



# **UNIVERSIDAD TÉCNICA DEL NORTE**

**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS  
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO  
DE INGENIERO EN SISTEMAS COMPUTACIONALES**

## **TEMA:**

**Estudio de la tecnología Oracle Application Express para el  
desarrollo de aplicaciones empresariales con el prototipo de un  
sistema de facturación electrónica**

## **AUTOR:**

**Jorge Luis Quiguango Terán**

## **DIRECTOR:**

**Ing. José Luis Rodríguez**

**Ibarra – Ecuador**

**2015**

## **CERTIFICACIÓN DEL ASESOR**

Certifico que la Tesis **ESTUDIO DE LA TECNOLOGÍA ORACLE APPLICATION EXPRESS PARA EL DESARROLLO DE APLICACIONES EMPRESARIALES** con el aplicativo **PROTOTIPO DE UN SISTEMA DE FACTURACIÓN ELECTRÓNICA** ha sido realizada en su totalidad por el señor: Jorge Luis Quiguango Terán  
Portador de la cédula de identidad número: 1002456877.

.....  
Ing. José Luis Rodríguez  
Director de la Tesis

# CERTIFICACIÓN DEL AUSPICIANTE

Ibarra, 1 de abril de 2014

Señores  
UNIVERSIDAD TÉCNICA DEL NORTE  
Presente

De mis consideraciones.-

Siendo auspiciante del proyecto de tesis del Egresado JORGE LUIS QUIGUANGO TERÁN con CI: 1002456877 quien desarrolló su trabajo con el tema **ESTUDIO DE LA TECNOLOGÍA ORACLE APPLICATION EXPRESS PARA EL DESARROLLO DE APLICACIONES EMPRESARIALES** con el aplicativo **PROTOTIPO DE UN SISTEMA DE FACTURACIÓN ELECTRÓNICA**, me es grato informar que se han superado con satisfacción las pruebas técnicas y la revisión de cumplimiento de los requerimientos funcionales, por lo que se recibe el proyecto como culminado y realizado por parte del egresado JORGE LUIS QUIGUANGO TERÁN . Una vez que hemos recibido la capacitación y documentación respectiva, nos comprometemos a continuar utilizando el mencionado aplicativo en beneficio de nuestra empresa.

El egresado JORGE LUIS QUIGUANGO TERÁN puede hacer uso de este documento para los fines pertinentes en la Universidad Técnica del Norte.

Atentamente,

---

Lic. Marco Merizalde  
Jefe Gestor Asesoría Empresarial



**UNIVERSIDAD TÉCNICA DEL NORTE  
CESIÓN DE DERECHOS DE AUTOR  
DEL TRABAJO DE INVESTIGACIÓN A  
FAVOR DE LA UNIVERSIDAD TÉCNICA  
DEL NORTE**

Yo, JORGE LUIS QUIGUANGO TERÁN, con cédula de identidad Nro. 1002456877, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad intelectual del Ecuador, artículo 4, 5 y 6, en calidad de autor del trabajo de grado denominado: **ESTUDIO DE LA TECNOLOGÍA ORACLE APPLICATION EXPRESS PARA EL DESARROLLO DE APLICACIONES EMPRESARIALES** con el aplicativo **PROTOTIPO DE UN SISTEMA DE FACTURACIÓN ELECTRÓNICA**, que ha sido desarrollada para optar por el título de Ingeniería en Sistemas Computacionales, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de la obra antes mencionada, aclarando que el trabajo aquí descrito es de mi autoría y que no ha sido previamente presentado para ningún grado o calificación profesional.

En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la biblioteca de la Universidad Técnica del Norte .

.....  
Firma

Nombre: JORGE LUIS QUIGUANGO TERÁN

Cédula: 1002456877

Ibarra, a los 2 días del mes de abril del 2014



**UNIVERSIDAD TÉCNICA DEL NORTE BIBLIOTECA  
UNIVERSITARIA  
AUTORIZACIÓN DE USO Y PUBLICACIÓN  
A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL  
NORTE**

## 1. IDENTIFICACIÓN DE LA OBRA

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital institucional determina la necesidad de disponer los textos completos de forma digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual ponemos a disposición la siguiente investigación:

DATOS DE CONTACTO	
CEDULA DE IDENTIDAD	1002456877
APELLIDOS Y NOMBRES	QUIGUANGO TERÁN JORGE LUIS
DIRECCIÓN	Canamvalle, calle los Higos
EMAIL	jorgequiguango@yahoo.com
TELÉFONO FIJO	062542173
TELÉFONO MOVIL	0969528480

DATOS DE LA OBRA	
TITULO	“ESTUDIO DE LA TECNOLOGÍA ORACLE APPLICATION EXPRESS PARA EL DESARROLLO DE APLICACIONES EMPRESARIALES”
AUTOR	QUIGUANGO TERÁN JORGE LUIS
FECHA	2 de abril del 2014
PROGRAMA	PREGRADO
TITULO POR EL QUE	INGENIERÍA EN SISTEMAS COMPUTACIONALES
DIRECTOR	Ing. José Luis Rodríguez

## 2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, QUIGUANGO TERÁN JORGE LUIS, con cedula de identidad Nro. 1002456877, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y el uso del archivo digital en la biblioteca de la universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión, en concordancia con la Ley de Educación Superior Artículo 143.

.....  
Firma

Nombre: QUIGUANGO TERÁN JORGE LUIS

Cédula: 1002456877

Ibarra, a los 2 días del mes de abril del 2014

## **DEDICATORIA**

A Martha Terán mi madre, por todo el amor brindado desde el momento en que nací, la dedicación para con sus hijos, cuidado, respeto por la vida y el apoyo en cada día de mi vida. A Jorge Quiguango mi padre, por siempre estar al pendiente de toda familia velando por nuestro bienestar y en triunfo de sus hijos. A mis compañeros y amigos que lo conocí en el transcurso de la carrera con los que estude muchos años.

## **AGRADECIMIENTO**

A mis padres, por darme todo lo que soy como persona, mis valores, mis principios, mi perseverancia y mi empeño.

A mis amigos y compañeros, por estar siempre a mi lado y compartir momentos especiales en la carrera.

A mis profesores, por su esfuerzo y dedicación. Sus conocimientos, orientaciones, tiempo, predisposición y motivación han sido fundamentales para mi formación como profesional. Ellos han inculcado en mí un sentido de seriedad, responsabilidad y rigor académico sin los cuales no podría tener una formación completa como ingeniero.

# Índice de Contenidos

<b>CERTIFICACIÓN DEL ASESOR.....</b>	<b>II</b>
<b>CERTIFICACIÓN DEL AUSPICIANTE.....</b>	<b>III</b>
<b>CESIÓN DE DERECHOS DE AUTOR.....</b>	<b>.....</b>
<b>AUTORIZACIÓN DE USO Y PUBLICACIÓN.....</b>	<b>I</b>
<b>DEDICATORIA.....</b>	<b>II</b>
<b>AGRADECIMIENTO.....</b>	<b>III</b>
<b>CAPITULO I.....</b>	<b>1</b>
<b>1 INTRODUCCIÓN, HERRAMIENTAS Y METODOLOGÍA.....</b>	<b>1</b>
1.1 Antecedentes.....	1
1.2 Problema.....	1
1.3 Objetivos del proyecto.....	1
1.4 Justificación.....	2
1.5 Alcance.....	3
<b>CAPITULO II.....</b>	<b>5</b>
<b>2 MARCO TEÓRICO.....</b>	<b>5</b>
2.1 Base de datos.....	5
2.1.1 Sistema gestor de base de datos.....	5
2.1.2 Modelo cliente – servidor.....	6
2.2 Oracle.....	6
2.3 Oracle 11g Express Edition.....	6
2.3.1 Modelo de desarrollo Oracle Application Builder.....	7
2.3.2 El lenguaje PL/SQL.....	8
2.3.2.1 Arquitectura de PL/SQL.....	8
2.3.3 Biblioteca jQuery y técnica de desarrollo web AJAX.....	9
2.4 Arquitectura de Oracle Application Express.....	10
2.5 Firma Electrónica.....	10
2.5.1 Base legal para la emisión de documentos electrónicos.....	11
2.5.2 Emisión de documentos electrónicos.....	12
2.5.3 Requisitos para la firma del documento electrónico.....	12
2.5.4 Autorización en línea de comprobantes electrónicos.....	12
2.6 Metodología XP.....	13
2.7 Herramienta de desarrollo Oracle Application Express.....	13
2.7.1 ¿Qué se necesita para desarrollar en APEX?.....	14
2.7.2 Primeros pasos con Oracle Application Express.....	14
2.7.3 Servicios de Administración de Application Express.....	14
2.8 Oracle Application Builder.....	14

2.9 La Aplicación.....	15
2.10 Los Componentes Compartidos.....	16
2.10.1 Componente Compartido: Lógica.....	17
2.10.1.1 Procesos de Aplicación.....	17
2.10.1.2 Referencia de Servicios Web.....	18
2.10.1.3 Carga de Datos.....	18
2.10.2 Componente Compartido: Navegación.....	18
2.10.2.1 Separadores.....	18
2.10.2.2 Listas.....	19
2.10.2.3 Rutas de Navegación.....	20
2.10.2.4 Entradas de Barra de Navegación.....	20
2.10.3 Componente Compartido: Seguridad.....	21
2.10.3.1 Esquema de Autenticación.....	21
2.10.3.2 Esquema de Autorización.....	21
2.10.4 Componente Compartido: Interfaz de Usuario.....	23
2.10.4.1 Atributos de Interfaz de Usuario.....	23
2.10.4.2 Temas.....	24
2.10.4.3 Plantillas.....	24
2.10.4.4 Listas de Valores.....	24
2.10.5 Gestión de las páginas en Oracle Application Builder.....	24
2.10.5.1 Páginas de Aplicación de Base de Datos.....	25
2.10.5.2 Páginas de Reportes Interactivos y Reportes Clásicos.....	26
2.10.6 Definición de construcción de la página en Oracle Application Builder.....	26
2.10.6.1 Los Elementos de una página en Oracle Application Builder.....	27
2.10.7 Haciendo referencia a los valores de los elementos de una página en Oracle Application Builder.....	30
2.10.8 Las Regiones en una página en Oracle Application Builder.....	30
2.10.9 Los Botones en una página en Oracle Application Builder.....	31
2.10.10 Las Validaciones en una página en Oracle Application Builder.....	31
2.10.11 Los Cálculos en una página en Oracle Application Builder.....	32
2.10.12 Los Procesos en una página en Oracle Application Builder.....	32
2.10.13 Los Plug-ins en Oracle Application Builder.....	33
2.10.14 Las Acciones Dinámicas en una página en Oracle Application Builder.....	33
2.10.15 El lenguaje JavaScript y AJAX en Oracle Application Builder.....	35
2.10.16 La librería jQuery en Oracle Application Builder.....	37
2.11 El servidor de aplicaciones GlassFish.....	42
2.11.1 Acerca de GlassFish.....	42
2.11.2 GlassFish y Oracle Application Express.....	42
<b>3 ANÁLISIS DEL SISTEMA.....</b>	<b>44</b>
3.1 Metodología.....	44
3.2 Gestión del Proyecto.....	44
3.3 Planificación Inicial.....	44
3.3.1 Iteración 1.....	44
3.3.2 Iteración 2.....	51
3.3.3 Iteración 3.....	53

3.4 Diagramas UML.....	58
3.4.1 Diagramas de Casos de Uso.....	58
3.4.1.1 Módulo de Administración.....	58
3.4.1.2 Módulo de SRI.....	61
3.4.1.3 Módulo de Clientes.....	62
3.4.1.4 Módulo de Ventas.....	64
3.4.2 Diagramas de Actividades.....	65
3.4.2.1 Módulo de Administración.....	66
3.4.2.2 Módulo de SRI.....	69
3.4.2.3 Módulo de Clientes.....	70
3.4.2.4 Módulo de Ventas.....	71
<b>CAPITULO IV.....</b>	<b>72</b>
<b>4 DISEÑO DEL SISTEMA.....</b>	<b>72</b>
4.1 Arquitectura del sistema.....	72
4.2 Diagrama de Navegación.....	73
4.3 Modelo Entidad Relación.....	74
4.4 Modelo Físico.....	75
4.5 Pantallas principales del Prototipo de Facturación Electrónica.....	76
4.6 Sitio web con para distribución del sistema prototipo de facturación electrónica.....	76
<b>CAPITULO V.....</b>	<b>77</b>
<b>5 SITIO WEB.....</b>	<b>77</b>
<b>6 CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>78</b>
6.1 Conclusiones.....	78
6.2 Recomendaciones.....	78
<b>7 ANEXOS.....</b>	<b>80</b>
7.1 Descarga, instalación y configuración de la base de datos Oracle 11G XE.....	80
7.2 Descarga, instalación y configuración de Oracle Application Express 4.....	81
7.3 Descarga, instalación y configuración de Oracle APEX Listener.....	83
7.4 Descarga, instalación y configuración de GlassFish.....	85
<b>8 REFERENCIAS.....</b>	<b>89</b>
<b>9 BIBLIOGRAFIA.....</b>	<b>90</b>

## Índice de Gráficos

Ilustración 1: Diagrama de Componentes del Proyecto.....	4
Ilustración 2: Sistema de Gestión de Base de datos.....	5
Ilustración 3: Arquitectura Cliente - Servidor.....	6
Ilustración 4: Esquema Oracle Application Express.....	7
Ilustración 5: Arquitectura de PL/SQL.....	9
Ilustración 6: Arquitectura de Oracle Application Express.....	10
Ilustración 7: Firma electrónica.....	11
Ilustración 8: Gestionar Espacios de Trabajo.....	14
Ilustración 9: Oracle Application Builder.....	15
Ilustración 10: Página principal del Application Builder.....	15
Ilustración 11: Herramientas y propiedades de la aplicación.....	16
Ilustración 12: Pantalla Principal de Componentes Compartidos.....	17
Ilustración 13: Carga de Datos.....	18
Ilustración 14: Gestión de Separadores.....	19
Ilustración 15: Vista del separador en el navegador web.....	19
Ilustración 16: Lista mostrada en el navegador web.....	20
Ilustración 17: Ruta de navegación.....	20
Ilustración 18: Entrada de Barra de Navegación.....	20
Ilustración 19: Ejemplo Esquema de Autorización.....	21
Ilustración 20: Tipo de Esquema de Autorización.....	22
Ilustración 21: Opciones para crear una página en Application Builder.....	25
Ilustración 22: Reporte Interactivo.....	26
Ilustración 23: Definición de Construcción de la Página.....	27
Ilustración 24: Elementos en una vista tipo árbol.....	27
Ilustración 25: Elementos en una vista tipo componente.....	28
Ilustración 26: Ejemplo de una pantalla con tres regiones.....	31
Ilustración 27: Pantalla de ejemplo que funciona con AJAX.....	35
Ilustración 28: Propiedades del proceso PL/SQL a ser llamado por AJAX.....	36
Ilustración 29: Propiedades de la Acción Dinámica que ejecuta el código JavaScript.....	37
Ilustración 30: Pantalla de ejemplo que funciona con jQuery y AJAX.....	38
Ilustración 31: Propiedades del proceso PL/SQL a ser llamado por AJAX.....	39
Ilustración 32: Propiedades de la Acción Dinámica que ejecuta el código JavaScript.....	41
Ilustración 33: Configuración standard de Oracle Application Express.....	43
Ilustración 34: Módulo de Administración.....	59
Ilustración 35: Administración de Usuarios.....	60
Ilustración 36: Creación, modificación y eliminación de módulos.....	60
Ilustración 37: Administración de accesos a usuarios.....	61
Ilustración 38: Parámetros de documentos autorizados.....	62
Ilustración 39: Módulo de Clientes.....	63
Ilustración 40: Módulo de Ventas.....	64
Ilustración 41: Generación de la firma electrónica en el documento.....	65
Ilustración 42: Diagrama de Actividades: Administrar Módulos.....	66
Ilustración 43: Diagrama de Actividades: Administrar Usuarios.....	67
Ilustración 44: Diagrama de Actividades: Administrar Accesos.....	68
Ilustración 45: Diagrama de Actividades: Parámetros de Documentos Autorizados.....	69
Ilustración 46: Diagrama de Actividades: Administrar Clientes.....	70

Ilustración 47: Grabar documento de venta.....	71
Ilustración 48: Generar archivo XML de documento de venta, firmar y enviar a verificar..	71
Ilustración 49: Arquitectura del Sistema.....	72
Ilustración 50: Diagrama de Navegación.....	73
Ilustración 51: Modelo Entidad – Relación.....	74
Ilustración 52: Modelo Físico.....	75
Ilustración 53: Pantalla para otorgar acceso a los usuarios.....	76
Ilustración 54: Validación de elementos en pantalla de grabar documento de venta.....	76
Ilustración 55: Comando rpm -Uvh para instalación de Oracle XE.....	80
Ilustración 56: Configuración de .profile con editor texto nano.....	81
Ilustración 57: Contenido de carpeta apex.....	82
Ilustración 58: Instalación del idioma español en APEX.....	83
Ilustración 59: Instalación de Java 7.....	84
Ilustración 60: Configuración de entorno Java.....	84
Ilustración 61: Apex Listener iniciado.....	85
Ilustración 62: Inicio de instalación de GlassFish.....	86
Ilustración 63: Parámetros de configuración de GlassFish.....	87
Ilustración 64: Desplegar archivo apex.war.....	88
Ilustración 65: Desplegar archivo i.war.....	88
Ilustración 66: Aplicaciones desplegadas en GlassFish.....	88

## Índice de tablas

Tabla 1: Entidades de certificación.....	12
Tabla 2: Ejemplo de código fuente de seguridad en PL/SQL.....	23
Tabla 3: Tipos de referencias a los elementos de la aplicación.....	30
Tabla 4: Métodos de validación de una aplicación.....	32
Tabla 5: Tipos de procesos de una aplicación.....	32
Tabla 6: Ejemplo PL/SQL para ser llamado desde una función AJAX.....	36
Tabla 7: Ejemplo de código JavaScript que implementa AJAX.....	37
Tabla 8: Ejemplo PL/SQL para ser llamado desde una función AJAX.....	40
Tabla 9: Ejemplo de código JavaScript que implementa AJAX con sintaxis jQuery.....	42
Tabla 10: Administración de Usuarios.....	45
Tabla 11: Administración de Módulos del Sistema.....	45
Tabla 12: Administración de Accesos al Sistema.....	46
Tabla 13: Gestión de Clientes.....	46
Tabla 14: Administración de Información de Documentos Autorizados por el SRI.....	47
Tabla 15: Creación de Factura de Servicios.....	48
Tabla 16: Eliminación y Anulación de Facturas.....	48
Tabla 17: Maestro de Facturas.....	49
Tabla 18: Mantenimiento de Lista de Precios.....	49
Tabla 19: Pantalla de Formas de Pago y Tipos de Formas de Pago.....	50
Tabla 20: Reportes de Ventas y Cierres de Caja.....	50
Tabla 21: Diseño de diagramas UML.....	51
Tabla 22: Diseño de la base de datos.....	52
Tabla 23: Añadir nueva tabla a la base de datos.....	52
Tabla 24: Añadir nueva tabla a la base de datos.....	53
Tabla 25: Creación de los menús de navegación.....	54
Tabla 26: Diseño de la interfaz.....	54
Tabla 27: Desarrollo del módulo de seguridad y administración.....	55
Tabla 28: Desarrollo del módulo de clientes.....	56
Tabla 29: Desarrollo del módulo de SRI.....	56
Tabla 30: Desarrollo del módulo de ventas.....	57
Tabla 31: Desarrollo de la generación del documento XML de venta.....	58
Tabla 32: Descripción caso de uso del módulo de administración.....	59
Tabla 33: Descripción caso de uso de administración de usuarios.....	60
Tabla 34: Descripción caso de uso de creación, modificación y eliminación de módulos..	61
Tabla 35: Descripción caso de uso de administración de accesos a usuarios.....	61
Tabla 36: Descripción de caso de uso parámetros de documentos autorizados.....	62
Tabla 37: Descripción caso de uso del módulo de clientes.....	63
Tabla 38: Descripción de caso de uso del módulo de ventas.....	64
Tabla 39: Descripción de caso de uso de generación de la firma electrónica en el documento de venta.....	65

# RESUMEN

Oracle Application Express funciona con la base de datos Oracle, es una herramienta de desarrollo rápido de aplicaciones web. En este documento comparte el conocimiento acerca de Oracle Application Builder que es parte de Oracle Application Express. Esta herramienta permite la fácil creación de pantallas con un mínimo conocimiento de lenguajes HTML, JavaScript o CSS que son necesarios para el desarrollo de pantallas web. El lenguaje de programación PL/SQL es obligatorio, es un lenguaje que combina el lenguaje de consulta SQL para la manipulación de los datos y la facilidad del procesamiento de los datos.

Se explica los componentes básicos que se necesita en la creación de una aplicación web, la mayoría de estos son creados a partir de asistentes, también se explica las tecnologías avanzadas de desarrollo web, AJAX y jQuery. La seguridad en una aplicación es importante, la herramienta proporciona esquemas de seguridad, fáciles de crear e implementar.

La forma de desarrollar aplicaciones web es diferente a otras herramientas. Al principio si está acostumbrado a trabajar con otros frameworks, le parecerá que no tiene todo lo que se necesita. Oracle Application Express tiene una metodología de desarrollo declarativa un poco difícil al principio, pero fácil de superar con el tiempo. Es una poderosa herramienta, si se tiene en cuenta que para desarrollar solo se necesita la base de datos Oracle.

## SUMMARY

Oracle Application Express works with the Oracle database, is a rapid web application development tool. This document shares the knowledge about Oracle Application Builder is part of Oracle Application Express. This tool allows the easy creation of web screens with minimal knowledge of HTML, JavaScript or CSS languages that are necessary for the development of web screens. The programming language PL/SQL is required, it is a language that combines the SQL query language for data manipulation and ease of data processing.

The basic components needed to create a web application is explained, most are created from wizards, advanced web development technologies, AJAX and jQuery is also explained. Safety is important in a web application, the tool provides security schemes, easy to create and implement.

The way to develop web applications is different from other tools. At first, if you are used to working with other frameworks, it will seem that has no everything that requires. Oracle Application Express has a declarative development methodology a little difficult at first, but easy to overcome with time. It is a powerful tool, if you consider that to develop only the Oracle database is needed.

# CAPITULO I

## 1 Introducción, Herramientas y Metodología

### 1.1 Antecedentes

Oracle Forms 6i es utilizado en el desarrollo de aplicaciones empresariales por la empresa Gestor Asesoría Empresarial a sido la herramienta principal, debido a que cuenta con ventajas tales como el lenguaje de programación de cuarta generación, la facilidad de interacción con la base de datos y la integración con herramientas que no necesitan de costo de licencias como es la base de datos Oracle Express Edition y tecnologías como Java. Pero con el tiempo ha surgido la necesidad de llegar a un medio necesario en este tiempo, como es el acceso a las aplicaciones informáticas por medio del internet, donde Oracle Forms 6i ya no es de utilidad.

Lo que orienta esta investigación es estudiar la tecnología Oracle Application Express y desarrollar un prototipo de software libre al que se pueda acceder por medio del internet, además de contar con ventajas nombradas.

### 1.2 Problema

En la provincia de Imbabura se tiene numerosas empresas entre medianas y pequeñas que utilizan en forma parcial o total, software para automatizar los procesos de facturación, contabilidad, inventario o por el contrario no cuentan con ninguna herramienta más allá de las hojas de cálculo.

En el aspecto tecnológico se puede decir que, el software actualmente empleado por estas empresas tiene problemas de fragmentación de la información, por ejemplo sistemas donde la información se guarda en archivos separados o se utiliza hojas de cálculo, también se tiene que el software utilizado es dependiente de la plataforma o incluso de la versión del sistema operativo, además no sirve para la web, el manejo de la seguridad puede causar que los sistemas sean vulnerables, se puede citar también el hecho de que la información esté dispersa en diferentes computadoras y las copias de seguridad inconsistentes.

Se pretende resolver todo lo expuesto a través de la herramienta Oracle Application Express, ya que encaja bien con el propósito de tener una herramienta de gestión sencilla, el desarrollo y el despliegue de la aplicación se lo hará a través del navegador web, no depende de componentes por parte del cliente y la facilidad con la que se programa en esta herramienta es semejante con los Oracle Forms.

### 1.3 Objetivos del proyecto

- **Objetivo General :**
  - Investigar la tecnología Oracle Application Express 4 para la creación de sistemas empresariales como software libre, que ayude a las empresas y personas que tienen una actividad económica.

- **Objetivos Específicos :**

- Estudiar la herramienta Oracle Application Builder para desarrollo de las pantallas.
- Estudiar el lenguaje de programación PL/SQL utilizado en el desarrollo de aplicaciones en Oracle Application Express 4.
- Estudiar la biblioteca JQuery y técnica de desarrollo web Ajax aplicado a Oracle Application Express.
- Desarrollar el prototipo sistema de facturación electrónica con la metodología de desarrollo xtreme programming.
- Configurar un entorno de producción standard con el servidor de aplicaciones GlassFish para el prototipo.
- Implementar la firma digital en la emisión de facturas electrónicas.
- Crear un sitio web con el CMS Drupal, donde se pueda acceder al fuente del prototipo para su descarga y distribución con licencia GPL.

#### **1.4 Justificación**

Oracle es una de las grandes empresas de software de base de datos, actualmente se pueden descargar sus herramientas Oracle 11g XE y Oracle Application Express para estudiarlas y desarrollar aplicaciones 100% web, sin la necesidad de pago de licencias, además es una tecnología que a estado en desarrollo desde el año 2004, hoy en día Oracle recomienda migrar de Oracle Forms a Oracle Application Express, significa que esta herramienta se considerada capaz de sustituir a los Forms que han estado en desarrollo desde los años 80, anterior al nacimiento del principio cliente - servidor. Forms ejecuta un applet Java que es llamado desde el navegador web, no es HTML, los Forms no puede mantener el paso con aplicaciones web modernas.

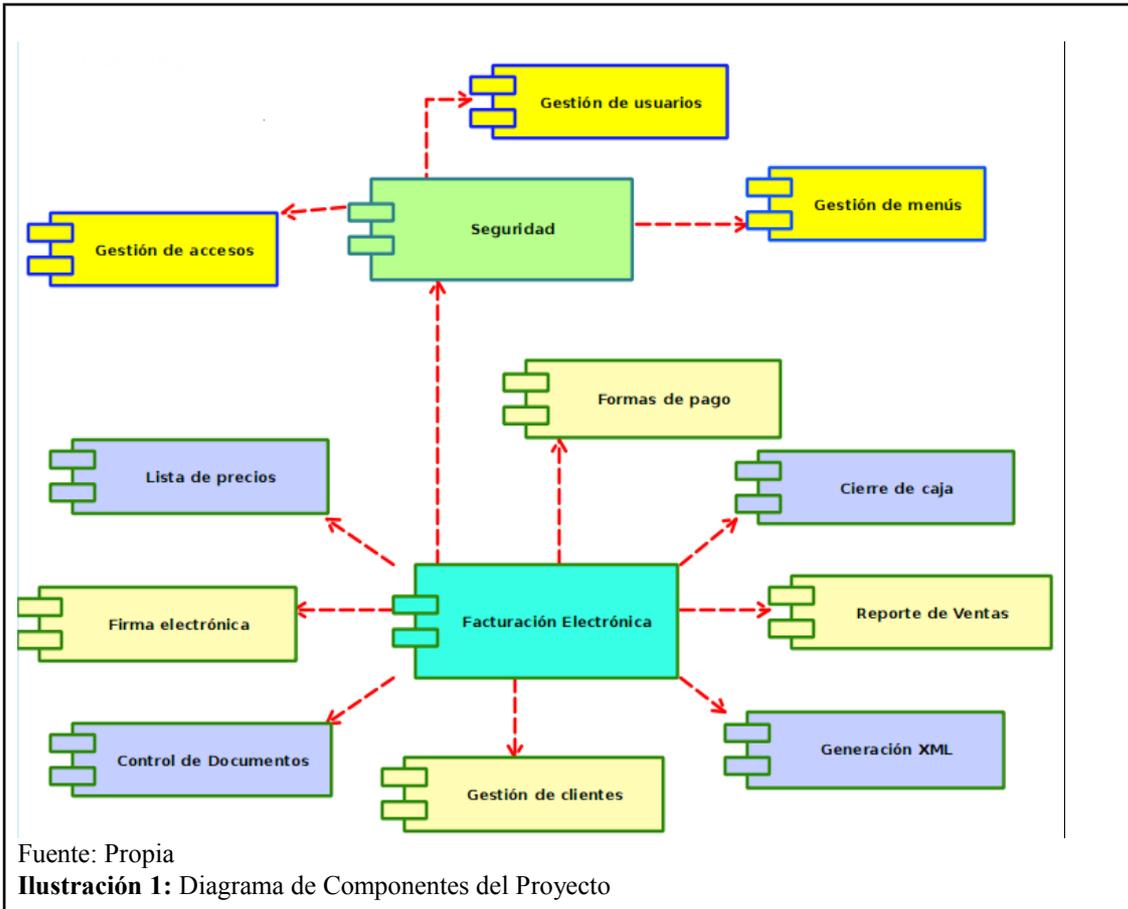
El aprendizaje de nuevas tecnologías informáticas para el desarrollo de aplicaciones que se puedan ejecutar en un gran número de dispositivos, sean móviles como los de hoy en día y los que vendrán, y que todavía falta por explotar, o los de sobremesa que utilizamos en lugares de trabajo y hogar. Esta tecnología permite que las aplicaciones se ejecuten en cualquier arquitectura gobernada con un sistema operativo con un navegador web sin la utilización de plugins, no como es el caso de los conocidos Oracle Forms, los cuales tiene esa desventaja de no ser 100% web y además el alto costo de licencias.

En la carrera de Ingeniería en Sistemas Computacionales se ha realizado aplicaciones similares a la mencionada en esta propuesta, las aportaciones más significativas serán el aprendizaje en una tecnología nueva que puede servir para que en las materias de la carrera que tienen relación con el desarrollo web se incluya también al Oracle Application Express como alternativa. La facturación electrónica es una opción que en el Ecuador existe desde hace un par de años, y no se había hecho un prototipo o una aplicación real. Por último el aporte más significativo es que el prototipo se una con otras posibles tesis de grado dentro o fuera de la universidad, o aportes independientes comentado anteriormente, se tratará de hacer posible licenciando el aplicativo con una licencia de software libre y publicando la herramienta en un repositorio o sitio web.

## 1.5 Alcance

Se estudiará las funcionalidades de desarrollo de Oracle Application Express 4 como son: el editor de pantallas, los reportes, las validaciones, la autenticación, el administrador de sesiones, las plantillas o templates, charts o gráficos estadísticos, los procesos condicionales, navegación de pantallas. Estas funcionalidades son las que ayudaran en la construcción del prototipo de facturación electrónica. A continuación se detalla lo que contendrá el prototipo.

- Facturación electrónica.
  - Emisión de facturas.
  - Eliminación y anulación de facturas.
  - Emisión de notas de crédito.
  - Eliminación y anulación de notas de crédito.
  - Maestro de facturas.
  - Maestro de notas de crédito.
  - Mantenimiento de lista de precios.
  - Parámetros de las formas de pago y tipos de formas de pago.
  - Parámetros de impuestos.
  - Parámetros de números de series, números de autorizaciones y secuenciales, que el Servicio de Rentas Internas emite al contribuyente.
  - Maestro de clientes.
- Administración de usuarios y menús.
  - Creación y eliminación de usuario.
  - Gestión de acceso a las diferentes pantallas.
  - Creación y eliminación de módulos y menús.
- Reportes
  - Reportes de cierres de caja.
  - Reportes de ventas (texto y gráfico estadístico).
- Distribución del software como software libre.
  - Calificarlo con una licencia GPL.
  - Distribuirlo libremente a través de un sitio web y repositorio de control de versiones.



# CAPITULO II

## 2 Marco Teórico

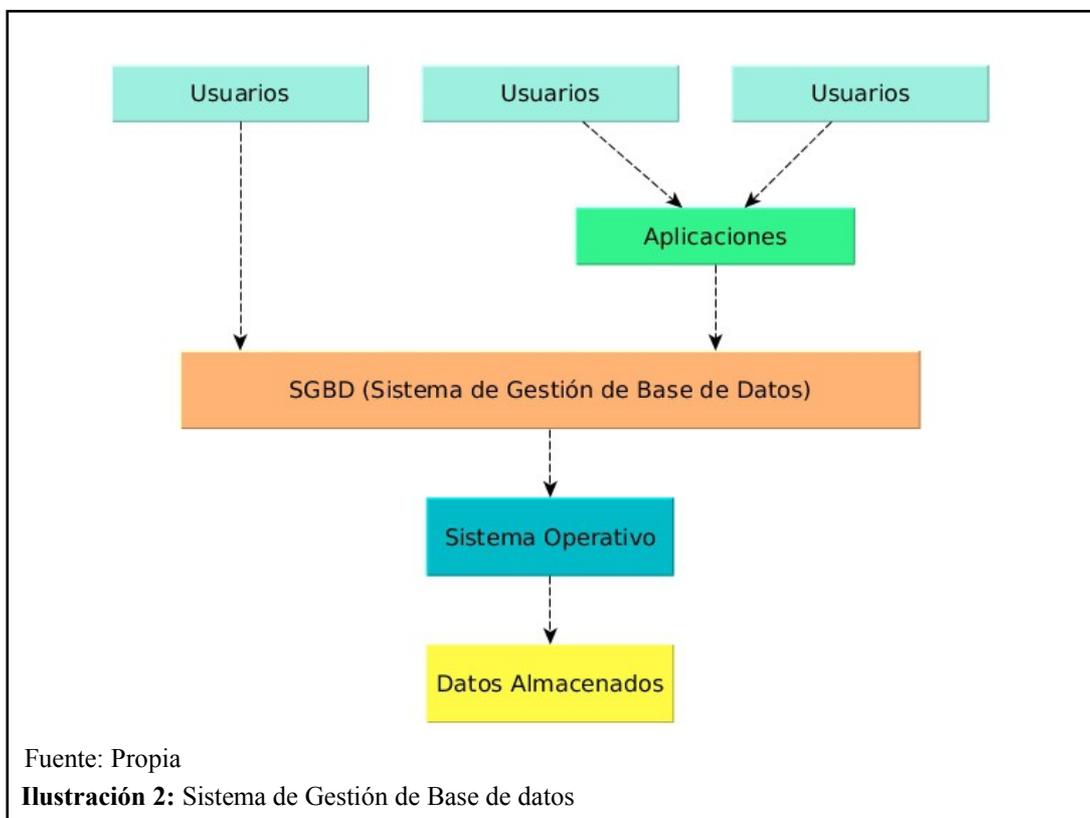
### 2.1 Base de datos

#### 2.1.1 Sistema gestor de base de datos

Un Sistema de Gestión de Bases de Datos es un software informático que permiten el almacenamiento, manipulación y control de la información. Los SGBD<sup>1</sup> permiten:

- Definir los datos a los distintos niveles de abstracción; físico, lógico y externo.
- Manipula los datos en la base de datos permitiendo; insertar, modificar, borrar y consultar los datos.
- Mantiene la integridad.
- Controla la seguridad en la base de datos.

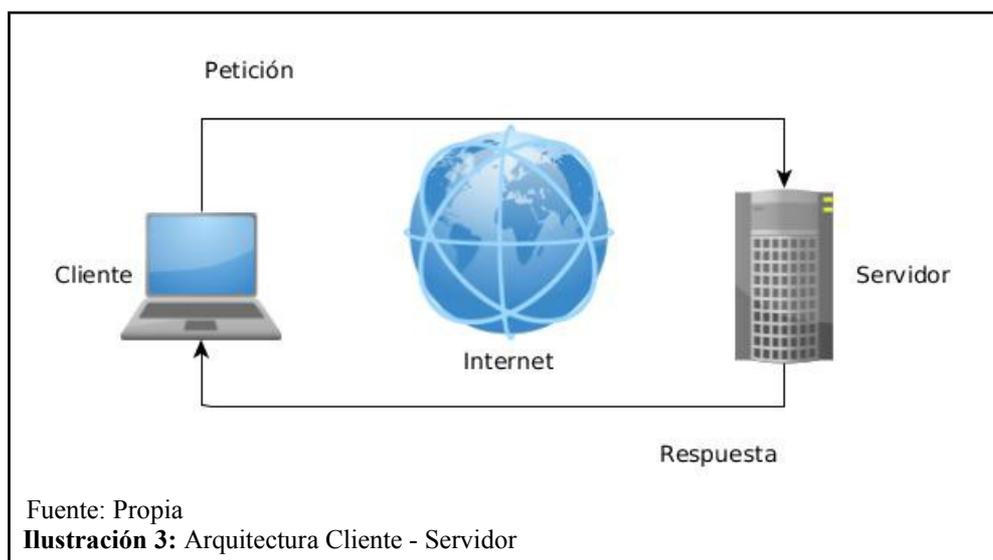
En la **Ilustración 2** se muestran a los usuarios que pueden acceder directamente al SGBD con cualquier lenguaje de manipulación de datos, o mediante aplicaciones que acceden a las base de datos, el SGBD funciona sobre el sistema operativo el cual accede a la información que se guarda en un medio de almacenamiento, puede ser el disco duro. (Cabello, 2010)



<sup>1</sup> Sistema de gestión de bases de datos

### 2.1.2 Modelo cliente – servidor

El modelo cliente - servidor es un modelo de sistema en el que los servicios que se encuentran en servidores son demandados por unos clientes. Un cliente realiza peticiones a otro programa, el servidor, es quien le da respuesta. En la **Ilustración 3** se representa al cliente que hace una petición desde el navegador web o una aplicación que se está ejecutando en el servidor, estas peticiones y respuestas forman un ciclo repetitivo mientras exista una conexión entre el cliente y servidor. (Sommerville & Galipienso, 2005)



## 2.2 Oracle

En el sitio web de Oracle se encuentra que la empresa se refiere a si misma como líder en software de base de datos que sigue desarrollando aún más las tecnologías que son adquiridas por las mejores empresas en su clase y posee un ampliado conjunto de tecnologías, de servidores y almacenamiento, base de datos y middleware.

Con más de 380.000 clientes y con despliegues en una amplia variedad de industrias en más de 145 países de todo el mundo. Oracle ofrece una amplio, optimizado y completamente integrado, sistema de hardware y software de negocios ([www.oracle.com](http://www.oracle.com)).

## 2.3 Oracle 11g Express Edition

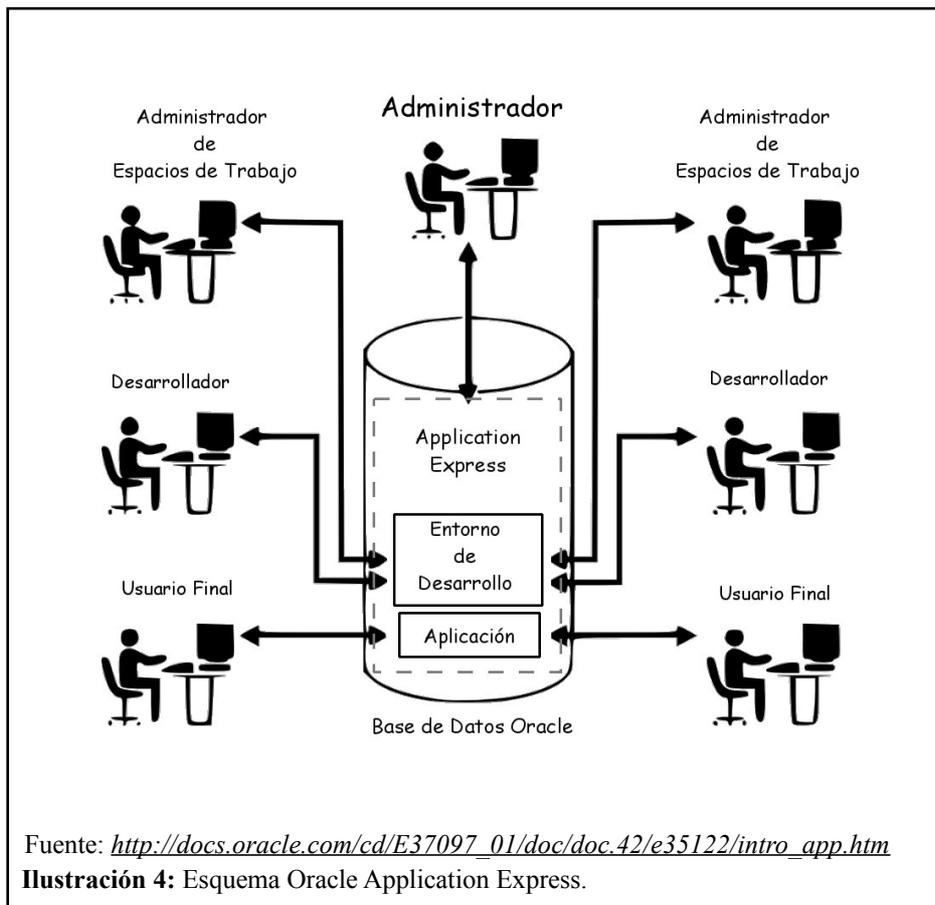
Oracle Database 11g Express Edition fue publicada en septiembre del 2011. Esta versión es gratuita para desarrollar, implementar y distribuir, puede ser instalado en cualquier computador con cualquier número de CPUs<sup>2</sup> pero con una sola instancia de base de datos, la base de datos solo almacenará hasta 11 GB de datos de usuario, puede utilizar hasta 1 GB de memoria, y utilizar una CPU o núcleo físico por computador, está disponible para las plataformas Windows y Linux. (Oracle Application Express)

Se considera a Oracle Database como uno de los sistemas de bases de datos más completos, destacando el soporte de transacciones, la estabilidad, escalabilidad y el soporte multiplataforma.

<sup>2</sup> Unidad Central de Procesamiento

### 2.3.1 Modelo de desarrollo Oracle Application Builder

Oracle Application Express es una herramienta de desarrollo rápido de aplicaciones web para la base de datos Oracle. Utilizando sólo el navegador web y poca experiencia en programación, se puede desarrollar aplicaciones profesionales que son a la vez rápidas y seguras. Gracias a sus características tales como; temas de interfaces de usuario, controles de navegación, controladores de formulario, generación de informes flexibles. El desarrollo es más fácil y dinámico. Desde la perspectiva del usuario final, las aplicaciones implementadas sólo necesitan un navegador y el acceso a una base de datos Oracle que se ejecuta en Oracle Application Express. En la **Ilustración 4** se representa al Oracle Application Express instalado en un servidor con la base de datos Oracle y como pueden acceder los administradores, desarrolladores o usuarios finales de las aplicaciones.



El Oracle Application Builder es un componente más de Oracle Application Express, este componente es específicamente para el desarrollo de los formularios, reportes, mapas, menús, listas, y demás objetos utilizados en una aplicación web, esto puede ser por medio del uso de asistentes o sin ellos, en su mayoría a partir de los objetos de la base de datos. (*Introducing Oracle Application Express*)

### 2.3.2 El lenguaje PL/SQL

PL/SQL<sup>3</sup> es un lenguaje de programación incorporado en las base de datos Oracle que soportara todas las consultas, ya que la manipulación de datos que se usa es la misma que en SQL<sup>4</sup>, incluyendo nuevas características:

- El manejo de variables.
- Estructuras modulares.
- Estructuras de control de flujo y toma de decisiones.
- Control de excepciones.

El lenguaje PL/SQL está incorporado en:

- Servidor de la base de datos.
- Herramientas de Oracle Forms<sup>5</sup>, Oracle Reports<sup>6</sup>, etc.

En un entorno de base de datos los programadores pueden construir bloques PL/SQL para utilizarlos como procedimientos o funciones, o bien pueden escribir estos bloques como parte de scripts SQL\*Plus.

Los programas o paquetes de PL/SQL se pueden almacenar en la base de datos como otro objeto, y todos los usuarios que estén autorizados tienen acceso a estos paquetes. Los programas se ejecutan en el servidor para ahorrar recursos a los clientes.

Este lenguaje es utilizado en Oracle Application Express cuando el navegador web envía una petición a una URL<sup>7</sup> que se traduce en una llamada a una instrucción PL/SQL apropiada, después de que la base de datos procesa el PL/SQL, los resultados se transmiten de vuelta al navegador como HTML<sup>8</sup>, este ciclo ocurre cada vez que se hace una solicitud en el navegador web o cuando se despliegue una página web. (*Oracle PL/SQL Programming, 2014*)

#### 2.3.2.1 Arquitectura de PL/SQL

Para la ejecución de programas PL/SQL existe la “PL/SQL runtime engine” en español motor de tiempo de ejecución, es conocida como la máquina virtual de PL/SQL (PVM). Esta maquina virtual de PL/SQL ejecuta el código o programa y lo traduce a código de máquina y devuelve los resultados desde donde fue ejecutado. La PVM esta escrito en lenguaje de programación C.

En la mayor parte el código PL/SQL se ejecuta en el lado del servidor incluso cuando la PVM se ha incluido del lado del cliente como los Oracle Forms, el motor de ejecución abre una sesión en la base de datos remota. Cada sesión es una entidad lógica en la memoria de la instancia de la base de datos que representa el estado del usuario que ha iniciado sesión en esa base de datos. Las sesiones conectadas a una base de datos son visibles a través de la vista V\$SESSION.

En la **Ilustración 5** se muestra la tarea que se realiza al ejecutar un bloque de programa en PL/SQL desde la aplicación SQL\*Plus, este envío de ejecución se produce sobre cualquier conexión de sesión que se haya establecido por ejemplo de red o de comunicación entre procesos. (*Feuerstein & Pribyl, 2014*)

---

<sup>3</sup> Lenguaje de procedimiento / Lenguaje de consulta estructurado.

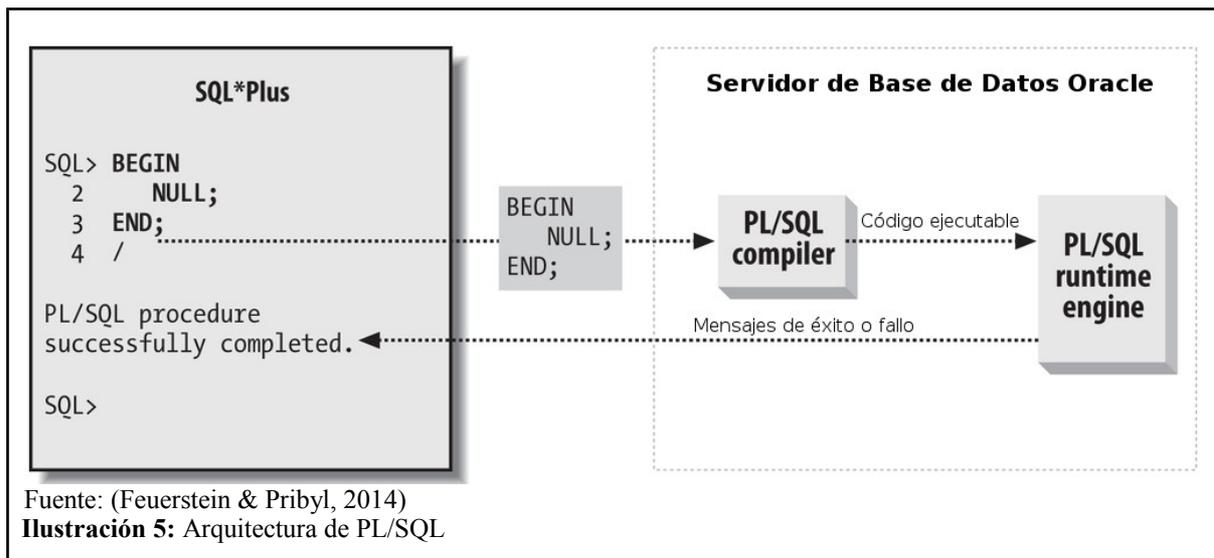
<sup>4</sup> Lenguaje de consulta estructurado.

<sup>5</sup> Herramienta de desarrollo de pantallas que pueden desplegarse en el escritorio o en el navegador.

<sup>6</sup> Herramienta de desarrollo de reportes que pueden desplegarse en el escritorio o en el navegador.

<sup>7</sup> Es una secuencia de caracteres, de acuerdo a un formato modélico y estándar, que se usa para nombrar recursos en Internet.

<sup>8</sup> Lenguaje de marcado de hipertexto.



El programa se dirige al compilador de PL/SQL (PL/SQL compiler) que intenta convertir este bloque de programa a lenguaje de máquina. Se comprueba la sintaxis para asegurar que el programa cumple con la gramática del lenguaje. Si la compilación tiene éxito, la base de datos pone en forma compilada el programa en un área de memoria compartida; si falla, el compilador devolverá los mensajes de error a la sesión de SQL \* Plus.

Al final el PVM (PL/SQL runtime engine) interpreta el lenguaje de máquina y finalmente vuelve un éxito o un código de error para la sesión de SQL \* Plus.

### 2.3.3 Biblioteca jQuery y técnica de desarrollo web AJAX

jQuery es una biblioteca escrita en JavaScript<sup>9</sup>, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM<sup>10</sup>, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX<sup>11</sup> a páginas web. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio. (*jquery, 2014*)

AJAX es un término que describe un nuevo modo de utilizar conjuntamente varias tecnologías existentes para crear aplicaciones interactivas. Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargar las páginas, mejorando la interoperatividad, velocidad y facilidad de uso en las aplicaciones. (*Mozilla Developer Network, 2012*) (*Eguiluz*)

Oracle Application Express proporciona acciones dinámicas como una manera de definir el comportamiento de la aplicación del lado del cliente sin la necesidad de conocimientos de Javascript o AJAX. En lo que se refiere a JQuery, Oracle Application Express sólo carga los componentes necesarios para la funcionalidad básica, sin embargo si es necesario hacer el uso de AJAX o JQuery la herramienta da la posibilidad de hacerlo.

<sup>9</sup> Lenguaje de programación interpretado que se ejecuta del lado del cliente, implementado como parte de un navegador web.

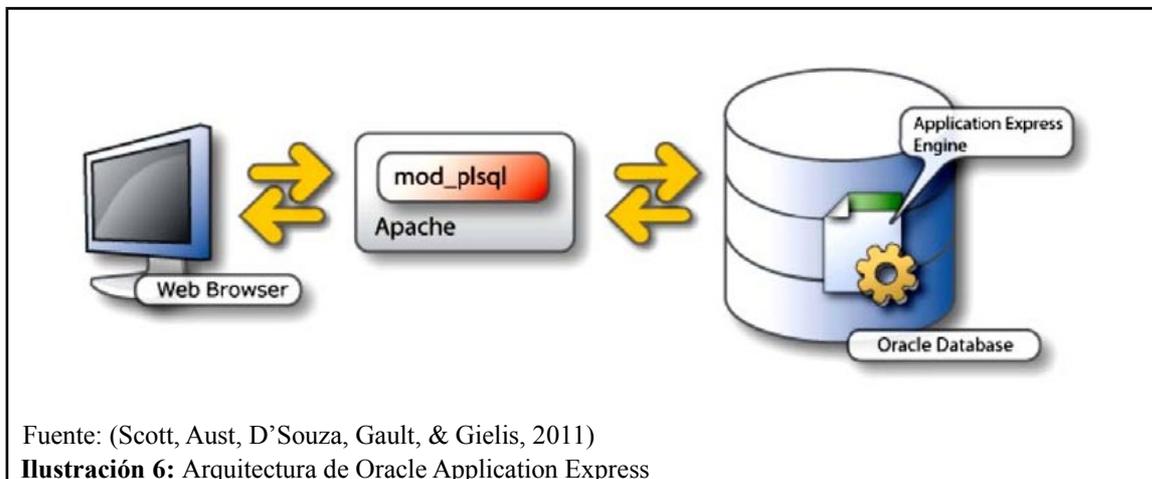
<sup>10</sup> Modelo en objetos para la representación de documentos HTML y XML.

<sup>11</sup> JavaScript asíncrono y XML.

## 2.4 Arquitectura de Oracle Application Express

Se mencionó en temas anteriores todo lo que Oracle Application Express tiene para ser una herramienta de desarrollo rápido de aplicaciones web, con un lenguaje de programación declarativo, lo que hace que esta herramienta sea fácil de usar en el desarrollo y producción de las aplicaciones. Se trata de un soporte completo con todas las ediciones de la base de datos Oracle.

La arquitectura de Oracle Application Express es muy simple y sin embargo una arquitectura de gran alcance. Se llama Oracle HTTP Server como se representa en la **Ilustración 6** al conjunto de, servidor web que se encuentra entre el navegador web y la base de datos, es responsable de manejar las peticiones desde el navegador web pasándolos a la base de datos, a continuación, Oracle Application Express procesa la solicitud y genera la respuesta en código HTML para enviar de vuelta al navegador web, se utiliza para la **Ilustración 6** el `mod_plsql`, es una extensión para Apache Oracle HTTP Server que no se utilizara en este desarrollo, en su lugar se lo hará con el servidor Glassfish, que se explica más adelante. (Scott, Aust, D'Souza, Gault, & Gielis, 2011)



## 2.5 Firma Electrónica

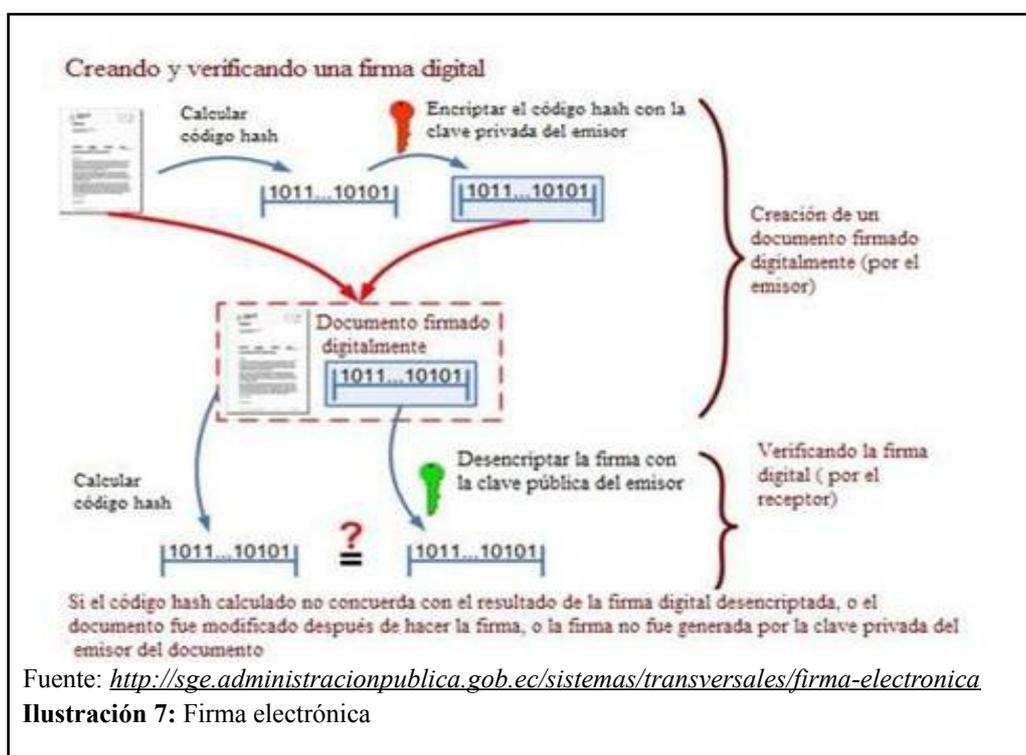
La firma electrónica es la equivalencia digital de la firma manuscrita, tiene la misma validez legal y se encuentra amparada por la Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos.

Desde el punto de vista técnico, la firma es un conjunto de datos digitales que se añaden a un archivo digital y que se obtienen del cifrado del mismo mediante programas de computadora.

La firma digital permite la transacción segura de documentos y operaciones en aplicaciones de computador garantizando los siguientes aspectos:

- Identidad, reconoce unívoco a un emisor como autor del mensaje.
- Integridad, el documento no puede ser alterado de forma alguna durante la transmisión.
- No repudio, el emisor no puede negar en ningún caso que un documento no fue firmado.
- Confidencialidad, solo el emisor y receptor puedan leer el documento si fuera el caso.

Con la firma electrónica se puede realizar diferentes tipos de transacciones a través de la Internet sin necesidad de desplazarse, ni hacer filas de forma que los trámites públicos se agilizan aumentando la transparencia, lo que se traduce en ahorros significativos de tiempo y dinero. Las aplicaciones de la firma digital son diversas, como pueden ser, las compras públicas, trámites ciudadanos, la gestión documental, operaciones bancarias, pago electrónico, trámites judiciales y notariales, comercio electrónico y el principal en esta investigación, la facturación electrónica. En la **Ilustración 7** se representa el proceso de como un documento se firma y posteriormente se verifica, para comenzar es necesario calcular el código hash del archivo a firmar y de la firma, con ambos datos se obtiene un documento firmado, para su verificación se vuelve a obtener el código hash del documento y debe coincidir con el archivo antes de ser firmado, sino es así el archivo pudo ser manipulado o modificado este proceso se realiza con un emisor que es el que firma y el receptor que es el que obtiene el archivo y verifica. *(Subsecretaria de Tecnologías de la Información)*



**Ilustración 7:** Firma electrónica

### 2.5.1 Base legal para la emisión de documentos electrónicos

La emisión de documentos electrónicos autorizados por el Servicio de Rentas Internas se basa en la Disposición General Décima del Reglamento de Comprobantes de Venta, Retención y Documentos Complementarios inciso tercero del número 3 del artículo 6.

Se basa también en la Ley de Comercio Electrónico, Firmas y Mensajes de Datos publicado en el Suplemento del Registro Oficial No. 557 del 17 de abril de 2002.

En el Reglamento a la Ley de Comercio Electrónico, Firmas y Mensajes de Datos, que fue publicado en el Registro Oficial No. 735 del 31 de diciembre de 2002.

En el Decreto No. 181 publicado en el Registro Oficial No. 553 del 11 de octubre del 2011, en el que norma la numeración de identificadores de campo y campos mínimos de los tipos de certificados.

En la Resolución No. NAC-DGERCGC12-00105 del 9 de marzo de 2012, publicada en Registro Oficial No. 666 de 21 de marzo de 2012.

En la Resolución No. NAC-DGERCGC13-00236 del 6 de mayo de 2013, publicada mediante Registro Oficial 956 del 9 de mayo de 2013, en la cual se establece el calendario de obligatoriedad para ingresar al esquema de facturación electrónica.

### 2.5.2 Emisión de documentos electrónicos

Para la emisión de comprobantes electrónicos la persona que posee el RUC<sup>12</sup> y tiene una actividad económica conocido como contribuyente debe solicitar al Servicio de Rentas Internas a través del portal web la autorización para emitir estos comprobantes. El contribuyente que se incorpore a esta modalidad debe obtener el certificado digital de firma electrónica que puede adquirirlo en una entidad de certificación autorizada. En la **Tabla 1** se muestran las entidades de certificación.

Entidad de certificación	Portal web
Banco Central del Ecuador	www.eci.bce.ec
Security Data	www.securitydata.net.ec
ANF	www.anf.ec

**Tabla 1:** Entidades de certificación

Para el desarrollo del prototipo de facturación electrónica se obtuvo el certificado digital para la firma electrónica en el Banco Central del Ecuador. La solicitud de autorización para la emisión de documentos electrónicos se la hizo a un ambiente de pruebas que es obligatorio para los solicitantes ya que el Servicio de Rentas Internas exige que cualquier sistema cumpla con los requisitos y pase las pruebas de verificación y calidad antes de trabajar con un ambiente de producción.

### 2.5.3 Requisitos para la firma del documento electrónico

Los contribuyentes tendrán que generar sus comprobantes electrónicos en formato XML<sup>13</sup>, este archivo contiene lo definido en los esquemas XSD<sup>14</sup> disponibles en el Portal web del Servicio de Rentas Internas.

Los documentos electrónicos emitidos deben estar firmados de forma obligada bajo el estándar de firma digital XadES\_BES.

### 2.5.4 Autorización en línea de comprobantes electrónicos

El Servicio de Rentas Internas a puesto a disposición canales WEB Services seguros con protocolos de seguridad y certificados SSL para la autorización en línea de los comprobantes electrónicos, son dos canales WEB Services:

1. Para ambiente de pruebas;

<https://celcer.sri.gob.ec/comprobantes-electronicos-ws/RecepcionComprobantes?wsdl>

<sup>12</sup> Registro Único de Contribuyentes, cuya función es registrar e identificar a los contribuyentes con fines impositivos y proporcionar información a la Administración Tributaria.

<sup>13</sup> Lenguaje de Marcas Extensible.

<sup>14</sup> Lenguaje de esquema para describir la estructura y las restricciones de los contenidos de los documentos XML.

<https://celcer.sri.gob.ec/comprobantes-electronicos-ws/AutorizacionComprobantes?wsdl>

2. Para ambiente de producción;

<https://cel.sri.gob.ec/comprobantes-electronicos-ws/RecepcionComprobantes?wsdl>

<https://cel.sri.gob.ec/comprobantes-electronicos-ws/AutorizacionComprobantes?wsdl>

## 2.6 Metodología XP

La metodología XP es una forma ligera, eficiente y de bajo riesgo, flexible, predecible, científico y divertido desarrollo de software. Se distingue de otras metodologías por:

- Su pronta, concreta, y continua retro alimentación de los ciclos cortos.
- Su enfoque de planificación gradual, que viene rápidamente con un plan general que se espera que evolucione a través de la vida del proyecto.
- Su capacidad de programar con flexibilidad las aplicaciones funcionales, en respuesta a las cambiantes las necesidades del negocio.
- Su confianza de las pruebas automatizadas escritas por los programadores y clientes para controlar el progreso del desarrollo, para permitir que el sistema evolucione, y para atrapar defectos tempranamente.
- Su confianza de un proceso de diseño evolutivo que dura mientras el sistema permanece.
- Su confianza de la estrecha colaboración de los programadores con conocimientos ordinarios.
- Su confianza de las prácticas que funcionan tanto con los sentidos de corto plazo de los programadores y los intereses a largo plazo del proyecto.

XP es una disciplina de desarrollo de software. Es una disciplina porque hay ciertas cosas que se tiene que hacer para estar haciendo XP. No puede elegir si se va a escribir las pruebas, si no, no es extrema. (*Beck, 2000*)

XP está diseñado para trabajar con los proyectos que se pueden construir por equipos programadores, que no se ven limitados considerablemente por el entorno informático existente, y donde un trabajo razonable de pruebas de ejecución se puede hacer en una fracción de un día. La innovación de XP es:

- Poniendo todas estas prácticas bajo un mismo paraguas.
- Asegurarse de que se practican tan a fondo como sea posible.
- Asegurarse de que las prácticas se apoyan mutuamente con el grado más grande posible.

## 2.7 Herramienta de desarrollo Oracle Application Express

En esta sección se explicará de manera teórica el modo de desarrollo, componentes, y funcionamiento de Oracle Application Express.

### 2.7.1 ¿Qué se necesita para desarrollar en APEX?

Para crear una aplicación en Oracle Application Express necesitamos descargar e instalar Oracle Database Express Edition 11g, es libre para el desarrollo, implementación y distribución de aplicaciones, es necesario también descargar e instalar en APEX 4 y el Apex Listener.

### 2.7.2 Primeros pasos con Oracle Application Express

Para comenzar se accede a la dirección <http://localhost:8585/apex> en el navegador web, Oracle Application Express es compatible con cualquier navegador web de cualquier sistema operativo. El puerto se define en la instalación y configuración de Oracle Application Express.

### 2.7.3 Servicios de Administración de Application Express

Este módulo sirve para acceder y realizar tareas administrativas de Oracle Application Express, principalmente la opción **Gestionar Espacios de Trabajo** que es la que permite crear un área de trabajo compartida donde varios desarrolladores pueden crear aplicaciones, cada espacio de trabajo tiene el privilegio para analizar uno o varios esquemas de base de datos.



Los administradores del servicio Oracle Application Express pueden gestionar cuentas de usuario para cualquier espacio de trabajo, el acceso a este módulo es el como se muestra en la **Ilustración 8**.

## 2.8 Oracle Application Builder

Este módulo de Oracle Application Express provee a los desarrolladores de aplicaciones el uso de asistentes para ensamblar las aplicaciones de forma declarativa organizadas por páginas. El contenido de la página se organiza en regiones. Las regiones pueden contener texto, PL/SQL personalizado, informes, gráficos, mapas, calendarios, pantallas o referencias del servicio web. Las pantallas están compuestas de campos denominados elementos, que se pueden seleccionar de entre los distintos tipos integrados como: los campos de texto, las áreas de texto, los grupos de radio, las listas de selección, las casillas de control, los selectores de fechas y la lista emergente de valores. Está incorporada la funcionalidad de actualización de la tabla y se puede utilizar PL/SQL para procesar los datos. El estado de la sesión o contexto de la aplicación se gestiona de forma transparente y se separa la presentación de la interfaz de usuario de la lógica de la aplicación, de forma

que sólo hay que seleccionar un tema distinto para cambiar la apariencia de la aplicación. En la **Ilustración 9** se muestra el icono de acceso para un entorno en idioma inglés.

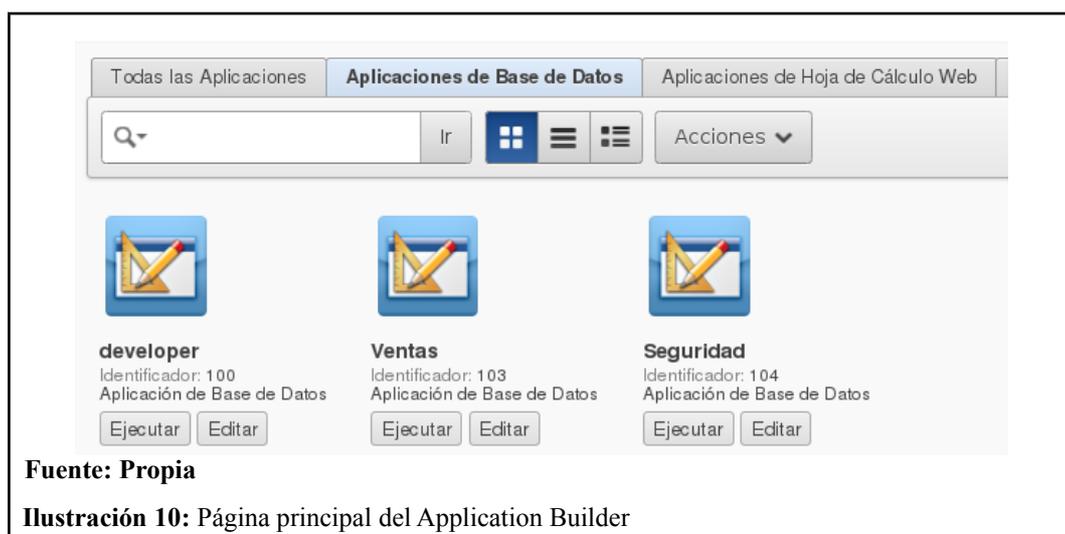


Con las aplicaciones de la base de datos, los desarrolladores pueden aprovechar directamente sus conocimientos de programación SQL y PL/SQL. Las aplicaciones de la base de datos utilizan control declarativo sobre el control de flujo y soportan controles de interfaz de usuario completos mediante el uso de plantillas y temas.

Una aplicación de la base de datos es una recopilación de páginas enlazadas utilizando separadores, botones o enlaces de hipertexto. Las páginas de una aplicación comparten un método de autenticación y una definición del estado de la sesión común.

## 2.9 La Aplicación.

En la página principal del Application Builder o en español **Creador de Aplicaciones** se muestran todas las aplicaciones creadas, son de dos categorías: Aplicaciones de Bases de Datos y Aplicaciones de Hoja de Cálculo Web, como se muestra en la **Ilustración 10**, pero en esta tesis solo se hablará de las aplicaciones de bases de datos.



Al seleccionar una aplicación específica en la página principal del **Creador de Aplicaciones**, se dirige a la página principal de la aplicación en donde se muestra un menú para manejar la aplicación y editar las propiedades de la aplicación, como en la **Ilustración 11**.



La definición de los iconos de opciones son:

- **Ejecutar Aplicación:** presenta en formato HTML la aplicación actual en el navegador, luego de interpretarla en el motor del Application Express.
- **Objetos de Soporte:** utilice esta utilidad para definir las definiciones de objetos de base de datos, imágenes y datos iniciales que se incluirán con la exportación de la aplicación.
- **Componentes Compartidos:** son elementos comunes que pueden mostrar o ser aplicados en cualquier página dentro de una aplicación. Puede utilizar las herramientas y asistentes en la página de componentes compartidos, ya sea en el nivel de aplicación o en páginas específicas, más adelante se detallara cuales son estos.
- **Utilidades:** Estas utilidades resumen información de toda la aplicación y facilitan el acceso a herramientas útiles.
- **Exportar/Importar:** llama al asistente para realizar la exportación o importación de la aplicación. Al exportar una aplicación se exportan todos sus atributos incluidas las plantillas, páginas, regiones, elementos, botones a un solo archivo. Al importar una aplicación se lo hará desde un archivo único previamente exportado anteriormente.

## 2.10 Los Componentes Compartidos

Acercas de los **Componentes Compartidos**, en la **Ilustración 12** se muestra la pantalla principal para acceder a las diferentes opciones, se describirá a las principales opciones utilizadas para la realización del prototipo.



### 2.10.1 Componente Compartido: Lógica

En la siguiente sección se describirá las opciones más importantes que están en el recuadro de **Lógica** como se ve en la **Ilustración 12**.

#### 2.10.1.1 Procesos de Aplicación

Los procesos de aplicación ejecutan lógica PL/SQL en puntos específicos para cada página de una aplicación, o como definen las condiciones en las que están definidas para arrancarse. Se tiene en cuenta que los procesos sólo se arrancan cuando se llaman desde AJAX o por procesos ejecutar a petición definidos para páginas concretas, por ejemplo se lista algunas de las opciones para la ejecución de estos procesos:

- Durante la carga, antes de cabecera de plantilla de página.
- Durante la carga, después de cabecera de plantilla de página.
- Durante la carga, antes de regiones de cuerpo de la página.
- Durante la carga, después de regiones de cuerpo de la página.
- Durante la carga, antes del pie de página de la plantilla de página.
- Durante la carga, después de pie de página de la plantilla de página.
- Durante la ejecución, después de ejecución de la página, antes de cálculos y validaciones.
- Durante la ejecución, después de ejecución de la página, después de cálculos y validaciones.
- En una nueva instancia o nuevo inicio de sesión.

- A Petición, se ejecuta este proceso de aplicación cuando se lo solicite el proceso de página.

### 2.10.1.2 Referencia de Servicios Web

El motor de Application Builder puede utilizar las referencias de servicio web para acceder a un servicio web a través de la red. El servicio web realiza una acción y a continuación devuelve una respuesta.

### 2.10.1.3 Carga de Datos

Una tabla de carga de datos es una tabla existente en el esquema que seleccione para utilizarla en el proceso de carga de datos, con el fin de cargar dichos datos. Se utiliza las tablas de carga de datos con el fin de definir tablas para utilizar en el asistente de creación de páginas de carga de datos.

El asistente de carga de datos crea un nuevo asistente de carga de datos, para gestionar la carga de datos en una tabla dentro de este esquema de análisis por defecto de la aplicación. Se crearán cuatro páginas dentro del nuevo asistente para obtener la posibilidad de cargar datos desde un archivo o mediante la opción de copiar y pegar, se define las asignaciones de tabla con datos, se validan los datos y, a continuación, se carga los datos en la tabla. Una vez creadas todas las páginas, éstas pueden personalizarse para que se ajusten a las necesidades de la aplicación, en la **Ilustración 13** se muestra un ejemplo de una carga de datos ya creada.



## 2.10.2 Componente Compartido: Navegación

En la siguiente sección se describirá las opciones más importantes que están en el recuadro de **Navegación** como se ve en la **Ilustración 12**.

### 2.10.2.1 Separadores

Los separadores son una manera eficaz para que los usuarios puedan navegar entre páginas de una aplicación. Se pueden crear dos tipos de separadores: separadores estándar o separadores padres. Un conjunto de separadores estándar se asocia a una página específica y al número de página. Un separador padre establece funciones como un contenedor para

almacenar un grupo de separadores estándar. En la siguiente **Ilustración 14** se muestra como se gestionan los separadores.



En la **Ilustración 15** se muestra como se ve el separador en el navegador web.



### 2.10.2.2 Listas

Una lista es una colección compartida de enlaces. El desarrollador controla la aparición de una lista a través de plantillas de lista. Cada elemento de la lista tiene una condición de visualización que le permite controlar cuando se muestre. En la **Ilustración 16** se muestra una lista, los elementos mostrados en esta lista pueden ser ocultos según el criterio de acceso o la inteligencia de negocio de la aplicación.



### 2.10.2.3 Rutas de Navegación

Las rutas de navegación proporcionan una navegación jerárquica con ilimitados niveles. Tras crear una ruta de navegación, se pueden definir sus entradas. Las entradas de la ruta de navegación se asocian a las páginas y también identifican una página principal. Las rutas de navegación se pueden mostrar en una página agregando una región de ruta de navegación mediante el uso de asistentes. En la **Ilustración 17** se muestra la ruta Inicio – Ventas – Administrar – Servicios.



### 2.10.2.4 Entradas de Barra de Navegación

Las entradas de barra de navegación proporcionan una navegación basada en enlaces de hipertexto. La ubicación se determina mediante plantillas de página. Las entradas de barra de navegación, normalmente iconos gráficos, se utilizan para proporcionar navegación a nivel de aplicación. Las entradas de barra de navegación se muestran en todas las páginas, aunque también se pueden mostrar de forma condicional. En la **Ilustración 18** se muestra el enlace **Administración** solo se muestra a usuarios administradores.



### 2.10.3 Componente Compartido: Seguridad

En la siguiente sección se describirá las opciones más importantes que están en el recuadro de **Seguridad** como se ve en la **Ilustración 12**.

#### 2.10.3.1 Esquema de Autenticación

La autenticación es el proceso que establece la identidad de cada usuario antes de que pueda acceder a la aplicación. Puede ser necesaria la introducción de un nombre de usuario y una contraseña por parte del usuario o podría implicar el uso de certificados digitales o de una clave segura.

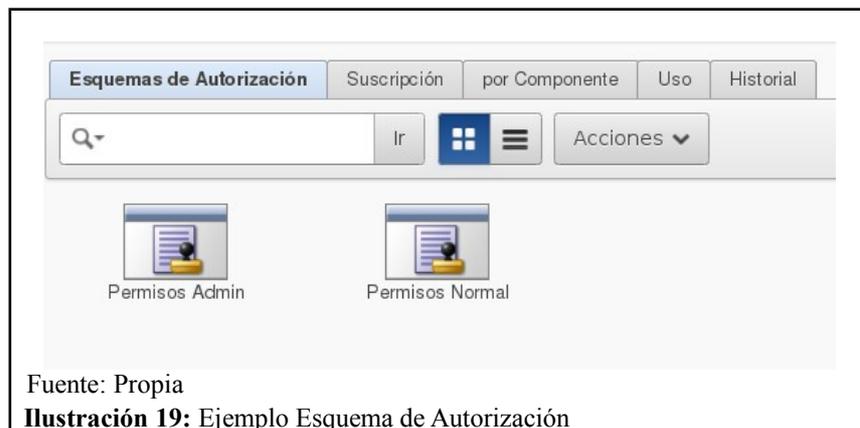
Se utiliza esta opción para crear o editar esquemas de autenticación y definir el que utiliza su aplicación. El esquema de autenticación actual determina la forma en que la aplicación identifica y verifica al usuario final. Dependiendo del tipo de esquema de autenticación seleccionado, puede utilizar diferentes configuraciones y enlaces de programa para ajustar la autenticación de la aplicación.

#### 2.10.3.2 Esquema de Autorización

El esquema de autorización se asocia a una aplicación, página o cualquier otro componente en la página de atributos del componente.

Los esquemas de autorización son comprobaciones de aprobación o fallo. Los tipos de esquema de autorización comunes son: consultas SQL Existe y No Existe y Función PL/SQL que devuelve un valor booleano.

El resultado satisfactorio o de error de los esquemas de autorización se almacenará en caché en cada sesión o en cada vista de página con el fin de mejorar el rendimiento. En la **Ilustración 19** se muestra un ejemplo de dos esquemas de autorización creados, ambos esquemas son comprobaciones con una función PL/SQL que devuelve un valor booleano.



A continuación se muestra en la **Ilustración 20** un ejemplo de como esta hecho el esquema de autorización **Permisos Admin** de la **Ilustración 19**.

**Esquema de Autorización**

\* Tipo de Esquema **Función PL/SQL que Devuelve Valor Booleano**

Cuerpo de la Función  
PL/SQL

```
begin
  return PKG_ADMINISTRACION.fun_is_user_admin(:APP_USER);
end;
```

\* Mensaje de error de identificación que aparece cuando se ha violado un esquema

No es un usuario Administrador

Fuente: Propia

**Ilustración 20: Tipo de Esquema de Autorización**

El uso de lenguaje de programación PL/SQL es el principal recurso para la mayoría de opciones y componentes del Oracle Application Express se uso en la **Ilustración 20**, se muestra a continuación el PL/SQL de la función en la **Tabla 2**:

```

1  /*
2  Retorna "true" si el usuario es admin o "false" si no lo
3  es
4  */
5  FUNCTION fun_is_user_admin(
6      p_username IN VARCHAR2)
7      RETURN BOOLEAN
8  AS
9      v_result VARCHAR2(1);
10 BEGIN
11     SELECT '1'
12     INTO v_result
13     FROM seg_usuarios
14     WHERE tipo          ='ADMIN'
15     AND upper(usuario)=upper(p_username);
16     IF v_result        ='1' THEN
17         RETURN true;
18     END IF;
19     RETURN false;
20 EXCEPTION
21 WHEN no_data_found THEN
22     RETURN false;
END;
```

**Tabla 2:** Ejemplo de código fuente de seguridad en PL/SQL

Se puede observar que la consulta se realiza a una tabla llamada **seg\_usuarios** donde tiene un campo **tipo**, este campo guarda la información si el usuario **p\_username** es o no **ADMIN**, de serlo retorna verdadero sino falso.

## 2.10.4 Componente Compartido: Interfaz de Usuario

En la siguiente sección se describirá las opciones más importantes que están en el recuadro de **Interfaz de Usuario** como se ve en la **Ilustración 12**.

### 2.10.4.1 Atributos de Interfaz de Usuario

Esta opción se utiliza para editar las definiciones de interfaz de usuario de la aplicación, como por ejemplo: logotipo de la aplicación, uso de archivos JavaScript, uso de archivos CSS<sup>15</sup> y para incluir una nueva interfaz de usuario.

<sup>15</sup> Es el lenguaje de hojas de estilo en cascada utilizado para describir el aspecto y el formato de un documento escrito en un lenguaje de marcas.

#### **2.10.4.2 Temas**

Un tema es una recopilación con nombre de plantillas que define la interfaz de usuario de la aplicación. Cada tema contiene plantillas para cada tipo de componente de aplicación y de control de página, incluidas las páginas individuales, las regiones, los informes, las listas, las etiquetas, las rutas de navegación, los botones y las listas de valores. Las plantillas contienen texto HTML y variables que el motor de Application Express. sustituye por valores dinámicos en tiempo de ejecución.

#### **2.10.4.3 Plantillas**

Las plantillas definen la forma en que se van a mostrar las páginas, los controles de página o los componentes de páginas. En esta opción puede editar los atributos de plantilla.

#### **2.10.4.4 Listas de Valores**

En esta opción se define las listas de valores, estas se pueden utilizar tanto en elementos de página como campos de informe para hacer referencia a una lista de valores. La lista de valores controla los valores mostrados y limita la selección del usuario. Las listas de valores pueden ser estáticas, basadas en los valores introducidos o dinámicas, basadas en una consulta SQL.

#### **2.10.5 Gestión de las páginas en Oracle Application Builder**

Cuando se genera una aplicación en Application Builder, se puede crear páginas que contienen elementos de la interfaz de usuario, tales como fichas, listas, botones, objetos y regiones. Se puede definir pantallas que permitan la entrada de datos para una amplia variedad de casos, incluidos:

- Maestro/Detalle
- Tabular
- Basada en Servicios Web
- Basada en Tablas
- Basada en Procedimientos Almacenados
- Pantallas de Asistente

El procesamiento de página consiste en:

- Cálculos: definir el valor de una aplicación o elemento de página.
- Validaciones: realizar comprobación de ediciones para garantizar la calidad de los datos.
- Procesos: interactuar con la base de datos, servicios web, otras páginas.
- Bifurcación: identificar si el usuario se toma después de procesar la página.

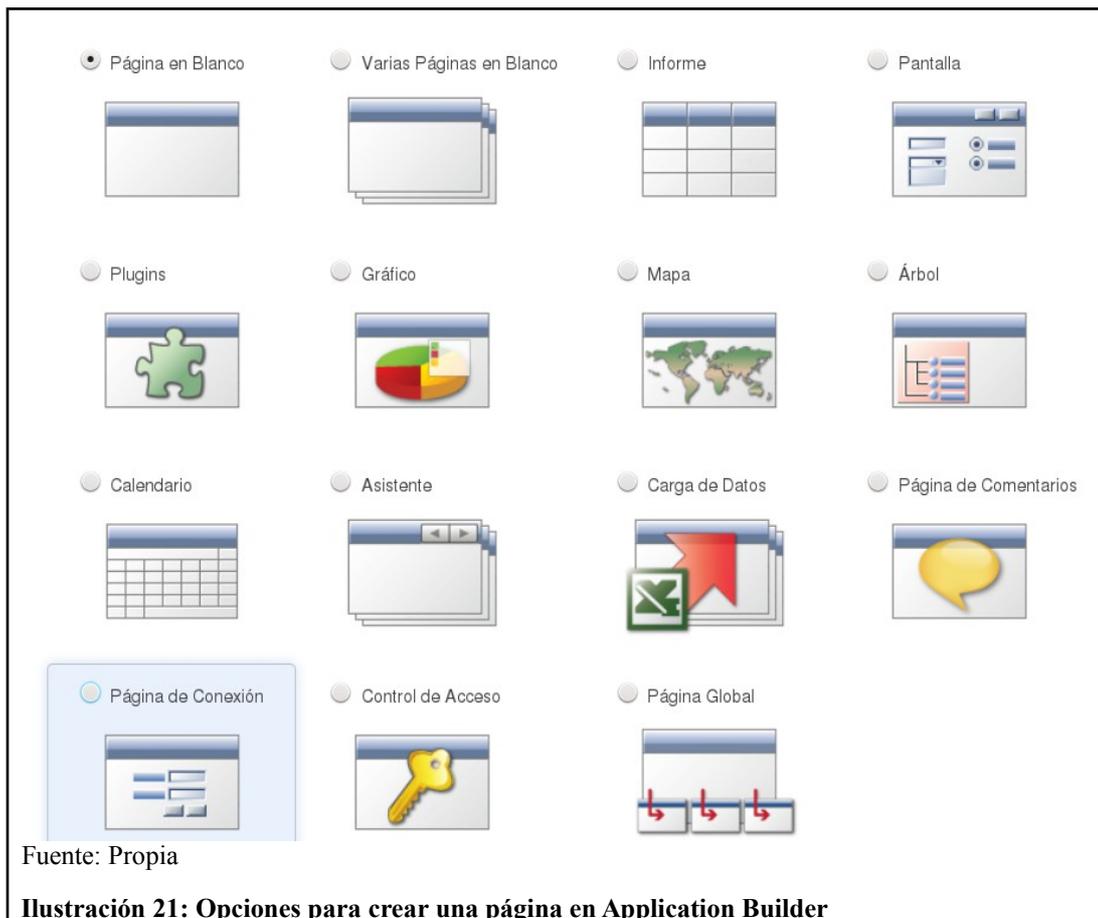
En las páginas también se puede crear regiones de informes, e informes interactivos que permiten a los usuarios finales personalicen los informes. Los usuarios pueden crear varias variaciones del informe y guardarlas como informes con nombre, exportarlas a archivos delimitados por comas e imprimirlas como documentos PDF. Los informes interactivos permiten:

- Selección de Columnas
- Filtros
- Ordenación
- Divisiones de Control
- Resaltado

- Columnas Calculadas
- Suscripciones
- Agrupar por Informes
- Descargar en CSV, Correo Electrónico, CSV y PDF
- Generación de Gráficos

Las regiones de gráficos en flash o HTML permiten crear una amplia variedad de gráficos de una o varias consultas SQL.

En la **Ilustración 21** se muestran las páginas que podemos crear para nuestra aplicación de base de datos.



### 2.10.5.1 Páginas de Aplicación de Base de Datos

Una página puede tener botones y campos llamados elementos, que se agrupan en contenedores llamados regiones. Las páginas también pueden tener lógica de la aplicación o procesos. Puede pasar de una página a la siguiente usando la navegación condicional, realizar cálculos, realizar validaciones como comprobaciones de edición, y visualizar informes y gráficos, a continuación se detalla los más importantes tipos de páginas.

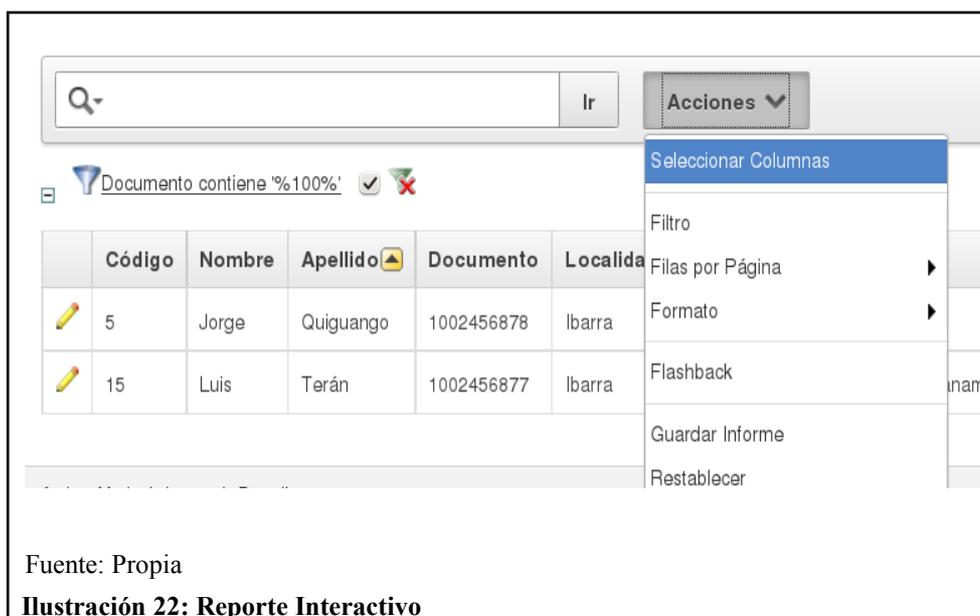
- **Página en Blanco:** crea una página con ninguna funcionalidad integrada.
- **Varias Páginas en Blanco:** crea varias páginas en blanco.
- **Pantalla:** crea una interfaz de formulario con el que los usuarios pueden actualizar una sola fila o varias filas dentro de una tabla.
- **Gráfico:** crea gráficos estadísticos en HTML5 o Flash

- **Árbol:** crea una estructura de datos gráfica tipo árbol para mostrar datos jerárquicos o datos con múltiples niveles.
- **Página de Conexión:** crea una página de conexión donde se ingresa el usuario y contraseña.

### 2.10.5.2 Páginas de Reportes Interactivos y Reportes Clásicos

Al crear una aplicación de base de datos, se puede incluir dos tipos de informes: un informe interactivo o un informe clásico. La principal diferencia entre estos dos tipos de informes es que los informes interactivos permiten al usuario personalizar la apariencia de los datos a través de la búsqueda, filtrado, clasificación, selección de columnas, poniendo de relieve, y otras manipulaciones de datos.

Los reportes interactivos permiten a los usuarios finales crear informes altamente personalizados. Los usuarios pueden cambiar el diseño del informe al ocultar o mostrar columnas específicas y la aplicación de filtros, aplicación de orden, y la clasificación. También pueden definir saltos, agregaciones, gráficos, apartaderos de grupo, y añadir sus propios cálculos. Una vez personalizado, el informe se puede guardar como un informe o bien privado o público. Cada informe interactivo incluye una barra de búsqueda, el menú Acciones, menú Encabezado de columna, y editar iconos en la primera columna de cada fila. En la **Ilustración 22** se muestra un ejemplo de como es un reporte interactivo.



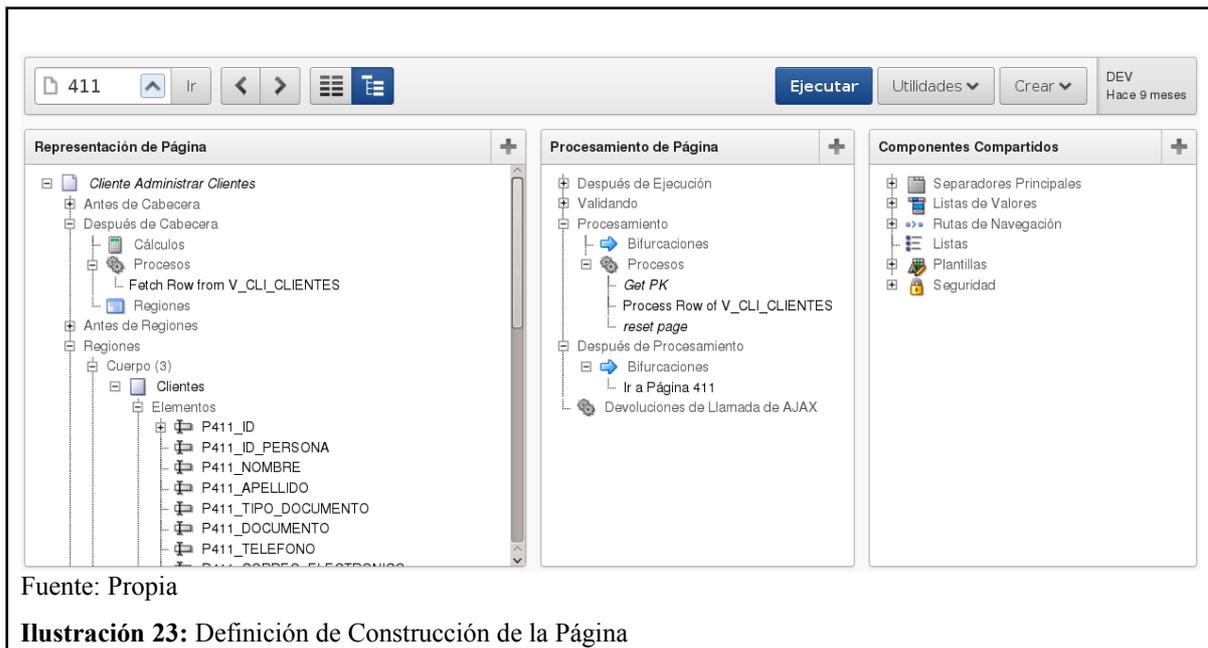
**Ilustración 22: Reporte Interactivo**

Los reportes clásicos son los contrario, no permite ninguna personalización pero conservan la capacidad de búsqueda.

### 2.10.6 Definición de construcción de la página en Oracle Application Builder

Se utiliza la definición de página para ver, crear y editar los controles y la lógica de aplicación que definen a una página. Cada página puede tener varios elementos, que se agrupan en contenedores llamados regiones. Las páginas también tienen la lógica de la aplicación que se puede editar o crear a través de esta interfaz. Puede pasar de una página a la siguiente usando la navegación condicional, crear cálculos, crear validaciones como comprobaciones de edición, y visualizar informes y gráficos. Se puede ver, crear y editar

los controles que definen a una página mediante el acceso a la definición de página. La **Ilustración 23** muestra la definición de construcción de la página.



### 2.10.6.1 Los Elementos de una página en Oracle Application Builder

Los elementos son parte de una pantalla HTML, pueden ser un campo de texto, área de texto, contraseña, selección de lista, casilla de control o cualquier otro elemento usado en páginas HTML. Los elementos tienen atributos que afectan a la visualización en una página. Por ejemplo, estos atributos pueden afectar lo grande que es un elemento, y si el elemento aparece al lado o debajo del elemento anterior, en la **Ilustración 24** se muestran algunos elementos para seleccionar.



En la **Ilustración 25** se muestran algunos elementos en una vista tipo componente estos elementos están ya dentro de una pantalla de aplicación.

Elementos		
Región <b>Cientes</b>		
10	<u>P411_ID</u>	Código SkillBuilders Super LOV (2.0.2) [Plugin]
20	<u>P411_ID_PERSONA</u>	Id Persona Oculto
30	<u>P411_NOMBRE</u>	Nombre Campo de Texto
40	<u>P411_APELLIDO</u>	Apellido Campo de Texto
50	<u>P411_TIPO_DOCUMENTO</u>	Tipo Documento Lista de Selección
60	<u>P411_DOCUMENTO</u>	Documento Campo de Texto
80	<u>P411_TELEFONO</u>	Teléfono Campo de Texto
90	<u>P411_CORREO_ELECTRONICO</u>	Correo Electrónico Campo de Texto
110	<u>P411_ID_LOCALIDAD</u>	Localidad Lista de Valores Emergente
120	<u>P411_DIRECCION</u>	Dirección Campo de Texto
150	<u>P411_ID_TIPO_CLIENTE</u>	Tipo de Cliente Lista de Selección
180	<u>P411_IVA</u>	Iva Oculto
190	<u>P411_OBSERVACION</u>	Observacion Área de Texto
200	<u>P411_ESTADO</u>	Estado Oculto
210	<u>P411_USUARIO</u>	Oculto

Fuente: Propia  
**Ilustración 25:** Elementos en una vista tipo componente.

Cuando se crea un elemento, se especifica un tipo de elemento. Se describe a continuación los tipos de elementos disponibles y los atributos del elemento que definen el comportamiento de los mismos.

#### a) Área de Texto

Se muestra un área de texto HTML. No existe una longitud máxima de un elemento que se muestra como un área de texto. Se controla la altura y el ancho del elemento.

#### b) Campo de Texto

Se muestra el campo de texto de uso normal en HTML.

#### c) Campo Numérico

Muestra un campo tipo numérico. Se puede configurar el atributo en un rango de números mínimo y máximo. Este tipo de elemento comprueba automáticamente si el valor introducido es un número.

#### d) Casilla de Control

Se muestra una casilla de control HTML, está basado en una lista de valores que se construye con una instrucción de selección de SQL .

#### e) Contraseña

Se muestra un campo de contraseña HTML.

### **f) Explorador de Archivos**

Se muestra un campo de texto con un botón de explorar que permite al usuario localizar y cargar el archivo a subir.

### **g) Grupo de Botones de Radio**

Se muestra un grupo de botones de radio HTML, está basado en una lista de valores. Por ejemplo se lo puede construir a través de una instrucción de selección SQL; **select codigo, documento from tipos\_documentos**.

### **h) Lista de Selección**

Se muestra una lista de selección HTML, está basado en una lista de valores. Por ejemplo se la puede construir a través de una instrucción de selección SQL; **select usuario as display\_value, USUARIO as return\_value from seg\_usuarios order by 1**, el valor para display\_value es el que se mostrara en la página y el valor return\_value es que se envía al ejecutar la página.

### **i) Lista de Valores Emergentes**

Se muestra un campo de texto con un icono. Cuando el usuario hace clic en el icono, aparece una ventana emergente con una lista de valores representado una serie de enlaces. Cuando el usuario selecciona un elemento de la lista, el valor seleccionado se coloca en el campo de texto. Las lista de valores emergentes se basan en una consulta SQL por ejemplo:

```
select c.apellido||' '||c.nombre "Nombres", c.ID "Código" from v_cli_clientes c.
```

### **j) Mostrar Imagen**

Permite la visualización de las imágenes almacenadas en una URL o en una columna BLOB<sup>16</sup> de la base de datos.

### **k) Oculto**

Crea un elemento oculto.

### **l) Selector de Color**

Se muestra un campo de texto con un icono. Cuando el usuario hace clic en el icono, aparece una ventana emergente. Cuando el usuario hace una selección de la paleta, se devuelve el valor de código HTML para el color seleccionado.

### **m) Selector de Fecha**

Se muestra un campo de texto con un icono de calendario. Al hacer clic en este icono, se muestra un pequeño calendario en el que el usuario puede seleccionar una fecha.

### **n) Solo Visualización**

Crea un elemento solo para mostrar texto.

---

<sup>16</sup> Binary Large Objects, objetos binarios grandes; se utilizan para guardar en la base de datos objetos de gran tamaño como imágenes.

### 2.10.7 Haciendo referencia a los valores de los elementos de una página en Oracle Application Builder

Se puede hacer referencia a los valores de los elementos; en las regiones, cálculos y procesos de validación. En la **Tabla 3** se describe las formas:

Tipo	Sintaxis	Descripción
SQL	:MI_ELEMENTO	Se utiliza esta sintaxis para las referencias dentro de una consulta SQL y en el código PL / SQL
PL/SQL	V('MI_ELEMENTO')	Se utiliza esta sintaxis en el código PL / SQL de paquetes o procedimientos almacenados y funciones.
PL/SQL	NV('MI_ELEMENTO_NUMERICO')	Se utiliza esta sintaxis en el código PL / SQL de paquetes o procedimientos almacenados y funciones.
Texto estático	&MI_ELEMENTO.	Se utiliza esta sintaxis para una sustitución exacta, se debe evitar en código SQL o PL / SQL, ya que puede dar lugar a vulnerabilidades de inyección SQL.

**Tabla 3:** Tipos de referencias a los elementos de la aplicación

### 2.10.8 Las Regiones en una página en Oracle Application Builder

Una región es un área de una página que sirve como un contenedor. Cada página puede tener cualquier número de regiones. Se controla la aparición de una región a través de una plantilla de región específica. La plantilla región controla el aspecto de la región, el tamaño, determina si existe un límite o un color de fondo, y qué tipo de fuente se mostrara. Una plantilla región también determina la ubicación estándar para los botones colocados en posiciones de la región.

Se puede utilizar las regiones para los controles la página como elementos o botones. Las regiones se muestran en secuencia en el cuerpo de la plantilla de página o se pueden colocar de forma explícita. En la **Ilustración 26** se muestra una pantalla con tres regiones; Datos, Artículos y Detalle.

**Datos** Crear

\* Fecha

\* Cliente

**Articulos** Agregar

\* Artículo

\* Cantidad

\* Precio Unitario

**Detalle**

	Código	Nombre	Cantidad	Precio	Precio Total	IVA	Total
X	7	Servicios Prestados	1	100	100	12	112
<b>Total de Informe:</b>							<b>112</b>

1 - 1

Fuente: Propia

**Ilustración 26:** Ejemplo de una pantalla con tres regiones

### 2.10.9 Los Botones en una página en Oracle Application Builder

Al diseñar una aplicación, puede utilizar los botones para dirigir a los usuarios a una página específica o URL, o para permitir a los usuarios enviar información a una página. Un botón puede ser implementado como un botón HTML, una imagen, o mediante el uso de una plantilla. Los botones pueden ser colocados en posiciones de las regiones plantillas predefinidas o entre los elementos de un formulario.

### 2.10.10 Las Validaciones en una página en Oracle Application Builder

La validación es un chequeo de ingreso o edición de información o datos. Existen las validaciones específicas de un solo elemento. Las validaciones que se aplican a toda una página. Las validaciones para pantallas tabulares específicas a una sola columna. Las validaciones para pantallas tabulares que no se aplican a una sola columna sino a una fila del formulario.

Se puede definir una validación de forma declarativa mediante la selección de un método de validación. Se debe tener en cuenta que si hay un error en la validación, los procesos de las páginas posteriores o cálculos no ocurren. En la **Tabla 4** se describe los métodos de validación.

Método de Validación	Descripción
Not Null	Comprueba si el valor de un elemento en estado de sesión es nulo.
Comparación	Compara el valor de un elemento de una cadena específica.
Expresión Regular	Las expresiones regulares proporcionan un método para describir los patrones de texto.
SQL	Puede comparar los valores con valores de las base de datos de las siguientes formas: <ul style="list-style-type: none"> <li>• Existe</li> <li>• No Existe</li> <li>• Expresiones SQL</li> </ul>
PL/SQL	Si se necesita de una validación compleja se puede utilizar: <ul style="list-style-type: none"> <li>• Expresiones PL/SQL</li> <li>• Errores PL/SQL</li> <li>• Funciones que retornan tipos de datos Boolean</li> <li>• Funciones que retornan mensajes de error</li> </ul>

**Tabla 4:** Métodos de validación de una aplicación.

### 2.10.11 Los Cálculos en una página en Oracle Application Builder

Se utilizan los cálculos en las páginas para asignar un valor a un elemento identificado cuando una página se presenta o se muestra. También puede utilizar los cálculos de nivel de aplicación para asignar valores a los elemento. La mayoría de los cálculos de nivel de página asignan valores a los elementos de página. Como, la mayoría de los cálculos a nivel de aplicación asignan valores a los elementos de aplicación.

### 2.10.12 Los Procesos en una página en Oracle Application Builder

Un proceso de la página es una unidad de la lógica que se ejecuta cuando se produce un evento específico, como la carga o la presentación de una página. En la **Tabla 5** se describe los tipos de procesos que se pueden realizar.

Tipo de Procesos	Descripción
PL/SQL	Ejecuta PL/SQL del desarrollador.
Manipulación de datos	Los tipos de procesos de manipulación de datos son creados por los asistentes para apoyar el lenguaje de manipulación de datos.
Servicios Web	Implementa un Servicio Web como un proceso en una página. La ejecución del proceso presenta una solicitud al proveedor del Servicio Web.
Enviar Correo Electrónico	Se proporciona una interfaz para realizar el envío de correos electrónicos.

**Tabla 5:** Tipos de procesos de una aplicación.

### 2.10.13 Los Plug-ins en Oracle Application Builder

Los plug-ins permiten a los desarrolladores extender, compartir y reutilizar de forma declarativa los componentes disponibles en Oracle Application Express. Debido a que los plug-ins están diseñados para su reutilización, los desarrolladores pueden crear, exportar e importar a otras áreas de trabajo, también se los encuentra compartidos a través de repositorios en la comunidad de plug-ins de Oracle Application Express.

Si bien siempre es posible crear funciones personalizadas utilizando herramientas como Ajax, el código podría ubicarse en varios lugares diferentes: en el interior de la base de datos o en archivos JavaScript. Se puede seguir utilizando todas las herramientas de personalización que el desarrollador está familiarizado, pero convertir ese código en un plug-in hace que sea mucho más fácil de usar y administrar, ya que todo el código está contenido en un solo nuevo tipo de objeto.

En vez de construir JavaScript, jQuery, Ajax en sus páginas de forma manual, se puede simplemente importar los plug-ins existentes que utilizan la interfaz estándar Oracle Application Express, si se encuentra un plug-in que casi se adapte a las necesidades del usuario o desarrollador, pero no exactamente, se puede extender un plug-in importado para obtener exactamente lo que se necesita. Como ventaja se ahorra tiempo y se puede aprender mucho mediante la revisión de código que otro desarrollador escribió.

### 2.10.14 Las Acciones Dinámicas en una página en Oracle Application Builder

Las acciones dinámicas proporcionan una forma para definir el comportamiento de la aplicación en el navegador web del cliente, se lo hace de forma declarativa con o sin necesidad de JavaScript. El uso de la acción dinámica creada desde el asistente, especifica una acción que se realiza cuando se produce una serie de condiciones definidas. También puede especificar qué elementos se ven afectados por la acción, cuándo y cómo se ven afectados.

Cuando se trabaja con acciones dinámicas, se debe tener en cuenta el hecho de que mientras más acciones dinámicas se agregan a una página, mayor será el tamaño total de la página. Esto es debido a que el marco de acción dinámica emite código adicional para la ejecución de dicha acción dinámica, entonces también debe ser descargado y ejecutado por el navegador web en el cliente.

Crear una acción dinámica implica especificar cuando la acción sucede, se pueden definir al desencadenante basado en eventos que suceden en la página. Hay cuatro categorías diferentes de eventos que pueden ser utilizados, eventos del explorador, eventos de marco, eventos de componentes y evento personalizado.

#### 1. Eventos del Explorador

- **Cambiar:** Se activa cuando un control pierde el foco de entrada y su valor ha sido modificado desde que obtuvo el enfoque.
- **Click:** Se activa cuando se hace click sobre el elemento desencadenante.
- **Doble Click:** Se activa cuando se hace doble click sobre el elemento desencadenante.
- **Obtener Enfoque:** Se activa cuando el elemento desencadenante recibe el foco, ya sea con un puntero o tabulando en el elemento.

- **Tecla Abajo:** Se activa cuando se pulsa una tecla en el teclado. Se utilice este evento cuando se desea capturar pulsaciones de teclas especiales, tales como las teclas de flecha, después de pulsar una tecla.
- **Pulsación de Tecla:** Se activa cuando se pulsa una tecla en el teclado que resulta ser el texto que se ingresa. Utilice este evento cuando se quiere captar el ingreso de texto actual.
- **Liberación de Tecla:** Se activa cuando se suelta una tecla del teclado. Utilice este evento cuando se desea capturar pulsaciones de teclas especiales, tales como las teclas de flecha, después de que haya sido liberada una tecla.
- **Perder Enfoque:** Se activa cuando el elemento desencadenante pierde el foco, ya sea por el dispositivo puntero o por presionar la tecla tabulador y quedar fuera del elemento.
- **Pulsación del Botón del Ratón:** Se activa cuando se pulsa el botón del ratón sobre el elemento desencadenante.
- **Liberación del Botón del Ratón:** Se activa cuando se suelta el botón del ratón sobre el elemento desencadenante.
- **Acercamiento del Ratón:** Se activa cuando el apuntador se mueve en el elemento desencadenante.
- **Alejamiento del Ratón:** Se activa cuando el apuntador se mueve alejándose del elemento de activación.
- **Movimiento del Ratón:** Se activa cuando el apuntador se mueve mientras está sobre el elemento desencadenante.
- **Carga de Página:** Se activa cuando la página se carga.
- **Anulación de Carga de Página:** Se activa cuando se interrumpe la carga de la página.
- **Cambiar Tamaño:** Se activa cuando la pantalla del navegador web cambia de tamaño.
- **Carga de Recurso:** Se activa cuando el navegador termina de cargar todo el contenido dentro de un documento, incluyendo ventanas, marcos, objetos e imágenes.

## 2. Eventos de Marco

- **Después de Refrescamiento:** Se activa cuando el elemento ha sido actualizado, solo se aplica para elementos que se refresquen parcialmente dentro de la página.
- **Antes de Ejecución de Página:** Se activa antes de que la página sea ejecutada o antes de haber realizado un envío de datos.
- **Antes de Refrescamiento:** Se activa antes de que el elemento haya sido cargado, solo se aplica para elementos que se refresquen parcialmente dentro de la página.

## 3. Eventos de Componentes

Estos eventos están disponibles cuando hay un componente o plug-in a disposición de la aplicación que provoca un evento personalizado.

## 4. Evento Personalizado

Se lo utiliza cuando los eventos definidos no son suficientes para el objetivo de la acción dinámica.

### 2.10.15 El lenguaje JavaScript y AJAX en Oracle Application Builder

Se puede utilizar JavaScript para proporcionar funcionalidades del lado del cliente, tales como mostrar y ocultar elementos de la página, o hacer peticiones HTTP JavaScript asíncrono conocido como AJAX.

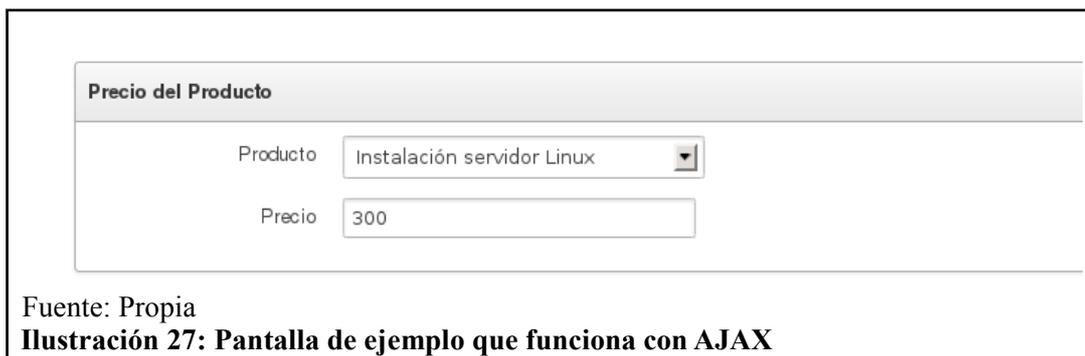
Además puede incluir archivos o librerías de JavaScript para una aplicación web es una manera de agregar características que imitan las que se encuentran en las aplicaciones cliente - servidor sin perder los beneficios de la implantación en la web. Oracle Application Express incluye múltiples interfaces integradas especialmente diseñados para añadir JavaScript.

En la práctica se puede usar para hacer referencia a un elemento de la página, el mejor enfoque es hacer referencia por id. Si se ve el código fuente HTML de una página de Oracle Application Express en un navegador web, todos los artículos tienen un atributo id. Este id corresponde al nombre del elemento, no la etiqueta del elemento. Por ejemplo, si crea un elemento con el nombre P1\_APELLIDO y una etiqueta de Apellido, el ID es P1\_APELLIDO.

Llamar a un código JavaScript desde un botón de una página es una gran manera de confirmar el evento a realizar como eliminar un registro o guardar. Oracle Application Express utiliza esta técnica para la operación de eliminación. Por ejemplo, cuando se elimina un registro, aparece un mensaje de JavaScript que le pedirá que confirme la solicitud.

Para la comunicación AJAX entre el cliente y el servidor en Oracle Application Express utiliza la función apex.server.process, esta función llama un proceso PL/SQL definido en la aplicación, lo que retorna la función es personalizable para el desarrollador, en el siguiente ejemplo se describe el funcionamiento:

Se tiene la pantalla mostrada en la **Ilustración 27**, se tiene dos elementos **P2\_PRODUCTO** que es una lista de selección y representa la descripción de un producto, y **P2\_PRECIO** es un campo de texto donde se muestra el precio del producto. Al cambiar el valor en **P2\_PRODUCTO** se hace una llamada AJAX para recuperar el valor para **P2\_PRECIO**.



A continuación se crea un proceso PL/SQL **AJAX\_GET\_PRECIO** con las propiedades que se muestran en la **Ilustración 28**.

Página: 2 Ejemplo AJAX

\* Nombre:

Tipo: **Bloque Anónimo PL/SQL**

**Punto de Proceso**

\* Secuencia:

Punto de Proceso: **A Petición: Ejecute este proceso cuando lo solicite AJAX** ▾

Proceso de Ejecución: **Una Vez por Visita de Página (valor por defecto)** ▾

Fuente: Propia

**Ilustración 28: Propiedades del proceso PL/SQL a ser llamado por AJAX**

El código PL/SQL del proceso se muestra en la **Tabla 6**, la variable **apex\_application.g\_x01** hace referencia a la variable de la llamada AJAX en el código JavaScript de la **Tabla 7**. Para devolver el valor consultado se usa la función **htp.prn()**, su uso principal es para imprimir código HTML desde PL/SQL en Oracle Application Express.

```

1  /*
2  Consulta el precio de la vista V_CLI_PRECIOS_DET
3  */
4  DECLARE
5      vPrecio NUMBER;
6  BEGIN
7      SELECT precio
8      INTO vPrecio
9      FROM V_CLI_PRECIOS_DET
10     WHERE ID=TO_CHAR(apex_application.g_x01);
11     htp.prn(vPrecio);
12 END;
```

**Tabla 6:** Ejemplo PL/SQL para ser llamado desde una función AJAX.

En la **Ilustración 29** se muestra las propiedades con las que se crea la Acción Dinámica que ejecuta el código JavaScript. El evento desencadenante será cuando el usuario cambie el valor del elemento **P2\_PRODUCTO**, si la acción de cambiar el valor es verdadera se ejecuta el código JavaScript que contiene la llamada AJAX al proceso de la **Tabla 6** que retorna el valor y se muestra en **P2\_PRECIO**.

Página: 2. Ejemplo AJAX

\* Nombre

\* Secuencia

**When**

\* Evento

\* Tipo de Selección

\* Elementos

Condición

Fuente: Propia

**Ilustración 29:** Propiedades de la Acción Dinámica que ejecuta el código JavaScript.

En la **Tabla 7** se muestra el código JavaScript donde, en la línea 1 se realiza la llamada al proceso **AJAX\_GET\_PRECIO** con la función **apex.server.process**, en la línea 2 se obtiene el valor de **P2\_PRODUCTO** con la función JavaScript **document.getElementById** y se asigna a **x01**, la variable **x01** es aquella que en la **Tabla 6** se lee **apex\_application.g\_x01** en el proceso PL/SQL. En la línea 3 se define el tipo de dato que retorna **AJAX\_GET\_PRECIO**, en este caso tipo texto. En caso de una ejecución satisfactoria se define una función que asigna la información a **P2\_PRECIO**.

```

1 apex.server.process ('AJAX_GET_PRECIO'
2   , {x01:
3     document.getElementById( "P2_PRODUCTO" ).value}
4     , {"datatype": "text"
5       , success: function(pData)
6         {
7       document.getElementById( "P2_PRECIO" ).value=pData;
8         }
9     });

```

**Tabla 7:** Ejemplo de código JavaScript que implementa AJAX.

### 2.10.16 La librería jQuery en Oracle Application Builder

La librería jQuery de código abierto es ampliamente aceptado y está cargado con útiles funcionalidades JavaScript y Ajax. Oracle Application Express sólo carga los componentes de interfaz de usuario jQuery que se requieren para la base de la funcionalidad de Oracle Application Express. No incluye toda la biblioteca jQuery UI, ya que hacerlo sería aumentar considerablemente descargar y tiempo de procesamiento para cada carga de página. Se incluye los siguientes componentes.

- jQuery UI Core - Requerido para todas las interacciones y widgets<sup>17</sup>.
- Query UI Widget - La fábrica widget que es la base para todos los widgets.

<sup>17</sup> Es una pequeña aplicación o programa, usualmente presentado en archivos o ficheros pequeños.

- jQuery UI Mouse - El widget de ratón es una clase base para todas las interacciones y widgets con la interacción de ratón.
- jQuery UI Position - Un plug-in de utilidad para los elementos de posicionamiento en relación con otros elementos.
- jQuery UI Draggable - Hace que cualquier elemento de la página se puede arrastrar.
- jQuery UI Resizable - Hace que cualquier elemento de la página sea de tamaño variable.
- jQuery UI de Dialog - Abre un marco existente en un cuadro de diálogo que pueden arrastrarse y de tamaño variable.
- jQuery UI datepicker - Un selector de fechas que se puede alternar desde una entrada o en línea mostrada.
- jQuery UI Effects - Extiende los efectos internos de jQuery, incluye el morphing<sup>18</sup> que es requerido por todos los demás efectos.
- jQuery UI Effects Drop - Un efecto de abandono, moviendo el elemento en una dirección y ocultando al mismo tiempo.

Si el desarrollador necesita cargar los demás componentes de jQuery se lo puede realizar adjuntando las librerías necesarias en las aplicación o página. A continuación se muestra un ejemplo aplicando jQuery y AJAX, no se cargo librerías adicionales para este ejemplo:

En la **Ilustración 30** se presenta la pantalla para recuperar la información de los totales vendidos en un periodo, se tiene seis elementos; **P231\_FECHA\_INICIAL** la fecha inicial, **P231\_FECHA\_FINAL** la fecha final, **P231\_RETENCION\_FUENTE** el valor en retenciones a la fuente, **P231\_RETENCION\_IVA** el valor en retenciones al IVA, **P231\_EFECTIVO** el valor del efectivo y **P231\_CHEQUE** el valor de cheques. De acuerdo al rango de fechas se sumaran los valores.

Ventas totales por fecha

* Fecha Inicial	01-03-2014	
* Fecha Inicial	31-03-2014	
* Retención Fuente	6.00	
* Retención IVA	1.56	
* Efectivo	291.93	
* Cheque	120.00	

Fuente: Propia  
**Ilustración 30: Pantalla de ejemplo que funciona con jQuery y AJAX**

A continuación se crea un proceso PL/SQL **AJAX\_GET\_TOTALES** con las propiedades que se muestran en la **Ilustración 31**.

<sup>18</sup> Es un efecto de animación que transforma un objeto en otro objeto.

Página:	231 VenRepVentasTotalesPorFecha
* Nombre	<input type="text" value="AJAX_GET_TOTALES"/>
Tipo:	Bloque Anónimo PL/SQL
<b>Punto de Proceso</b>	
* Secuencia	<input type="text" value="10"/>
Punto de Proceso	A Petición: Ejecute este proceso cuando lo solicite AJAX ▾
Proceso de Ejecución	Una Vez por Visita de Página (valor por defecto) ▾
Fuente: Propia	
<b>Ilustración 31: Propiedades del proceso PL/SQL a ser llamado por AJAX</b>	

El código PL/SQL del proceso se muestra en la **Tabla 8**, se consulta la información en la base de datos y se devuelve en formato JSON<sup>19</sup> en lugar de XML con la ayuda de la función `apex_util.json_from_sql`.

<sup>19</sup> Notación de Objetos de JavaScript, es un formato ligero de intercambio de datos.

```

1 DECLARE
2   vRetencionFuente NUMBER;
3   vRetencionIVA     NUMBER;
4   vEfectivo         NUMBER;
5   VCheque           NUMBER;
6 BEGIN
7   SELECT NVL(SUM(d.RETENCION_FUENTE), 0)
8   RETENCION_FUENTE,
9     NVL(SUM(d.RETENCION_IVA), 0) RETENCION_IVA,
10    NVL(SUM(d.EFECTIVO), 0) EFECTIVO,
11    NVL(SUM(d.CHEQUE), 0) CHEQUE
12 INTO vRetencionFuente,
13      vRetencionIVA,
14      vEfectivo,
15      VCheque
16 FROM CLI_DOCUMENTOS_CAB d
17      WHERE TRUNC(d.FECHA) between
18      TRUNC(to_date(apex_application.g_x01, 'dd/mm/yyyy'
19      )) and
20      TRUNC(to_date(apex_application.g_x02, 'dd/mm/yyyy'
21      ));
22 apex_util.json_from_sql('select ''||
23 to_char(vRetencionFuente, '9990.90')|| ''
24 RETENCION_FUENTE, ''||
25 to_char(vRetencionIVA, '9990.90')|| ''
26 RETENCION_IVA, ''||
27 to_char(vEfectivo, '9990.90')|| '' EFECTIVO, ''||
28 to_char(VCheque, '9990.90')|| '' CHEQUE from
29 dual');
30 EXCEPTION
31 WHEN OTHERS THEN
32 apex_util.json_from_sql('select 0 RETENCION_FUENTE, 0
33 RETENCION_IVA, 0 EFECTIVO, 0 CHEQUE from dual');
34 END;

```

**Tabla 8:** Ejemplo PL/SQL para ser llamado desde una función AJAX.

**En la Ilustración 32** se muestra las propiedades con las que se crea la Acción Dinámica que ejecuta el código JavaScript. El evento desencadenante será cuando el usuario cambie el valor del elemento **P231\_FECHA\_FINAL**, si la acción de cambiar el valor es verdadera se ejecuta el código JavaScript que contiene la llamada AJAX al proceso de la **Tabla 8** que retorna el conjunto de valores.

Página: 231. VenRepVentasTotalesPorFecha

\* Nombre

\* Secuencia

**When**

\* Evento

\* Tipo de Selección

\* Elementos

Condición

Fuente: Propia

**Ilustración 32:** Propiedades de la Acción Dinámica que ejecuta el código JavaScript.

En la **Tabla 9** se muestra JavaScript con sintaxis jQuery donde, en la línea 1 se realiza la llamada al proceso **AJAX\_GET\_TOTALES** con la función `apex.server.process`, de la línea 2 a la 3 se obtienen los valores de **P231\_FECHA\_INICIAL** y **P231\_FECHA\_FINAL** usando la función de jQuery y se asigna a **x01** y **x02** respectivamente, las variables **x01** y **x02** son aquellas que en la **Tabla 8** se lee **apex\_application.g\_x01** y **apex\_application.g\_x02** en el proceso PL/SQL. En la línea 4 se define el tipo de dato que retorna **AJAX\_GET\_TOTALES**, en este caso tipo JSON. En caso de una ejecución satisfactoria se define una función que extrae los valores que están en formato JSON y los asigna a sus respectivos elementos de la pantalla, también con sintaxis jQuery. Entre la comunicación asíncrona a la base de datos se muestran mensajes usando las funciones de jQuery UI Dialog.

```

1 apex.server.process('AJAX_GET_TOTALES'
2 ,{x01: apex.jQuery('#P231_FECHA_INICIAL').val(),
3   x02: apex.jQuery('#P231_FECHA_FINAL').val() }
4 ,{"dataType": "json", success: function(pData) {
5   $("#dialog").dialog();
6   $("#P231_RETENCION_FUENTE").text("");
7   $("#P231_RETENCION_IVA").text("");
8   $("#P231_EFECTIVO").text("");
9   $("#P231_CHEQUE").text("");
10  $.each(pData.row, function(key, val) {
11    $
12    ($("#P231_RETENCION_FUENTE").append(val.RETENCION_FUENTE);
13    $("#P231_RETENCION_IVA").append(val.RETENCION_IVA);
14    $("#P231_EFECTIVO").append(val.EFECTIVO);
15    $("#P231_CHEQUE").append(val.CHEQUE);
16  });
17  $("#dialog").dialog("close");
  });});

```

**Tabla 9:** Ejemplo de código JavaScript que implementa AJAX con sintaxis jQuery.

## 2.11 El servidor de aplicaciones GlassFish

### 2.11.1 Acerca de GlassFish

El origen de GlassFish se remontan a los primeros días de Tomcat<sup>20</sup> cuando la empresa Sun<sup>21</sup> dono esta tecnología a Apache<sup>22</sup>. Desde entonces, Sun ha reutilizado Tomcat en sus diferentes productos. En 2005, Sun creo el proyecto GlassFish, el objetivo inicial era producir un servidor de aplicaciones Java EE totalmente certificado. La parte del contenedor web de GlassFish tiene una gran cantidad de patrimonio de Tomcat, una aplicación que se ejecuta en Tomcat debería ejecutarse sin modificaciones en GlassFish. (Goncalves, 2010)

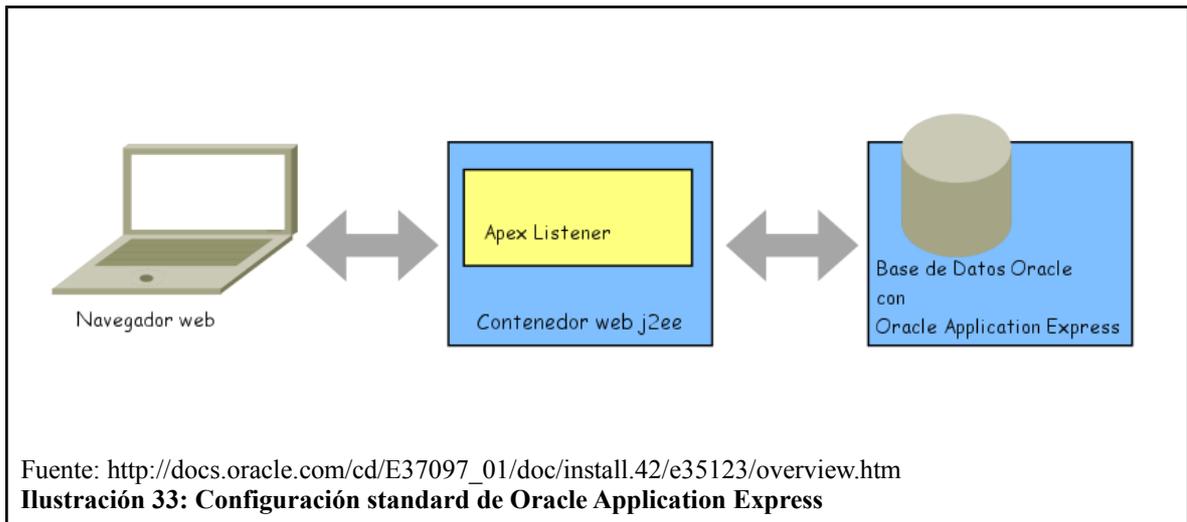
### 2.11.2 GlassFish y Oracle Application Express

En la **Ilustración 33** se representa la configuración standart para redes privadas, el Apex Listener es una aplicación que se instala en un servidor de aplicaciones j2ee GlassFish, el Apex Listener se encarga de la comunicación entre el navegador web de los usuarios y la base de datos donde se encuentra instalado el Oracle Application Express. (Oracle Application Express Installation Overview, 2013).

<sup>20</sup> Apache Tomcat funciona como un contenedor de servlets y JSP.

<sup>21</sup> Sun Microsystems fue una empresa informática que fue adquirida en el año 2009 por Oracle Corporation.

<sup>22</sup> Apache Software Foundation es una organización no lucrativa creada para dar soporte a los proyectos de software bajo la denominación Apache.



## CAPITULO III

### 3 Análisis del Sistema

#### 3.1 Metodología

En este capítulo se mostrara la metodología de desarrollo que sigue el proceso de desarrollo de software basado en la metodología ágil de Programación Extrema. *(Usero Ruiz & Valero Vergara, 2004)*

#### 3.2 Gestión del Proyecto

En esta sección se describe la planificación que ha seguido el proyecto a lo largo de su desarrollo, junto con la documentación asociada, incidentes del proceso y finalmente los diarios de actividades del equipo de programación. *(Usero Ruiz & Valero Vergara, 2004)*

#### 3.3 Planificación Inicial

Esta es la planificación de historias que se realizo al inicio del proyecto, tras estudiar el proyecto y mantener conversaciones con el patrocinador. De esta redacción inicial de historias de usuario se realizó una planificación inicial y posteriormente fue cambiada a lo largo del proyecto. Algunas de estas historias fueron eliminadas o cambiadas a lo largo del proyecto, a medida que cambiaban los requisitos del patrocinador o se tenía una concepción más clara del proyecto. *(Usero Ruiz & Valero Vergara, 2004)*

##### 3.3.1 Iteración 1

Historia de Usuario	
Número: 1	Usuario: Administrador
Nombre historia: Administración de Usuarios	
Prioridad en negocio: Alta (Alta / Media / Baja)	Riesgo en desarrollo: Media (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Creación, modificación y eliminación de un usuario para el acceso al sistema de facturación electrónica con la información de sus datos personales y contraseña.</p>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 10:** Administración de Usuarios.

Historia de Usuario	
Número: 2	Usuario: Administrador
Nombre historia: Administración de Módulos del Sistema	
Prioridad en negocio: Alta (Alta / Media / Baja)	Riesgo en desarrollo: Media (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Creación, modificación y eliminación de los módulos que tendrá el sistema para que se muestren en los menús de la navegación.</p>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 11:** Administración de Módulos del Sistema.

Historia de Usuario	
Número: 3	Usuario: Administrador
Nombre historia: Administración de Accesos al Sistema	
Prioridad en negocio: Alta (Alta / Media / Baja)	Riesgo en desarrollo: Media (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se muestra una pantalla donde el administrador puede escoger un usuario y otorga el permiso de acceso a la pantalla.</p>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 12:** Administración de Accesos al Sistema.

Historia de Usuario	
Número: 4	Usuario: Usuario Final
Nombre historia: Gestión de Clientes	
Prioridad en negocio: Alta (Alta / Media / Baja)	Riesgo en desarrollo: Media (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Creación, modificación y eliminación de un cliente con la información de sus datos personales.</p>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 13:** Gestión de Clientes.

Historia de Usuario	
Número: 5	Usuario: Usuario Final
Nombre historia: Administración de Información de Documentos Autorizados por el SRI	
Prioridad en negocio: Alta (Alta / Media / Baja)	Riesgo en desarrollo: Media (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se muestra una pantalla donde se maneja la información de los números de documentos autorizados por el SRI tales como:</p> <ul style="list-style-type: none"> <li>• Número de documento autorizado</li> <li>• Serie</li> <li>• Secuenciales</li> </ul>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 14:** Administración de Información de Documentos Autorizados por el SRI.

Historia de Usuario	
Número: 6	Usuario: Usuario Final
Nombre historia: Creación de Factura de Servicios	
Prioridad en negocio: Alta (Alta / Media / Baja)	Riesgo en desarrollo: Media (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se ingresa a la pantalla de facturación o ventas y se crea el documento el que es emitido a un cliente previamente creado en el sistema.</p>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 15: Creación de Factura de Servicios.**

Historia de Usuario	
Número: 7	Usuario: Usuario Final
Nombre historia: Eliminación y Anulación de Facturas	
Prioridad en negocio: Media (Alta / Media / Baja)	Riesgo en desarrollo: Baja (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se ingresa a la pantalla de eliminación facturación y se procede con la eliminación o anulación de la factura.</p>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 16: Eliminación y Anulación de Facturas.**

Historia de Usuario	
Número: 8	Usuario: Usuario Final
Nombre historia: Maestro de Facturas	
Prioridad en negocio: Baja (Alta / Media / Baja)	Riesgo en desarrollo: Media (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>En esta pantalla se puede consultar las facturas ya grabadas anteriormente.</p>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 17:** Maestro de Facturas.

Historia de Usuario	
Número: 9	Usuario: Usuario Final
Nombre historia: Mantenimiento de Lista de Precios	
Prioridad en negocio: Media (Alta / Media / Baja)	Riesgo en desarrollo: Media (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Es una pantalla donde se puede crear o actualizar los precios de los servicios que se van a facturar.</p>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 18:** Mantenimiento de Lista de Precios.

Historia de Usuario	
Número: 10	Usuario: Usuario Final
Nombre historia: Pantalla de Formas de Pago y Tipos de Formas de Pago	
Prioridad en negocio: Media (Alta / Media / Baja)	Riesgo en desarrollo: Media (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Es una pantalla donde se guardan las formas de pago que puede manejar.</p>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 19:** Pantalla de Formas de Pago y Tipos de Formas de Pago.

Historia de Usuario	
Número: 11	Usuario: Usuario Final
Nombre historia: Reportes de Ventas y Cierres de Caja	
Prioridad en negocio: Media (Alta / Media / Baja)	Riesgo en desarrollo: Media (Alta / Media / Baja)
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se ingresa a las pantallas de reportes y consulta la información en un rango de fechas inicial y final.</p>	
Observaciones: Confirmado por el patrocinador.	

**Tabla 20:** Reportes de Ventas y Cierres de Caja.

### 3.3.2 Iteración 2

Tarea	
Número tarea: 1	Número historia: 2
Nombre tarea: Diseño de diagramas UML	
Tipo de tarea : Desarrollo	Puntos estimados: 10
Fecha inicio: 18 de agosto del 2012	Fecha fin: 18 de septiembre del 2012
Programador responsable: Jorge Quiguango	
Descripción: Se diseña los diagramas UML de casos de uso y diagramas de actividades para el mejor entendimiento del prototipo a desarrollar.	

**Tabla 21:** Diseño de diagramas UML.

Tarea	
Número tarea: 2	Número historia: 2
Nombre tarea: Diseño de la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 10
Fecha inicio: 19 de septiembre del 2012	Fecha fin: 29 de octubre del 2012
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se diseña la base de datos según el alcance especificado y se generan el archivo con los objetos de base de datos que se ejecutan en Oracle 11g XE.</p>	

**Tabla 22:** Diseño de la base de datos.

Tarea	
Número tarea: 3	Número historia: 2
Nombre tarea: Añadir nueva tabla a la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 30 de septiembre del 2012	Fecha fin: 30 de septiembre del 2012
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Nueva tabla en la base de datos, requerida para el ingreso de localidades donde irá la provincia y ciudad.</p>	

**Tabla 23:** Añadir nueva tabla a la base de datos.

Tarea	
Número tarea: 4	Número historia: 2
Nombre tarea: Añadir nueva tabla a la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 20 de octubre del 2012	Fecha fin: 20 de octubre del 2012
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Nueva tabla en la base de datos, requerida para el ingreso de la información de los datos de la empresa o persona, como el RUC, dirección.</p>	

**Tabla 24:** Añadir nueva tabla a la base de datos.

### 3.3.3 Iteración 3

Tarea	
Número tarea: 1	Número historia: 3
Nombre tarea: Creación de los menús de navegación.	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 31 de octubre del 2012	Fecha fin: 31 de octubre del 2012
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se crea un componente compartido llamado separador, este separador es el menú de navegación que aparecerá en las pantallas de la aplicación y que el usuario usara para navegar.</p>	

**Tabla 25:** Creación de los menús de navegación.

Tarea	
Número tarea: 2	Número historia: 3
Nombre tarea: Diseño de la interfaz	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 31 de octubre del 2012	Fecha fin: 31 de octubre del 2012
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Diseño de las interfaces primarias para el módulo de administración para las pantallas de datos de empresa, creación de módulos, creación de usuarios y creación de localidades. Se define los componentes a utilizar como navegación tipo árbol, pantallas modales y listas desplegables.</p>	

**Tabla 26:** Diseño de la interfaz.

Tarea	
Número tarea: 3	Número historia: 3
Nombre tarea: Desarrollo del módulo de seguridad y administración	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 1 de noviembre del 2012	Fecha fin: 1 de diciembre del 2012
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se diseña el módulo con las siguientes aplicaciones:</p> <ul style="list-style-type: none"> <li>• Administración de usuarios.</li> <li>• Administración de módulos.</li> <li>• Permisos de acceso a usuarios.</li> <li>• Parámetros.</li> <li>• Reportes de permisos de accesos.</li> </ul>	

**Tabla 27:** Desarrollo del módulo de seguridad y administración.

Tarea	
Número tarea: 4	Número historia: 3
Nombre tarea: Desarrollo del módulo de clientes	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 2 de diciembre del 2012	Fecha fin: 2 de enero del 2013
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se diseña el módulo con las siguientes aplicaciones:</p> <ul style="list-style-type: none"> <li>• Parámetros de localidades.</li> <li>• Administración de clientes.</li> <li>• Reportes de clientes.</li> </ul>	

**Tabla 28:** Desarrollo del módulo de clientes.

Tarea	
Número tarea: 5	Número historia: 3
Nombre tarea: Desarrollo del módulo de SRI	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 3 de enero del 2013	Fecha fin: 31 de enero del 2013
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se diseña el módulo con las siguientes aplicaciones:</p> <ul style="list-style-type: none"> <li>• Parámetros de tipos de documentos.</li> <li>• Parámetros de series de documentos autorizados.</li> <li>• Parámetros de impuestos.</li> </ul>	

**Tabla 29:** Desarrollo del módulo de SRI.

Tarea	
Número tarea: 6	Número historia: 3
Nombre tarea: Desarrollo del módulo de ventas	
Tipo de tarea : Desarrollo	Puntos estimados: 3
Fecha inicio: 1 de febrero del 2013	Fecha fin: 28 de febrero del 2013
Programador responsable: Jorge Quiguango	
<p>Descripción:</p> <p>Se diseña el módulo con las siguientes aplicaciones:</p> <ul style="list-style-type: none"> <li>• Administrar Servicios o Productos</li> <li>• Administrara precios de servicios</li> <li>• Crear documento de venta</li> </ul>	

**Tabla 30:** Desarrollo del módulo de ventas.

Tarea	
Número tarea: 7	Número historia: 3
Nombre tarea: Desarrollo de la generación del documento XML de venta	
Tipo de tarea : Desarrollo	Puntos estimados: 8
Fecha inicio: 1 de marzo del 2013	Fecha fin: 31 de marzo del 2013
Programador responsable: Jorge Quiguango	
Descripción: Para emitir el comprobante digital se realizara: <ul style="list-style-type: none"> <li>• Generación del documento en formato XML</li> <li>• Firma digital del documento XML</li> </ul>	

**Tabla 31:** Desarrollo de la generación del documento XML de venta.

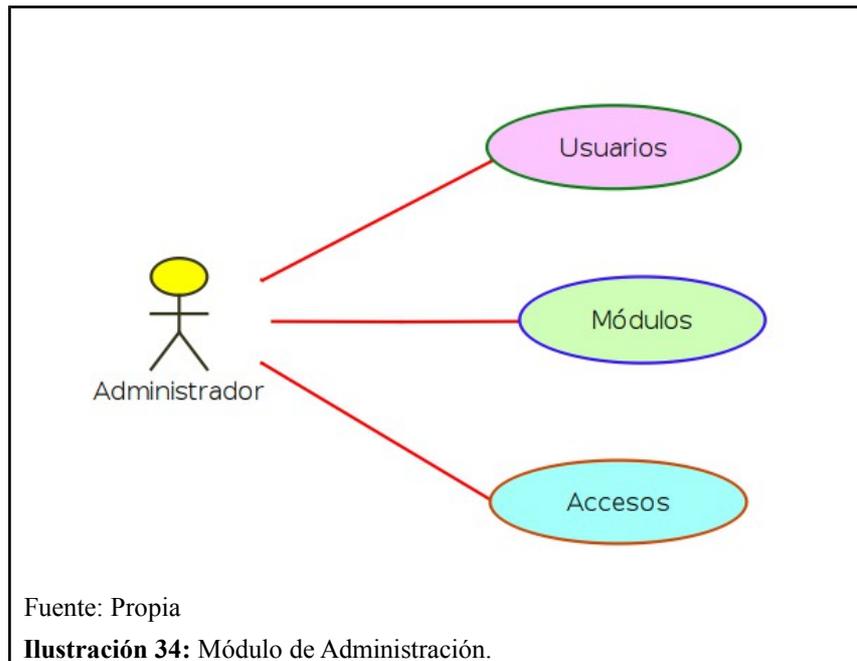
### 3.4 Diagramas UML

#### 3.4.1 Diagramas de Casos de Uso

En esta sección se ilustra y describen los casos de uso que fueron creados a partir de las especificaciones del sistema prototipo a desarrollar.

##### 3.4.1.1 Módulo de Administración

Los siguientes casos de uso describen el módulos de administración y seguridad del sistema.



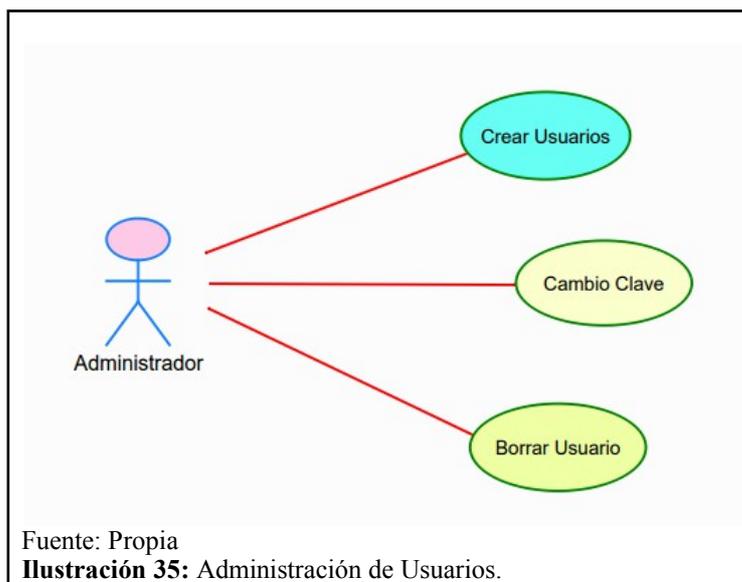
A continuación se describen los casos de uso:

Caso de Uso	Descripción de Caso de Uso
Usuarios	Se encarga de gestionar a los usuarios de sistema
Módulos	Se encarga de gestionar a los módulos del sistema
Accesos	Se encarga de gestionar a los accesos de los usuarios a los módulos del sistema

**Tabla 32:** Descripción caso de uso del módulo de administración.

#### a) Administración de Usuarios

Los siguientes casos de uso describen la función de gestión de usuarios del sistema.



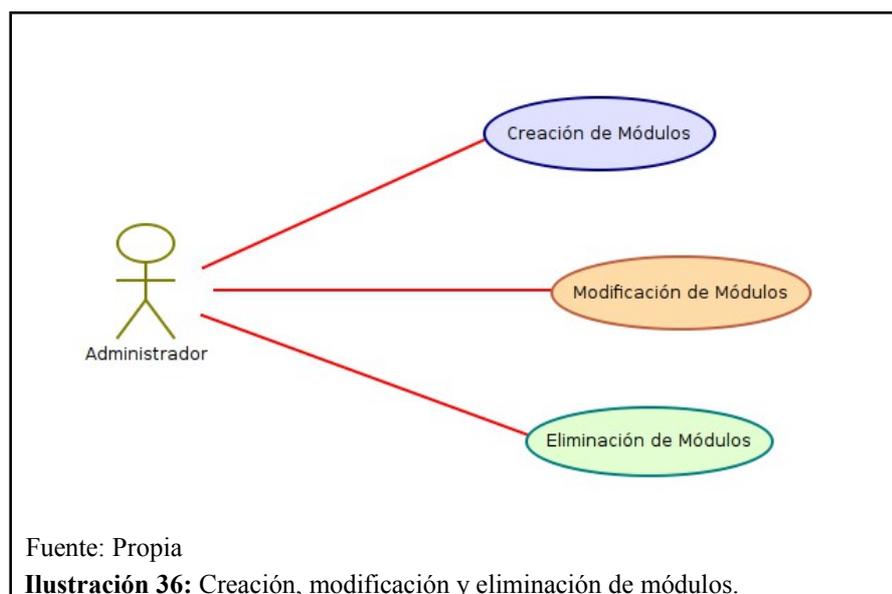
A continuación se describen los casos de uso:

Caso de Uso	Descripción de Caso de Uso
Crear Usuarios	Se encarga de la creación del usuario en el sistema
Cambio Clave	Se encarga de el cambio de clave de un usuario
Borrar Usuario	Se encarga de la eliminación del usuario

**Tabla 33:** Descripción caso de uso de administración de usuarios.

### b) Creación, modificación y eliminación de módulos

Los siguientes casos de uso describen la función de gestión de módulos del sistema.



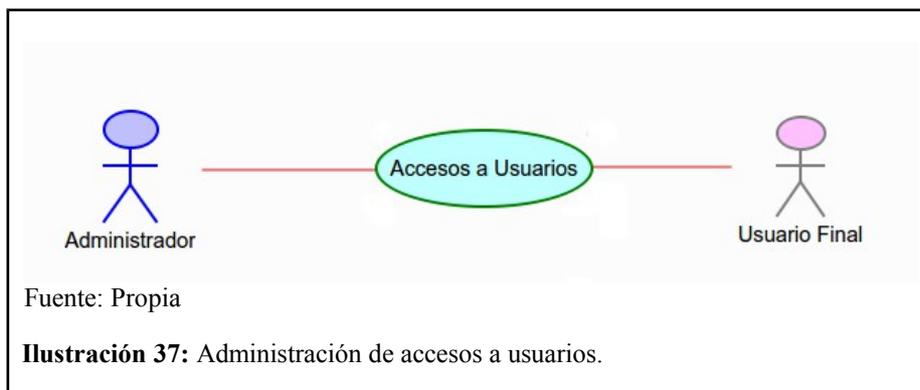
A continuación se describen los casos de uso:

Caso de Uso	Descripción de Caso de Uso
Creación de Módulos	Se encarga de la creación de un módulos en el sistema o una pantalla con el enlace de acceso a la misma
Modificación de Módulos	Se encarga de la actualización de la información del módulo o pantalla
Eliminación de Módulos	Se encarga de la eliminación del módulo o pantalla

**Tabla 34:** Descripción caso de uso de creación, modificación y eliminación de módulos.

### c) Administración de accesos a usuarios

El siguiente caso de uso describe la función de la gestión de accesos a los usuarios del sistema.



A continuación se describen los casos de uso:

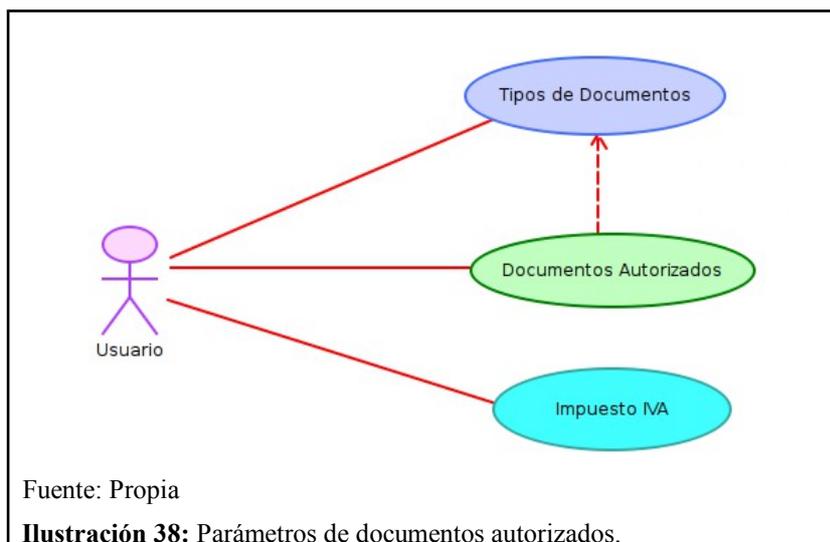
Caso de Uso	Descripción de Caso de Uso
Accesos a Usuarios	Se encarga de gestionar los accesos a los usuarios del sistema a los diferentes módulos y pantallas

**Tabla 35:** Descripción caso de uso de administración de accesos a usuarios.

### 3.4.1.2 Módulo de SRI

#### a) Parámetros de Documentos Autorizados

Los siguientes casos de uso describen la función de definición de parámetros para los documentos autorizados del sistema.



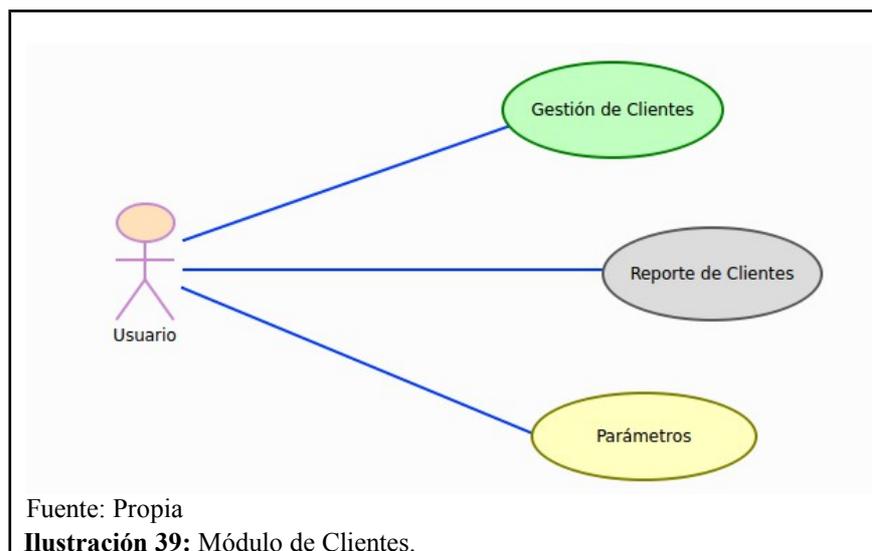
A continuación se describen los casos de uso:

Caso de Uso	Descripción de Caso de Uso
Tipos de documentos	Se encarga de la gestión de los tipos de documentos a grabar, pueden ser facturas, notas de crédito.
Documentos Autorizados	Se encarga de la gestión de los parámetros definidos por el Servicio de Rentas Internas que debe contener el documento emitido
Impuesto IVA	Se encarga de definir el parámetro de impuesto IVA

**Tabla 36:** Descripción de caso de uso parámetros de documentos autorizados.

### 3.4.1.3 Módulo de Clientes

Los siguientes casos de uso describen el módulos de clientes.



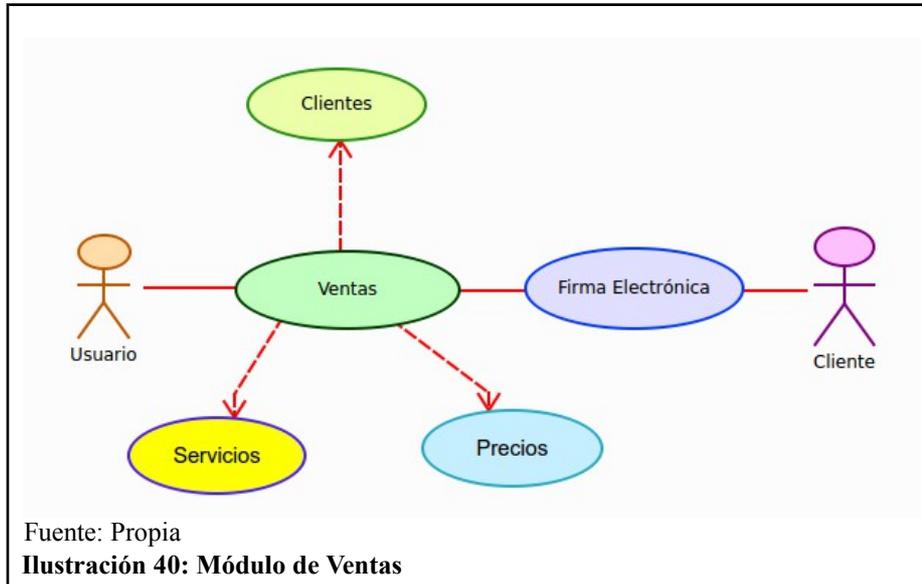
A continuación se describen los casos de uso:

Caso de Uso	Descripción de Caso de Uso
Gestión de Clientes	Se encarga de la gestión de los clientes
Reporte de Clientes	Se encarga de generar un reporte de los clientes
Parámetros	Se encarga de la definición de los parámetros necesarios para la creación de clientes

**Tabla 37:** Descripción caso de uso del módulo de clientes.

### 3.4.1.4 Módulo de Ventas

Los siguientes casos de uso describen el módulos de ventas.



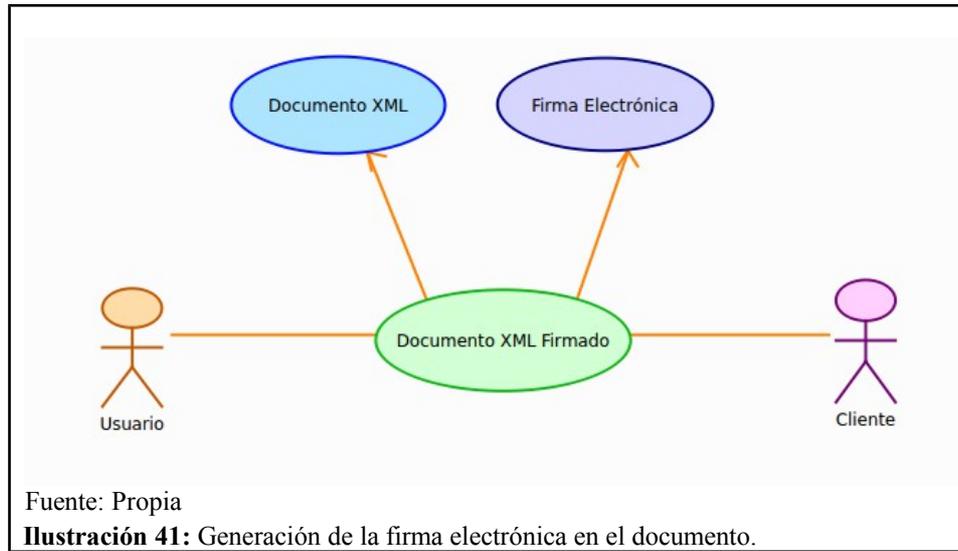
A continuación se describen los casos de uso:

Caso de Uso	Descripción de Caso de Uso
Clientes	Se encarga de la búsqueda del cliente en la aplicación de ventas
Servicios	Se encarga de la gestión de servicios como un producto.
Precios	Se encarga del mantenimiento de los precios del servicio o producto
Ventas	Se encarga de generar la transacción de venta y generar el comprobante en formato XML
Firma Electrónica	Se encarga de firmar el documento XML

**Tabla 38:** Descripción de caso de uso del módulo de ventas.

### a) Generación de la firma electrónica en el documento de venta

Los siguientes casos de uso describen la aplicación de firmar al documento



A continuación se describen los casos de uso:

Caso de Uso	Descripción de Caso de Uso
Documento XML	Se encarga de generar el documento de venta en XML de acuerdo con las especificaciones del Servicio de Rentas Internas
Firma Electrónica	Se encarga de la comunicación del software con el Token
Documento XML Firmado	Se encarga de firmar el documento con la firma digital.

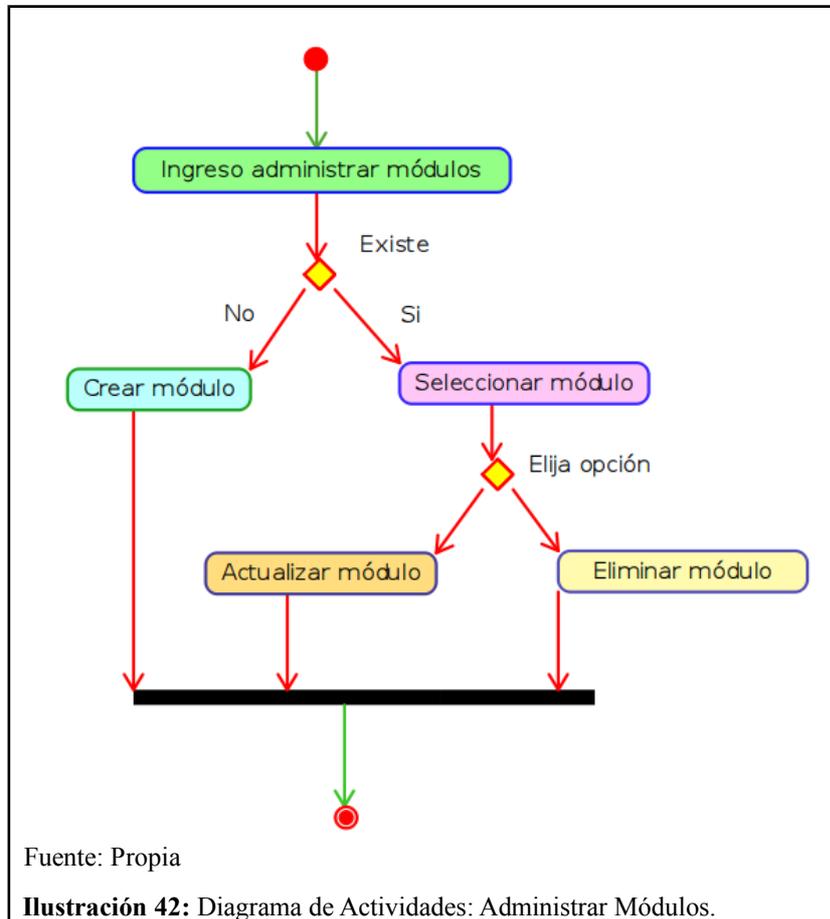
**Tabla 39:** Descripción de caso de uso de generación de la firma electrónica en el documento de venta.

### 3.4.2 Diagramas de Actividades

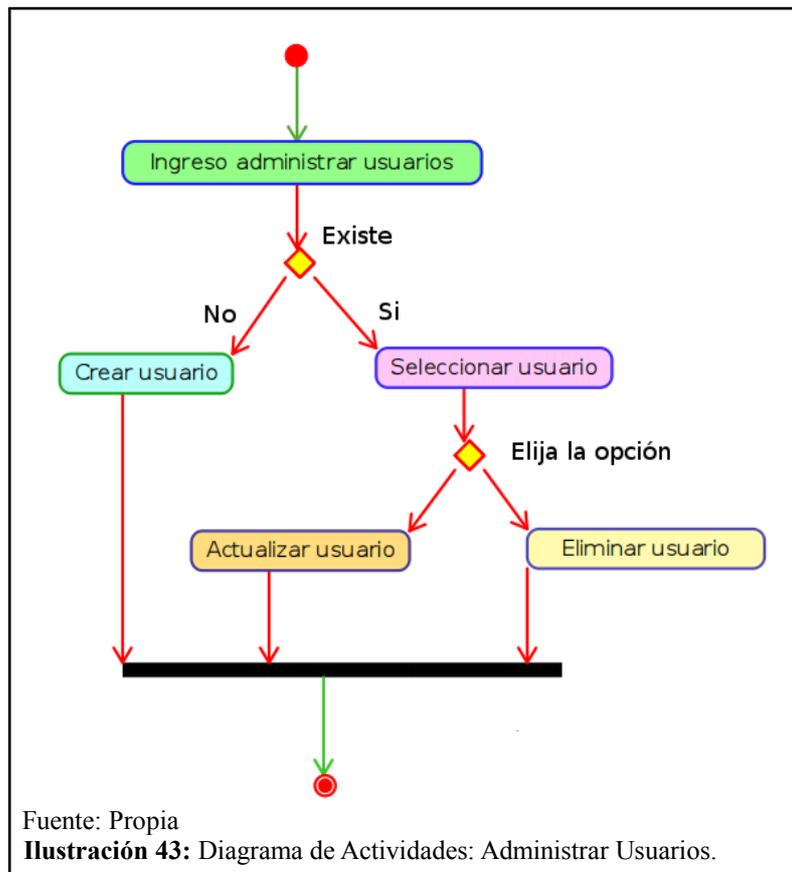
En esta sección se ilustra y describen los diagramas de actividades que fueron creados a partir de las especificaciones del sistema prototipo a desarrollar.

### 3.4.2.1 Módulo de Administración

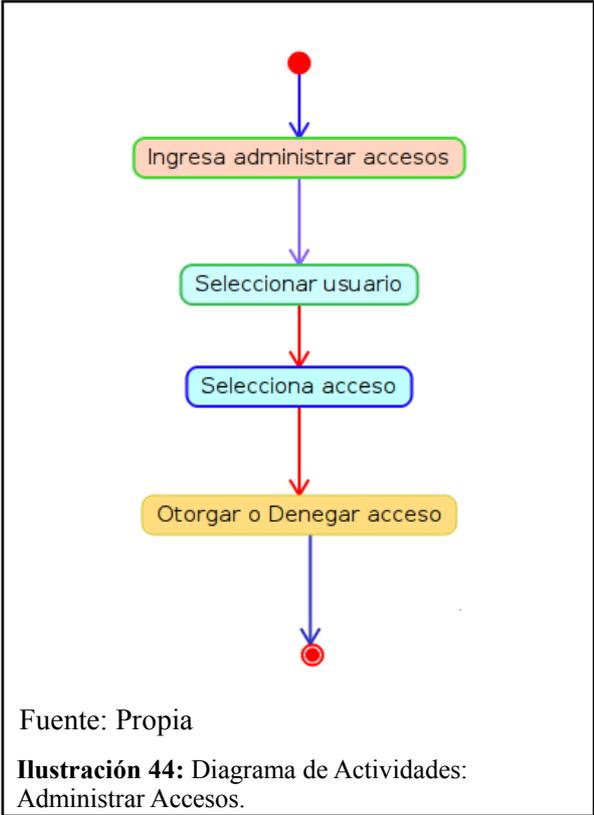
#### a) Administrar Módulos



## b) Administrar Usuarios

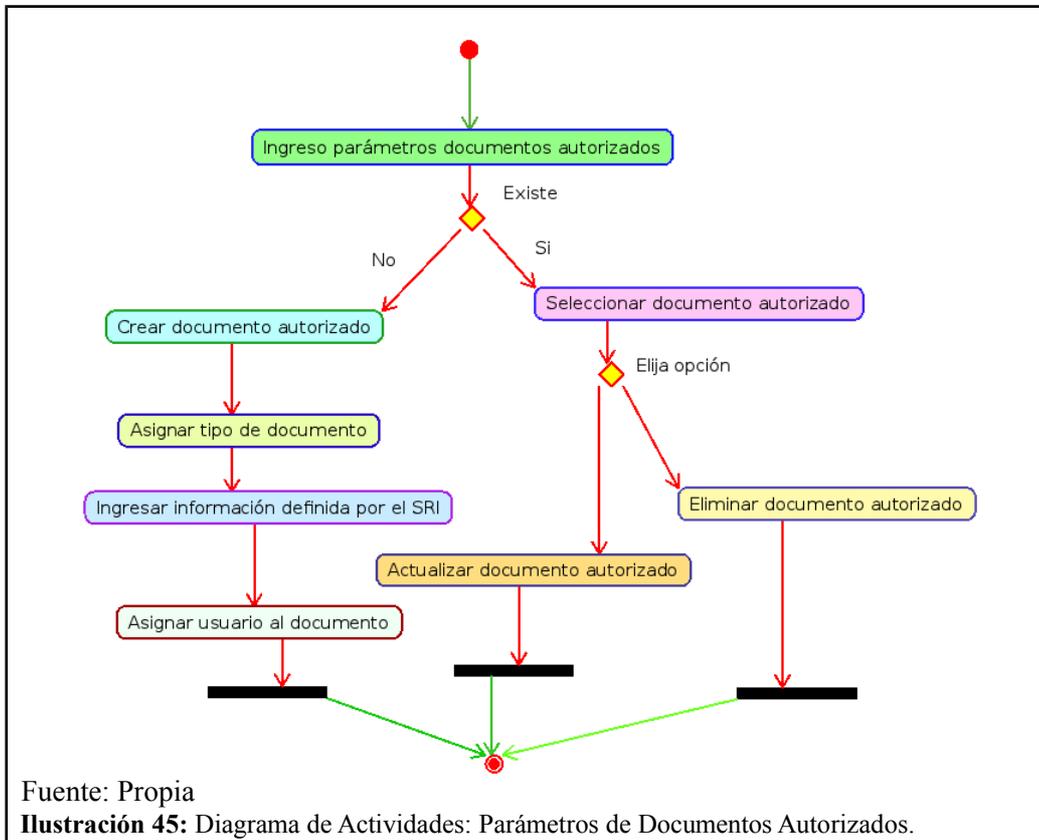


c) Administrar Accesos



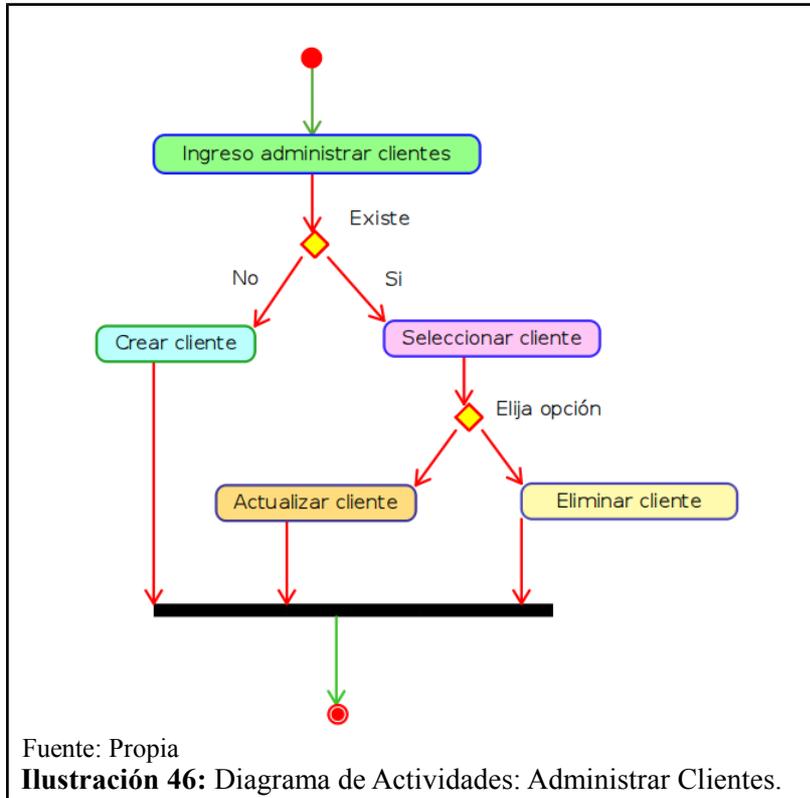
### 3.4.2.2 Módulo de SRI

#### a) Parámetros de Documentos Autorizados



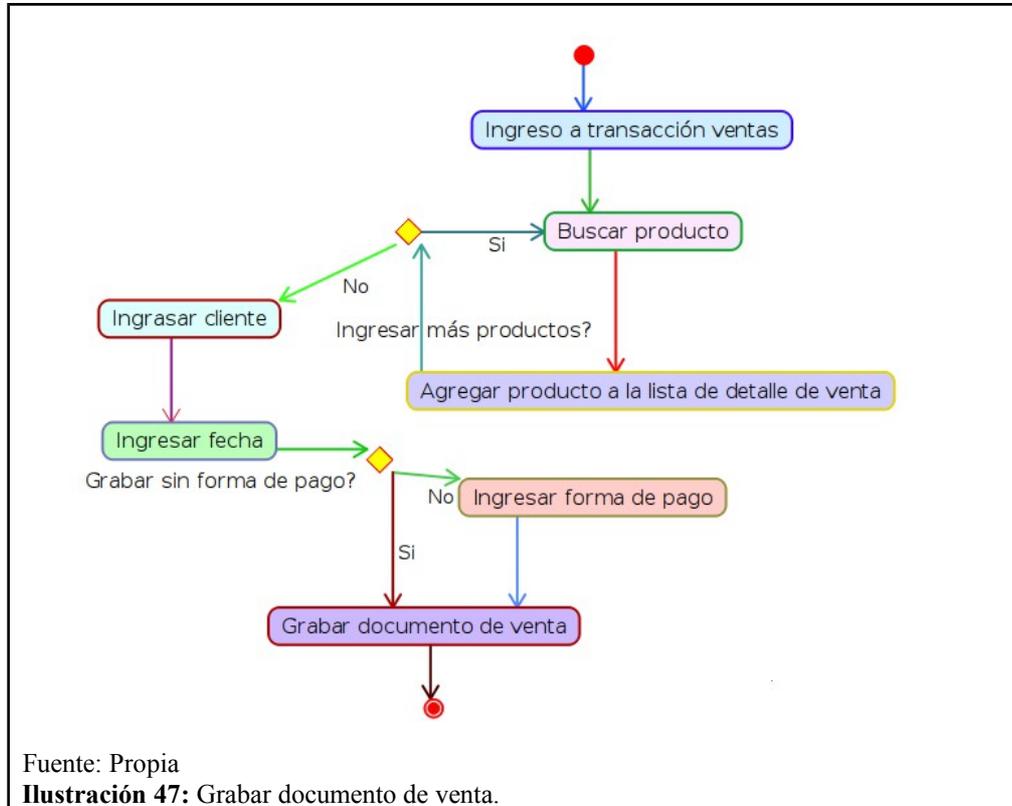
### 3.4.2.3 Módulo de Clientes

#### a) Administrar Clientes

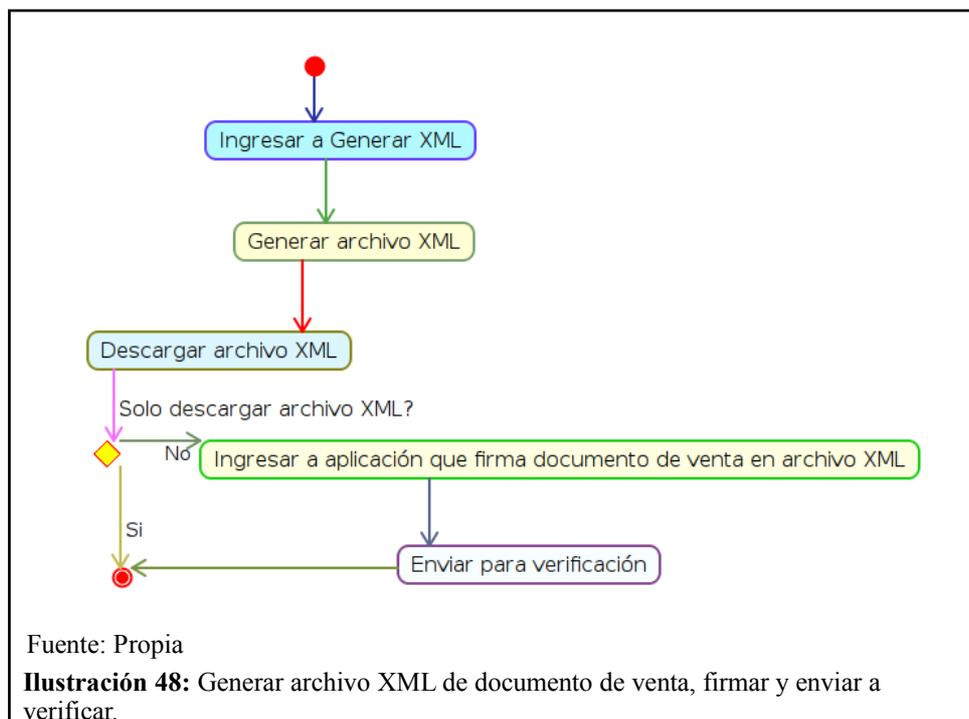


### 3.4.2.4 Módulo de Ventas

#### a) Grabar documento de venta



#### b) Generar archivo XML de documento de venta, firmar y enviar a verificar



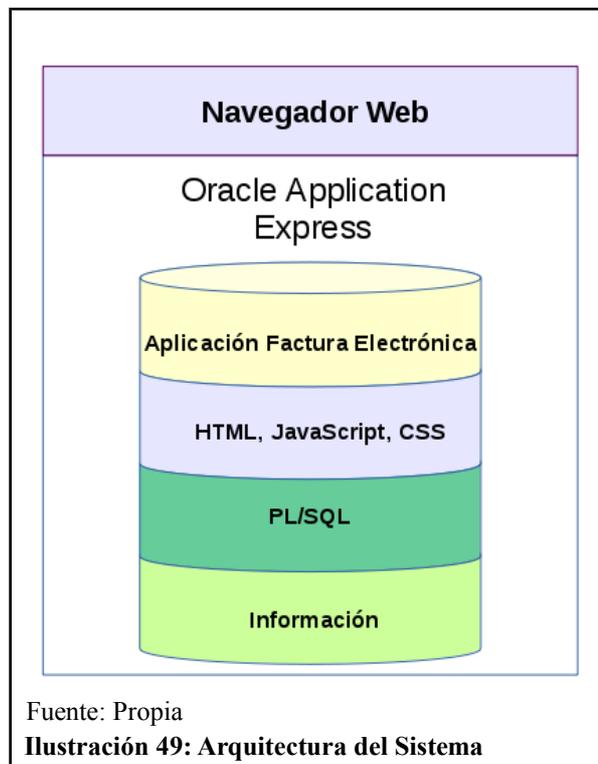
## CAPITULO IV

### 4 Diseño del Sistema

#### 4.1 Arquitectura del sistema

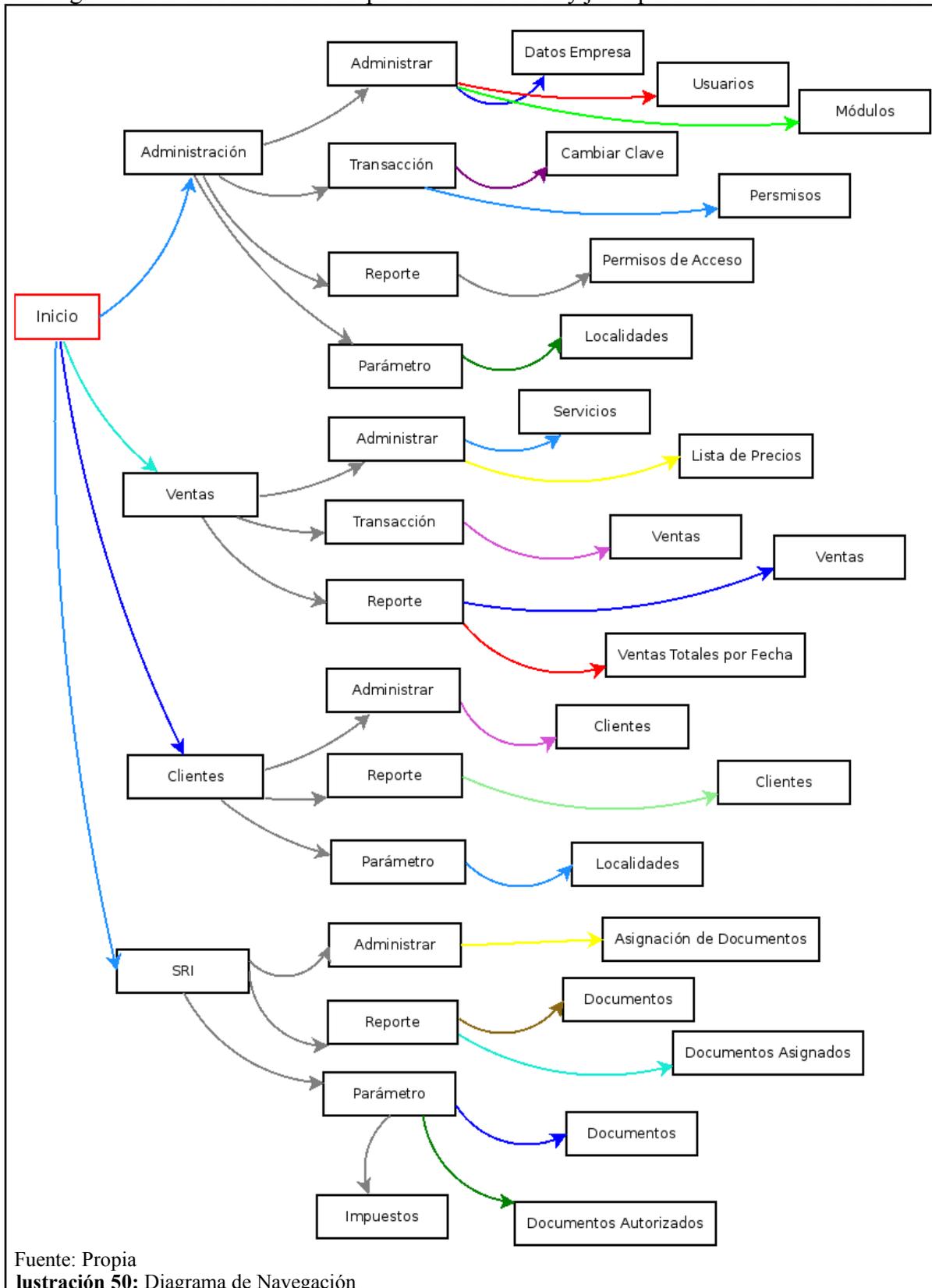
En esta sección se describe el diseño arquitectónico de la aplicación. El Oracle Application Express está contenido en la base de datos lo que significa que todo lo que se desarrolle se guardará en la base de datos incluidas las imágenes que mostremos en la aplicación desarrollada. Para desplegar este contenido en el navegador web, Oracle Application Express se encarga de ejecutar el programa en la base de datos y traer la información al navegador web con los programas que se ejecuten del lado del navegador en este caso los programas JavaScript.

En el diagrama de la **Ilustración 49** describe las capas de los componentes que son parte de la aplicación.



## 4.2 Diagrama de Navegación

El diagrama de la **Ilustración 50** representa la relación y jerarquía de la interfaz de usuario.







#### 4.5 Pantallas principales del Prototipo de Facturación Electrónica

Las siguientes pantallas son parte de la aplicación desarrollada en Oracle Application Express. En la **Ilustración 53** se muestra la pantalla de otorgar acceso a los usuarios del sistema.

Fuente: Propia

**Ilustración 53:** Pantalla para otorgar acceso a los usuarios

En la Ilustración 54 se muestra la validación de los elementos en la grabación de la factura electrónica.

Fuente: Propia

**Ilustración 54:** Validación de elementos en pantalla de grabar documento de venta.

#### 4.6 Sitio web con para distribución del sistema prototipo de facturación electrónica

El sitio web se crea con el propósito de una libre distribución del sistema prototipo, para que cualquier persona pueda descargar, modificar o distribuir la aplicación. La intención es calificarlo con una licencia de software libre.

El sitio web donde se puede encontrar este prototipo, y la documentación es:

[www.quijotelu.com](http://www.quijotelu.com)

## CAPITULO V

### 5 Sitio Web

Para la publicación de los fuentes de las aplicaciones desarrolladas, se construyo un sitio web en Drupal<sup>23</sup>, su dirección es [www.quijotelu.com](http://www.quijotelu.com), desde esté sitio se puede descargar y distribuir con licencia GPL, a continuación se describen los archivos publicados:

- Código fuente de la aplicación para firmar los comprobantes electrónicos en formato XadES\_BES.
- Respaldo de la base de datos, contienen las tablas, vistas, funciones, procedimiento y paquetes.
- Respaldo de la aplicación desarrollada en Oracle Application Express versión 4.
- Código fuente de la aplicación que se comunica con los servicios web del Servicio de Rentas Internas.

---

<sup>23</sup> Marco de gestión de contenidos, cuya información se almacena en una base de datos y se maneja usando un entorno web.

## 6 CONCLUSIONES Y RECOMENDACIONES

### 6.1 Conclusiones

- La herramienta Oracle Application Builder, es fácil de aprender y usar tanto para los desarrolladores con experiencia como para los desarrolladores novatos. Es una buena herramienta para comenzar a aprender y desarrollar aplicaciones web, no necesita conocimientos más allá del lenguaje de consulta SQL y el lenguaje de programación PL/SQL.
- La interfaz gráfica para el usuarios final, en la versión 4, se la construye con la idea de bloques que contienen elementos. Es una construcción de la interfaz de forma que el desarrollador no necesita mayor conocimiento de HTML, una ventaja para los que inician ya que el tiempo que se toma en construir la interfaz es significativo, pero una limitación con forme se requiere de pantallas más complejas.
- En cuanto al prototipo realizado se buscó una aplicación que ayude a un segmento de la sociedad económicamente activa, con un tema actual como la facturación electrónica.
- Para la puesta en producción de las aplicaciones desarrolladas en Oracle Application Express, solo se necesita de la base de datos y un contenedor de aplicaciones Java, puede trabajar en cualquier sistema operativo y las herramientas son de libre distribución, en sistemas Linux es donde se encontró mayor fluidez en comparación con windows, como apreciación en el desarrollo de aplicaciones.
- Al levantar la infraestructura de Oracle Application Express en el servidor de aplicaciones Glassfish, se puede escalar la aplicación o las aplicaciones con la tecnología Java. Si es necesario, se puede desarrollar en Java lo que no se pueda o sea más complejo de hacer en Oracle Application Express e integrar las aplicaciones en un solo servidor, con la ayuda del servidor aplicaciones Glassfish.

### 6.2 Recomendaciones

- En cuanto a la inteligencia de negocio de la aplicación a desarrollar, se recomendaría que toda o casi toda la programación se lo haga directamente en la base de datos, de esta manera solo se referencia a las funciones, procedimientos o paquetes desde la interfaz web. El editor de programación de Oracle Application Builder es bastante malo en comparación a otros entornos de desarrollo, y es más fácil de realizar mantenimiento y corrección de errores de la aplicación si el código fuente es creado desde la base de datos y no desde Oracle Application Builder.
- Acerca de las técnicas avanzadas de desarrollo web, Oracle Application Express proporciona funciones que integra con AJAX y jQuery, importante para realizar aplicaciones más complejas y recomendable para desarrolladores más avanzados. En tiempos que los dispositivos móviles tienen acceso a la web, por lo tanto a las aplicaciones que se desarrolla. Pero los dispositivos móviles tienen una debilidad, es su autonomía de energía. Las tecnologías de software han evolucionado a favor para que el contenido web se presente mucho más rápido y con el uso de menos energía de estos dispositivos, AJAX y jQuery ayudan a que la página web no tenga que recargarse nuevamente y solo transferir la información necesaria, lo que se

traduce en funcionamiento óptimo y menos consumo de energía, bueno para los exigentes usuarios finales, pero para los desarrolladores una tarea difícil.

## 7 ANEXOS

### 7.1 Descarga, instalación y configuración de la base de datos Oracle 11G XE

Para esta instalación y configuración se trabajara con sistemas operativos basados en RedHat, como Fedora, CENTOS o Scientific Linux con arquitectura de 64 bits.

a) El siguiente enlace es donde se encuentra el instalador de la base de datos: <http://www.oracle.com/technetwork/database/database-technologies/express-edition/downloads/index.html>.

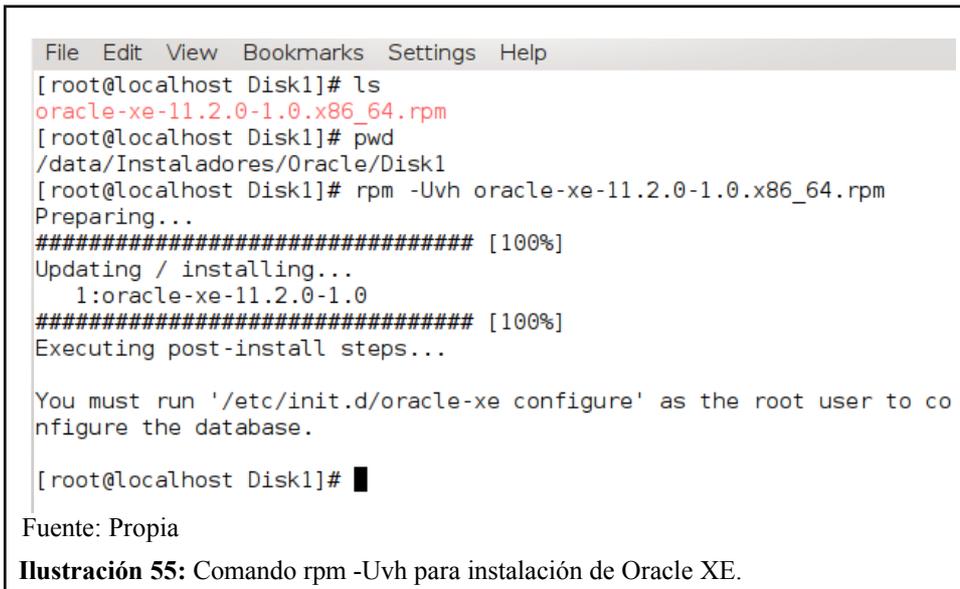
b) Se acepta el acuerdo de licencia.

c) Se descarga **Oracle Database Express Edition 11g Release 2 for Linux x64**.

d) Se graba en el computador el archivo **oracle-xe-11.2.0-1.0.x86\_64.rpm.zip**.

e) Se extrae el archivo .zip, dentro de la carpeta Disk1 se encuentra el instalador.

f) Se ejecuta el comando **rpm -Uvh oracle-xe-11.2.0-1.0.x86\_64.rpm**, como se muestra en la **Ilustración 55**.



```
File Edit View Bookmarks Settings Help
[root@localhost Disk1]# ls
oracle-xe-11.2.0-1.0.x86_64.rpm
[root@localhost Disk1]# pwd
/data/Instaladores/Oracle/Disk1
[root@localhost Disk1]# rpm -Uvh oracle-xe-11.2.0-1.0.x86_64.rpm
Preparing...
##### [100%]
Updating / installing...
 1:oracle-xe-11.2.0-1.0
##### [100%]
Executing post-install steps...

You must run '/etc/init.d/oracle-xe configure' as the root user to co
nfigure the database.

[root@localhost Disk1]# █
```

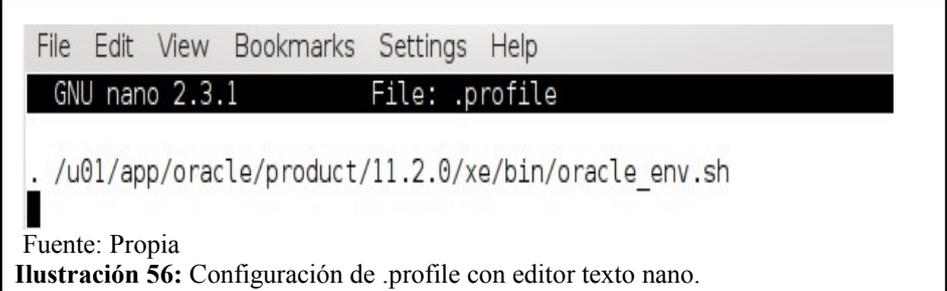
Fuente: Propia

**Ilustración 55:** Comando rpm -Uvh para instalación de Oracle XE.

g) Se ejecuta el comando **/etc/init.d/oracle-xe configure**, se elige las opciones de configuración por defecto y se especifica una contraseña.

h) Después de la instalación y configuración de Oracle 11g XE, se crea automáticamente un usuario oracle, desde el usuario root se ingresa con el comando **su - oracle** y se crea el archivo **.profile** con la siguiente instrucción **nano .profile** en este caso se creo el archivo con el editor **nano**.

i) Para configurar las variables de la base de datos Oracle 11g XE ingresamos lo siguiente en el archivo **./u01/app/oracle/product/11.2.0/xe/bin/oracle\_env.sh** y grabar, se muestra en la **Ilustración 56**.



```
File Edit View Bookmarks Settings Help
GNU nano 2.3.1 File: .profile

./ /u01/app/oracle/product/11.2.0/xe/bin/oracle_env.sh

```

Fuente: Propia

**Ilustración 56:** Configuración de .profile con editor texto nano.

## 7.2 Descarga, instalación y configuración de Oracle Application Express 4

Para la instalación de Oracle Application Express se tiene iniciada la base de datos Oracle 11g XE.

a) El siguiente enlace es donde se encuentra el instalador:

<http://www.oracle.com/technetwork/developer-tools/apex/downloads/index.html>.

b) Se acepta el acuerdo de licencia.

c) Descargar **Oracle Application Express 4.x - All languages**.

d) Se graba en el computador el archivo **apex\_4.x.zip**.

e) Se crea una carpeta con usuario root **mkdir /app** y se otorga privilegios a dicha carpeta con la siguiente instrucción **chmod 777 /app**, en esta carpeta se extrae el contenido del archivo .zip.

f) Se ingresa con usuario desde roor **su – oracle**, se accede a la carpeta apex **cd /app/apex/**, a continuación se ingresa a sqlplus como sysdba con el comando **sqlplus / as sysdba**, en la **Ilustración 57** se muestra el contenido de la carpeta y el comando ejecutado.

```
File Edit View Bookmarks Settings Help
[root@localhost apex]# su - oracle
-bash-4.2$ cd /app/apex/
-bash-4.2$ ls
apex_epg_config_con.sql      apxdvins.sql                apxrtins.sql
apex_epg_config_core.sql    apxe101.sql                 apxsqler.sql
apex_epg_config.sql         apxe102.sql                 apxxemig.sql
apexins_con.sql             apxe111.sql                 apxxepwd.sql
apexins.sql                 apxe112.sql                 builder
apex_rest_config_con.sql    apxexit.sql                 catapx.sql
apex_rest_config_core.sql   apxldimg_con.sql           core
apex_rest_config.sql        apxldimg.sql                coreins.sql
apexvalidate.sql           apxpatch.sql                devins.sql
appins.sql                 apxprereq.sql               endins.sql
apxchpwd.sql               apxrelod.sql                images
apxconf.sql                apxremov1.sql               load_trans.sql
apxdbmig.sql               apxremov2.sql               owa
apxdevrm_con.sql           apxremov_con.sql           utilities
apxdevrm.sql               apxremov.sql
apxdvins_con.sql           apxrtins_con.sql
-bash-4.2$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.2.0 Production on Wed Oct 2 17:39:06 2013

Copyright (c) 1982, 2011, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Pro
duction

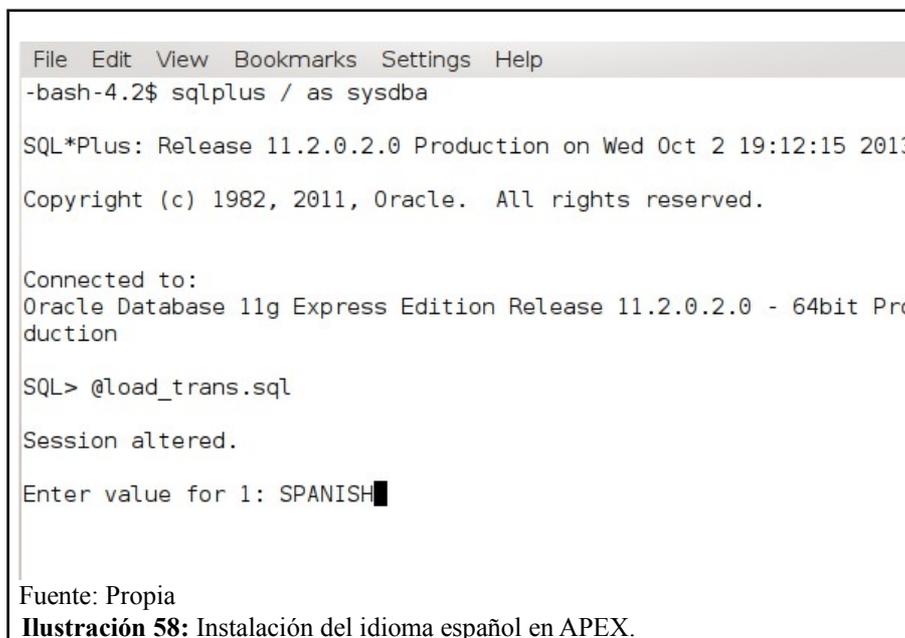
SQL> █
```

Fuente: Propia

**Ilustración 57:** Contenido de carpeta apex.

g) Se instala el entorno de desarrollo completo con la siguiente instrucción **@apexins SYSAUX SYSAUX TEMP /i/**

h) Para instalar el idioma español se ejecuta **@load\_trans**, y se ingresa el idioma **SPANISH**, como se muestra en la **Ilustración 58**.



```
File Edit View Bookmarks Settings Help
-bash-4.2$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.2.0 Production on Wed Oct 2 19:12:15 2013

Copyright (c) 1982, 2011, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Pro
duction

SQL> @load_trans.sql

Session altered.

Enter value for 1: SPANISH
```

Fuente: Propia  
**Ilustración 58:** Instalación del idioma español en APEX.

i) Para cambiar la contraseña de admin, se ejecuta en sqlplus **@apxchpwd**.

j) Para habilitar el usuario **APEX\_PUBLIC\_USER**, se ejecuta en sqlplus **ALTER USER APEX\_PUBLIC\_USER ACCOUNT UNLOCK;** y **ALTER USER APEX\_PUBLIC\_USER IDENTIFIED BY 'nueva\_password';** para asignar una contraseña.

### 7.3 Descarga, instalación y configuración de Oracle APEX Listener.

La arquitectura de Oracle Application Express requiere de algún tipo de servidor Web para las solicitudes entre el navegador web y el motor de Oracle Application Express, se instalará y configurará este componente en modo desarrollador.

a) El siguiente enlace es donde se encuentra el instalador:

<http://www.oracle.com/technetwork/developer-tools/rest-data-services/downloads/index.html>.

b) Se acepta el acuerdo de licencia.

c) Se descarga **Oracle APEX Listener 2.0.5**.

d) Se graba en el computador el archivo **apex\_listener.2.0.5.287.04.27.zip**.

e) En **/app** se crea una carpeta llamada **apex\_listener** y se extrae el contenido del archivo **.zip**.

f) Es necesario tener instalado el JDK, si no es así, se accede a la siguiente dirección <http://www.oracle.com/technetwork/es/java/javase/downloads/index.html>.

g) Se acepta el acuerdo de licencia.

h) Se guarda el archivo, en este caso se descargó el archivo con extensión rpm.

i) Se instala con el comando `rpm -Uvh jdk-7u40-linux-x64.rpm`, como se muestra en la **Ilustración 59** y `rpm -qpl jdk-7u40-linux-x64.rpm` para verificar la ruta de instalación.

```
File Edit View Bookmarks Settings Help
[root@localhost java]# rpm -Uvh jdk-7u40-linux-x64.rpm
Preparing...
##### [100%]
Updating / installing...
 1:jdk-2000:1.7.0_40-fcs
##### [100%]
Unpacking JAR files...
  rt.jar...
  jsse.jar...
  charsets.jar...
  tools.jar...
  localedata.jar...
  jfxrt.jar...
[root@localhost java]#
```

Fuente: Propia  
**Ilustración 59:** Instalación de Java 7.

j) Para configurar el jdk en el entorno de usuario, se edita el archivo `.bash_profile` y se añade `JAVA_HOME=/usr/java/jdk1.7.0_40; export JAVA_HOME` y `PATH=/usr/java/jdk1.7.0_40/bin/:$PATH; export PATH`, en la **Ilustración 60** se representa.

```
File Edit View Bookmarks Settings Help
GNU nano 2.3.1 File: .bash_profile

PATH=$PATH:$HOME/.local/bin:$HOME/bin

export PATH

#JAVA
JAVA_HOME=/usr/java/jdk1.7.0_40; export JAVA_HOME
PATH=/usr/java/jdk1.7.0_40/bin/:$PATH; export PATH
█
```

Fuente: Propia  
**Ilustración 60:** Configuración de entorno Java.

k) Se ejecuta `java -jar apex.war` dentro de la carpeta `/app/apex_listener`.

l) Se ingresa como directorio de configuración, si el asistente lo requiere.

m) Se ingresa el nombre del servidor de base de datos, en este caso `localhost`.

n) Se ingresa por defecto el puerto `1521` que es el puerto de la base de datos Oracle 11g XE.

ñ) Se ingresa el nombre del servicio de base de datos, en este caso es `XE`.

- o) Se ingresa el usuario de base de datos, por defecto **APEX\_PUBLIC\_USER** y la contraseña para **APEX\_PUBLIC\_USER**, la misma que se utilizó en la configuración de Oracle Application Express.
- p) Para las contraseñas de los usuarios **APEX\_LISTENER** y **APEX\_REST\_PUBLIC\_USER**, elegimos la opción número 2, que es la misma que el usuario **APEX\_PUBLIC\_USER**, pero se puede ingresar otra.
- q) En el modo de servidor, se elige la opción número 1 que es **standalone mode**, este modo es para trabajar como desarrollador, no recomendado para producción.
- r) Para la ubicación de los recursos estáticos de APEX, se elige **/app/apex/images**
- s) Se ingresa el puerto 8585, pero se puede ingresar cualquier otro.
- t) Al final el **APEX Listener** inicia correctamente mostrando esta dirección **http://localhost:8585/apex** al final, como se muestra en la **Ilustración 61**.

```

File Edit View Bookmarks Settings Help
log.maxEntries=50
misc.compress=
misc.defaultPage=apex
security.disableDefaultExclusionList=false
security.maxEntries=2000
db.username=APEX_PUBLIC_USER
Using JDBC driver: Oracle JDBC driver version: 11.2.0.3.0
Oct 02, 2013 9:50:00 PM oracle.dbtools.rt.web.SCListener contextIn
itialized
INFO: Oracle Application Express Listener initialized
Application Express Listener version : 2.0.4.259.08.57
Application Express Listener server info: Grizzly/1.9.49

Oct 02, 2013 9:50:00 PM com.sun.grizzly.Controller logVersion
INFO: GRIZZLY0001: Starting Grizzly Framework 1.9.49 - 10/2/13 9:5
0 PM
Oct 02, 2013 9:50:00 PM oracle.dbtools.standalone.Standalone execu
te
INFO: http://localhost:8585/apex started.
■
Fuente: Propia

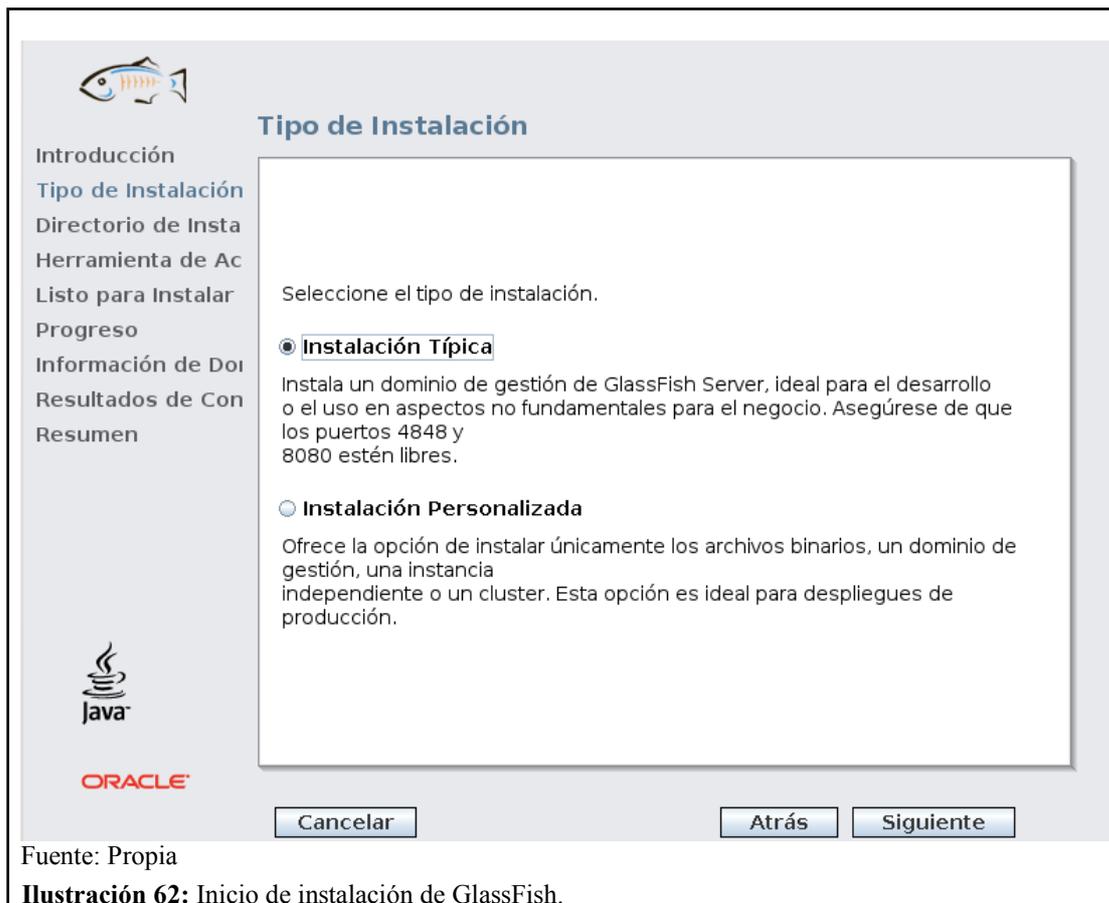
```

**Ilustración 61:** Apex Listener iniciado.

#### 7.4 Descarga, instalación y configuración de GlassFish

Para que una aplicación hecha en Oracle Application Express funcione en un entorno de producción, es necesario que se despliegue en un servidor de aplicaciones.

- a) El siguiente enlace es donde se encuentra el instalador:  
<http://www.oracle.com/technetwork/java/javaee/downloads/index.html>.
- b) Se acepta el acuerdo de licencia.
- c) Se descarga **Oracle GlassFish Server**
- d) Se graba en el computador el archivo **ogs-3.1.2.2-unix-ml.sh**.
- e) Para comenzar la instalación se ejecuta el siguiente comando **./ogs-3.1.2.2-unix-ml.sh**.
- f) Se despliega la siguiente pantalla que se muestra en la **Ilustración 62**, por defecto se elige instalación típica y se pulsa siguiente.



**g)** Se ingresa el directorio donde se instala GlassFish, es este caso en la dirección `/app/glassfish/`, y se pulsa siguiente.

**h)** Se activa las actualizaciones automáticas, y siguiente.

**i)** Se pulsa en el botón instalar, y siguiente.

**j)** En la **Ilustración 63** se muestra la configuración que se usa para el servidor, y se pulsa siguiente. Se tiene que verificar que los puertos no estén ocupados

Configure los valores de administración del servidor. Proporcione el nombre de usuario y la contraseña del servidor. Puede dejar las contraseñas vacías si desea configurar el servidor para conexiones no autenticadas.

Nombre de Dominio	<input type="text" value="domain"/>
Puerto de Administración	<input type="text" value="4848"/>
Puerto HTTP	<input type="text" value="8080"/>
Usuario	<input type="text" value="admin"/>
Contraseña	<input type="password" value="....."/>
Volver a Introducir Contraseña	<input type="password" value="....."/>
<input type="checkbox"/> Crear Servicio de Sistema Operativo para el Dominio	
Nombre de Servicio	<input type="text" value="domain1Service"/>
<input checked="" type="checkbox"/> Iniciar Dominio después de Crearlo	

Fuente: Propia

**Ilustración 63:** Parámetros de configuración de GlassFish.

**k)** Se muestra un mensaje de instalación y configuración satisfactoria, y salir.

**l)** Para desplegar Apex Listener en GlassFish, se siguen los pasos de la sección 6.3 excepto el paso **q)** que en lugar de iniciar, se selecciona la opción de no iniciar.

**m)** Se genera la configuración de los componentes estáticos, para ello se copia las imágenes de **/app/apex/images/** en **/app/glassfish/glassfish/domains/dominio/docroot/i/**, el directorio **i** hay que crearlo.

**n)** Se ejecuta el comando:

```
java -jar apex.war static /app/glassfish/glassfish/domains/dominio/docroot/i/
```

**ñ)** Se genera el archivo **i.war**

**o)** Se tiene dos archivos; el **apex.war** y el **i.war**

**p)** Se dirige a la dirección **localhost:4848**, es la consola de administración de GlassFish.

**q)** Se dirige a la opción **Aplicaciones** y se despliega el archivo **apex.war**, como se muestra en la **Ilustración 64**.

**Tipo:** \* Aplicación Web

**Raíz de Contexto:** apex  
Ruta de acceso relativa a la URL base del servidor.

**Nombre de Aplicación:** \* apex

**Servidores Virtuales:** server  
Asocia un nombre de dominio de Internet a un servidor físico.

**Estado:**  **Activada**  
Permite a los usuarios acceder a la aplicación.

Fuente: Propia  
**Ilustración 64:** Desplegar archivo apex.war.

r) A continuación se despliega el archivo **i.war**, como se muestra en la **Ilustración 65**

**Tipo:** \* Aplicación Web

**Raíz de Contexto:**  
Ruta de acceso relativa a la URL base del servidor.

**Nombre de Aplicación:** \* i

**Servidores Virtuales:** server  
Asocia un nombre de dominio de Internet a un servidor físico.

**Estado:**  **Activada**

Fuente: Propia  
**Ilustración 65:** Desplegar archivo i.war.

s) En la **Ilustración 66** se muestran las aplicaciones desplegadas.

Aplicaciones Desplegadas (2)					
	Nombre	Activada	Motores	Acción	
<input type="checkbox"/>	apex	<input checked="" type="checkbox"/>	web	<a href="#">Iniciar</a>   <a href="#">Volver</a>	
<input type="checkbox"/>	i	<input checked="" type="checkbox"/>	web	<a href="#">Iniciar</a>   <a href="#">Volver</a>	

Fuente: Propia  
**Ilustración 66:** Aplicaciones desplegadas en GlassFish.

t) Para verificar el funcionamiento se hace click en **Iniciar**.

u) Para iniciar el servidor GlassFish se lo hace con el comando:

**/app/glassfish/bin/asadmin stop-domain dominio**

## 8 REFERENCIAS

- 2.1.1 Sistema gestor de base de datos (Cabello, 2010)
- 2.1.2 Modelo cliente – servidor (Sommerville & Galipienso, 2005)
- 2.2 Oracle (www.oracle.com)
- 2.4 Arquitectura de Oracle Application Express (Scott, Aust, D’Souza, Gault, & Gielis, 2011)
- 2.3 Oracle 11g Express Edition (Oracle Application Express)
- 2.3.1 Modelo de desarrollo Oracle Application Builder (Introducing Oracle Application Express, 2013)
- 2.3.2 El lenguaje PL/SQL (Oracle PL/SQL Programming, 2014)
- 2.3.3 Biblioteca JQuery y técnica de desarrollo web Ajax (jquery, 2014)
- 2.3.3 Biblioteca JQuery y técnica de desarrollo web Ajax (Mozilla Developer Network, 2012) (Eguiluz)
- 2.4 Oracle Application Express (Oracle Application Express Installation Overview, 2013)
- 2.5 Firma electrónica (Subsecretaría de Tecnologías de la Información)
- 2.6 Metodología (Beck, 2000)
- Ilustración 5 (Feuerstein & Pribyl, 2014)
- 2.10.14 Las Acciones Dinámicas en una Página en Oracle Application Builder
- 3.1 Acerca de GlassFish (Goncalves, 2010)
- 3 Programación Extrema (Usero Ruiz & Valero Vergara, 2004)
  
- Libro de amazon (Ahmed, 2011)

## 9 BIBLIOGRAFIA

- Overview of PL/SQL. (1996). Recuperado el 20 de 2 de 2013, de Overview of PL/SQL: docs.oracle.com/cd/E11882\_01/appdev.112/e25519/overview.htm
- Mozilla Developer Network. (2012). Recuperado el 13 de 1 de 2014, de developer.mozilla.org/es/docs/AJAX
- Introducing Oracle Application Express. (2013). Recuperado el 4 de 2 de 2013, de Oracle Documentation: [http://docs.oracle.com/cd/E37097\\_01/doc/doc.42/e35122/intro\\_app.htm](http://docs.oracle.com/cd/E37097_01/doc/doc.42/e35122/intro_app.htm)
- Oracle Application Express Installation Overview. (2013). Recuperado el 18 de 6 de 2012, de docs.oracle.com/cd/E37097\_01/doc/install.42/e35123/overview.htm
- Jquery. (2014). Recuperado el 5 de 9 de 2013, de jquery.com
- Ahmed, R. (2011). Create Rapid Web Application Using Oracle Application Express. Lexington.
- Beck. (2000). Extreme Programming Explained: Embrace Change. USA: Addison-Wesley.
- Cabello, M. (2010). Introducción a Las Bases de Datos Relacionales. Madrid: Visión Libros.
- Eguiluz, J. (s.f.). Introducción a AJAX. Recuperado el 4 de 12 de 2013, de librosweb.es/ajax/
- Feuerstein, S., & Pribyl, B. (2014). Oracle PL/SQL Programming. Gravenstein Highway North: O'Reilly Media, Inc.
- Goncalves. (2010). Beginning Java™ EE 6 Platform with GlassFish™ 3. New York: Apress.
- Oracle Application Express. (s.f.). Recuperado el 2 de 2 de 2013, de Oracle Application Express: <http://www.oracle.com/technetwork/database/database-technologies/express-edition/overview/index.html>
- Scott, J. E., Aust, D., D'Souza, M. G., Gault, D., & Gielis. (2011). Expert Oracle Application Express. New York: Apress.
- Sommerville, & Galipienso. (2005). Ingeniería del Software. Madrid: Pearson Educación.
- Subsecretaria de Tecnologías de la Información. (s.f.). Recuperado el 17 de 5 de 2013, de [www.gobiernoelectronico.gob.ec](http://www.gobiernoelectronico.gob.ec)
- Usero Ruiz, A., & Valero Vergara, F. (2004). Ejemplo de desarrollo software utilizando la metodología XP.
- www.oracle.com. (s.f.). Recuperado el 2 de 2 de 2013, de <http://www.oracle.com/>