado ot – Gráfico narrativo extendido

Caso de Uso:	Descargar Orden de Trabajo.
Pre-Condición:	El Técnico debe estar logueado.
Actor Principal(s):	Técnico
Actor Secundario (s):	Vendedor, Asistente de Operaciones
Propósito:	Descargar Órdenes de Trabajo a realizar.
Tipo:	Principal.
Descripción:	El usuario del dispositivo móvil podrá descargar todas las órdenes de trabajo correspondientes a él para el día en curso, estas órdenes se encontrarán listadas por horario para facilitar la labor de búsqueda.

Curso normal de los eventos:	
ACTORES	RESPUESTA DEL SISTEMA
1. El Técnico selecciona del menú principal la opción “Descargar Órdenes de Trabajo”.	2. Sistema envía información del técnico al servicio web y este retorna de existir las órdenes de trabajo asociadas al técnico

3. Técnico selecciona una de las órdenes de trabajo a realizar desde el ComboBox	4. Sistema despliega toda la información referente a la orden de trabajo seleccionada (datos del cliente, hora del trabajo, descripción y obras a realizar).
5. Técnico tras realizar la obra selecciona estado “cerrada” en el dispositivo o “pendiente” agregando observaciones de ser necesarias, y presiona ingresar	5. El sistema verifica el estado y envía la información al servicio web. Salta al paso 6
6. Menú Principal	
Curso alternativo de los eventos:	
	<p>2' <i>No se encuentran órdenes de trabajo asociadas al técnico:</i> El sistema arroja un mensaje indicando de que no existen órdenes de trabajo asociadas al técnico. Salta al paso 6</p> <p>2'' <i>No hay Internet:</i> El sistema despliega mensaje indicando problemas con la base de datos</p>
	<p>5' <i>Estado no seleccionado:</i> Sistema despliega mensaje indicando que no existe ningún estado seleccionado</p> <p>5'' <i>No hay Internet:</i> El sistema despliega mensaje indicando problemas con la base de datos</p>

Tabla 12.16. Caso de uso gestionar orden de trabajo móvil.

12.9.2 Caso de uso gestionar orden de trabajo móvil - Descarga listado ot pendiente – Gráfico narrativo extendido

Caso de Uso:	Descargar Orden de Trabajo Pendiente.
Pre-Condición:	El Técnico debe estar logueado.
Actor Principal(s):	Técnico
Actor Secundario (s):	Vendedor, Asistente de Operaciones.
Propósito:	Descargar Órdenes de Trabajo pendientes.
Tipo:	Principal.
Descripción:	El usuario del dispositivo móvil podrá descargar todas las órdenes de trabajo pendientes correspondientes a él para el día en curso, estas órdenes se encontrarán listadas por horario para facilitar la labor de búsqueda.

Curso normal de los eventos:	
ACTORES	RESPUESTA DEL SISTEMA
1. El Técnico selecciona del menú principal la opción “Descargar Órdenes de Trabajo Pendiente”.	2. Sistema envía información del técnico al servicio web y este retorna de existir las órdenes de trabajo pendientes asociadas al técnico.
3. Técnico selecciona una de las órdenes de trabajo a realizar desde el ComboBox.	4. Sistema despliega toda la información referente a la orden de trabajo seleccionada (datos del cliente, hora del trabajo, descripción, obras a realizar y las observaciones ingresadas previamente del porqué no se pudo concretar la orden de trabajo).
5. Técnico tras realizar la obra selecciona estado “cerrada” en el dispositivo o “pendiente” agregando observaciones de ser necesarias, y presiona ingresar.	5. El sistema verifica el estado y envía la información al servicio web. Salta al paso 6
6. Menú Principal	
Curso alternativo de los eventos:	
	2' <i>No se encuentran órdenes de trabajo asociadas al técnico:</i> El sistema arroja un mensaje indicando de que no existen órdenes de trabajo asociadas al técnico. Salta al paso 6 2'' <i>No hay Internet:</i> El sistema despliega mensaje indicando problemas con la base de datos.
	5' <i>Estado no seleccionado:</i> Sistema despliega mensaje indicando que no existe ningún estado seleccionado 5'' <i>No hay Internet:</i> El sistema despliega mensaje indicando problemas con la base de datos.

Tabla 12.17. Caso de uso gestionar orden de trabajo móvil.

12.10 Caso de uso gestionar presupuesto – Grafico extendido

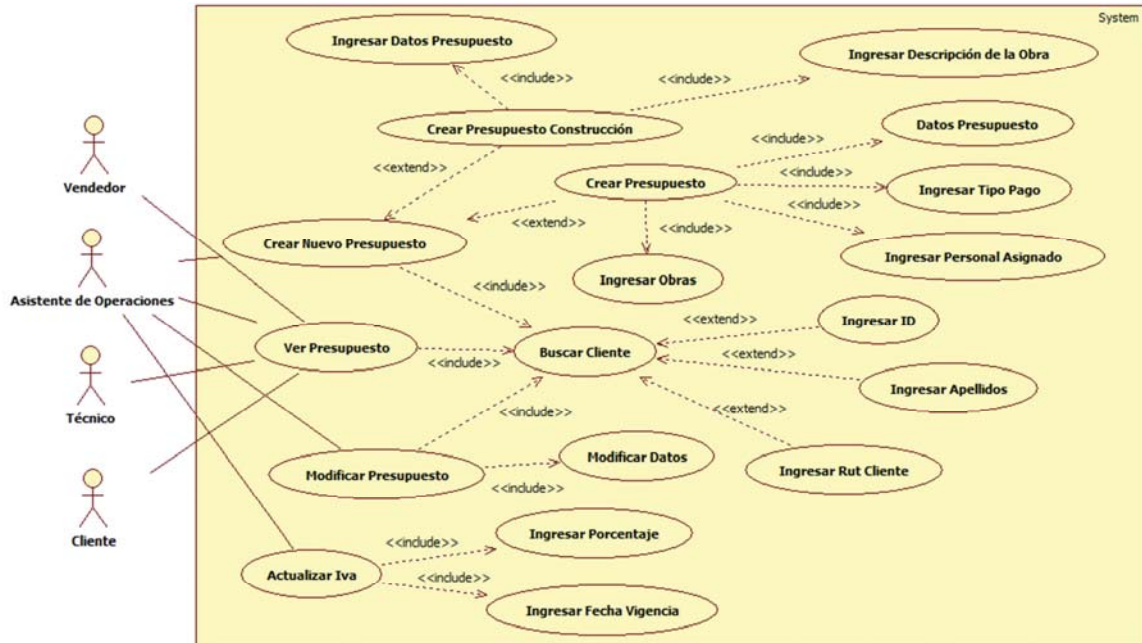


Figura 12.10. Caso de Uso Gestionar Presupuesto.

12.10.1 Caso de uso gestionar presupuesto – Gráfico narrativo extendido

Caso de Uso:	Gestionar Presupuesto
Pre-Condición:	Los usuarios deben estar logueados.
Actor Principal(s):	Asistente de Operaciones, Vendedor, Técnico, Cliente.
Actor Secundario (s):	
Propósito:	Gestionar Presupuestos.
Tipo:	Principal.
Descripción:	Los usuarios podrán crear nuevos presupuestos, modificar presupuestos, buscar presupuesto y actualizar el IVA en caso de que este varié.

Curso normal de los eventos:	
ACTORES	RESPUESTA DEL SISTEMA
1. El usuario selecciona del menú principal la opción "Presupuesto".	2. Retorna las opciones: A. Nuevo Presupuesto. B. Ver Presupuesto. C. Modificar. D. Actualizar IVA.

3. Si el usuario selecciona: A. Nuevo Presupuesto. B. Ver Presupuesto. C. Modificar. D. Actualizar IVA.	4. A. Va al paso 5 B. Va al paso 9 C. Va al paso 12 D. Va al paso 17
	5. Se solicita ingresar los datos del Presupuesto.
6. El usuario ingresa los datos del Presupuesto.	7. El sistema verifica que los datos ingresados sean válidos.
	8. Se genera el presupuesto y se muestra un mensaje éxito. Salta al paso 20.
	9. El sistema pide ingresar el rut del cliente o los apellidos o el id del presupuesto.
10. El usuario ingresa los datos solicitados.	11. El sistema verifica si los datos ingresados son válidos, desplegando por pantalla el presupuesto. En el caso de ser la búsqueda por apellidos el sistema despliega una lista con los apellidos y sus respectivos nombres que encuentre.
	12. El sistema pide ingresar el rut del cliente o los apellidos o el id del presupuesto.
13. El usuario ingresa los datos solicitados.	14. El sistema verifica si los datos ingresados son válidos, desplegando por pantalla el presupuesto a modificar. En el caso de ser la búsqueda por apellidos el sistema despliega una lista con los apellidos y sus respectivos nombres que encuentre.
15. El usuario modifica los datos en pantalla que desee y selecciona el botón modificar. En caso de ser la búsqueda por apellidos selecciona el presupuesto que desea modificar.	16. El sistema verifica que los nuevos datos ingresados son válidos y guarda las modificaciones realizadas arrojando un mensaje de éxito. Salta al paso 20.
	17. Se pide ingresar la fecha de vigencia y porcentaje de IVA.
18. El usuario ingresa los datos solicitados.	19. El sistema valida los datos ingresados y despliega por pantalla un mensaje de éxito. Salta al paso 20.
20. Menú Principal.	
Curso alternativo de los eventos:	
	7' <i>Los datos ingresados no son válidos:</i> El sistema arroja un mensaje indicando de que los datos ingresados no son válidos. Salta al paso 5.

	11' <i>No existen coincidencias para los datos ingresados:</i> El sistema despliega mensaje indicando que no existen coincidencias para los datos ingresados. Salta al paso 9.
	16' <i>Los datos ingresados son inválidos</i> El sistema arroja un mensaje indicando de que los datos ingresados no son válidos. Salta al paso 12.
	19' <i>Los datos ingresados para el IVA son inválidos</i> El sistema arroja un mensaje indicando de que los datos ingresados no son válidos. Salta al paso 17.

Tabla 12.18. Caso de uso gestionar presupuesto.

CAPÍTULO 13: DIAGRAMA DE CLASES Y MODELO DE DATOS

13.1 Diagrama de clases persistentes

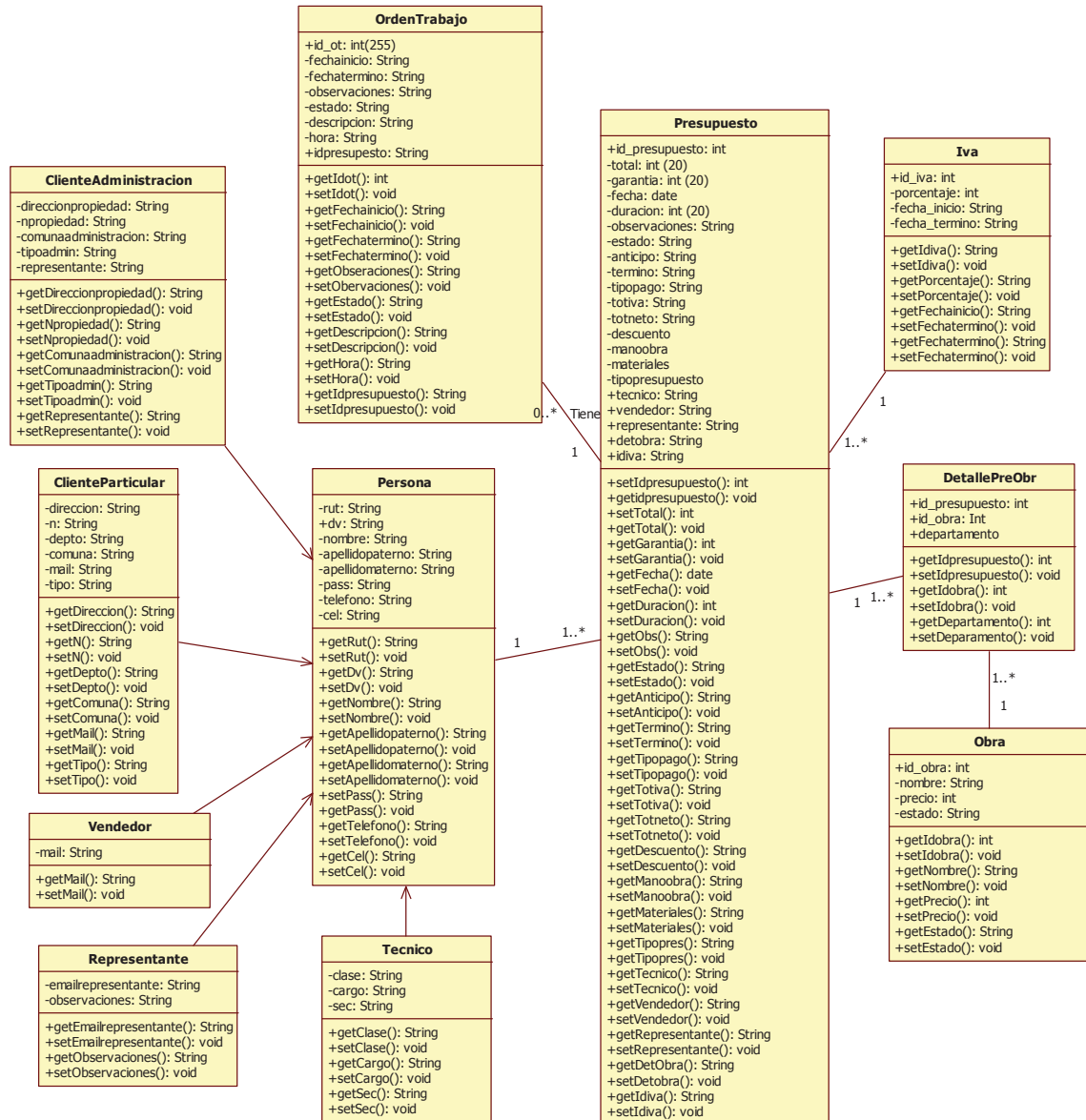


Figura 13.1. Diagrama de clases persistentes.

13.2 Diagrama de clases de lógica y acceso a datos

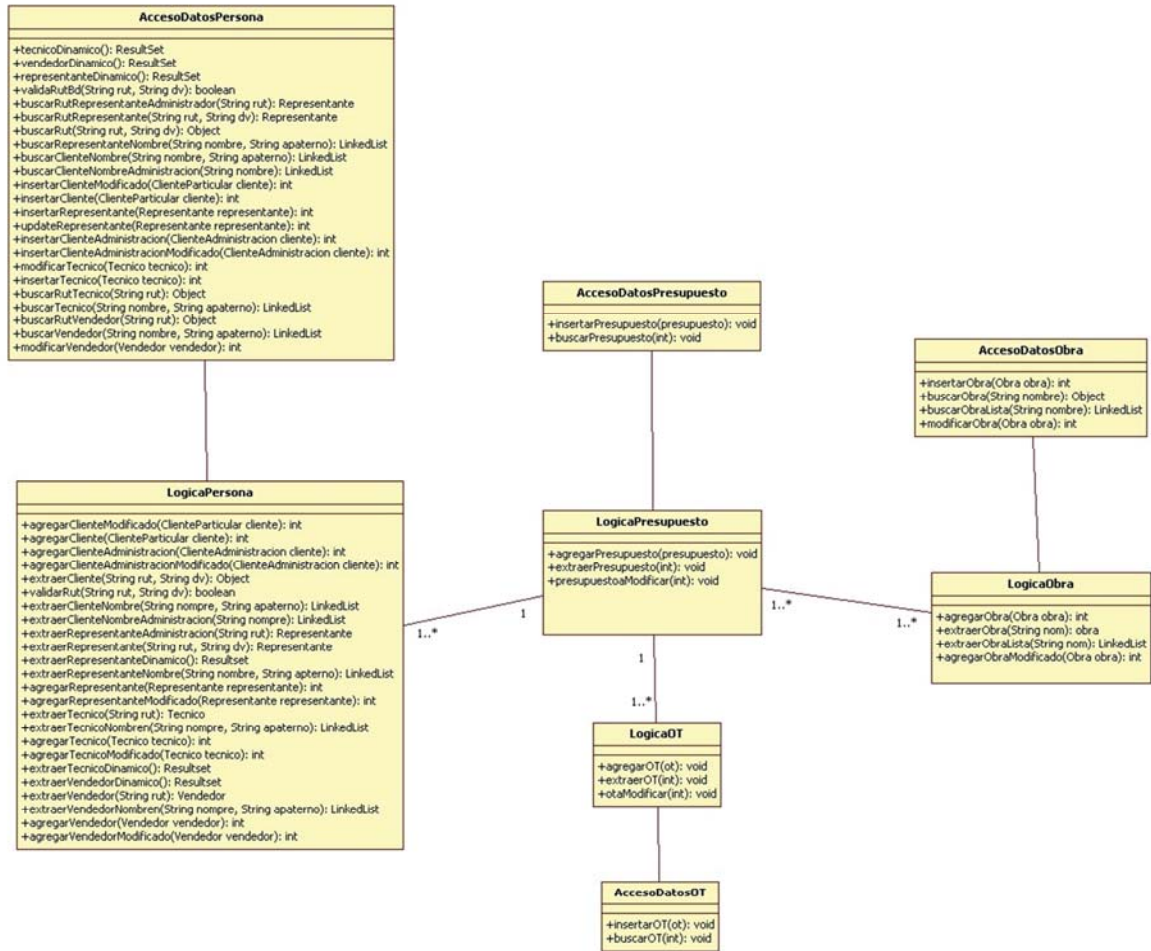


Figura 13.2. Diagrama de clases de lógica y acceso a datos.

13.3 Modelo entidad – relación

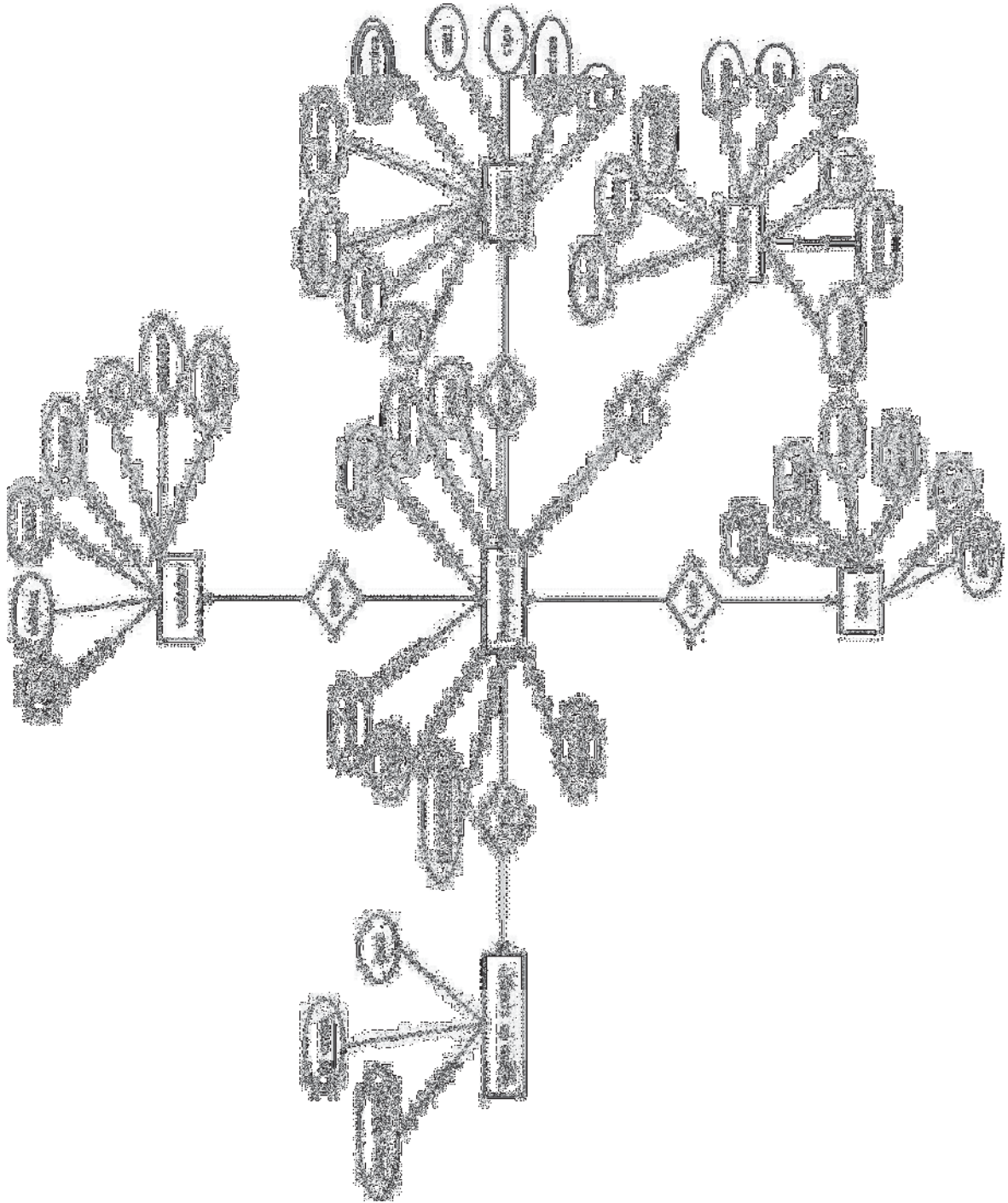


Figura 13.3. Modelo Entidad-Relación.

13.4 Modelo entidad-relación - refinado

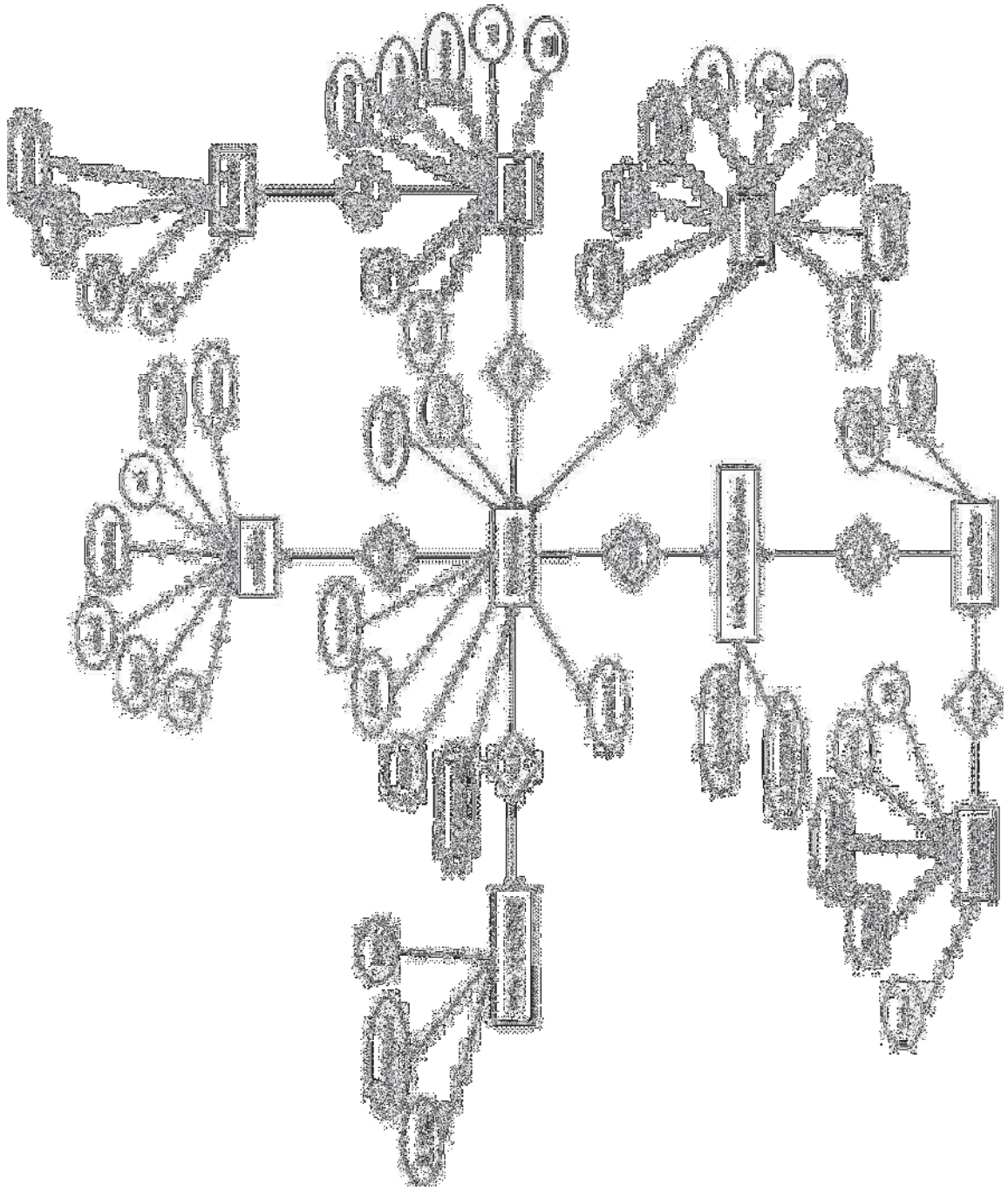


Figura 13.4. Modelo Entidad-Relación – Refinado.

13.5 Modelo relacional

El presente modelo tiene como función la de gestionar una base de datos, nos permite modelar problemas reales y administrar datos dinámicamente, en este modelo cada relación es considerada una dupla la cual puede ser tomada como una tabla que está compuesta por registros y columnas donde se pueden almacenar los datos.

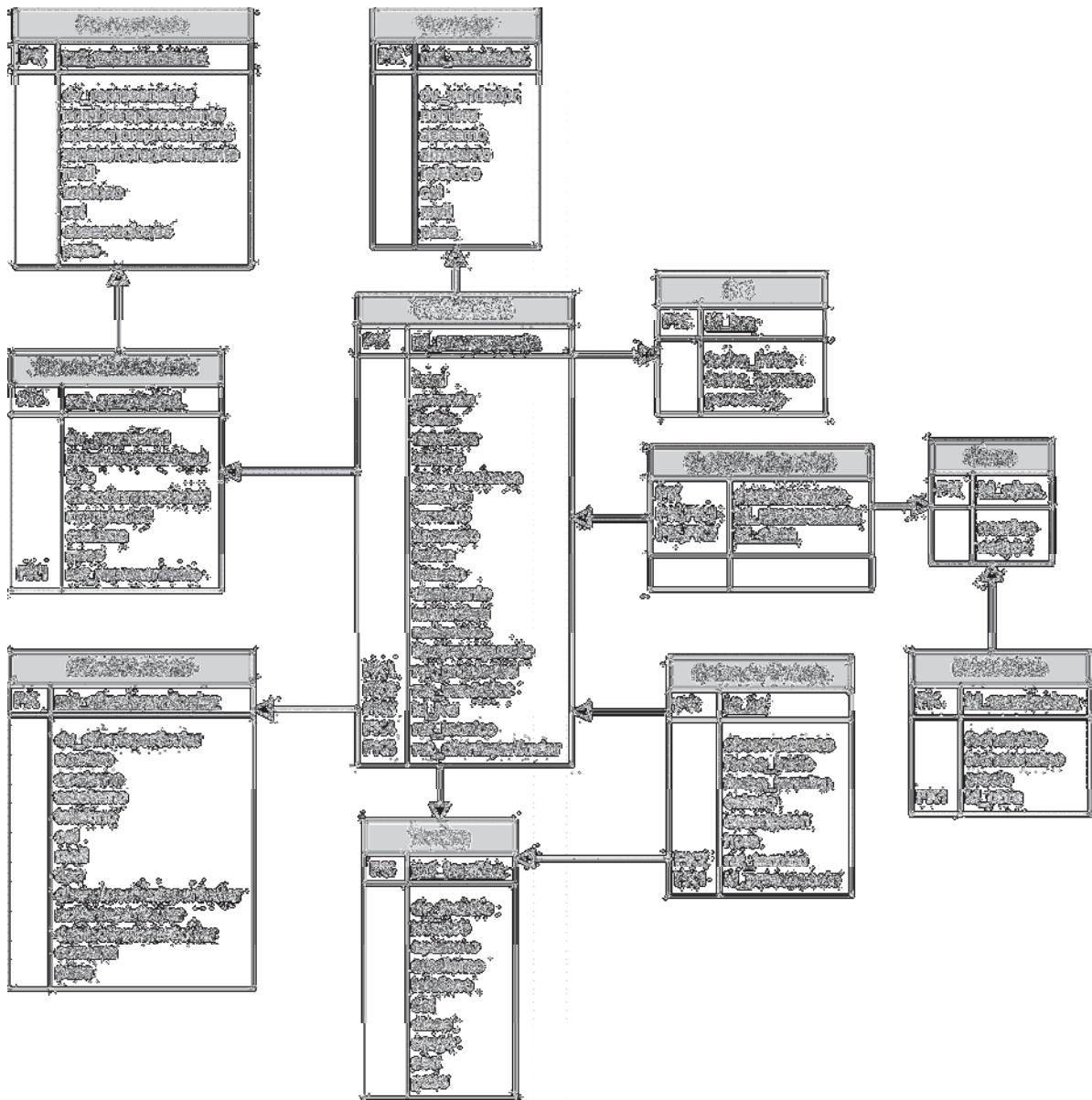


Figura 13.5. Modelo Relacional.

CAPÍTULO 14: DIAGRAMAS DE SECUENCIA

Un diagrama de secuencia muestra un conjunto de mensajes ordenados en una secuencia temporal. Cada rol se muestra como una línea de vida – es decir, una línea vertical que representa al a lo largo del tiempo a través de la interacción completa. Los mensajes se muestran flechas entre líneas de vida. Un diagrama de secuencia puede mostrar un escenario-una historia individual de una transacción. Las construcciones de control estructurado, como los bucles, las condiciones y las ejecuciones en paralelo, se muestran como rectángulos anidados con palabras clave y una o más regiones [8].

14.1 Diagramas de secuencia: Gestionar cliente administración

14.1.1 Diagrama de secuencia: Ingresar cliente administración

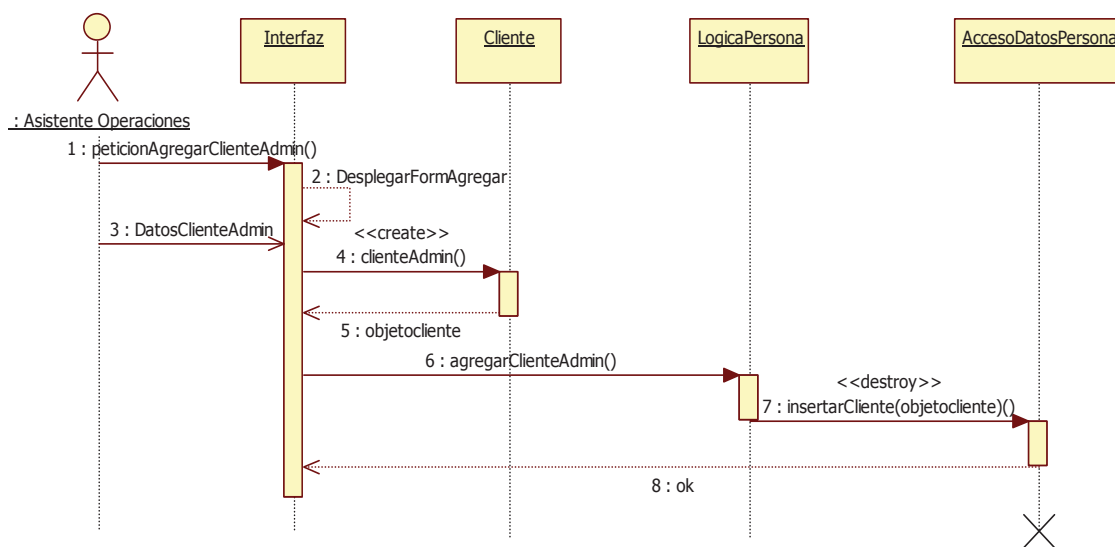


Figura 14.1. Diagrama de secuencia: ingresar cliente administración

14.1.2 Diagrama de secuencia: Buscar cliente administración por rut

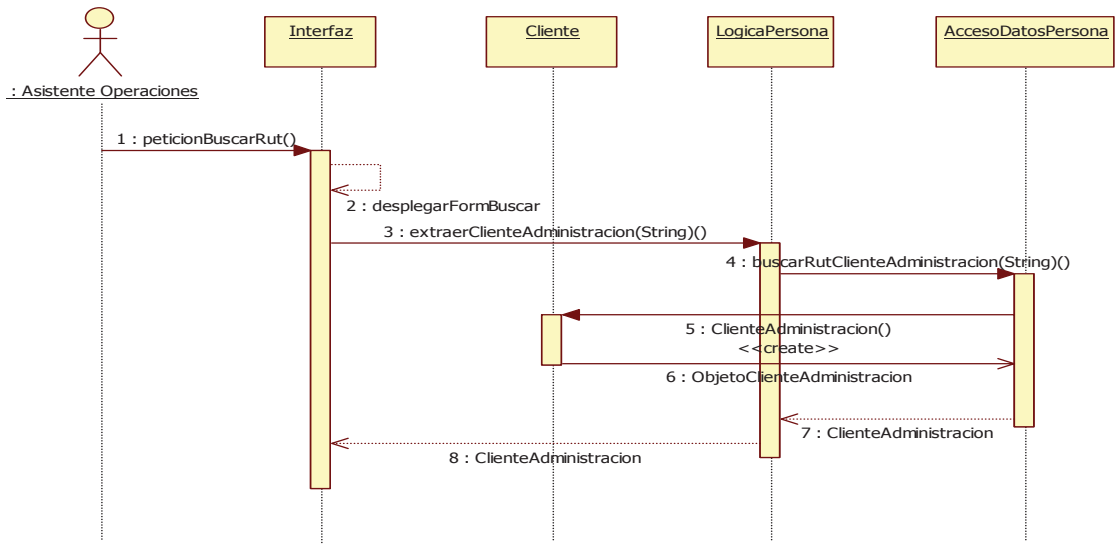


Figura 14.2. Diagrama de secuencia: buscar cliente administración por rut.

14.1.3 Diagrama de secuencia: Buscar cliente administración por nombre/apellido

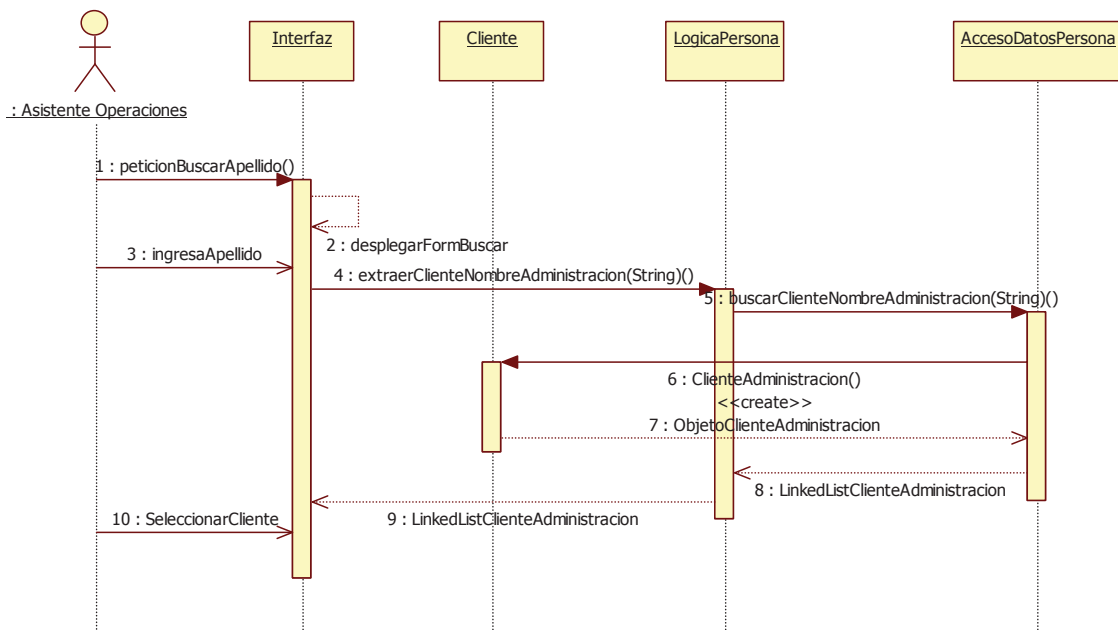


Figura 14.3. Diagrama de secuencia: buscar cliente administración por nombre/apellido

14.1.4 Diagrama de secuencia: Modificar cliente administración búsqueda por rut

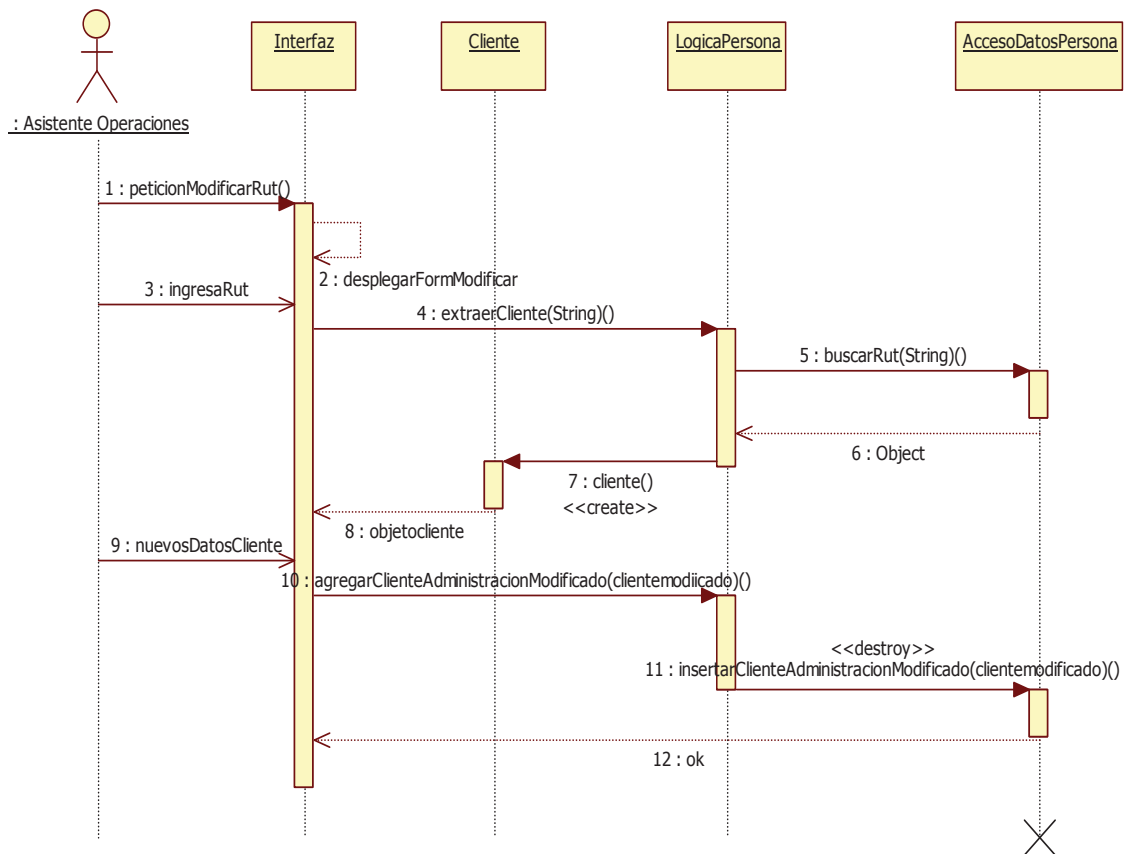


Figura 14.4. Diagrama de secuencia: modificar cliente administración, búsqueda por rut.

14.1.5 Diagrama de secuencia: Modificar cliente administración búsqueda por nombre/apellido

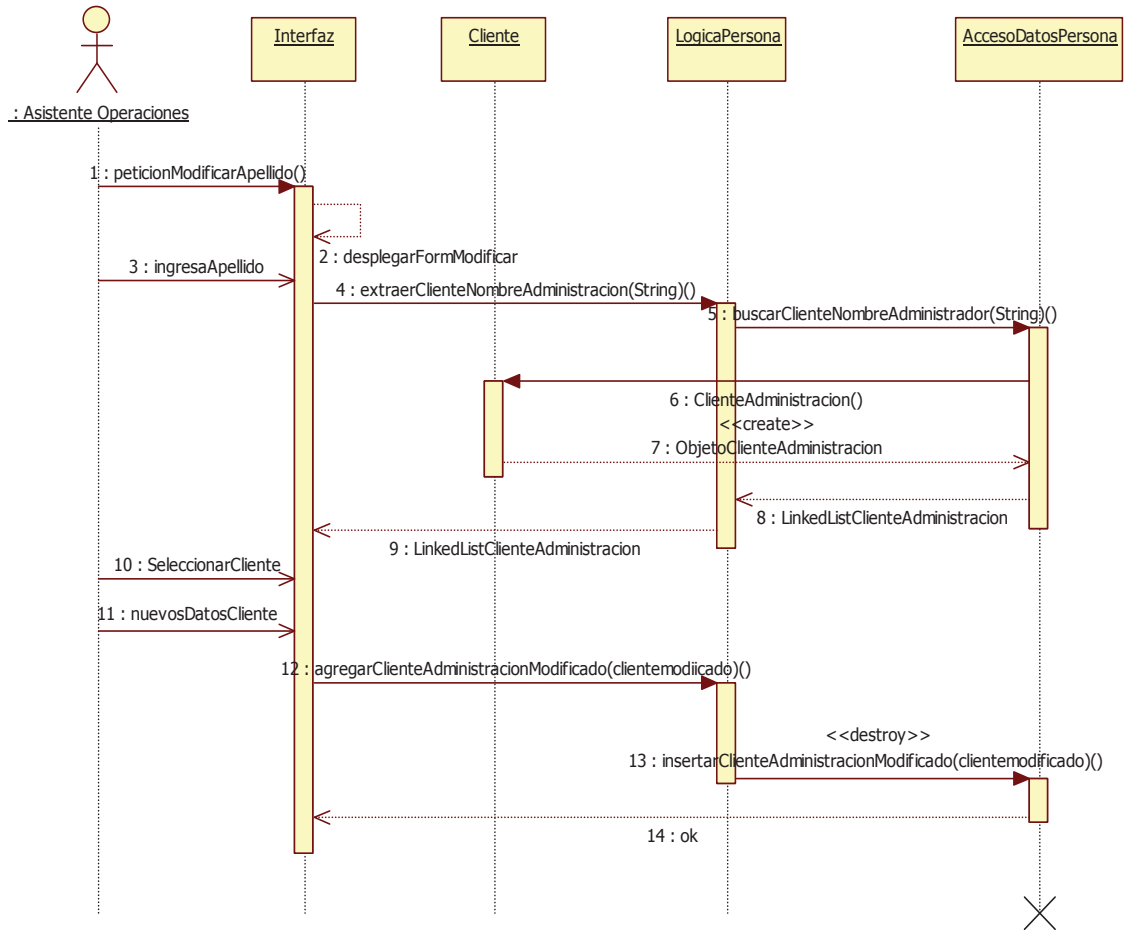


Figura 14.5. Diagrama de secuencia: modificar cliente administración por nombre/apellido.

14.2 Diagramas de secuencia: gestionar cliente particular

14.2.1 Diagrama de secuencia: Ingresar cliente particular

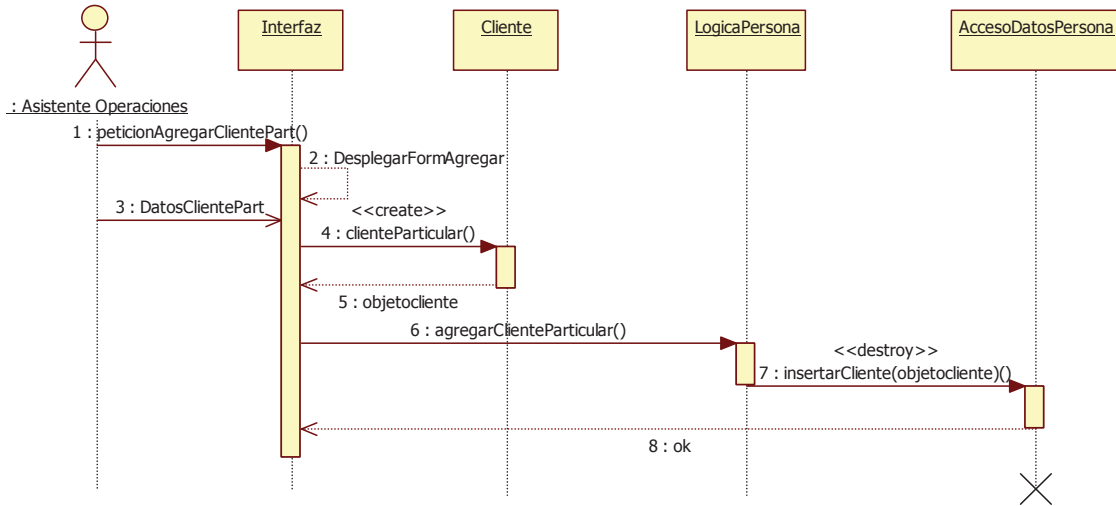


Figura 14.6. Diagrama de secuencia: Ingresar cliente particular.

14.2.2 Diagrama de secuencia: Buscar cliente particular por rut

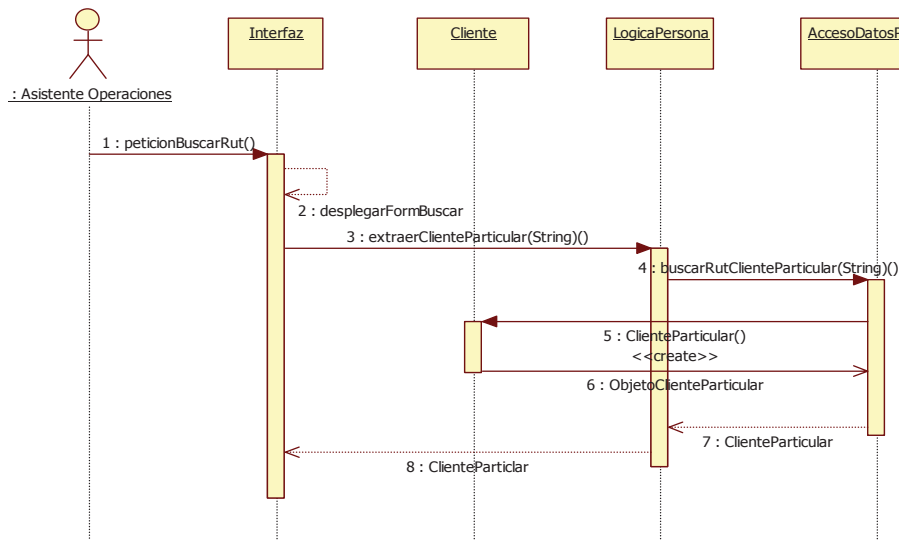


Figura 14.7. Diagrama de secuencia: buscar cliente particular por rut.

14.2.3 Diagrama de secuencia: Buscar cliente particular por nombre/apellido

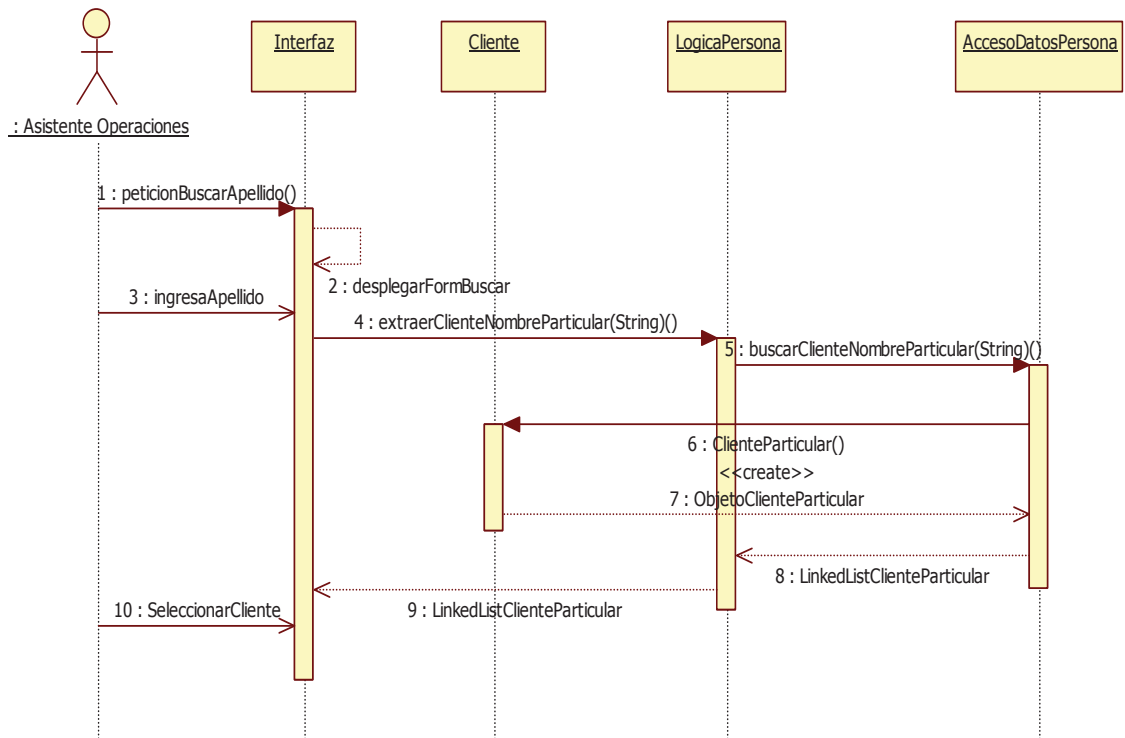


Figura 14.8. Diagrama de secuencia: buscar cliente particular por nombre/rut.

14.2.4 Diagrama de secuencia: Modificar cliente particular búsqueda por rut

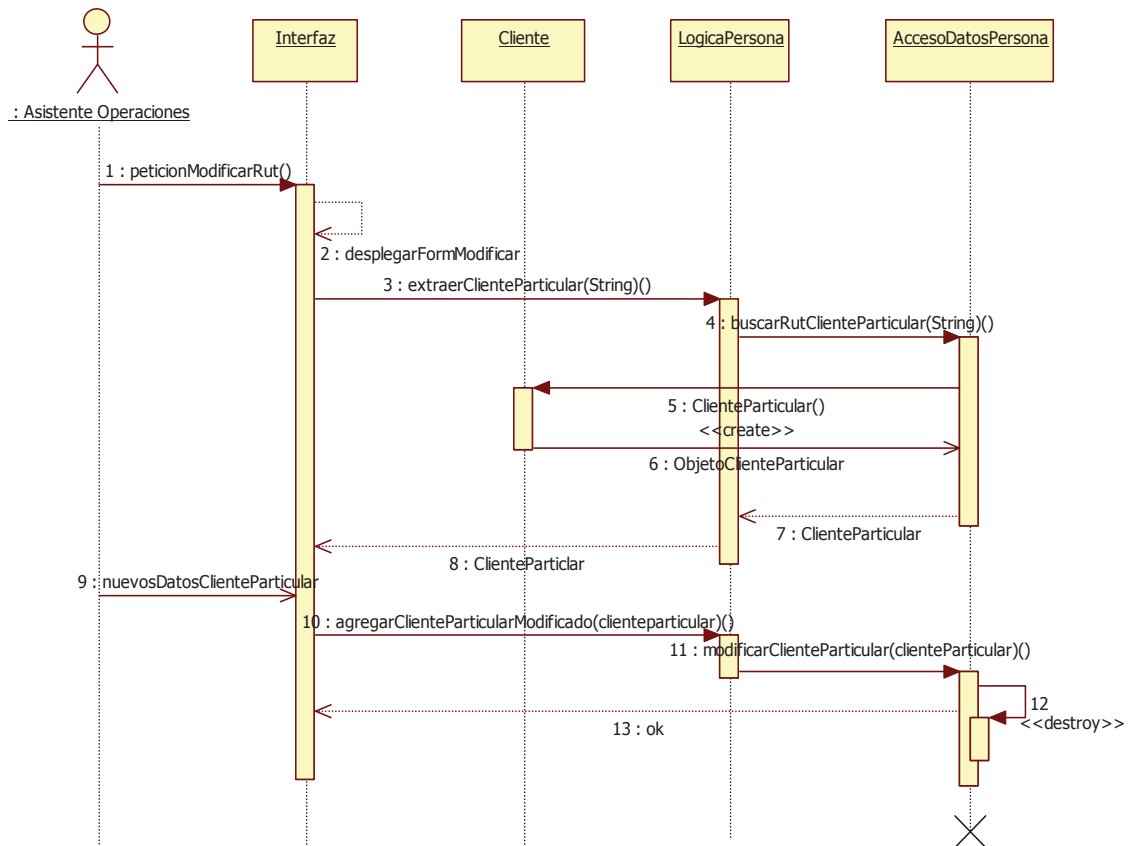


Figura 14.9. Diagrama de secuencia: modificar cliente particular por rut.

14.2.5 Diagrama de secuencia: Modificar cliente particular búsqueda por nombre/apellido

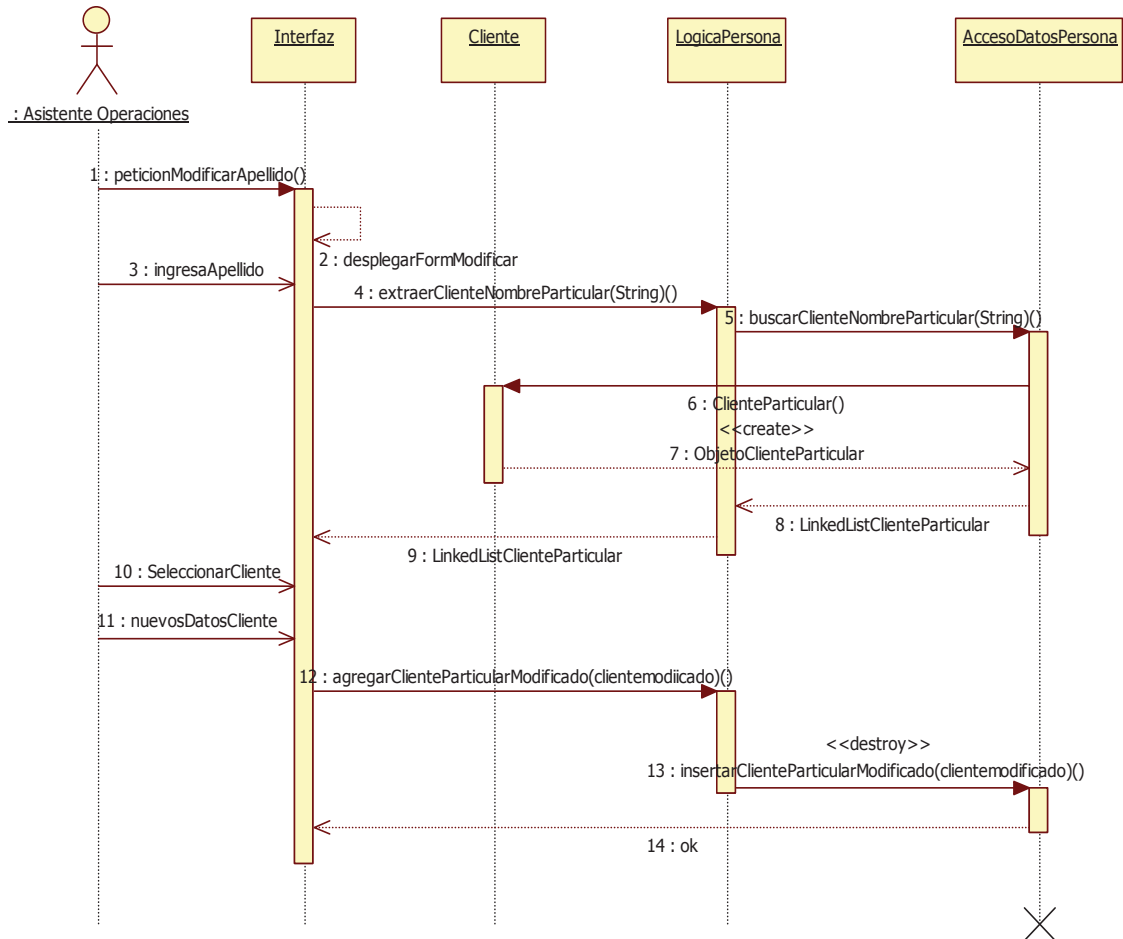


Figura 14.10. Diagrama de secuencia: Modificar cliente particular por nombre/apellido.

14.3 Diagramas de secuencia: Gestionar representante

14.3.1 Diagrama de secuencia: Ingresar representante

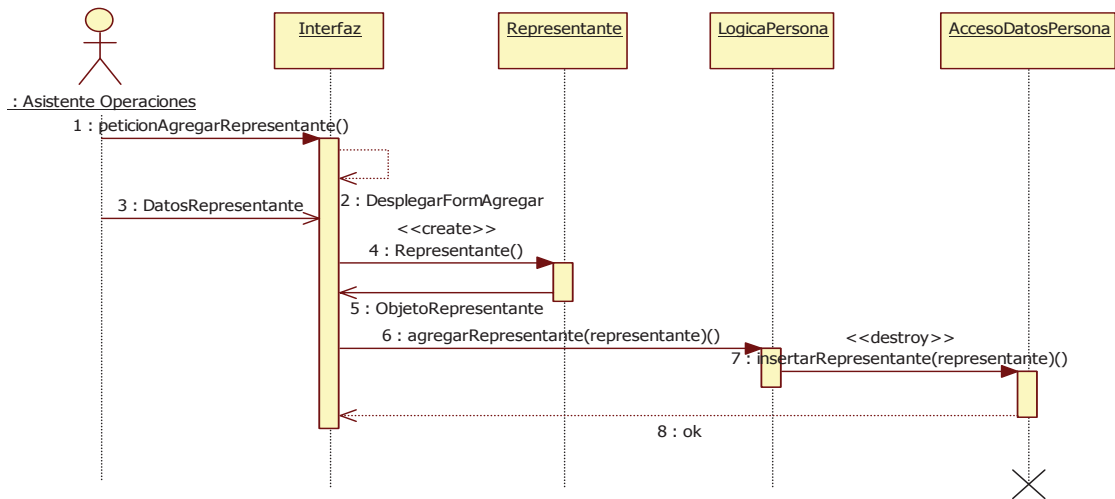


Figura 14.11. Diagrama de secuencia: Ingresar representante.

14.3.2 Diagrama de secuencia: Buscar representante por rut

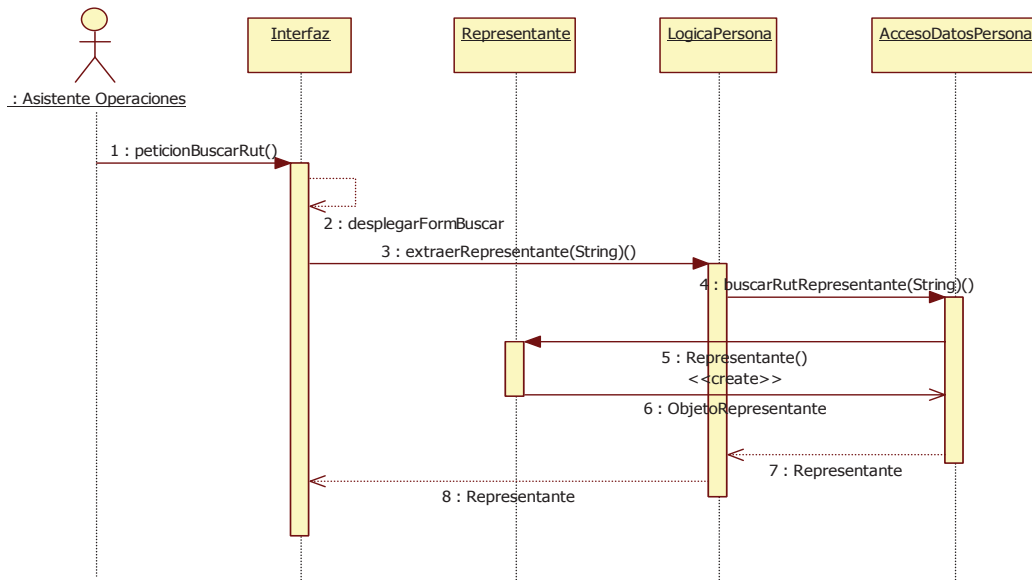


Figura 14.12. Diagrama de secuencia: buscar representante por rut.

14.3.3 Diagrama de secuencia: Buscar representante por nombre/apellido

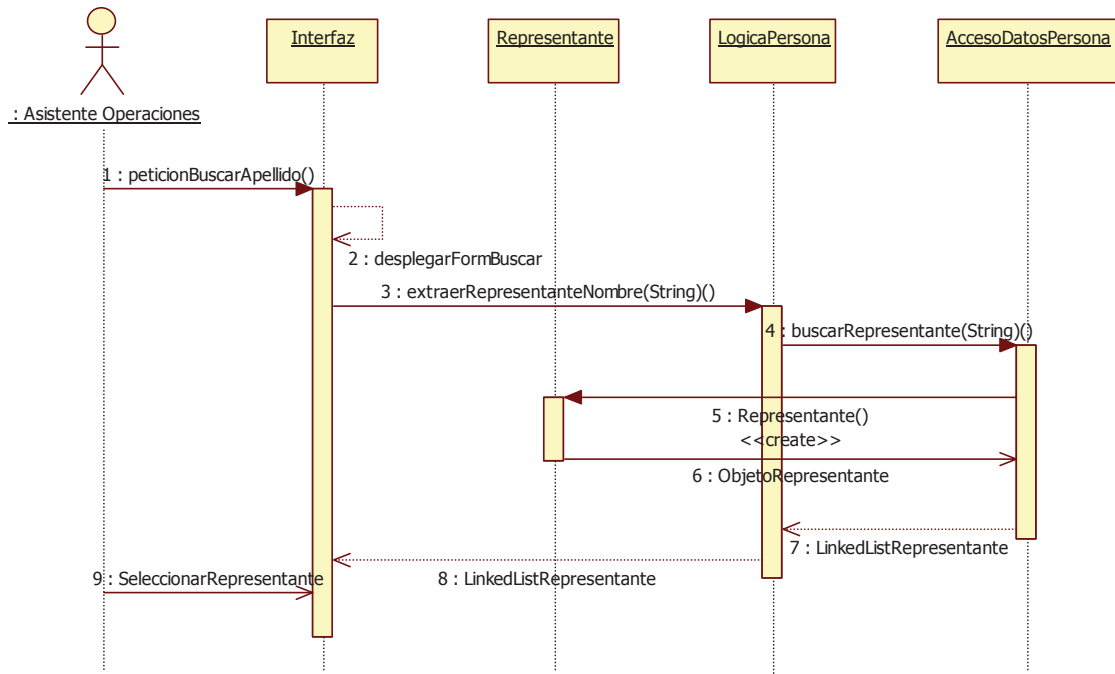


Figura 14.13. Diagrama de secuencia: buscar representante por nombre/apellido.

14.3.4 Diagrama de secuencia: Modificar representante búsqueda por rut

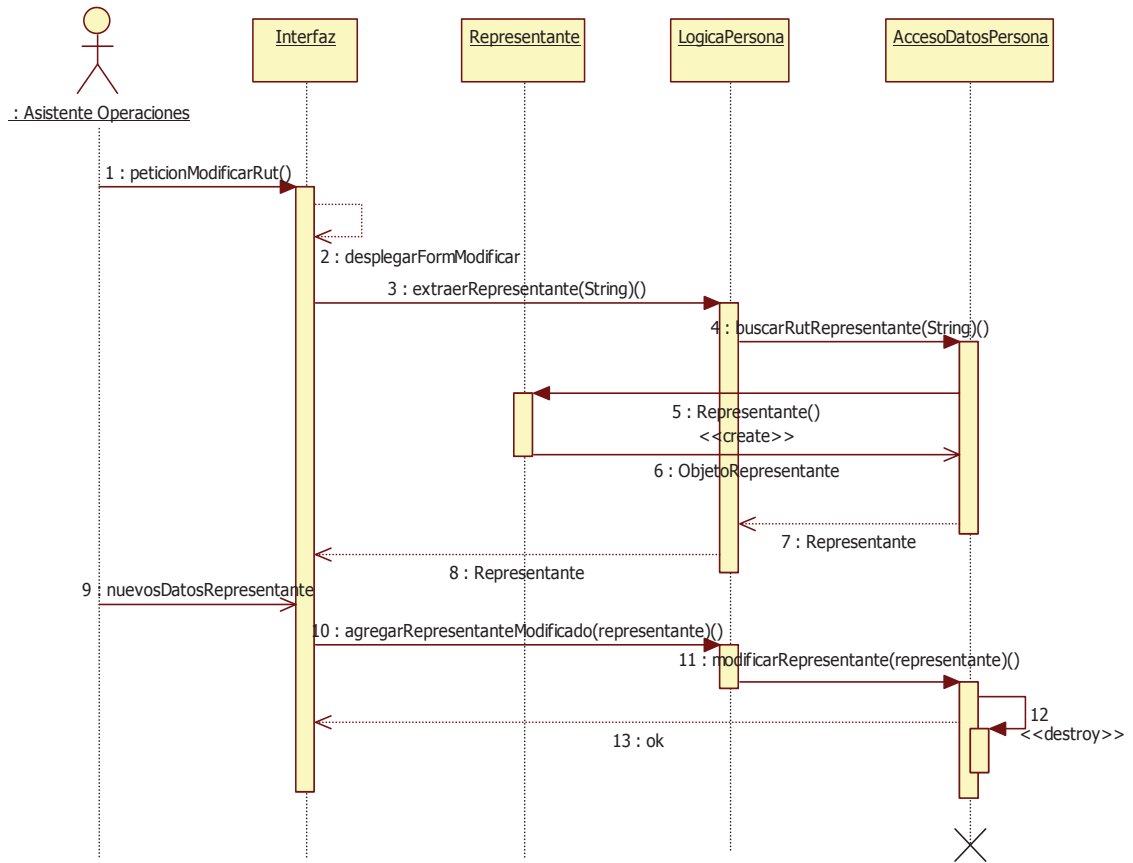


Figura 14.14. Diagrama de secuencia: Modificar representante por rut.

14.3.5 Diagrama de secuencia: Modificar representante búsqueda por nombre/apellido

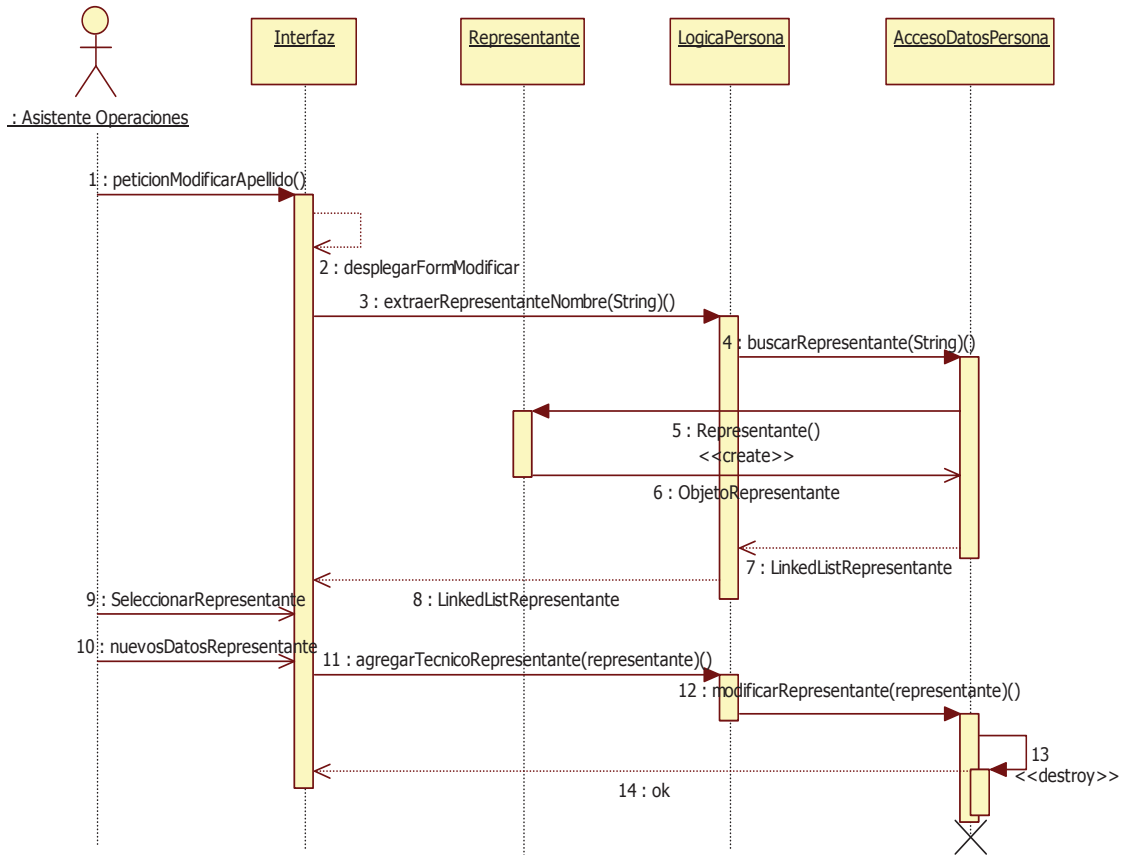


Figura 14.15. Diagrama de secuencia: Modificar Representante por nombre/apellido.

14.4 Diagramas de secuencia: Gestionar vendedor

14.4.1 Diagrama de secuencia: Ingresar vendedor

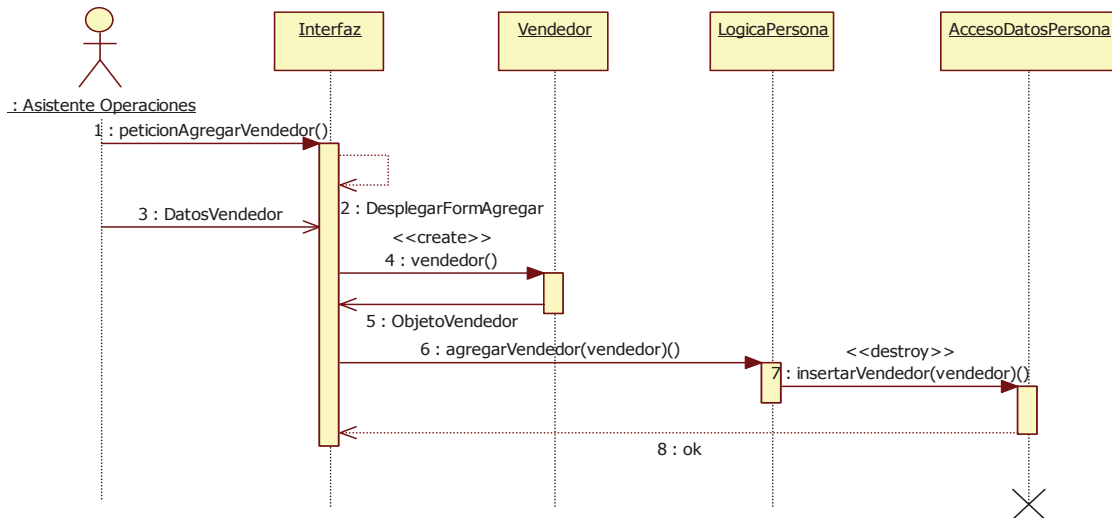


Figura 14.16. Diagrama de secuencia: Ingresar vendedor.

14.4.2 Diagrama de secuencia: Buscar vendedor por rut

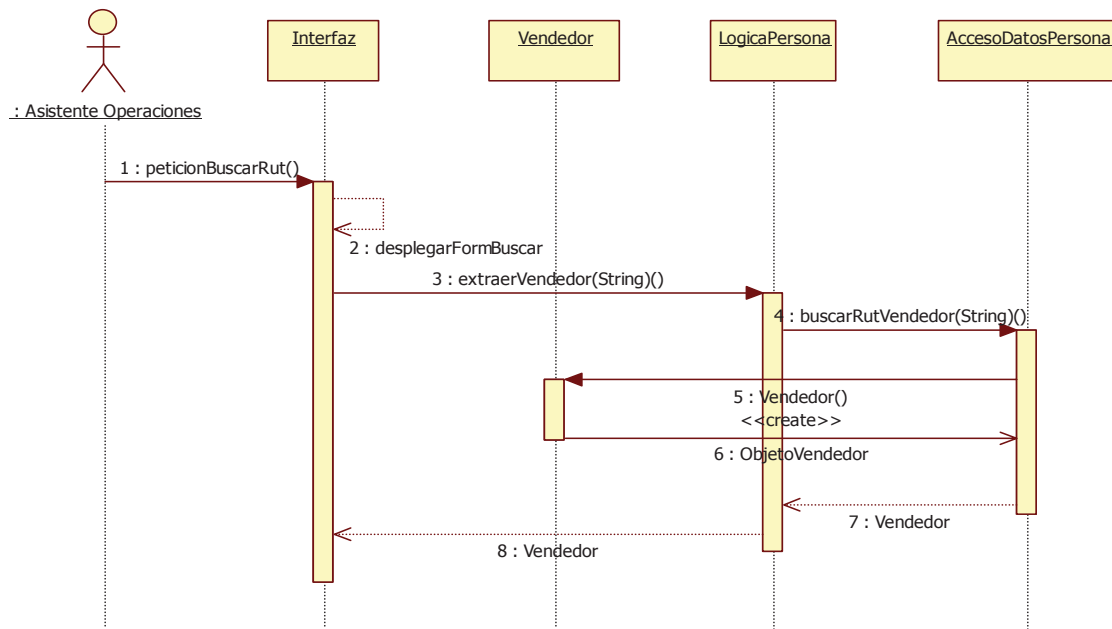


Figura 14.17. Diagrama de secuencia: buscar vendedor por rut.

14.4.3 Diagrama de secuencia: Buscar vendedor por nombre/apellido

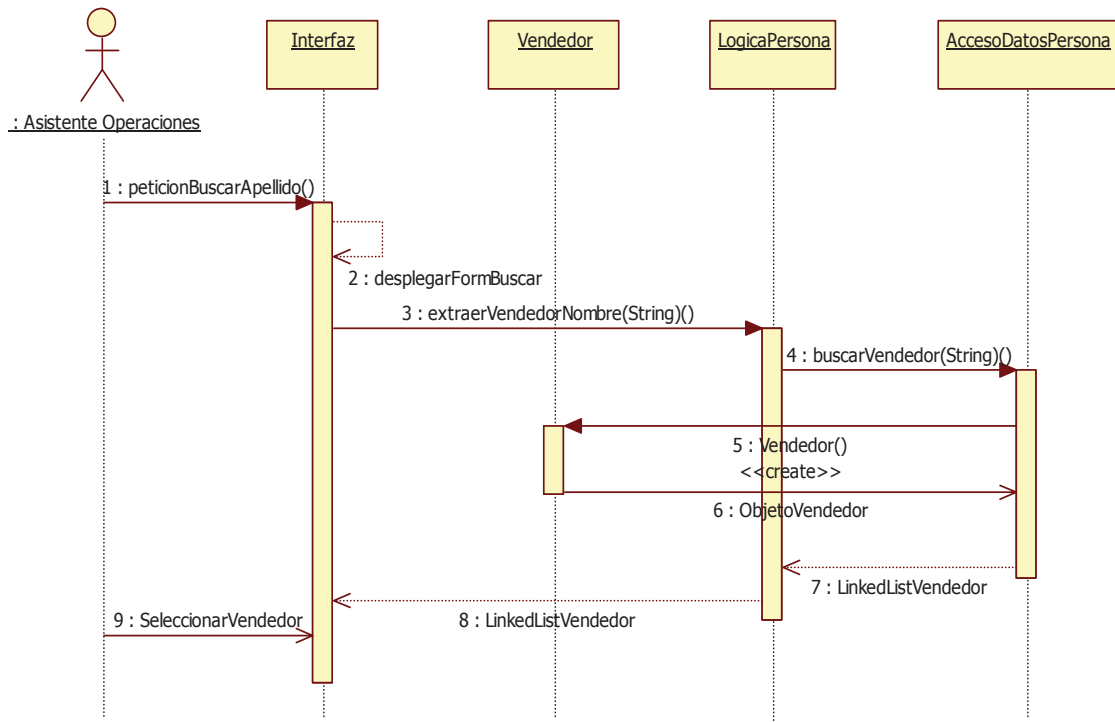


Figura 14.18. Diagrama de secuencia: buscar vendedor por nombre/apellido.

14.4.4 Diagrama de secuencia: Modificar vendedor búsqueda por rut

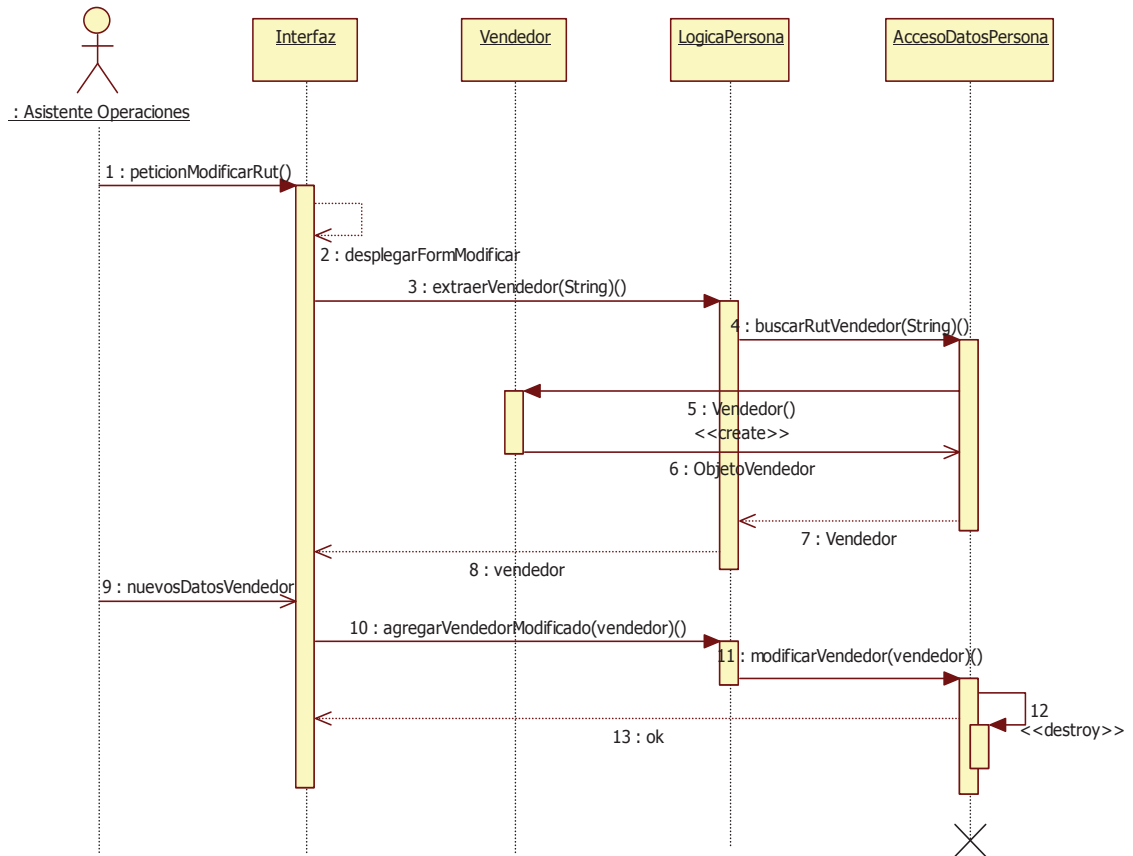


Figura 14.19. Diagrama de secuencia: Modificar vendedor por rut.

14.4.5 Diagrama de secuencia: Modificar vendedor búsqueda por nombre/apellido

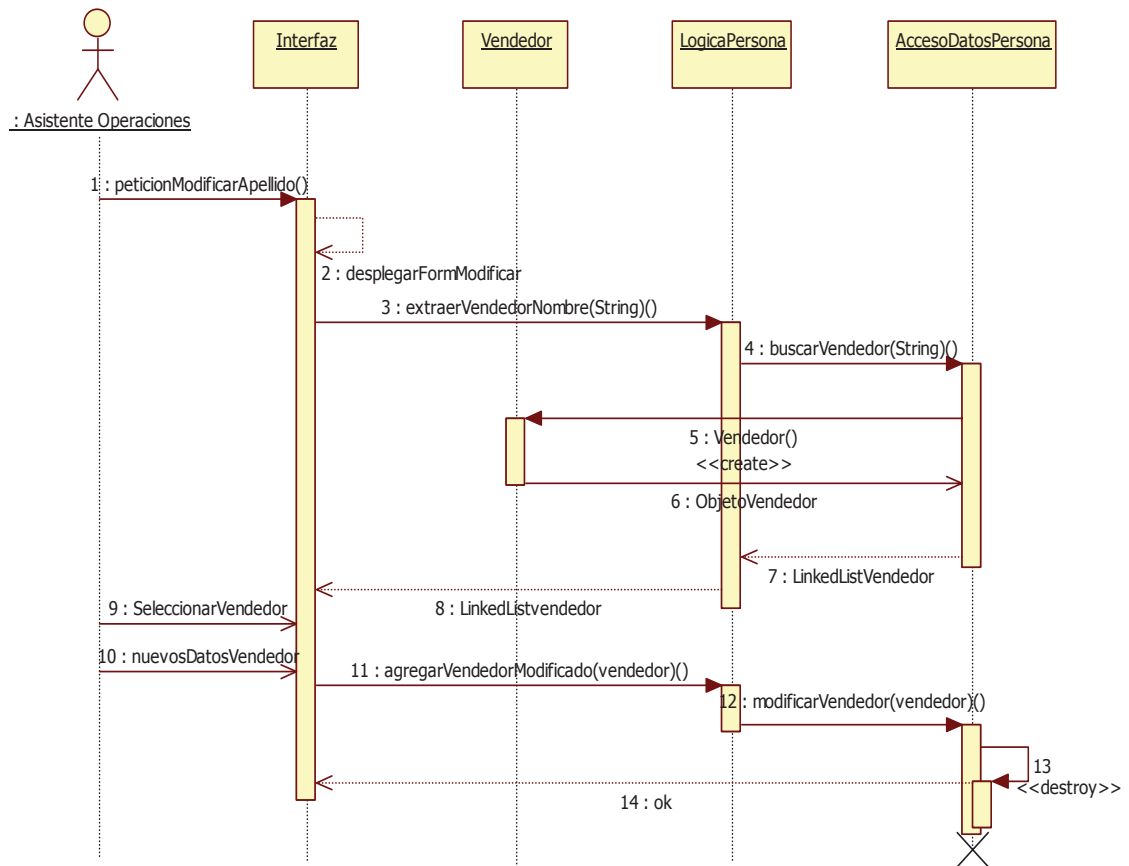


Figura 14.20. Diagrama de secuencia: Modificar vendedor por nombre/apellido.

14.5 Diagramas de secuencia: Gestionar técnico

14.5.1 Diagrama de secuencia: Ingresar técnico

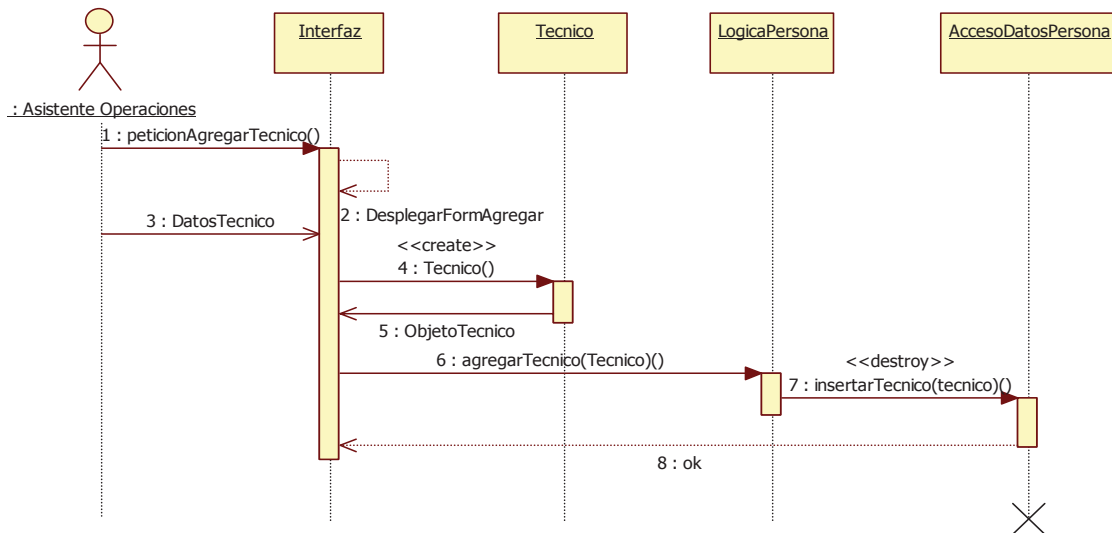


Figura 14.21. Diagrama de secuencia: Ingresar técnico.

14.5.2 Diagrama de secuencia: Buscar técnico por rut

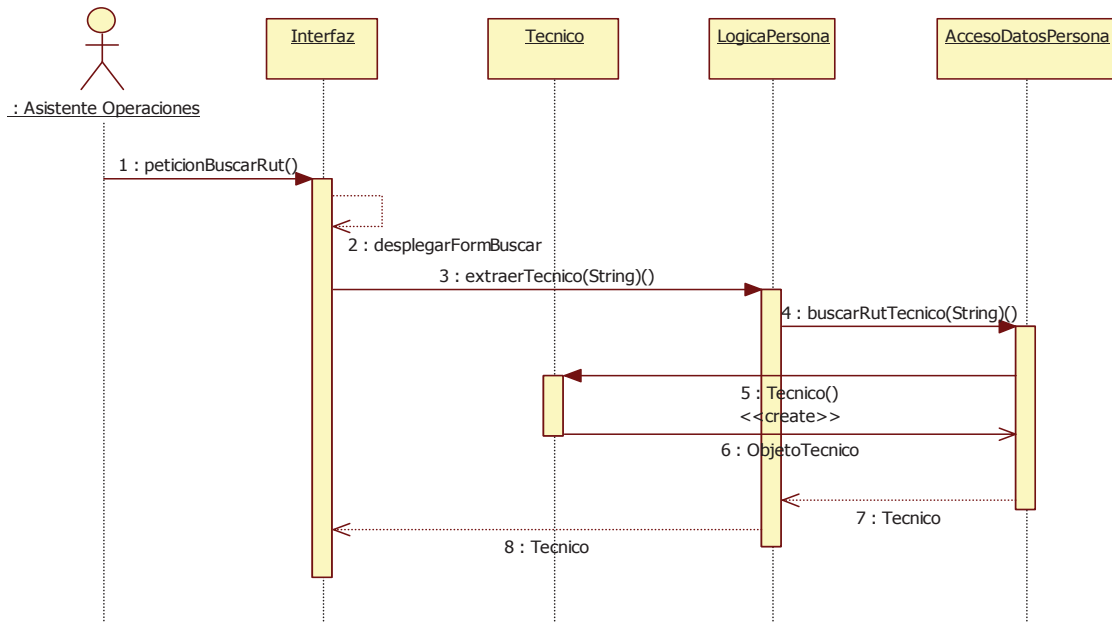


Figura 14.22. Diagrama de secuencia: buscar técnico por rut.

14.5.3 Diagrama de secuencia: Buscar técnico por nombre/apellido

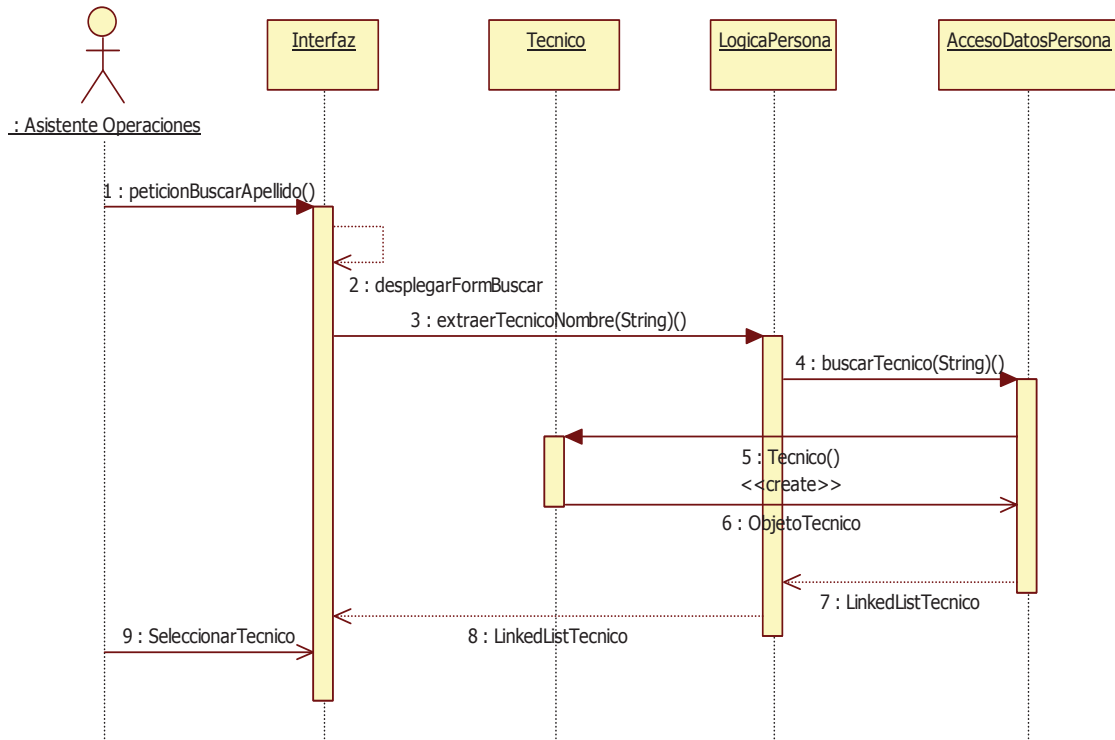


Figura 14.23. Diagrama de secuencia: buscar técnico por nombre/apellido.

14.5.4 Diagrama de secuencia: Modificar técnico búsqueda por rut

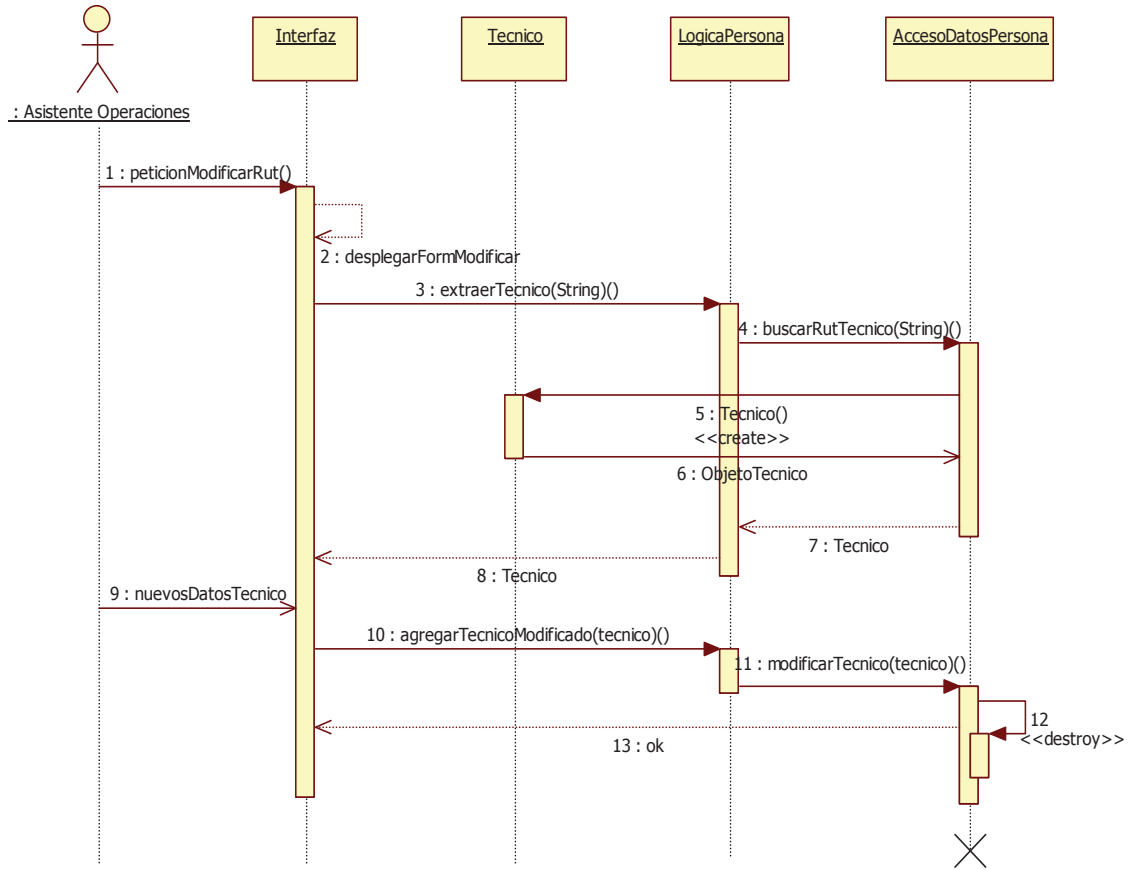


Figura 14.24. Diagrama de secuencia: Modificar técnico por rut.

14.5.5 Diagrama de secuencia: Modificar técnico búsqueda por nombre/apellido

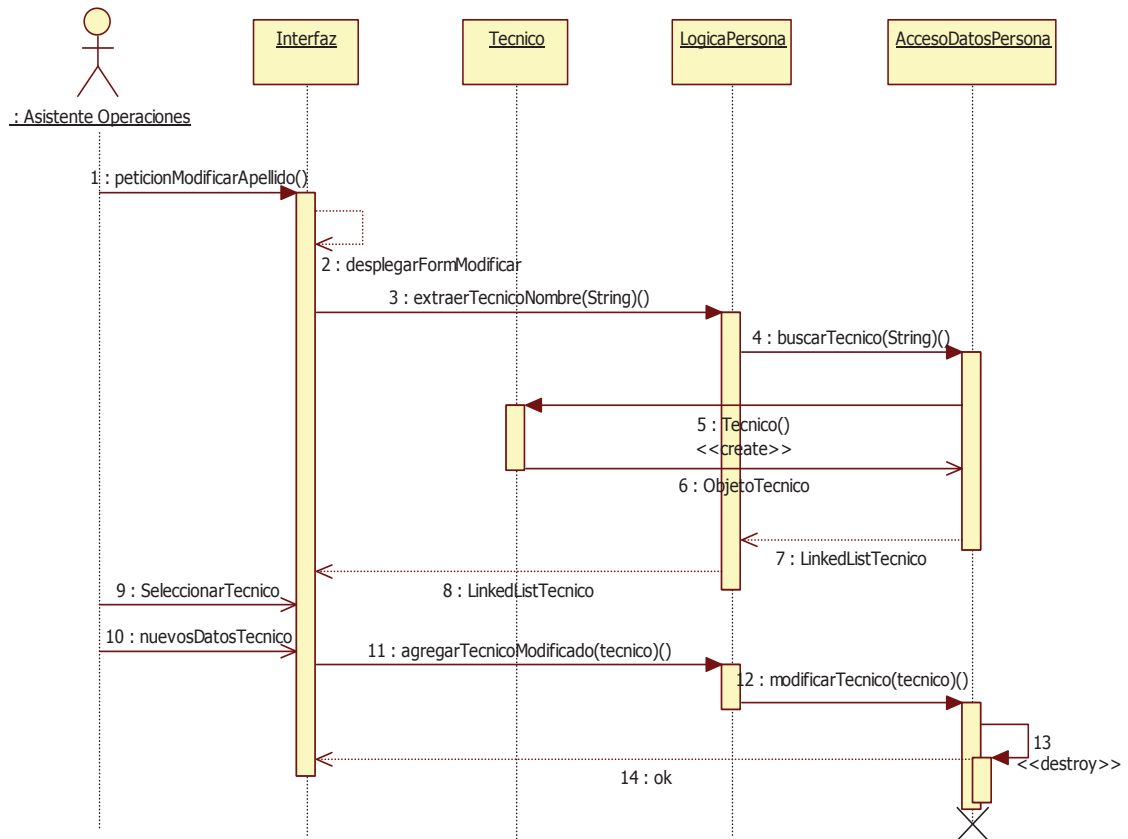


Figura 14.25. Diagrama de secuencia: Modificar técnico por nombre/apellido.

14.6 Diagramas de secuencia: Gestionar obras

14.6.1 Diagrama de secuencia: Ingresar obra

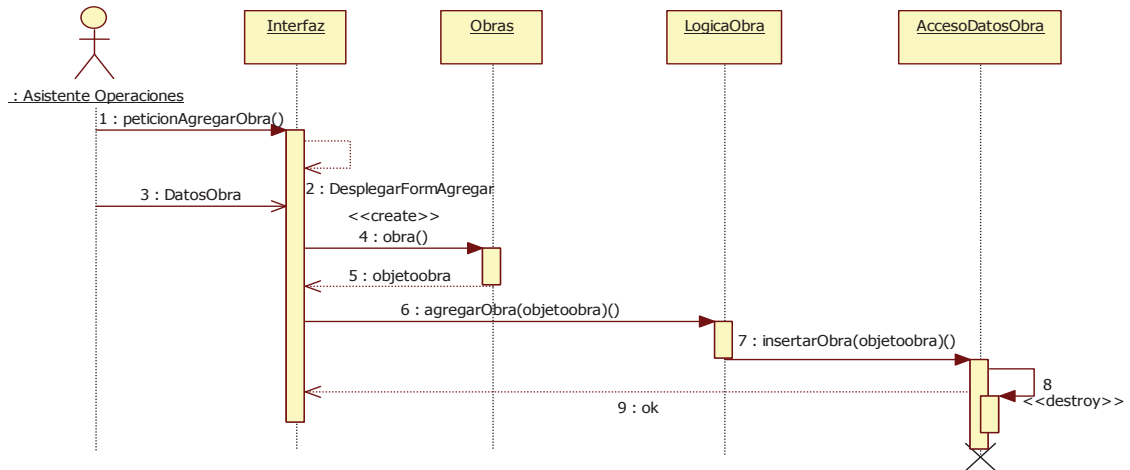


Figura 14.26. Diagrama de secuencia: ingresar obra

14.6.2 Diagrama de secuencia: Buscar obra por su nombre

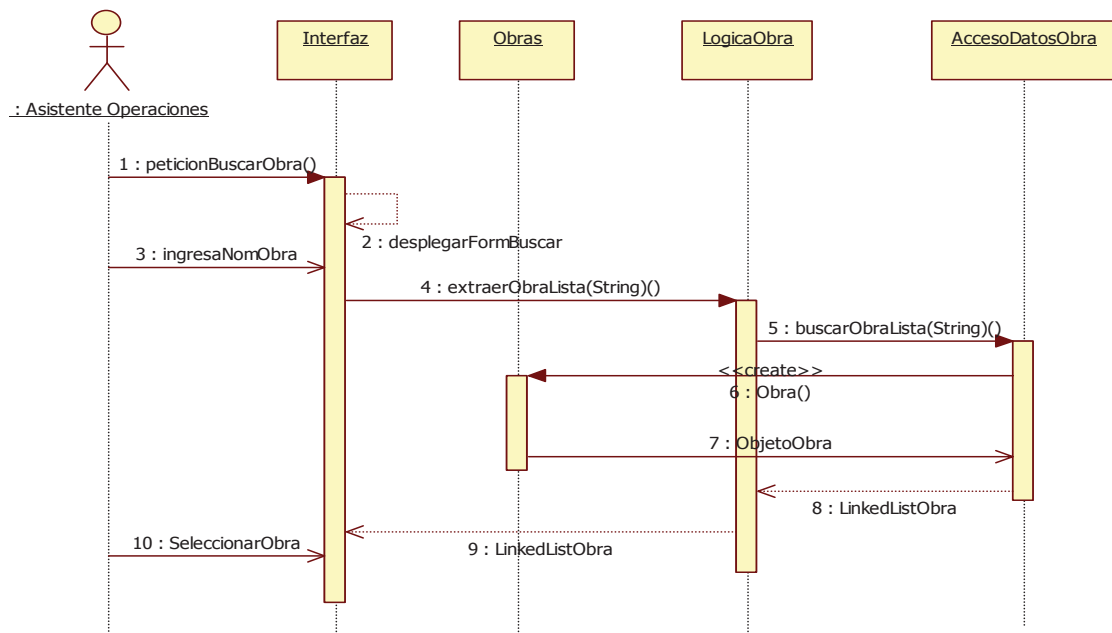


Figura 14.27. Diagrama de secuencia: buscar obra por nombre.

14.6.3 Diagrama de secuencia: Modificar obra búsqueda por su nombre

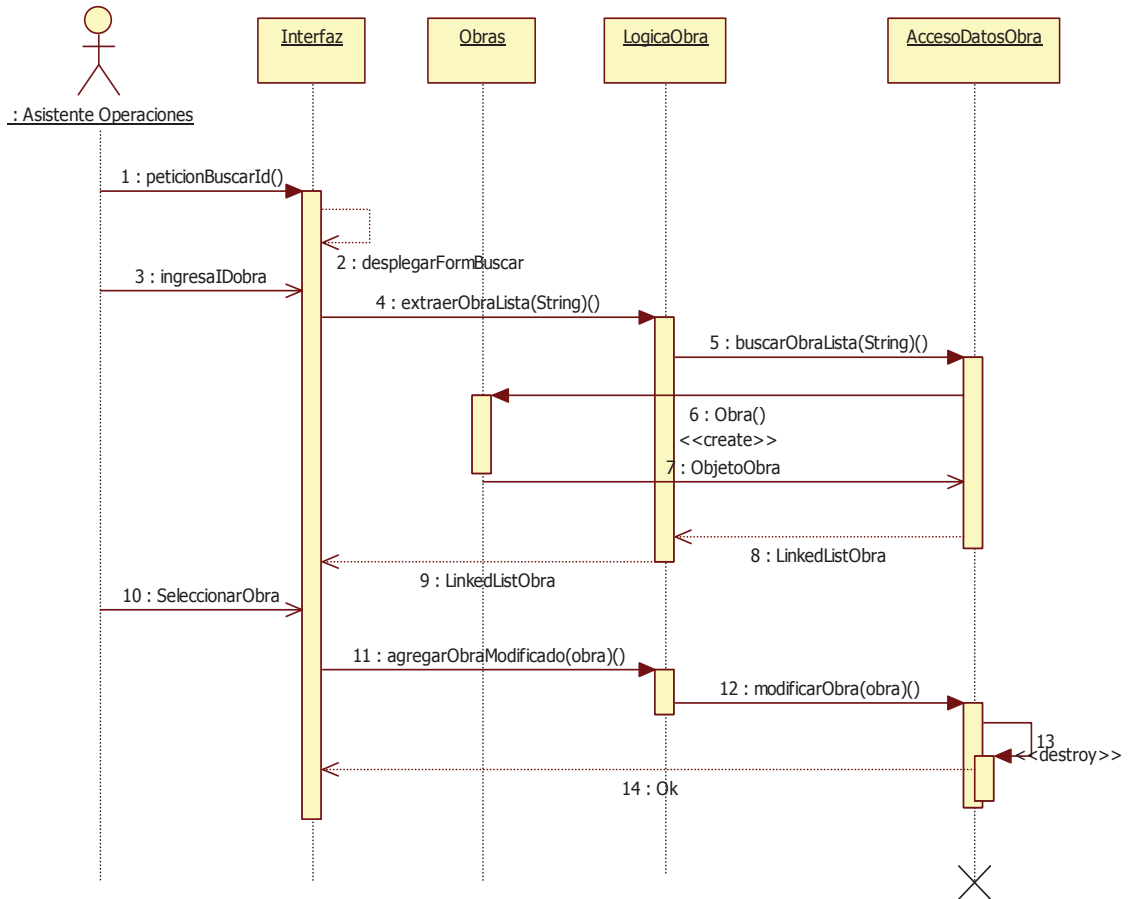


Figura 14.28. Diagrama de secuencia: modificar obra, búsqueda por nombre.

14.6.4 Diagrama de secuencia: Dar de baja obra búsqueda por su nombre

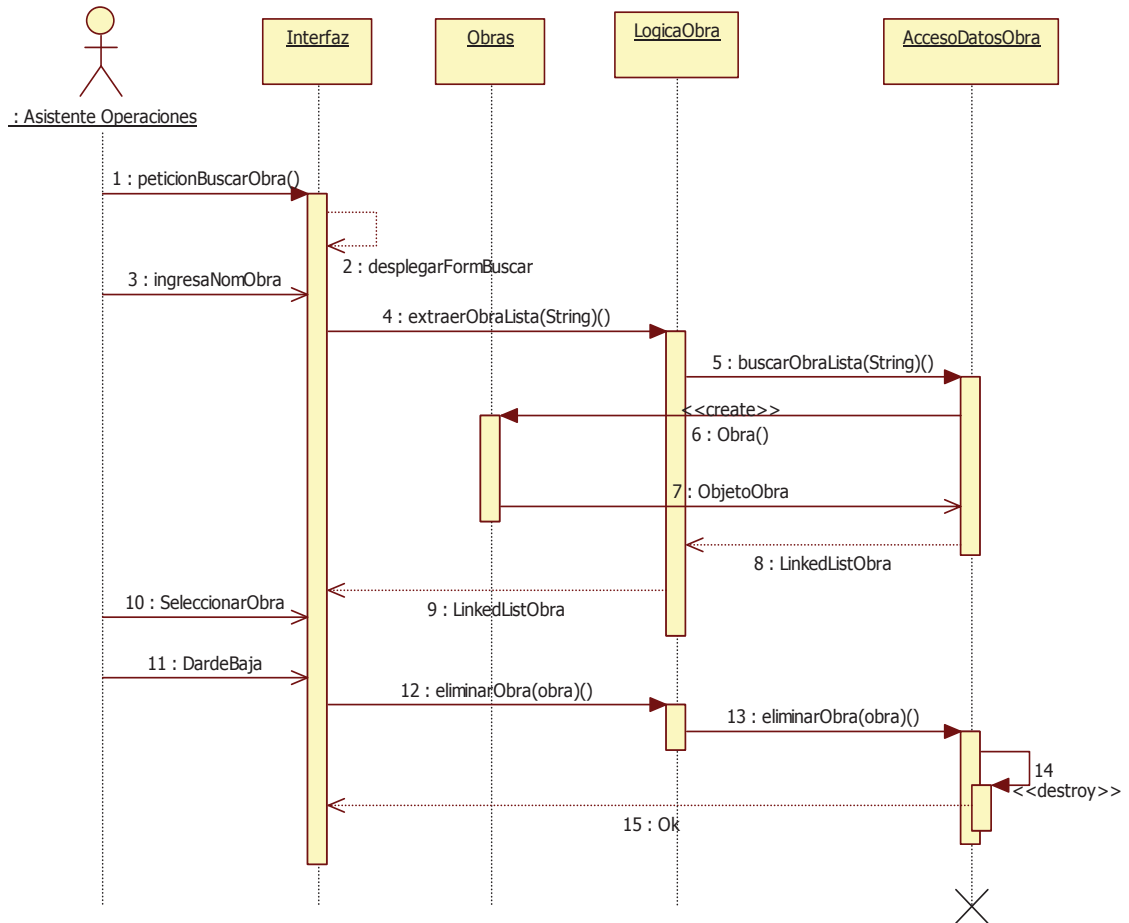


Figura 14.29. Diagrama de secuencia: dar de baja una obra, búsqueda por nombre.

14.7 Diagramas de secuencia: OT móvil

14.7.1 Diagrama de secuencia: Descargar orden de trabajo

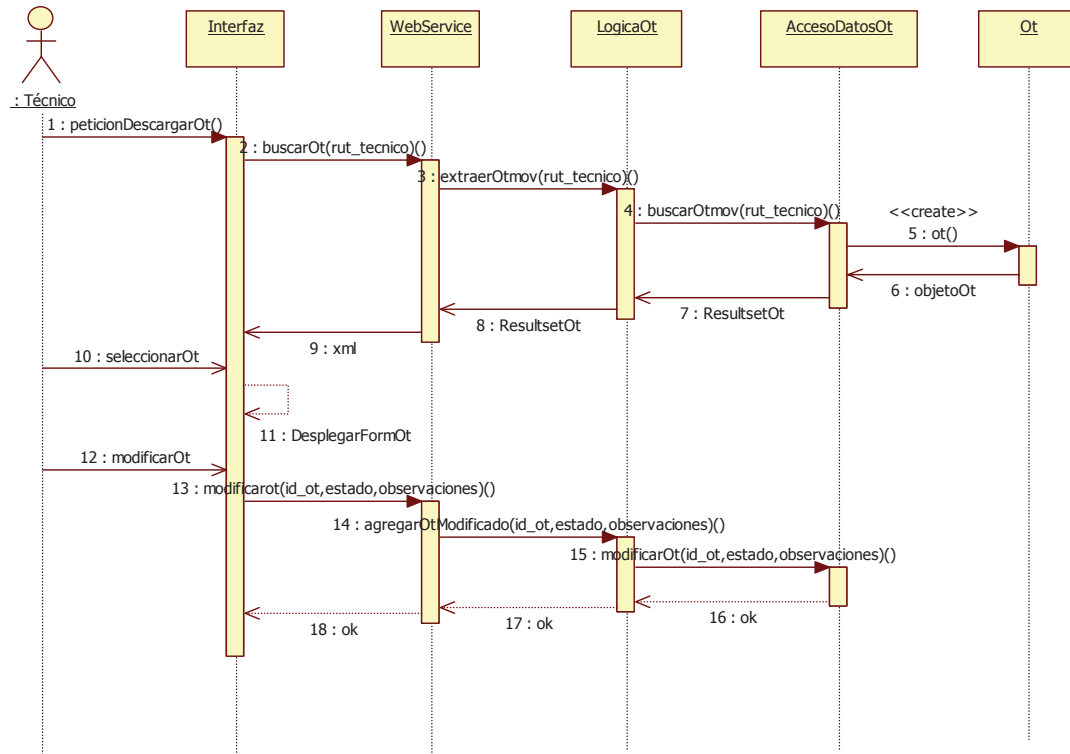


Figura 14.30. Diagrama de secuencia: Descargar orden de trabajo dispositivo móvil

14.7.2 Diagrama de secuencia: Descargar orden de trabajo pendiente

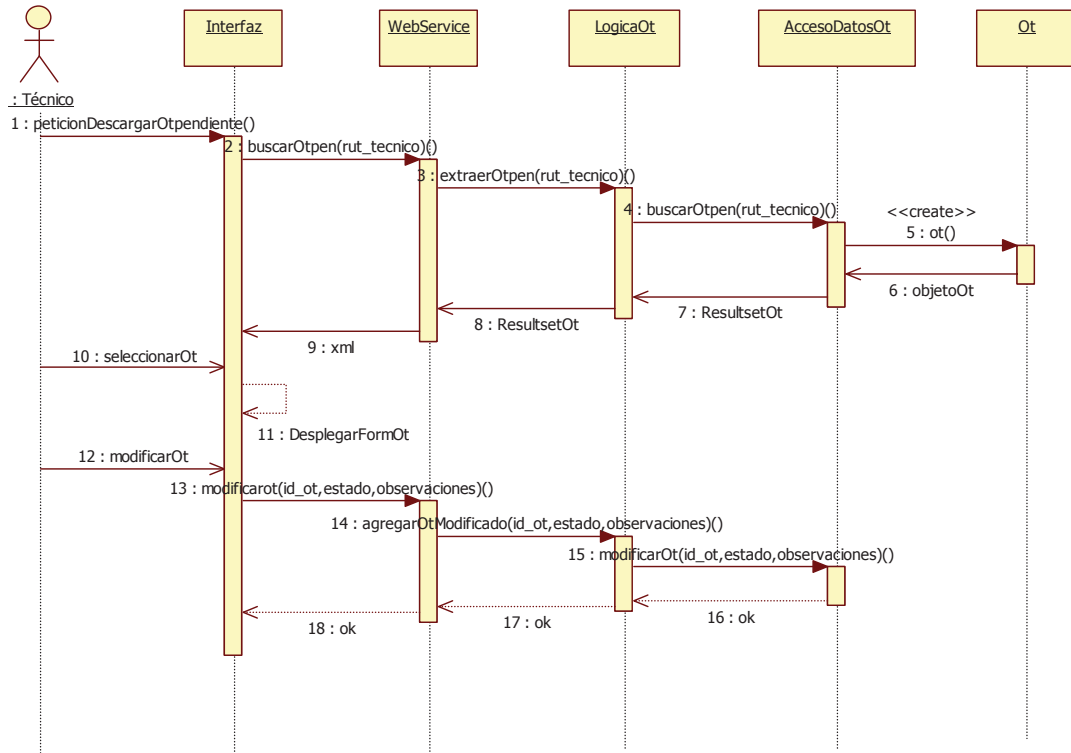


Figura 14.31. Diagrama de secuencia: Descargar orden de trabajo pendiente dispositivo móvil

14.8 Diagramas de secuencias: Gestionar Presupuesto

14.8.1 Diagrama de secuencia: Crear Presupuesto

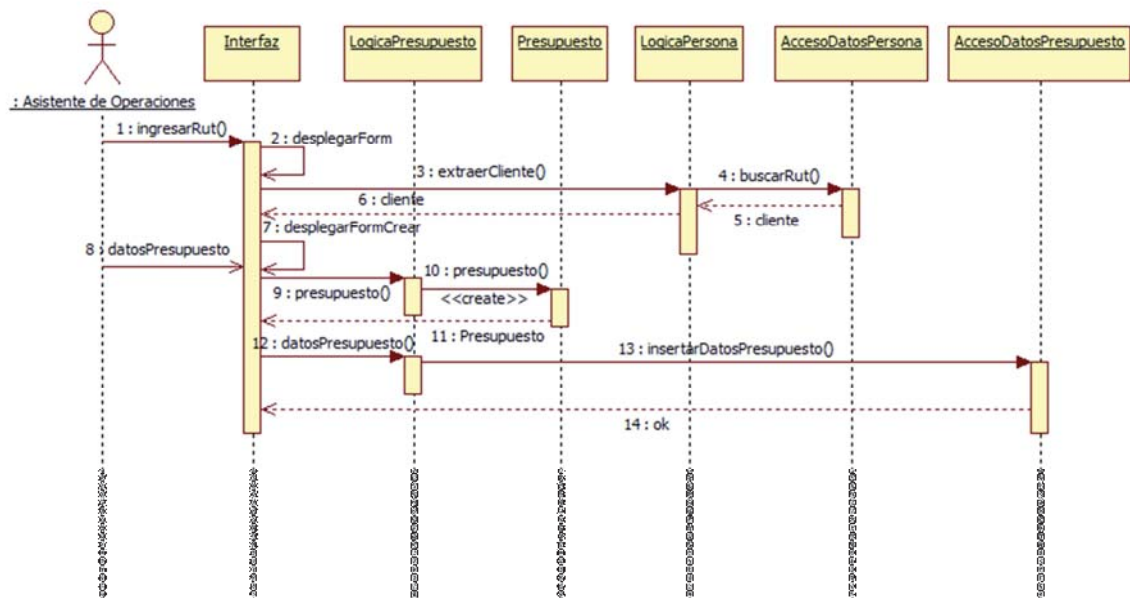


Figura 14.32. Diagrama de secuencia: Crear Presupuesto

CAPITULO 15: DIAGRAMA DE COMPONENTES Y DESPLIEGUE

15.1 Diagrama de componentes

Un diagrama de componentes muestra los componentes de un sistema-es decir, las unidades software con las que se construye la aplicación- así como las dependencias entre componentes, de forma que se pueda valorar el impacto de un cambio propuesto[8]. La figura 15.1 muestra el diagrama de componentes para el sistema. Las líneas discontinuas indican dependencia entre cada paquete.

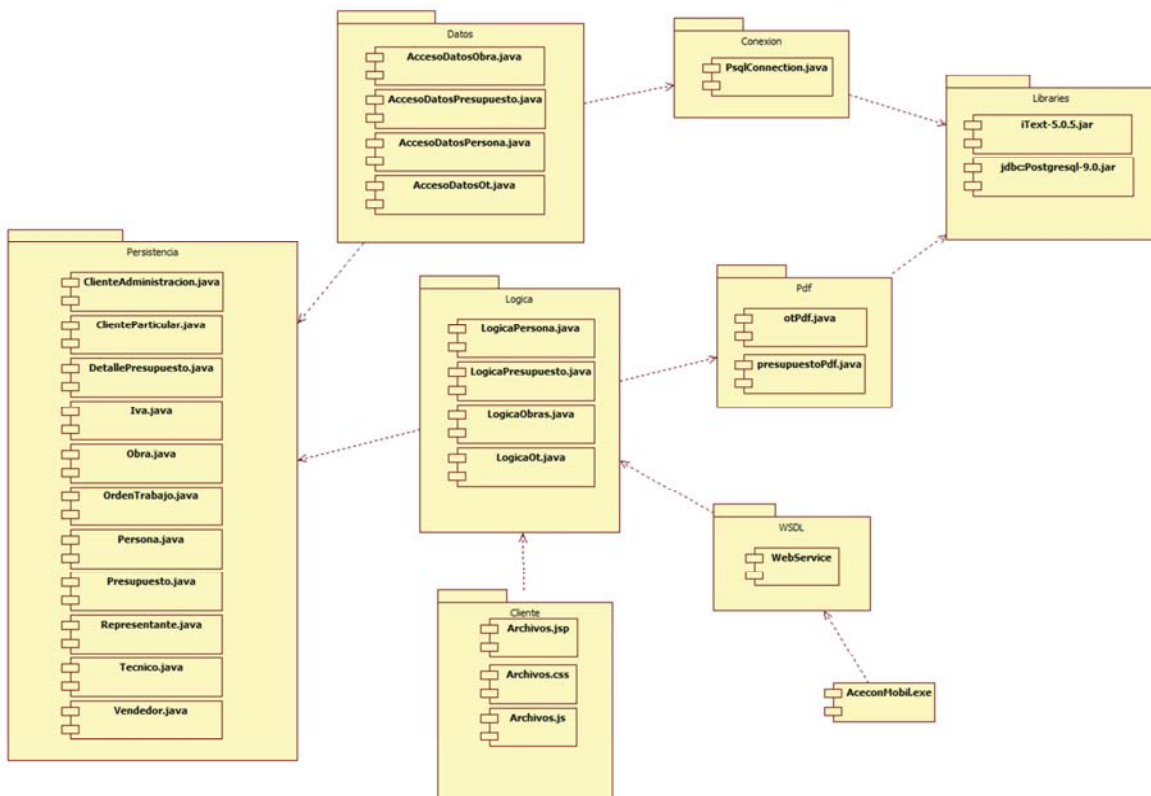


Figura 15.1. Diagrama de componentes del sistema.

15.2 Diagrama de despliegue

Un diagrama de despliegue representa el despliegue de artefactos de tiempo de ejecución sobre nodos. Un artefacto es una unidad de implementación física, como un archivo. Un nodo es un recurso de tiempo de ejecución, como una computadora, un dispositivo o la memoria. Un artefacto puede ser una manifestación (implementación) de uno o más componentes. Esta vista permite valorar las consecuencias de la distribución y de la asignación de recursos [8]. La figura 15.2 muestra un diagrama de despliegue del sistema. Este muestra los tipos de nodos que hay en el sistema y los artefactos que contiene.

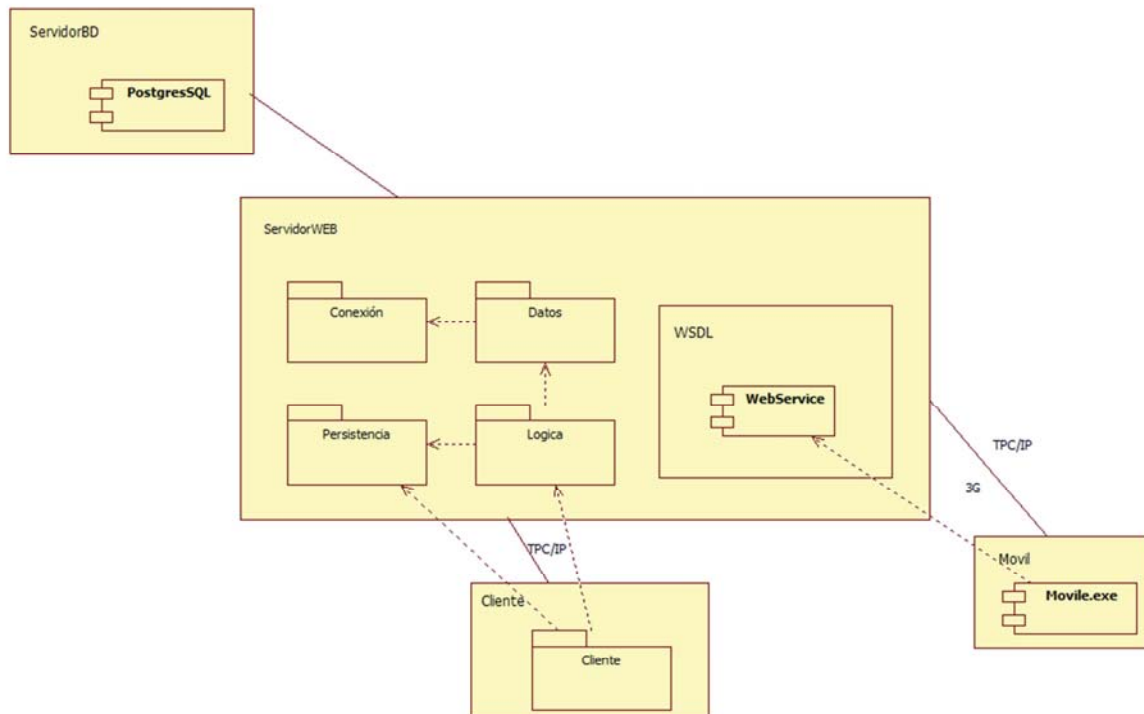


Figura 15.2. Diagrama de despliegue del sistema.

CAPÍTULO 16: DISEÑO DE INTERFACES

16.1 Elementos de interfaz

A continuación se detallan los distintos elementos presentes en cada una de las interfaces de usuario del sistema.

En la Figura N podemos apreciar la estructura aplicada en el diseño de las interfaces para el sitio Acecon Web, esta estructura se ha desarrollado pensando en las necesidades del usuario.

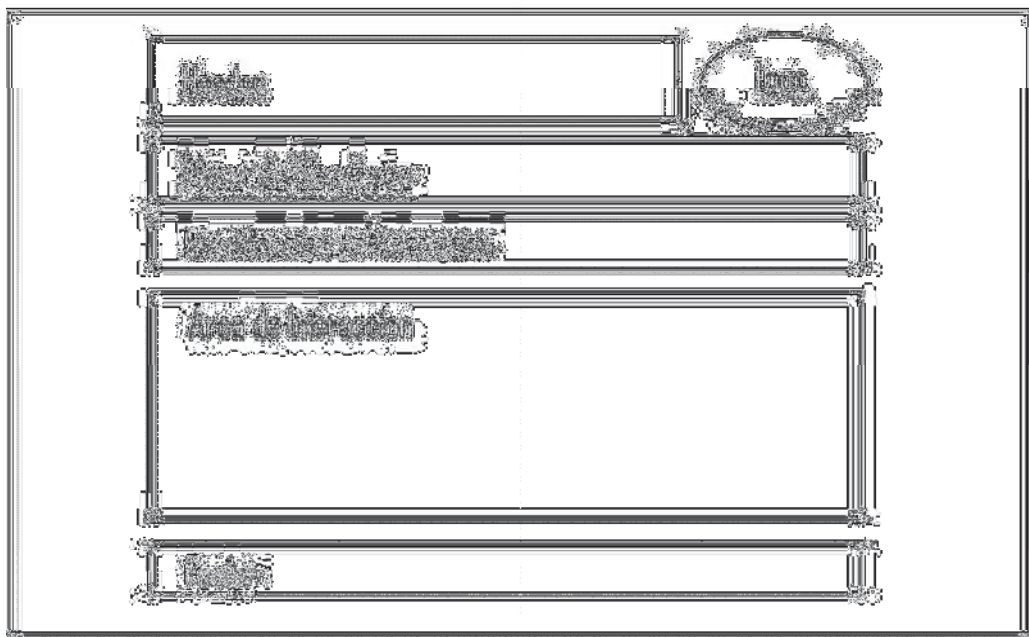


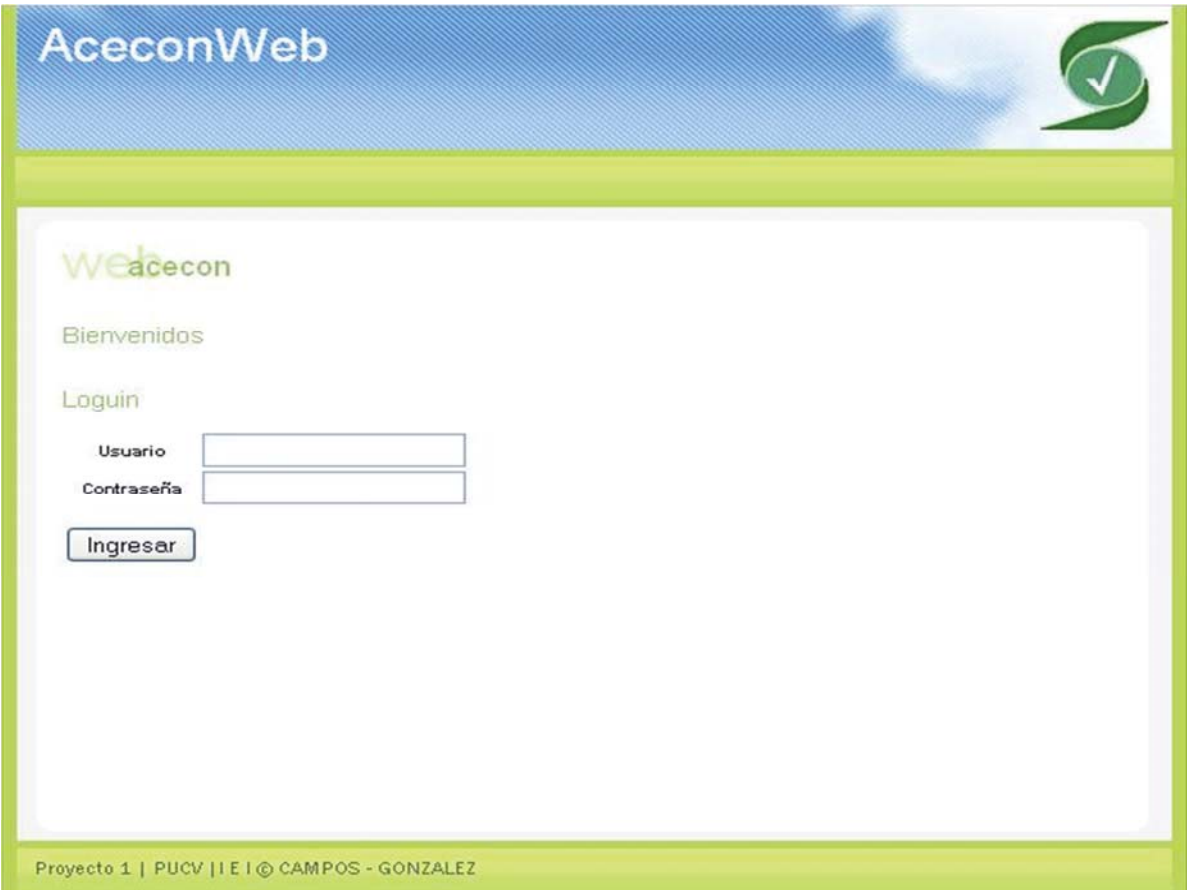
Figura 16.1. Estructura Interfaz.

- Header: Posee el nombre del Sistema en conjunto con el Logo distintivo de la empresa.
- Menú Secciones: En esta zona se detallan las secciones o categorías en las que está dividido el sitio web. Es recomendado ubicarlo en la parte superior izquierda del sitio y se debe evitar el uso de nombres complejos y preferir palabras de fácil y rápida comprensión.
- Identificación de Secciones: Debe estar en la zona superior de la página, puede ser gráfico y por lo mismo tener alguna imagen alusiva a la sección o categoría o bien ser una solución que incorpore sólo texto y color, el texto debe ser de fácil entendimiento y el color debe generar contraste con el fondo para facilitar su lectura.

- Área de Interacción: Como su nombre lo indica es el área donde se ofrece la realización de acciones por parte de los usuarios del sitio web. Las áreas de interacción son las que muestran botones y en ellas no se lee, sino que se ingresa información y se desarrolla la actividad que el sitio ofrece llevar a cabo.
- Footer: Se entiende que la zona inferior de cada pantalla cumple el relevante papel de completar su la información que se ofrece en las zonas superiores de navegación, al entregar datos relativos a la organización (nombre, direcciones, teléfonos), política de privacidad y repetir enlaces que se han entregado en la zona superior, todo esto para facilitar el contacto del usuario con el sitio.

16.2 Interfaces

16.2.1 Interfaz de inicio



The image shows a web browser window displaying the login page for 'AceconWeb'. The page has a blue header with the text 'AceconWeb' and a green logo on the right. Below the header is a green horizontal bar. The main content area is white and contains the following elements: the 'webacecon' logo, the text 'Bienvenidos', the heading 'Loguin', two input fields labeled 'Usuario' and 'Contraseña', and a button labeled 'Ingresar'. At the bottom of the page, there is a green footer with the text 'Proyecto 1 | PUCV | I E I © CAMPOS - GONZALEZ'.

Figura 16.2. Interfaz de Inicio del Sistema

16.2.2 Interfaz Portada



Figura 16.3. Interfaz de bienvenida del Sistema.

Una vez que el usuario se haya logueado, el sistema desplegará la interfaz Portada con el menú de acciones que se puede apreciar en la parte superior.

16.2.3 Interfaz Agregar Cliente Administrador

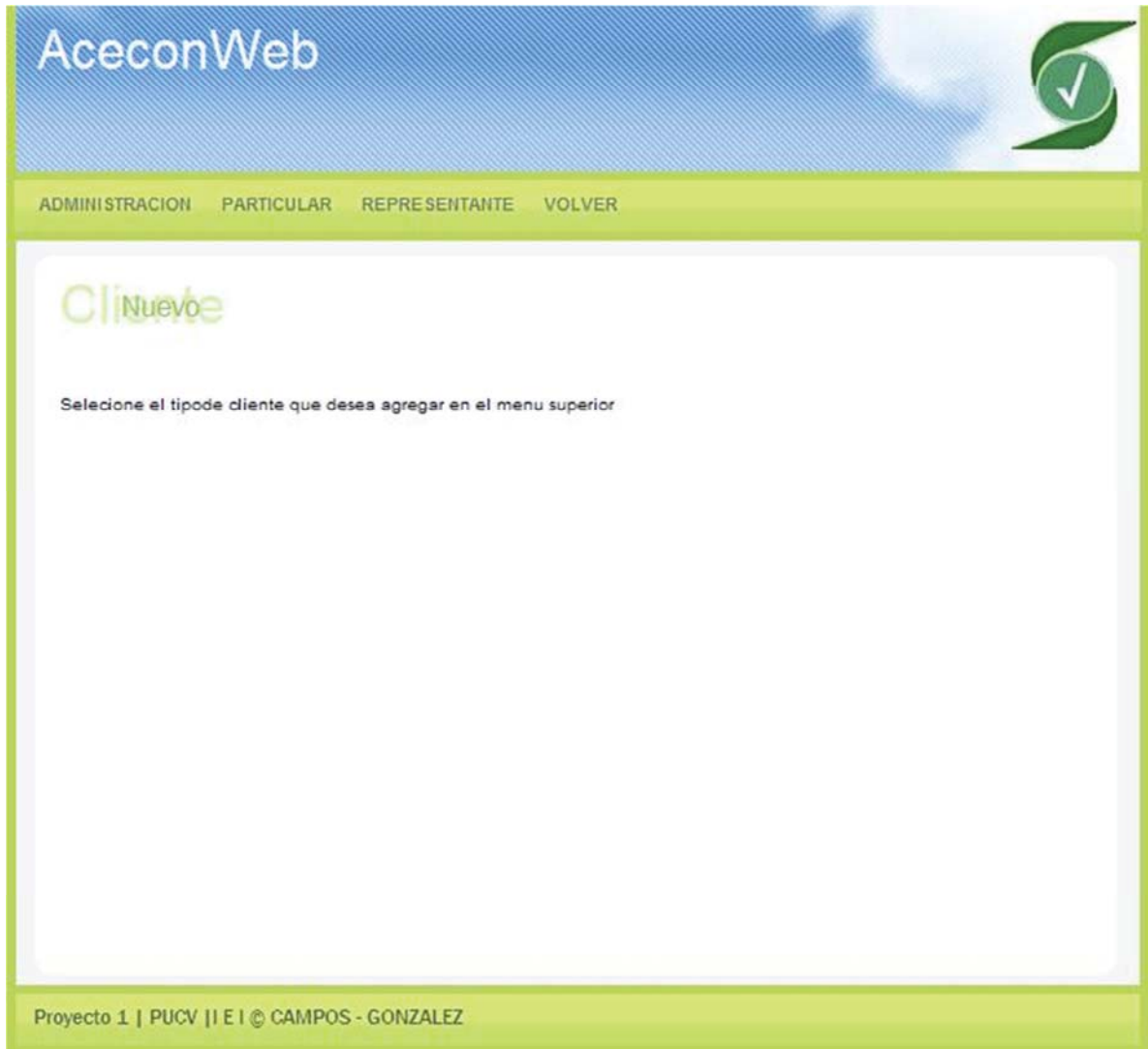


Figura 16.4. Interfaz Agregar Cliente

Interfaz Agregar Cliente: En esta interfaz el usuario podrá ingresar un nuevo cliente ya sea Cliente Particular o Cliente Administración (en el caso de condominios) o un nuevo Representante Legal de condominios al sistema, llenando todos los datos correspondientes incluyendo el tipo de cliente (cliente Acecon o Mademsa).

16.2.4 Interfaz Modificar Cliente Administrador

AceconWeb

AGREGAR BUSCAR/MODIFICAR VOLVER

CliNuevo

Datos de la propiedad

Rut (*) 6423852 -- 3

Direccion(*) N Comuna SELECCIONE

Nombre Comunidad

Tipo Cliente Acecon

Seleccione el representante

15340102-0 GABRIEL GONZALEZ

Agregar

Proyecto 1 | PUCV | I E I © CAMPOS - GONZALEZ

Figura 16.5. Interfaz Modificar Cliente.

Interfaz Agregar Cliente Administrador, para poder utilizar esta interfaz el usuario debe previamente haber ingresado el rut del cliente el cual será validado para registrar su existencia, una vez ingresado se deben llenar los campos y seleccionar de una lista dinámica el representante legal del edificio o condominio el cual deberá ser ingresado previamente, en esta imagen se pueden apreciar los datos de Gabriel González como Representante seleccionado.

16.2.5 Interfaz Buscar Clientes

AceconWeb

Inicio Presupuesto OT Clientes Vendedor Tecnico Obras Salir

Buscar Cliente

Ingrese rut del cliente

Validar

Seleccione cliente

Tipo Cliente: Administrador ▼

Nombre

Propiedad Buscar

Proyecto 1 | PUCV | I E | © CAMPOS - GONZALEZ

Figura 16.6. Interfaz Buscar Cliente.

Interfaz Buscar/Modificar Cliente: El usuario puede buscar un cliente en particular mediante dos filtros disponibles por Rut o por Nombre y Apellido, en la primera instancia el sistema mostrará los datos inmediatamente de encontrar coincidencias ya sean de Cliente Particular como de Cliente Administración, en la búsqueda por apellido el sistema mostrará una lista de clientes y previa selección del adecuado mostrará los datos correspondientes, seleccionando previamente si el cliente es Particular o Administración.

16.2.6 Interfaz Modificar Obra

AceconWeb

Inicio Presupuesto OT Clientes Vendedor Tecnico Obras Salir

Obra Modificar

Ingrese los nuevos datos para la obra.

Nombre AFIANZA LLAVE DE PASO O CAÑERÍA

Precio (NETO) 6000

Estado ACTIVO

Modificar

Proyecto 1 | PUCV | I E I © CAMPOS - GONZALEZ

Figura 16.7. Interfaz Modificar Obra.

Interfaz Modificar Obra, podemos apreciar el menú superior que nos da las distintas opciones sobre las obras que la empresa puede realizar, en este caso solo se podrá modificar el precio de la obra ya que puede variar con el tiempo, con motivo de persistencia en la base de datos se le ha asignado un estado a las obras, si el precio cambia la obra anterior se mantiene como invalida para que en posteriores búsquedas solo se muestren los últimos valores de obras.

16.2.7 Interfaz Agregar Presupuesto

AceconWeb

Inicio Presupuesto OT Cientes Vendedor Tecnico Obras Salir

Nuevo Presupuesto

Datos Clientes

Rut	<input type="text" value="11111111"/>	-	<input type="text" value="1"/>	Tipo Cliente	<input type="text" value="acecon"/>
Nombre	<input type="text" value="sebastian"/>	Apellido Paterno	<input type="text" value="perez"/>	Apellido Materno	<input type="text" value="davila"/>
Direccion	<input type="text" value="cerro alegre"/>	N°	<input type="text" value="1276"/>	Depto	<input type="text" value="767"/>
Comuna	<input type="text" value="pedro aguirre cerd"/>	Email	<input type="text" value="seba@hotmail.cor"/>		
Telefono	<input type="text" value="45678"/>	Cel	<input type="text" value="345678"/>		

Datos presupuesto

Fecha Garantia Duracion

Tipo de Pago

Anticipo Termino Tipo Pago

Personal Asignado

Vendedor Tecnico

Ingrese las obras a realizar y presione ENTER

ID	Nombre	Neto	Iva	
0				

TOTAL IVA	0	TOTAL NETO	0	TOTAL	0	DSCTO CLIENTE	S/N	TOTAL C/N DSCTO	0
-----------	---	------------	---	-------	---	---------------	-----	-----------------	---

Proyecto 1 | PUCV | I E I © CAMPOS - GONZALEZ

Figura 16.8. Interfaz Agregar Presupuesto.

16.2.8 Interfaces Aplicación Móvil

Las imágenes que se mostrarán a continuación corresponden a la aplicación móvil desarrollada para la empresa. La aplicación permite diariamente descargar las tareas a realizar por el trabajador, mostrando datos importantes como lo son el cliente dirección y hora de cada tarea.

a) Interfaz “Portada”

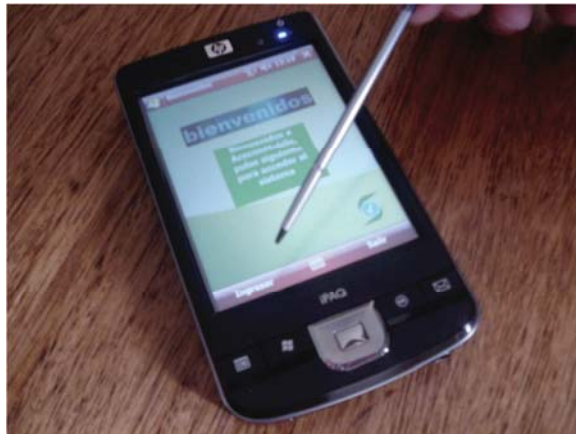


Figura 16.9. Interfaz Móvil Bienvenidos.

Primera interfaz mostrada al momento de iniciar el sistema, su única función es la de dar la bienvenida al usuario.

b) Interfaz “Login”

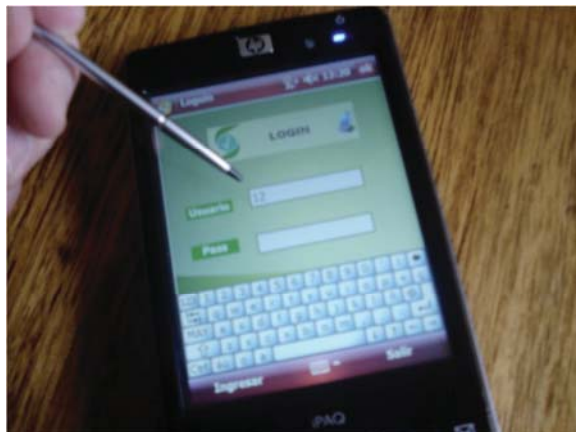


Figura 16.10. Interfaz Móvil Login.

La presente interfaz es la encargada de validar al usuario que desea ingresar al sistema, solicitando nombre usuario y password, una vez estos campos son validados da paso al menú principal de la aplicación.

c) Interfaz “Menú Principal”

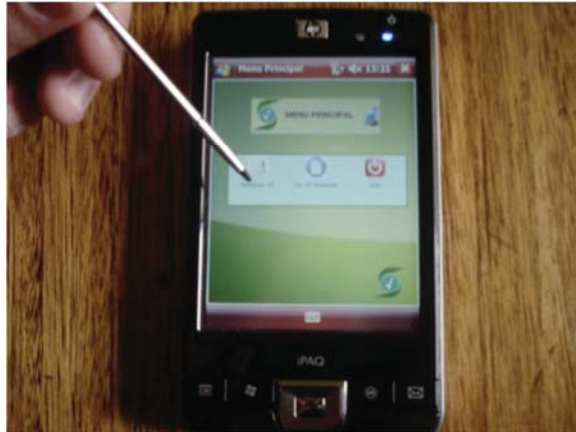


Figura 16.11. Interfaz Móvil Menú Principal.

Menú principal donde el usuario puede seleccionar descargar órdenes de trabajo, estas órdenes de trabajo especifican las obras a realizar durante el día, en una segunda opción el usuario puede descargar las órdenes de trabajo pendientes a realizar.

d) Interfaz “Ordenes de Trabajo”

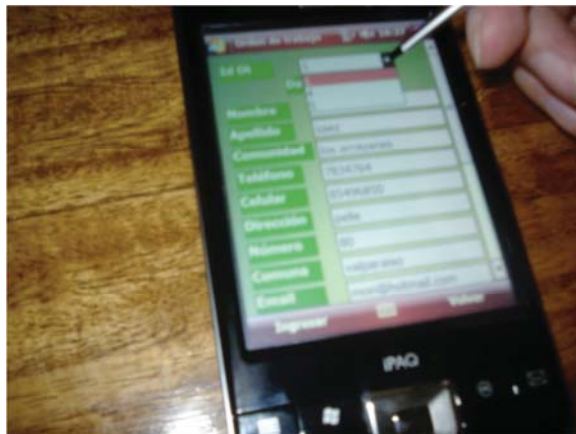


Figura 16.12. Interfaz Móvil Orden de Trabajo.

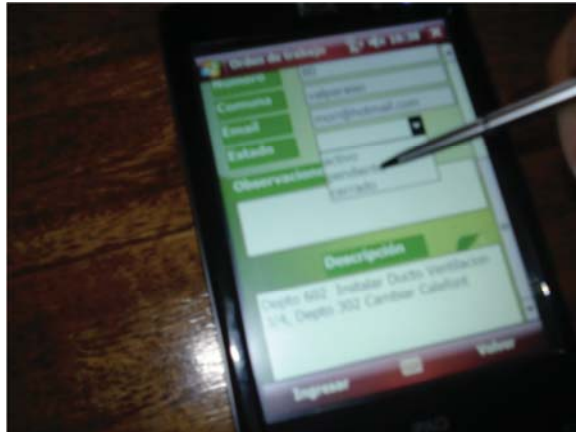


Figura 16.13. Interfaz Móvil Orden de Trabajo Pendiente.

La interfaz cuenta con un CheckBox el cual tiene como función la de seleccionar la orden de trabajo a realizar mediante la ID de la misma, estas OT (ordenes de trabajo) se cargarán dinámicamente al sistema y una vez realizada la OT el usuario tiene la posibilidad de cambiar el estado de la misma ha cerrado, dejando constancia de la finalización de la obra, esta interfaz también da la posibilidad de ingresar un campo observaciones, donde el usuario podrá registrar todo suceso no planificado.

CAPÍTULO 17: PLAN DE PRUEBAS

Mediante el presente plan de pruebas se espera Verificar (Actividades que aseguran la implementación correcta de una función) y Validar (Actividades que aseguran que el software construido corresponde a los requerimientos) una serie de módulos que se encuentran en etapas finales de desarrollo, gracias a este plan de pruebas podremos también asegurar la calidad del sistema en construcción ya que permite la detección de defectos (pero no su ausencia) los cuales pueden ir siendo corregidos en iteraciones consecutivas [1]. A continuación se pasará a detallar lo que es un plan de pruebas de software antes de mostrar las conclusiones obtenidas.

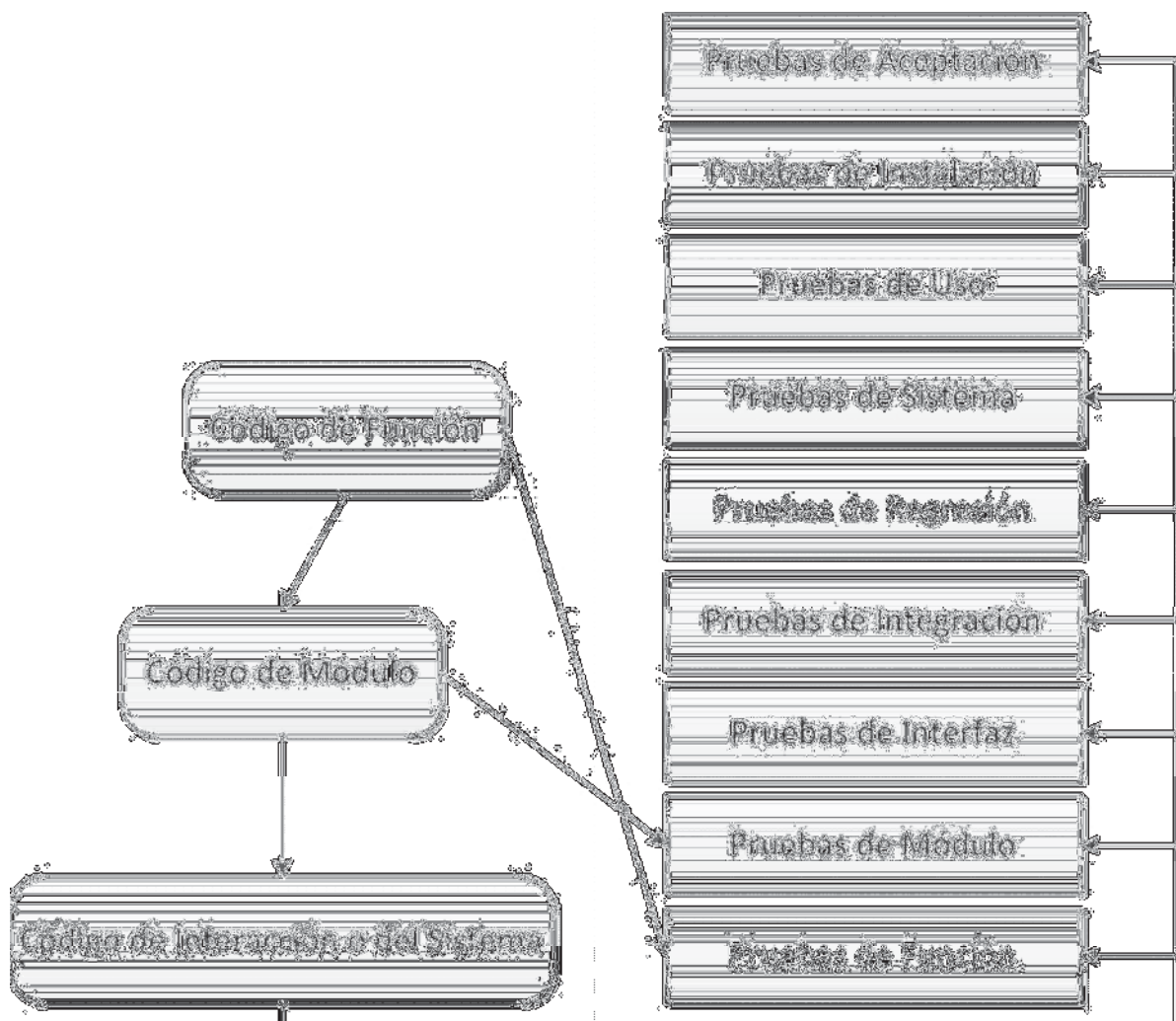


Figura 17.1. Tipos de Pruebas.

• Pruebas de Función:

Las Pruebas de Función son las que se realizan a nivel de Métodos. En estas pruebas verificamos las operaciones con valores normales de los parámetros, valores límites y valores que excedan estos límites, con esto se espera revisar el comportamiento del sistema, también se registra que se ejecuten todas las instrucciones requeridas verificando el uso de todos los objetos llamados.

• Pruebas de Módulo:

Pruebas de Módulo son las que se realizan a nivel de Clases donde ejecutamos los métodos de una clase en combinación a otros registrando y comprobando la secuencia de llamados entre unos con otros. Se puede decir que se prueban los objetos en términos de estados.

• Pruebas de Interfaz:

Una vez desarrollados los módulos se deben probar las interfaces generando tráfico entre las mismas mediante el llamado a funciones, se espera corroborar y validar el traspaso de mensajes entre clases, entregando los datos esperados.

• Pruebas de Integración:

Se realiza una prueba de integración verificando que el ensamblado de las partes se haya hecho según el plan, se prueban las funcionalidades en el contexto completo.

Para el proceso de integración se recomienda:

- Implementación de casos de uso completos
- Construir la interfaz pronto (prioridad a lo importante) para probar desde ella
- Organización metódica de un número grande de pruebas
- Congelar versiones para las pruebas.
- Generar nuevas versiones después de pruebas.
- Construir módulos en forma ascendente, así más tarde nuevos módulos ya poseen los de más abajo validados para su utilización.

• Pruebas de Regresión:

Una vez se ha realizado la integración de los nuevos módulos se debe verificar que los cambios no hayan afectado la funcionalidad ya existente, por lo cual se deben pasar el mismo tipo de pruebas antes de generar los cambios, y estas pruebas de regresión se deben realizar con frecuencia para validar la funcionalidad del sistema.

- **Pruebas de Sistema:**

Para las pruebas de sistema se ejecutan las llamadas Pruebas de Caja Negra donde se busca representar de mejor manera un conjunto infinito de posibilidades con un número finito representativo de pruebas, el problema existente es que es imposible agotar el número de posibilidades de entradas por lo cual se debe:

- Analizar valores comunes en el sistema y valores de frontera
- Valores fuera de rango

- **Pruebas de Uso:**

Son pruebas de transición realizadas antes de la liberación del sistema comercialmente, se genera una retroalimentación por parte de los usuarios del sistema la cual no afecta la reputación del producto no liberado. Para estas pruebas se selecciona una población de usuarios internos de confianza así como una población de clientes con entendimiento en lo que se está realizando, se espera cierta recolección de defectos no descubiertos en las pruebas anteriores para luego ser corregidos.

En estas pruebas también se pueden evaluar:

- **Accesibilidad:** Facilidad con la que entran, navegan y salen los usuarios
- **Rapidez de respuesta:** Que tan rápido logra el usuario sus metas
- **Eficiencia:** Que tan pequeños son los pasos para una funcionalidad
- **Comprensión:** Entendimiento a partir del uso y documentación

- **Pruebas de Instalación:**

Se prueba el sistema en el ambiente de hardware final, realizando las pruebas de sistema en el mismo.

- **Pruebas de Aceptación:**

Se entrega documentación a los usuarios sobre el sistema y se realizan las mismas pruebas de sistema pero con el cliente presente, estas pruebas ayudan al cliente a estar seguro de que se entregó es lo solicitado, se pueden realizar pruebas de aceptación parciales del producto de ser necesario, implementando distintas funcionalidades del sistema.

17.1 Plan de pruebas

Para el presente plan de pruebas se realizarán pruebas de caja negra tanto para la aplicación web como para la aplicación móvil, estas pruebas serán de sistema e interfaz ya que las pruebas de función y módulos fueron desarrollándose al mismo tiempo que la programación mediante revisiones de codificación.

17.1.1 Módulo Técnico

Módulo a Probar	Datos de Entrada	Datos esperados de Salida	Datos de Salida
Agregar Técnico (Rut)	Rut válido	Sistema da paso a ingresar datos del Técnico	Ok
Agregar Técnico (Rut)	Llenar Campos con datos erróneos o en blanco	Mensaje de Alerta: Campo N datos inválidos	Ok
Agregar Técnico (Rut)	Nombre: Francisco Apellido Paterno: González Apellido Materno: Urrutia Teléfono: 223355 Cel: 95936805 Clase: 1 Cargo Jefe: Licencia Sec: 3456	Mensaje de Alerta: Ingreso Exitoso	Ok
Agregar Técnico (Rut)	Rut: 12345678-9	Mensaje de Alerta: El rut Ingresado es Inválido	Ok
Agregar Técnico (Rut)	Datos ya Existentes Rut: Rut ya existente	Mensaje de Alerta: El Técnico ya Existe	Ok
Modificar Técnico (Buscar Técnico por Rut o Nombre)	Campos en Blanco	Mensaje de Alerta: Rut Invalido o Campo N caracteres Inválidos	Ok
Modificar Técnico (Buscar Técnico por Rut)	Rut No Existente Rut: válido	Mensaje de Alerta: Técnico no Existe	O k
Modificar Técnico (Buscar Técnico por	Técnico Existente Rut: válido	Sistema muestra datos del Técnico a modificar	Ok

Rut)			
Modificar Técnico (Buscar Técnico por Nombre)	Técnico Existente Nombre: Fr Apellido Paterno: Gon	Sistema muestra lista de Técnicos que contengan estos datos para selección	Ok
Modificar Técnico (Buscar Técnico por Nombre)	Seleccionar Técnico de la lista previamente mostrada	Sistema muestra datos del Técnico a modificar	Ok
Modificar Técnico (Buscar Técnico por Nombre)	Técnico Inexistente Nombre: U Apellido Paterno: U	Mensaje de Alerta: Técnico no Existe	Ok
Modificar Técnico	Nombre: Francisco Nuevo Nombre: Gabriel	Mensaje de Alerta: Modificación realizada con Éxito	Ok
Buscar Técnico (Buscar Técnico por Rut o Nombre)	Campos en Blanco	Mensaje de Alerta: Rut Invalido O Campo N caracteres Inválidos	Ok
Buscar Técnico (Buscar Técnico por Rut)	Rut No Existente Rut: válido	Mensaje de Alerta: Técnico no Existe	Ok
Buscar Técnico (Buscar Técnico por Rut)	Técnico Existente Rut: válido	Sistema muestra datos del Técnico a modificar	Ok
Buscar Técnico (Buscar Técnico por Nombre)	Técnico Existente Nombre: Ga Apellido Paterno: Gon	Sistema muestra lista de Técnicos que contengan estos datos para selección	Ok
Buscar Técnico (Buscar Técnico por Nombre)	Seleccionar Técnico de la lista previamente mostrada	Sistema muestra datos del Técnico	Ok

Tabla 17.1. Plan de pruebas módulo técnico.

17.1.2 Módulo Vendedor

Módulo a Probar	Datos de Entrada	Datos esperados de Salida	Datos de Salida
Agregar Vendedor (Rut)	Rut: válido	Sistema da paso a ingresar datos del Vendedor	Ok
Agregar Vendedor (Rut)	Llenar Campos con datos erróneos o en blanco	Mensaje de Alerta: Campo N datos inválidos	Ok
Agregar Vendedor (Rut)	Nombre: Francisco Apellido Paterno: González Apellido Materno: Urrutia Email: voodo17@hotmail.com Teléfono: 223355 Cel: 95936805	Mensaje de Alerta: Ingreso Exitoso	Ok
Agregar Vendedor (Rut)	Rut: 12345678-9	Mensaje de Alerta: El rut Ingresado es Inválido	Ok
Agregar Vendedor (Rut)	Datos ya Existentes Rut: válido	Mensaje de Alerta: El Vendedor ya Existe	Ok
Modificar Vendedor (Buscar Vendedor por Rut o Nombre)	Llenar Campos con datos erróneos o en blanco	Mensaje de Alerta: Rut Invalido o Campo N caracteres Inválidos	Ok
Modificar Vendedor (Buscar Vendedor por Rut)	Rut No Existente Rut: válido	Mensaje de Alerta: Vendedor no Existe	Ok
Modificar Vendedor (Buscar Vendedor por Rut)	Vendedor Existente Rut: válido	Sistema muestra datos del Vendedor a modificar	Ok
Modificar Vendedor (Buscar Vendedor por Nombre)	Vendedor Existente Nombre: Fr Apellido Paterno: Gon	Sistema muestra lista de Vendedores que contengan estos datos para selección	Ok
Modificar	Seleccionar Vendedor de la	Sistema muestra	Ok

Vendedor (Buscar Vendedor por Nombre)	lista previamente mostrada	datos del Vendedor a modificar	
Modificar Vendedor (Buscar Vendedor por Nombre)	Vendedor Inexistente Nombre: U Apellido Paterno: U	Mensaje de Alerta: Vendedor no Existe	Ok
Modificar Vendedor	Nombre: Francisco Nuevo Nombre: Gabriel	Mensaje de Alerta: Modificación realizada con Éxito	Ok
Buscar Vendedor (Buscar Vendedor por Rut o Nombre)	Campos en Blanco	Mensaje de Alerta: Rut Invalido o Campo N caracteres Inválidos	Ok
Buscar Vendedor (Buscar Vendedor por Rut)	Rut No Existente Rut: válido	Mensaje de Alerta: Vendedor no Existe	Ok
Buscar Vendedor (Buscar Vendedor por Rut)	Vendedor Existente Rut: válido	Sistema muestra datos del Vendedor a modificar	Ok
Buscar Vendedor (Buscar Vendedor por Nombre)	Vendedor Existente Nombre: Ga Apellido Paterno: Gon	Sistema muestra lista de Vendedores que contengan estos datos para selección	Ok
Buscar Vendedor (Buscar Vendedor por Nombre)	Seleccionar Vendedor de la lista previamente mostrada	Sistema muestra datos del Vendedor	Ok

Tabla 17.2. Plan de pruebas módulo vendedor.

17.1.3 Módulo Obras

Módulo a Probar	Datos de Entrada	Datos esperados de Salida	Datos de Salida
Agregar Obra	Nombre: Reparación Fuga Precio: 50000 Estado: Activo	Mensaje de Alerta: Ingreso Exitoso	Ok
Agregar Obra	Campos en Blanco	Mensaje de Alerta: Campo N datos inválidos	Ok
Agregar Obra	Precio Menor a 0 o Mayor a 9999999	Mensaje de Alerta: Precio Invalido	Ok
Agregar Obra	Estado Activo	Mensaje de Alerta: Ingreso Exitoso	Ok
Modificar Obra	Dejar Campo en Blanco	Listado de todas las obras existentes	Ok
Modificar Obra	Nombre: Rep	Sistema muestra lista de Obras que contengan este string en su nombre para selección	Ok
Modificar Obra	Nombre: Xxxx	Mensaje de Alerta: Obra no Existe	Ok
Modificar Obra	Nombre: Reparación Fuga Precio: 50000 Nuevo Precio: 99999	Mensaje de Alerta: Modificación Exitosa	Ok
Modificar Obra	Nombre: Reparación Fuga Precio: 50000 Nuevo Precio: -1 o 999999999999	Mensaje de Alerta: Precio Invalido	Ok
Buscar Obra	Dejar Campo en Blanco	Listado de todas las obras existentes	Ok
Buscar Obra	Nombre: Rep	Sistema muestra lista de Obras que contengan este string en su nombre para selección	Ok
Buscar Obra	Nombre: Xxxx	Mensaje de Alerta: Obra no Existe	Ok
Dar de Baja Obra	Dejar Campo en Blanco	Listado de todas las obras existentes	Ok
Dar de Baja Obra	Nombre: Xxxx	Mensaje de Alerta: Obra no Existe	Ok
Dar de Baja Obra	Nombre: Rep	Sistema muestra lista	Ok

Obra		de Obras que contengan este string en su nombre para selección	
Dar de Baja Obra	Estado: De Baja	Mensaje de Alerta: Obra de Baja	Ok
Dar de Baja Obra	Estado: Activo	Mensaje de Alerta: Obra Activa	Ok

Tabla 17.3. Plan de pruebas módulo obras.

17.1.4 Módulo Representante

Módulo a Probar	Datos de Entrada	Datos esperados de Salida	Datos de Salida
Agregar Representante (Rut)	Rut: válido	Sistema da paso a ingresar datos del Representante	Ok
Agregar Representante (Rut)	Llenar Campos con datos erróneos o en blanco	Mensaje de Alerta: Campo N datos inválidos	Ok
Agregar Representante (Rut)	Nombre: Francisco Apellido Paterno: González Apellido Materno: Urrutia Email: voodoo17@hotmail.com Teléfono: 223355 Cel: 95936805 Observaciones: asdasd	Mensaje de Alerta: Ingreso Exitoso	Ok
Agregar Representante (Rut)	Rut: 12345678-9	Mensaje de Alerta: El rut Ingresado es Inválido	Ok
Agregar Representante (Rut)	Datos ya Existentes Rut: válido	Mensaje de Alerta: El Representante ya Existe	Ok
Modificar Representante (Buscar Representante por Rut o Nombre)	Llenar Campos con datos erróneos o en blanco	Mensaje de Alerta: Rut Invalido o Campo N caracteres Inválidos	Ok
Modificar Representante (Buscar Representante por Rut)	Rut No Existente Rut: válido	Mensaje de Alerta: Representante no Existe	Ok

Modificar Representante (Buscar Representante por Rut)	Representante Existente Rut: válido	Sistema muestra datos del Representante a modificar	Ok
Modificar Representante (Buscar Representante por Nombre)	Representante Existente Nombre: Fr Apellido Paterno: Gon	Sistema muestra lista de Representantes que contengan estos datos para selección	Ok
Modificar Representante (Buscar Representante por Nombre)	Seleccionar Representante de la lista previamente mostrada	Sistema muestra datos del Representante a modificar	Ok
Modificar Representante (Buscar Representante por Nombre)	Representante Inexistente Nombre: U Apellido Paterno: U	Mensaje de Alerta: Representante no Existe	Ok
Modificar Representante	Nombre: Francisco Nuevo Nombre: Gabriel	Mensaje de Alerta: Modificación realizada con Éxito	Ok
Buscar Representante (Buscar Representante por Rut o Nombre)	Campos en Blanco	Mensaje de Alerta: Rut Invalido o Campo N caracteres Inválidos	Ok
Buscar Representante (Buscar Representante por Rut)	Rut No Existente Rut: válido	Mensaje de Alerta: Representante no Existe	Ok
Buscar Representante (Buscar Representante por Rut)	Representante Existente Rut: válido	Sistema muestra datos del Representante	Ok
Buscar Representante (Buscar Representante por Nombre)	Representante Existente Nombre: Ga Apellido Paterno: Gon	Sistema muestra lista de Representantes que contengan estos datos para selección	Ok
Buscar Representante	Seleccionar Representante de la lista previamente mostrada	Sistema muestra datos del	Ok

(Buscar Representante por Nombre)		Representante	
-----------------------------------	--	---------------	--

Tabla 17.4. Plan de pruebas módulo representante.

17.1.5 Módulo Cliente Particular

Módulo a Probar	Datos de Entrada	Datos esperados de Salida	Datos de Salida
Agregar Cliente Particular (Rut)	Rut: válido	Sistema da paso a ingresar datos del Cliente Particular	Ok
Agregar Cliente Particular (Rut)	Llenar Campos con datos erróneos o en blanco	Mensaje de Alerta: Campo N datos inválidos	Ok
Agregar Cliente Particular (Rut)	Nombre: Francisco Apellido Paterno: González Apellido Materno: Urrutia Dirección: Pelle N: 80 Depto: 602c Comuna: San Bernardo Email: voodo17@hotmail.com Teléfono: 223355 Cel: 95936805 Tipo: Acecon	Mensaje de Alerta: Ingreso Exitoso	Ok
Agregar Cliente Particular (Rut)	Rut: 12345678-9	Mensaje de Alerta: El rut Ingresado es Inválido	Ok
Agregar Cliente Particular (Rut)	Datos ya Existentes Rut: válido	Mensaje de Alerta: El Cliente ya Existe	Ok
Buscar/Modificar Cliente Particular (Buscar Cliente por Rut o Nombre)	Llenar Campos con datos erróneos o en blanco	Mensaje de Alerta: Rut Invalido o Campo N caracteres Inválidos	Ok
Buscar/Modificar Cliente Particular (Buscar Cliente por Rut)	Rut No Existente Rut: válido	Mensaje de Alerta: Cliente no Existe	Ok
Buscar/Modificar Cliente Particular	Cliente Existente	Sistema muestra datos del Cliente y	Ok

(Buscar Cliente por Rut)	Rut:15340102-0	permite seleccionar modificar	
Buscar/Modificar Cliente Particular (Buscar Cliente por Nombre)	Cliente Existente Nombre: Fr Apellido Paterno: Gon	Sistema muestra lista de Clientes que contengan estos datos para selección	Ok
Buscar/Modificar Cliente Particular (Buscar Cliente por Nombre)	Seleccionar Cliente de la lista previamente mostrada	Sistema muestra datos del Cliente a y permite seleccionar modificar	Ok
Buscar/Modificar Cliente Particular (Buscar Cliente por Nombre)	Cliente Inexistente Nombre: U Apellido Paterno: U	Mensaje de Alerta: Cliente no Existe	Ok
Modificar Cliente Particular	Nombre: Francisco Nuevo Nombre: Gabriel	Mensaje de Alerta: Modificación realizada con Éxito	Ok

Tabla 17.5. Plan de pruebas módulo cliente particular.

17.1.6 Módulo Cliente Administración

Módulo a Probar	Datos de Entrada	Datos esperados de Salida	Datos de Salida
Agregar Cliente Administración (Rut)	Rut: válido	Sistema da paso a ingresar datos del Cliente Administrador	Ok
Agregar Cliente Administración (Rut)	Llenar Campos con datos erróneos o en blanco	Mensaje de Alerta: Campo N datos inválidos	Ok
Agregar Cliente Administración (Rut)	Dirección: San Rafael N: 80 Comunidad: Comunidad San Rafael Tipo Cliente: Acecon Selección Representante: Gabriel González (previamente ingresado)	Mensaje de Alerta: Ingreso Exitoso	Ok
Agregar Cliente Administración (Rut)	Rut: 12345678-9	Mensaje de Alerta: El rut Ingresado es Inválido	Ok
Agregar Cliente Administración (Rut)	Datos ya Existentes Rut: válido	Mensaje de Alerta: El Cliente ya Existe	Ok
Buscar/Modificar	Llenar Campos con datos	Mensaje de Alerta:	Ok

Cliente Administración (Buscar Cliente por Rut o Nombre)	erróneos o en blanco	Rut Invalido o Campo N caracteres Inválidos	
Buscar/Modificar Cliente Administración (Buscar Cliente por Rut)	Rut No Existente Rut: válido	Mensaje de Alerta: Cliente no Existe	Ok
Buscar/Modificar Cliente Administración (Buscar Cliente por Rut)	Cliente Existente Rut: válido	Sistema muestra datos del Cliente y permite seleccionar modificar	Ok
Buscar/Modificar Cliente Administración (Buscar Cliente por Nombre)	Cliente Existente Nombre: Fr Apellido Paterno: Gon	Sistema muestra lista de Clientes que contengan estos datos para selección	Ok
Buscar/Modificar Cliente Administración (Buscar Cliente por Nombre)	Seleccionar Cliente de la lista previamente mostrada	Sistema muestra datos del Cliente a y permite seleccionar modificar	Ok
Buscar/Modificar Cliente Administración (Buscar Cliente por Nombre)	Cliente Inexistente Nombre: U Apellido Paterno: U	Mensaje de Alerta: Cliente no Existe	Ok
Modificar Cliente Administración	Nombre: Francisco Nuevo Nombre: Gabriel	Mensaje de Alerta: Modificación realizada con Éxito	Ok

Tabla 17.6. Plan de pruebas módulo cliente administración.

17.1.7 Módulo Presupuesto

Módulo a Probar	Datos de Entrada	Datos esperados de Salida	Datos de Salida
Nuevo Presupuesto (Rut)	Rut: válido	Sistema da paso a ingresar datos del Nuevo Presupuesto	Ok
Nuevo Presupuesto	Llenar campos con datos erróneos o en blanco	Mensaje de Alerta: Campo N datos	Ok

		inválidos ó Campo N llenar datos	
Nuevo Presupuesto	Botón Ver Obras	Sistema despliega pop-up con las obras actuales de la empresa	Ok
Nuevo Presupuesto	Seleccionar Fecha, Garantía, Duración, Anticipo, Tipo de Pago, Vendedor , Técnico, Seleccionar Obras	Mensaje de Alerta: Ingreso Exitoso	Ok
Ver Presupuesto	Campos Inválidos en alguno de los filtros	Mensaje de Alerta: Campo N datos inválidos	Ok
Ver Presupuesto	Llenar filtro de búsqueda con datos válidos	Sistema despliega información del presupuesto	Ok
Ver Presupuesto	Cambiar estado del presupuesto	Mensaje de Alerta: Cambio de estado realizado con éxito	Ok
Modificar Presupuesto	Campos Inválidos en alguno de los filtros	Mensaje de Alerta: Campo N datos inválidos	Ok
Modificar Presupuesto	Llenar filtro de búsqueda con datos válidos	Sistema despliega información del presupuesto	Ok
Modificar Presupuesto	Cambiar alguno de los campos mostrados, incluyendo técnico y vendedor	Mensaje de Alerta: Modificación realizada con éxito	Ok
Actualizar IVA	Cambiar porcentaje de IVA	Mensaje de Alerta: ¿Desea cambiar el IVA?	Ok
Actualizar IVA	Aceptar mensaje de validación	Mensaje de Alerta: Modificación realizada con Éxito	Ok

Tabla 17.7. Plan de pruebas módulo presupuesto.

17.1.8 Módulo Órdenes de Trabajo

Módulo a Probar	Datos de Entrada	Datos esperados de Salida	Datos de Salida
Nueva OT (Rut)	Rut: válido	Sistema despliega datos del presupuesto si el cliente posee uno asociado	Ok
Nueva OT (Rut)	Llenar campos con datos erróneos o en blanco	Mensaje de Alerta: Campo N datos inválidos o Campo N llenar datos	Ok
Nueva OT (Rut)	Seleccionar Crear OT	Sistema despliega formulario de Orden de Trabajo	Ok
Nueva OT (Rut)	Llenar campos con datos erróneos o en blanco	Mensaje de Alerta: Campo N datos inválidos o Campo N llenar datos	Ok
Nueva OT (Rut)	Ingresar datos válidos	Mensaje de Alerta: Ingreso Exitoso	Ok
Buscar OT	Llenar filtro de búsqueda con datos válidos	Sistema despliega la información de la orden de trabajo	Ok
Buscar OT	Seleccionar Convertir a PDF	Sistema despliega pop-up con la orden de trabajo para ser guardada en el sistema	Ok
Buscar OT	Cambiar estado de la orden de trabajo	Mensaje de Alerta: Cambio de estado realizado con éxito	Ok
Modificar OT	Llenar filtro de búsqueda con datos válidos	Sistema despliega información de la orden de trabajo	Ok
Modificar OT	Cambiar alguno de los campos mostrados, personal encargado, fecha y hora	Mensaje de Alerta: Modificación realizada con éxito	Ok

Tabla 17.8. Plan de pruebas módulo técnico.

17.1.9 Dispositivo Móvil

Requerimientos previos para el uso del dispositivo móvil:

- El usuario del sistema debe ser un técnico y encontrarse registrado en la base de datos.
- El dispositivo móvil debe estar conectado a internet.

Módulo a Probar	Datos de Entrada	Datos esperados de Salida	Datos de Salida
Login	Usuario: técnico válido	Sistema despliega el Menú Principal	Ok
Login	Rut Incorrecto Usuario:12345678	Mensaje de Alerta: El Rut y/o la contraseña ingresada no son válidos. Verifique los datos e intente de nuevo.	Ok
Login	Sin Internet	Mensaje de Alerta: Problemas con la Base de Datos	Ok
Descargar Ot	Seleccionar Descargar Ot en el Menú Principal	Mensaje de Alerta: No se encontraron Órdenes de trabajo asignadas.	Ok
Descargar Ot	Seleccionar Descargar Ot en el Menú Principal	Sistema despliega formulario de Orden de Trabajo	Ok
Descargar Ot	Seleccionar Botón Ingresar	Mensaje de Alerta: No existe Ot seleccionada	Ok
Descargar Ot	Seleccionar Ot del Combo box	Sistema completa los campos con la información referente a la Ot seleccionada	Ok
Descargar Ot	Seleccionar Botón Ingresar	Mensaje de Alerta: No existe estado seleccionado	Ok
Descargar Ot	Datos erróneos en campo Observaciones	Mensaje de Aviso: Caracteres Inválidos en Campo Observaciones.	Ok
Descargar Ot	Datos ingresados correctamente	Mensaje de Aviso: Modificación	Ok

		Realizada con Éxito	
Ver Ot Pendiente	Seleccionar Ver Ot Pendiente en el Menú Principal	Mensaje de Alerta: No se encontraron Órdenes de trabajo asignadas.	Ok
Ver Ot Pendiente	Seleccionar Ver Ot Pendiente en el Menú Principal	Sistema despliega formulario de Ordenes de Trabajo	Ok
Ver Ot Pendiente	Seleccionar Botón Ingresar	Mensaje de Alerta: No existe Ot seleccionada	Ok
Ver Ot Pendiente	Seleccionar Ot del Combo box	Sistema completa los campos con la información referente a la Ot seleccionada	Ok
Ver Ot Pendiente	Seleccionar Botón Ingresar	Mensaje de Alerta: No existe estado seleccionado	Ok
Ver Ot Pendiente	Datos erróneos en campo Observaciones	Mensaje de Aviso: Caracteres Inválidos en Campo Observaciones.	Ok
Botón Salir	Seleccionar Salir en el Menú Principal	Aplicación Cerrada	Ok

Tabla 17.9. Plan de pruebas dispositivo móvil.

CONCLUSIONES Y TRABAJO FUTURO

Dada la problemática de una empresa de construcción se logró gracias a un análisis previo, el desarrollo de una aplicación web basada en una tecnología JSP (Java Server Pages), esta aplicación cumple con las expectativas solicitadas por el cliente y se espera pueda optimizar la labor de la empresa, mejorando y facilitando en gran medida el flujo de trabajo de la compañía.

La principal ventaja de JSP frente a otros lenguajes es que el lenguaje Java es un lenguaje de propósito general que excede el mundo web y que es apto para crear clases que manejen lógica de negocio y acceso a datos de una manera detallada. Esto permite separar en niveles las aplicaciones web, dejando la parte encargada de generar el documento HTML en el archivo JSP.

Gracias a la plataforma de desarrollo Visual Studio 2008 y utilizando el .NET Compact Framework 3.5 de Microsoft se logró la elaboración de un sistema para un dispositivo móvil, el cual permitió integrar estas aplicaciones mediante servicios web. Demostrando así la interoperabilidad que brinda esta tecnología al conectar distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma.

A modo de trabajo a futuro cabe destacar que al ser un desarrollo orientado a objetos, se puede escalar en el sistema, estandarizándolo para cualquier empresa de servicios basada en presupuestos y órdenes de trabajo, a su vez agregando nuevos módulos ya sea a la aplicación web como al dispositivo móvil, en este último se espera adicionar un módulo que permita la generación de presupuestos en terreno para la empresa, permitiendo la creación de documentos PDF los cuales pueden ser fácilmente enviados vía internet a los clientes desde el mismo dispositivo móvil, agilizando aún más la gestión de la empresa.

REFERENCIAS

- [1] **Iván Rivera González.** Desarrollo de Aplicaciones para Dispositivos móviles – Tesis. 2003. UNIVERSIDAD
- [2] **Douglas Boling.** Programación de Dispositivos Móviles de Windows.- Microsoft - Jul 29, 2009
- [3] **Sitio oficial del sistema operativo Android.** Link <http://www.android.com>. Abril 26,2010
- [4] **Sommerville, I.** Ingeniería del software septima edicion. PEARSON EDUCATION, S.A. 2005.
- [5] **James Rumbaugh, Ivar Jacobson, Grady Booch.** El lenguaje unificado de modelado manual de referencia UML 2.0 2ª Edición.- Pearson - (2007)
- [6] **A. Pérez Martínez, N. Sierra Romero.** Servicios Web para Dispositivos Móviles en Sistemas de Información. – 5º Congreso Internacional de innovación y desarrollo tecnológico – Cuernavaca, Morelos, Mexico 2007.
- [7] **Joseph Schmuller.** Aprendiendo UML en 24 Horas. - Prentice Hall - 2001.
- [8] **Sergio Gálvez Rojas, Luis Ortega Díaz.** Java a tope: J2ME. Dpto. de Lenguajes y Ciencias de la Computación, E.T.S. de Ingeniería Informática Universidad de Málaga. 2003.