

# UNIVERSIDAD TÉCNICA DEL NORTE



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

### PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERA EN SISTEMAS COMPUTACIONALES

#### TEMA:

Desarrollo de una Aplicación Web utilizando la tecnología JSF  
con una arquitectura MVC para La Viandé.

---

#### APLICATIVO

Desarrollo – Software

AUTOR: Ana Belén Obando Varas

DIRECTOR: Ing. Pedro Granda.

Ibarra – Ecuador

2013

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



**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 formato 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 información:

<b>DATOS DE CONTACTO</b>	
<b>CEDULA DE IDENTIDAD</b>	100357258-1
<b>APELLIDOS Y NOMBRES</b>	OBANDO VARAS ANA BELÉN
<b>DIRECCIÓN</b>	IBARRA, SANCHEZ Y CIFUENTES Y REINALDO CHAVEZ
<b>EMAIL</b>	<a href="mailto:anitabelen88@hotmail.es">anitabelen88@hotmail.es</a>
<b>TELÉFONO FIJO</b>	2650446
<b>TELÉFONO MOVIL</b>	094119240

<b>DATOS DE LA OBRA</b>	
<b>TÍTULO</b>	DESARROLLO DE UNA APLICACIÓN WEB, UTILIZANDO LA TECNOLOGÍA JSF CON UNA ARQUITECTURA MVC PARA LA VIANDÉ.
<b>AUTOR</b>	OBANDO VARAS ANA BELÉN
<b>FECHA</b>	06 DE MARZO DE 2013
<b>PROGRAMA</b>	PREGRADO
<b>TÍTULO POR EL</b>	INGENIERÍA EN SISTEMAS COMPUTACIONALES
<b>DIRECTOR</b>	ING. PEDRO GRANDA

.....

Firma:

Nombre: Ana Belén Obando Varas

Cédula: 100357258-1

Ibarra, a los 06 días del mes de marzo de 2013

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

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

Yo, Ana Belén Obando Varas, con cédula de identidad Nro. 100357258-1, en calidad de autora 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 uso del archivo digital en la biblioteca de la universidad con fines académicos, investigación y extensión, en concordancia con la Ley de Educación Superior Artículo 144.



.....

Firma:

Nombre: Ana Belén Obando Varas

Cédula: 100357258-1

Ibarra, a los 06 días del mes de marzo de 2013

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



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

Yo, Ana Belén Obando Varas, con cédula de identidad Nro. 100357258-1, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad del Ecuador, artículos 4, 5 y 6 en calidad de autora del trabajo de grado denominado "DESARROLLO DE UNA APLICACIÓN WEB, UTILIZANDO LA TECNOLOGÍA JSF CON UNA ARQUITECTURA MVC PARA LA VIANDÉ", que ha sido desarrollado para optar por el título de: **Ingeniera en Sistemas Computacionales**, quedando la Universidad Técnica del Norte facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autora me reservo los derechos morales de la obra antes citada.

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

A handwritten signature in blue ink, appearing to read "Ana Belén Obando Varas", is written over a dotted line.

Firma:

Nombre: Ana Belén Obando Varas

Cédula: 100357258-1

Ibarra, a los 06 días del mes de marzo de 2013

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## DECLARACIÓN

Yo, Ana Belén Obando Varas, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; y que éste no ha sido previamente presentado para ningún grado o calificación profesional.

A través de la presente declaración cedo los derechos de propiedad intelectual correspondiente a este trabajo, a la Universidad Técnica del Norte, según lo establecido por las Leyes de Propiedad Intelectual, Reglamentos y Normatividad vigente de la Universidad Técnica del Norte.

Ana Belén Obando Varas

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## CONSTANCIA

El autor manifiesta que la obra de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en la defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 06 días del mes de marzo del 2013

Ana Belén Obando Varas

ACEPTACIÓN

Firma

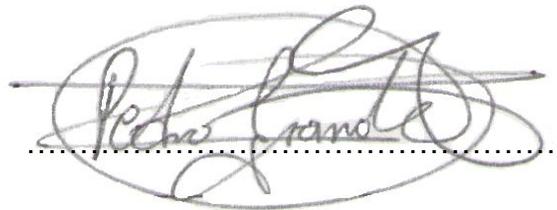


\_\_\_\_\_

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## CERTIFICACIÓN

En mi calidad de Asesor del Trabajo de Grado presentado por la egresada ANA BELÉN OBANDO VARAS, para optar por el título de INGENIERA EN SISTEMAS COMPUTACIONALES. Cuyo tema es: “Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé”. Considero que el trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del tribunal examinador.

A handwritten signature in black ink, enclosed within a hand-drawn oval. The signature is stylized and appears to read 'Pedro Granda'. Below the signature is a horizontal dotted line.

Ing. Pedro Granda

**DIRECTOR DE TESIS**

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## CERTIFICACIÓN

Ibarra, 06 de marzo del 2013

Señores

UNIVERSIDAD TÉCNICA DEL NORTE

Presente

De mis consideraciones.-

Siendo auspiciantes del proyecto de tesis de la Egresada Ana Belén Obando Varas con C.I. 100357258-1, quien desarrolló su trabajo con el tema "DESARROLLO DE UNA APLICACIÓN WEB, UTILIZANDO LA TECNOLOGÍA JSF CON UNA ARQUITECTURA MVC PARA LA VIANDÉ", 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 de la egresada Ana Belén Obando Varas. Una vez que hemos recibido la capacitación y documentación respectiva, nos comprometemos a continuar utilizando el mencionado aplicativo en beneficio de nuestra empresa.

La egresada Ana Belén Obando Varas puede hacer uso de este documento para los fines pertinentes en la Universidad Técnica del Norte.

Atentamente,



A handwritten signature in blue ink, appearing to be "Paulina Obando", written over a horizontal dotted line.

Tecnóloga Paulina Obando

GERENTE

La Viandé

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## DEDICATORIA

Primeramente a Dios por haberme dado la fuerza y conocimiento para realizar este trabajo y por poner en mí camino a personas tan hermosas y valiosas que iluminaron mi vida.

Dedico este trabajo de Tesis a mis padres que con esfuerzo y cariño supieron darme una educación digna, siempre me apoyaron en todo momento y creyeron en mí, porque me sacaron adelante, dándome ejemplos dignos de superación y entrega, porque en gran parte gracias a ustedes, hoy puedo ver alcanzada mi meta, ya que siempre estuvieron impulsándome en los momentos más difíciles de mi carrera, y porque el orgullo que sienten por mí, fue lo que me hizo ir hasta el final.

A mis Hermanos, porque siempre he contado con ellos para todo, gracias a la confianza que siempre nos hemos tenido; por el apoyo y amistad.

A mis maestros, gracias por su tiempo, por su apoyo así como por la sabiduría que me transmitieron en el desarrollo de mi formación profesional.

A la **Universidad Técnica del Norte** y en especial a la Facultad de Ingeniería en Ciencias Aplicadas que me dieron la oportunidad de formar parte de ella.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## **AGRADECIMIENTO**

Primeramente doy infinitamente gracias a Dios, por haberme dado fuerza y valor para culminar mis estudios.

Agradezco también la confianza y el apoyo de mi familia, porque han contribuido positivamente e impulsaron al desarrollo de esta tesis, logrando así cumplir el anhelo de culminar esta carrera.

Un agradecimiento especial al Ing. Pedro Granda por su tiempo dedicado, paciencia, su invaluable aporte académico y don de ser que hace de él una persona excepcional, por su orientación en el estudio y culminación de la presente tesis.

A mis amigos, conocer a personas como ustedes fue algo grandioso, gracias por estar en las buenas y en las malas apoyándonos mutuamente, cosechar su amistad fue una de las mejores cosas que me pasó en la universidad y que llevaré por siempre.

A la Universidad Técnica del Norte, a los docentes de la facultad de Ingeniería en Ciencias Aplicadas quienes, en su debido momento, compartieron sus conocimientos y me dieron la oportunidad de culminar mis estudios, al igual que por su incansable labor en beneficio de la juventud estudiosa del norte del país.

## ÍNDICE DE CONTENIDOS

AUTORIZACIÓN DE USO Y PUBLICACIÓN.....	iv
CESIÓN DE DERECHOS DE AUTOR .....	ii
DECLARACIÓN.....	v
CONSTANCIA .....	vi
CERTIFICACIÓN.....	vii
CERTIFICACIÓN.....	viii
DEDICATORIA .....	ix
AGRADECIMIENTO .....	x
ÍNDICE DE CONTENIDOS .....	xi
RESUMEN .....	xxiii
SUMMARY.....	xxiv
CAPÍTULO I.....	1
1.1    INTRODUCCIÓN AL CATERING SERVICE .....	1
1.1.1.    QUE ES CATERING SERVICE .....	1
1.1.1.1.    TIPOS DE CATERING.....	2
1.1.1.1.1.    Catering aperitivo.....	2
1.1.1.1.2.    Catering cocktail.....	2
1.1.1.1.3.    Catering recepción.....	3
1.1.1.1.4.    Catering recepción buffet.....	3
1.1.1.1.5.    Catering recepción, comida o banquete .....	4
1.1.1.1.6.    Catering desayuno de trabajo .....	4
1.1.1.1.7.    Catering brunch.....	5
1.1.1.1.8.    Catering vernissage.....	5
1.1.1.1.9.    Catering vino de honor.....	6
1.1.1.2.    TIPOS DE EVENTOS .....	7
1.2    EMPRESA LA VIANDÉ.....	7
CAPÍTULO II.....	8
2.1    MODELO VISTA CONTROLADOR .....	8
2.1.1    INTRODUCCIÓN AL PATRÓN MVC.....	8
2.1.1.1    MVC EN APLICACIONES WEB .....	10
2.1.2    PATRÓN MVC .....	10

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

2.1.3	FRAMEWORKS MVC .....	13
2.2	JAVA SERVER FACES .....	20
2.2.1	DEFINICIÓN DE JSF .....	20
2.2.2	COMPONENTES PRINCIPALES DE JSF .....	20
2.2.3	DEFINICIÓN DE UNA APLICACIÓN JSF .....	23
2.2.4	ESTRUCTURA DE UNA APLICACIÓN JSF .....	25
2.2.5	PASOS PARA DESARROLLAR UNA APLICACIÓN JSF .....	26
2.2.6	CICLO DE VIDA DE UNA PÁGINA JSF .....	26
2.2.7	REQUERIMIENTOS PARA TRABAJAR CON JSF .....	29
2.2.8	HERRAMIENTAS PARA LA CREACIÓN Y UTILIZACIÓN DE JSF .....	30
CAPÍTULO III .....		31
3.1	ANÁLISIS DE LOS ENTORNOS DE DESARROLLO ECLIPSE, JBUILDER Y NETBEANS.....	31
3.1.1	INTRODUCCIÓN.....	31
3.1.1.1	ECLIPSE .....	32
3.1.1.1.1	Eclipse como IDE.....	34
3.1.1.1.2	Servidores de Aplicaciones .....	36
3.1.1.1.3	Bases de datos .....	38
3.1.1.1.4	Plataforma de desarrollo J2EE .....	39
3.1.1.1.5	Lenguajes .....	40
3.1.1.1.6	Modelamiento.....	40
3.1.1.1.7	Investigación.....	41
3.1.1.1.8	Interfaces de usuario.....	41
3.1.1.2	NETBEANS .....	42
3.1.1.2.1	Bases de datos .....	48
3.1.1.2.2	Aplicaciones distribuidas.....	48
3.1.1.2.3	Refactorización .....	49
3.1.1.2.4	Modeling Tools.....	49
3.1.1.2.5	Web .....	49
3.1.1.2.6	Herramientas J2EE .....	50
3.1.1.2.7	J2ME .....	50
3.1.1.3	JBUILDER.....	50

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

3.1.2	COMPARACIÓN.....	53
3.1.3	CONCLUSIONES.....	56
3.2.1	Información general.....	57
3.2.2	Soporte del sistema operativo .....	58
3.2.3	Características fundamentales .....	58
3.2.5	Índices.....	59
3.2.6	Otros objetos.....	59
3.2.7	Particionamiento .....	60
3.2.8	Conclusiones:.....	60
	CAPÍTULO IV.....	63
4.1	DESARROLLO DEL SISTEMA .....	63
4.1.4	DOCUMENTO DE VISIÓN .....	63
4.1.4.1	Introducción .....	63
4.1.4.2	Propósito .....	63
4.1.4.3	Alcance.....	63
4.1.4.4	Definiciones, Acrónimos, y Abreviaciones .....	64
4.1.4.5	Referencias.....	64
4.1.4.6	Posicionamiento.....	64
4.1.4.6.1	Oportunidad de Negocio .....	64
4.1.4.6.2	Sentencia que define el problema .....	64
4.1.4.6.3	Sentencia que define la posición del Producto.....	65
4.1.4.7	Descripción de Stakeholders y Usuarios.....	66
4.1.4.7.1	Resumen de Stakeholders .....	66
4.1.4.7.2	Resumen de Usuarios.....	67
4.1.4.7.3	Entorno de usuario .....	67
4.1.4.8	Perfil de los Stakeholders.....	68
4.1.4.8.1	Representante del área técnica y sistemas de información .....	68
4.1.4.9	Perfiles de Usuario .....	69
4.1.4.9.1	María Paulina Obando Varas.....	69
4.1.4.9.2	Guadalupe Varas Pozo.....	69
4.1.4.10	Descripción Global del Producto .....	70
4.1.4.10.1	Perspectiva del producto .....	70

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

4.1.4.11	Resumen de características .....	70
4.1.4.12	Suposiciones y dependencias.....	71
4.1.4.13	Costo y precio .....	72
4.1.4.14	Descripción Global del Producto.....	74
4.1.4.15	Restricciones.....	76
4.1.4.16	Precedencia y Prioridad.....	76
4.1.4.17	Otros Requisitos del Producto .....	76
4.1.4.17.1	Estándares Aplicables.....	76
4.1.4.17.2	Requisitos de Sistema.....	77
4.1.4.17.3	Requisitos de Desempeño.....	77
4.1.4.17.4	Requisitos de Entorno.....	77
4.1.4.17.5	Requisitos de Documentación .....	77
4.1.4.17.6	Manual de Usuario.....	77
4.1.4.17.7	Ayuda en Línea .....	77
4.1.4.18	Guías de Instalación, Configuración, y Fichero Léame.....	78
4.1.4.19	Atributos de Características .....	78
4.3.1	Iniciar Sesión .....	94
4.3.1.1	Descripción .....	94
4.3.1.2	Flujo de Eventos .....	95
4.3.1.2.1	Flujo Básico .....	95
4.3.1.2.2	Flujos Alternativos .....	95
4.3.1.2.2.1	Datos Inválidos.....	95
4.3.1.2.3	Precondiciones.....	95
4.3.1.2.4	Pos condiciones.....	95
4.4.1	Crear Usuario.....	96
4.4.1.1	Descripción .....	96
4.4.1.2	Flujo de Eventos .....	96
4.4.1.2.1	Flujo Básico .....	96
4.4.1.2.2	Flujos Alternativos .....	96
4.4.1.3	Precondiciones.....	96
4.5.1	Crear Usuario.....	97
4.5.1.1	Descripción .....	97

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

4.5.1.2 Flujo de Eventos .....	97
4.5.1.2.1 Flujo Básico .....	97
4.5.1.2.2 Flujos Alternativos .....	97
4.5.1.3 Precondiciones.....	97
4.5.1.4 Pos condiciones.....	97
4.6.1 Eliminar Usuario.....	98
4.6.1.1 Descripción .....	98
4.6.1.2 Flujo de Eventos .....	98
4.6.1.2.1 Flujo Básico .....	98
4.6.1.2.2 Flujos Alternativos .....	98
4.6.1.3 Precondiciones.....	98
4.6.1.4 Pos condiciones.....	98
4.7.1 Buscar Usuario .....	99
4.7.1.1 Descripción .....	99
4.7.1.2 Flujo de Eventos .....	99
4.7.1.2.1 Flujo Básico .....	99
4.7.1.2.2 Flujos Alternativos .....	99
4.7.1.3 Precondiciones.....	99
4.7.1.4 Pos condiciones.....	99
4.8.1 Crear Grupo de Productos .....	100
4.8.1.1 Descripción .....	100
4.8.1.2 Flujo de Eventos .....	100
4.8.1.2.1 Flujo Básico .....	100
4.8.1.2.2 Flujos Alternativos .....	100
4.8.1.3 Precondiciones.....	100
4.8.1.4 Pos condiciones.....	100
4.9.1 Eliminar Grupo de Productos .....	101
4.9.2 Descripción .....	101
4.9.2.1 Flujo de Eventos .....	101
4.9.2.1.1 Flujo Básico .....	101
4.9.2.1.2 Flujos Alternativos .....	101
4.9.2.2 Precondiciones.....	101

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

4.9.2.3 Pos condiciones.....	101
4.10.1 Crear Producto.....	102
4.10.2 Descripción .....	102
4.10.3 Flujo de Eventos.....	102
4.10.3.1 Flujo Básico .....	102
4.10.3.2 Flujos Alternativos .....	102
4.10.4 Precondiciones .....	102
4.10.5 Pos condiciones.....	102
4.11.1 Eliminar Producto .....	103
4.11.2 Descripción .....	103
4.11.3 Flujo de Eventos .....	103
4.11.3.1 Flujo Básico .....	103
4.11.3.2 Flujos Alternativos .....	103
4.11.4 Precondiciones .....	103
4.11.5 Pos condiciones.....	104
4.12.1 Crear Cliente.....	104
4.12.2 Descripción .....	104
4.12.3 Flujo de Eventos .....	104
4.12.3.1 Flujo Básico .....	104
4.12.3.2 Flujos Alternativos .....	105
4.12.4 Precondiciones .....	105
4.12.5 Pos condiciones.....	105
4.13.1 Eliminar Cliente.....	105
4.13.2 Descripción .....	105
4.13.3 Flujo de Eventos .....	105
4.13.3.1 Flujo Básico .....	105
4.13.3.2 Flujos Alternativos .....	106
4.13.4 Precondiciones .....	106
4.13.5 Pos condiciones.....	106
4.14.1 Crear Proforma .....	106
4.14.1.1 Descripción .....	106
4.15.1 Crear Proforma .....	107
4.15.1.1 Descripción .....	107

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

4.15.2 Flujo de Eventos .....	107
4.15.2.1 Flujo Básico .....	107
4.15.2.2 Flujos Alternativos .....	107
4.15.3 Precondiciones .....	107
4.15.4 Pos condiciones.....	108
4.16.1 Crear Proforma .....	108
4.16.1.1 Descripción .....	108
4.16.2 Flujo de Eventos .....	108
4.16.2.1 Flujo Básico .....	108
4.16.2.2 Flujos Alternativos .....	108
4.17.1 Crear Proforma .....	109
4.17.1.1 Descripción .....	109
4.17.2 Flujo de Eventos .....	109
4.17.2.1 Flujo Básico .....	109
4.17.2.2 Flujos Alternativos .....	109
4.17.3 Precondiciones .....	109
4.17.4 Pos condiciones.....	110
4.18.1 Eliminar Proforma.....	110
4.18.2 Descripción .....	110
4.18.3 Flujo de Eventos .....	110
4.18.3.1 Flujo Básico .....	110
4.18.3.2 Flujos Alternativos .....	110
4.18.4 Precondiciones .....	110
4.18.5 Pos condiciones.....	110
4.19.1 Generar Orden.....	111
4.19.2 Descripción .....	111
4.19.3 Flujo de Eventos .....	111
4.19.3.1 Flujo Básico .....	111
4.19.3.2 Flujos Alternativos .....	111
4.19.4 Precondiciones .....	111
4.19.5 Pos condiciones.....	111
4.20.1 Generar Factura .....	112
4.20.2 Descripción .....	112

4.20.3 Flujo de Eventos .....	112
4.20.3.1 Flujo Básico .....	112
4.20.3.2 Flujos Alternativos .....	112
4.20.4 Precondiciones .....	113
4.20.5 Pos condiciones.....	113
4.21.1 Eliminar Factura.....	113
4.21.2 Descripción .....	113
4.21.3 Flujo de Eventos .....	113
4.21.3.1 Flujo Básico .....	113
4.21.3.2 Flujos Alternativos .....	114
4.21.4 Precondiciones .....	114
4.21.5 Pos condiciones.....	114
4.22.1 Descripción .....	114
4.22.2 Crear una factura y guardar .....	114
4.22.2.1 Descripción .....	114
4.22.2.2 Condiciones de ejecución.....	115
4.22.2.3 Entrada.....	115
4.22.2.4 Resultado esperado .....	115
4.22.2.5 Evaluación de la Prueba.....	115
4.22.3 Crear Factura y Buscar .....	115
4.22.3.1 Descripción .....	115
4.22.3.2 Condiciones de ejecución.....	116
4.22.3.3 Entrada.....	116
4.22.3.4 Resultado esperado .....	116
4.22.3.5 Evaluación de la Prueba .....	117
4.23 Fase de construcción .....	117
4.23.1 Vista Lógica.....	117
4.23.1.1 Diagrama de Base de Datos .....	117
4.23.1.2 Fase de transición.....	119
4.24 Glosario .....	119
4.24.1 Introducción .....	119
4.24.2 Propósito .....	119

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

4.24.3	Alcance .....	119
4.24.4	Referencias .....	119
4.24.5	Organización del Glosario .....	119
4.24.6	Definiciones .....	120
4.24.6.1	Artefacto RUP .....	120
4.24.6.2	Base de Datos .....	120
4.24.6.3	MVC.....	120
4.24.6.4	Navegador Web .....	120
4.24.6.5	RUP .....	120
4.24.6.6	Servidor .....	120
4.24.6.7	Sistema Web .....	121
4.24.6.8	Validación .....	121
4.24.6.9	Estereotipos UML .....	121
CAPÍTULO V.....		122
5. CONCLUSIONES Y RECOMENDACIONES.....		122
CAPÍTULO VI.....		124
6. BIBLIOGRAFÍA.....		124
ANEXOS .....		126
A.	INSTALACIÓN JDK.....	126
B.	INSTALACIÓN DEL CONTENEDOR APACHE TOMCAT 6.0.20 .....	135
C.	INSTALACIÓN DE ECLIPSE HELIOS .....	138
D.	MANUAL DE USUARIO SISTEMA DE GESTIÓN DE INFORMACIÓN .....	140
D.1.	INTRODUCCIÓN.....	140
D.2.	OBJETIVO DE ESTE MANUAL .....	140
D.3.	DIRIGIDO A .....	141
D.4.	LO QUE DEBE CONOCER.....	141
D.5.	CONVENCIONES Y ESTÁNDARES A UTILIZAR.....	141
5.1.	CONVENCIONES DEL USO DEL MOUSE .....	141
5.2.	CONVENCIONES DEL USO DEL TECLADO.....	142
D.6.	INGRESO AL SISTEMA.....	142
6.1.	PASOS PARA ENCENDER LA COMPUTADORA.....	142
6.2.	COMO ACCEDER AL SISTEMA DE GESTIÓN DE LA INFORMACIÓN.....	143

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

D.7.	SISTEMA DE GESTIÓN DE INFORMACIÓN .....	144
7.1.	LOGIN .....	144
7.2.	ARCHIVO .....	145
7.3.	ADMINISTRACIÓN .....	145
7.3.1.	USUARIOS .....	145
7.3.2.	GRUPOS PRODUCTOS .....	148
7.3.3.	PRODUCTOS .....	150
7.3.4.	PLATOS .....	153
7.3.5.	MENÚS .....	156
7.3.6.	AJUSTE DE INVENTARIO.....	159
7.4.	PROFORMAS .....	161
7.4.1.	CALENDARIO.....	161
7.4.2.	CLIENTES .....	162
7.4.3.	PROFORMA.....	163
7.4.4.	STATUS ORDEN.....	167
7.5.	FACTURACIÓN.....	168
7.5.1.	FACTURA.....	168
7.5.2.	ANULAR FACTURA.....	170
7.5.3.	REPORTE DE VENTAS .....	171
7.5.4.	REPORTE DE CLIENTE-EVENTOS .....	172
7.6.	CUENTAS POR COBRAR .....	172
7.6.1.	ABONOS.....	172
7.6.2.	PAGO .....	174
7.6.3.	PAGOS POR CLIENTES.....	176
D.8.	PÁGINA WEB LA VIANDE .....	177
E.	GLOSARIO DE TÉRMINOS.....	179

## ÍNDICE DE FIGURAS

Figura N° 1. 1 Catering Service.....	1
Figura N° 1. 2. Catering Aperitivo. ....	2
Figura N° 1. 3. Catering Cocktail.....	2
Figura N° 1. 4. Catering Recepción. ....	3
Figura N° 1. 5. Catering recepción buffet.....	3
Figura N° 1. 6. Catering recepción, comida o banquete .....	4
Figura N° 1. 7. Catering desayuno de trabajo .....	4
Figura N° 1. 8. Catering brunch.....	5
Figura N° 1. 9. Catering vernissage.....	5
Figura N° 1. 10. Catering vernissage.....	6
Figura N° 1. 11. Catering coffee break.....	6
Figura N° 1. 12. Logotipo Empresa La Viandé .....	7
Figura N° 2. 1. Arquitectura del Patrón MVC (Modelo – Vista – Controlador).....	8
Figura N° 2. 2. Elementos del patrón MVC (Modelo – Vista – Controlador).....	11
Figura N° 2. 3. Diagrama de secuencia patrón MVC (Modelo - Vista – Controlador). ....	13
Figura N° 2. 4 Componentes de una aplicación JSF. ....	21
Figura N° 2. 5. Ciclo de vida de una página JSF.....	27
Figura N° 4. 1 Sentencia que define el problema. ....	65
Figura N° 4. 2 Sentencia que define la posición del Producto. ....	66
Figura N° 4. 3. Descripción del Proceso unificado. ....	94
Figura N° 4. 4. Caso de Uso Iniciar Sesión .....	94
Figura N° 4. 5. Caso de Uso Crear Usuario.....	96
Figura N° 4. 5. Caso de Uso Crear Usuario.....	97
Figura N° 4. 6. Caso de Uso Eliminar Usuario. ....	98
Figura N° 4. 7. Caso de Uso Buscar Usuario.....	99
Figura N° 4. 8. Caso de Uso Crear Grupo de Productos.....	100
Figura N° 4. 9. Caso de Uso Eliminar Grupo de Productos.....	101
Figura N° 4. 10. Caso de Uso Crear Producto.....	102
Figura N° 4. 11. Caso de Uso Eliminar Producto.....	103
Figura N° 4. 12. Caso de Uso Crear Cliente .....	104
Figura N° 4. 13. Caso de Uso Eliminar Cliente.....	105
Figura N° 4. 14. Caso de Uso Crear Proforma. ....	106
Figura N° 4. 14. Caso de Uso Crear Proforma. ....	107
Figura N° 4. 14. Caso de Uso Crear Proforma. ....	108
Figura N° 4. 14. Caso de Uso Crear Proforma. ....	109
Figura N° 4. 15. Caso de Uso Eliminar Proforma.....	110
Figura N° 4. 16. Caso de Uso Generar Orden.....	111
Figura N° 4. 17. Caso de Uso Generar Factura.....	112
Figura N° 4. 18. Caso de Uso Eliminar Factura.....	113

## ÍNDICE DE TABLAS

Tabla N° 3. 1. Descripción Servidores de Aplicación. ....	38
Tabla N° 3. 2. Descripción Base de Datos. ....	39
Tabla N° 3. 3. Descripción Plataforma de desarrollo J2EE. ....	40
Tabla N° 3. 4. Descripción Lenguajes ....	40
Tabla N° 3. 5. Descripción Modelamiento.....	40
Tabla N° 3. 6. Descripción Investigación. ....	41
Tabla N° 3. 7. Descripción Interfaz de Usuario. ....	41
Tabla N° 4. 1 Resumen de Stakeholders.....	67
Tabla N° 4. 2. Resumen de Usuarios.....	67
Tabla N° 4. 3. Representante de Área Técnica .....	68
Tabla N° 4. 4. Perfil de Usuario – Paulina Obando .....	69
Tabla N° 4. 5. Perfil de Usuario – Guadalupe Varas.....	70
Tabla N° 4. 6. Resumen de Características .....	71
Tabla N° 4. 7. Inversión .....	73
Tabla N° 4. 8. Capital de Trabajo .....	73
Tabla N° 4. 9. Descripción Global del Producto .....	75
Tabla N° 4. 10. Atributos de Características. ....	79
Tabla N° 4. 11. Roles y responsabilidades. ....	90
Tabla N° 4. 12. Plan de las Fases.....	91
Tabla N° 4. 13. Hitos.....	93

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## RESUMEN

La empresa la Viandé, enfrenta un problema en cuanto a los procesos de facturación, control de productos, clientes del servicio de catering y banquetes de la provincia de Imbabura.

Todos estos procesos de manejo de información lo realizan de manera manual, o con la ayuda de herramientas no seguras, por tal motivo este proyecto pretende dar solución mediante la automatización de estos procesos de información.

La necesidad de este aspecto surge debido a que no se tiene un mayor control de la información, surge pérdida de información, los datos obtenidos no son tan reales.

Este trabajo trata de llegar a una solución de calidad, logrando así que el sistema permita tener un mayor control de estos datos y presentar al usuario información real en tiempo real.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## **SUMMARY**

The Viandé Company, has a problem in terms of billing processes, product control, customer catering service and banquet of the province of Imbabura.

All these processes of information management are done manually or with the help of tools unsecured Therefore this project aims to address this by automating information processes.

The need for this aspect arises because there is no further control information, information loss arises, the data are not real.

This paper tries to reach a quality solution, thus allowing the system to have more control of the data and present the user with real information in real time.

## CAPÍTULO I

---

### 1.1 INTRODUCCIÓN AL CATERING SERVICE

#### 1.1.1. QUE ES CATERING SERVICE



Figura N° 1. 1 Catering Service

Fuente: <http://www.stmarys-ca.edu/for-community-visitors/catering-services>

La palabra catering, proveniente del idioma inglés, hace referencia a la actividad de proveer con un servicio de comida especialmente contratado a clientes en situaciones específicas, tales como eventos de diverso tipo. La palabra catering viene del verbo inglés 'to cater', que significa servir, atender. El catering es el servicio que supone la atención y la oferta de un variado conjunto de alimentos en situaciones excepcionales como suelen ser los eventos. Si bien esta palabra no es perteneciente al idioma español, se encuentra hoy en día aceptada al menos en el lenguaje informal y es de uso común.

Catering Service, se define como el servicio profesional que se dedica principalmente a la prestación externa del suministro del servicio de comida preparada, así como también puede tomar parte en abastecer de todo lo necesario para la organización de un banquete o una fiesta. Dicha comida puede ser bien preparada en cocinas centrales, toda la gestión corresponde

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

a la empresa de catering, o subcontratando a terceros en la elaboración total o parcial. También se considera catering a la preparación de comidas en grandes cantidades para ser vendidas en puntos de consumo separados del lugar donde se elaboran. Es una actividad de externalización de servicios de comidas.

### **1.1.1.1. TIPOS DE CATERING**

#### **1.1.1.1.1. Catering aperitivo**



**Figura N° 1. 2. Catering Aperitivo.**

Fuente: <http://ideasmuchogusto.blogspot.com/>

Este servicio tiene una duración de entre 15 a 30 minutos. Se realiza un cálculo de dos bocados por persona.

#### **1.1.1.1.2. Catering cocktail**



**Figura N° 1. 3. Catering Cocktail.**

Fuente: <http://www.infojardin.com/foro/showthread.php?p=4679263>

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

El horario estimado de realización es de 11 a 13 o de 19 a 21 horas. Se calculan de quince a veinte bocados por personas; salados 2/3 y dulce 1/3.

#### 1.1.1.1.3. Catering recepción



Figura N° 1. 4. Catering Recepción.

Fuente: <http://loa1130.wordpress.com/catering/>

Dura aproximadamente tres horas y frecuentemente se realiza a las 20 horas. Se calculan de dieciocho a veinte bocadillos por persona; de los cuales 2/3 son salados y 1/3 dulce.

#### 1.1.1.1.4. Catering recepción buffet



Figura N° 1. 5. Catering recepción buffet.

Fuente: <http://www.alamaula.pe/lima/eventos/eventos-buffets-catering-parrilladas/1220445>

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Al igual que el anterior, su duración es de tres horas y el horario estimado es a las 20 horas. Pero en este caso, se ofrecen variedades de platos fríos, calientes y guarniciones.

#### 1.1.1.1.5. Catering recepción, comida o banquete



Figura N° 1. 6. Catering recepción, comida o banquete

Fuente: [http://es.123rf.com/photo\\_14497375\\_catering-comida-en-una-fiesta-de-bodas.html](http://es.123rf.com/photo_14497375_catering-comida-en-una-fiesta-de-bodas.html)

Se contrata para eventos de larga duración, ya que se calculan entre seis y ocho horas y su horario estimado es de 22 horas. En este catering se calculan seis bocados para la recepción y luego, en la cena, encontramos la entrada, el plato principal, el postre, un servicio de café, el brindis, la mesa dulce y el final de la fiesta.

#### 1.1.1.1.6. Catering desayuno de trabajo



Figura N° 1. 7. Catering desayuno de trabajo

Fuente: <http://metropolitana.evisos.cl/fotos-del-anuncio/andes-catering-coffee-breaks-desayunos-empresariales-id-360387>

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Se calculan entre tres a cinco bocados por hora por persona y se puede elegir entre continental o americano, entre otros.

#### 1.1.1.1.7. Catering brunch



Figura N° 1. 8. Catering brunch

Fuente: <http://www.cateringmalena.es/index.php/id//objeto/34>

Es la combinación del desayuno y el almuerzo (breakfast – lunch), su horario estimado de servicio es a las 11 horas y su duración aproximada es de una hora y media.

#### 1.1.1.1.8. Catering vernissage



Figura N° 1. 9. Catering vernissage

Fuente: [http://www.anunico.com.ar/aviso/eventos\\_fiestas\\_catering/cordoba\\_catering-966402.html](http://www.anunico.com.ar/aviso/eventos_fiestas_catering/cordoba_catering-966402.html)

Se ofrecen dos bocados por persona, champagne y vino. Este catering se realiza mayormente en inauguraciones de muestras de pinturas.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

#### 1.1.1.1.9. Catering vino de honor



Figura N° 1. 10. Catering vernissage

Fuente: <http://lafetteeventos.com.ar/servicios/inauguraciones>

El horario de servicio de este catering es aproximadamente de 11.30 a 12.30 o de 18 a 19 horas. Se calculan dos bocados por persona y se sirven también vinos, y dulces.

#### 1.1.1.1.10. Catering coffee break



Figura N° 1. 11. Catering coffee break

Fuente: <http://vinadelmar.olx.cl/servicios-de-coffeebreak-cafeteria-coctel-catering-colaciones-region-metropolitan-iiid-113630600>

Se ofrece café, té, jugos y bocaditos salados y dulces. Dura entre 15 y 30 minutos y es básicamente un corte o recreo.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### 1.1.1.2. TIPOS DE EVENTOS

**Evento:** es una actividad social determinada, un festival, una fiesta, una ceremonia, una competición, una convención.

Aquella actividad que se desarrolla en un momento puntual en el tiempo, con un principio y un final previamente conocido, independientemente de que pueda poseer una periodicidad establecida.

Entre los tipos de eventos tenemos:

- Reunión.
- Matrimonio.
- Bautizo.
- Boda.
- Graduación.
- Primera comunión.
- Entre otros.

## 1.2 EMPRESA LA VIANDÉ



Figura N° 1. 12. Logotipo Empresa La Viandé

Fuente: La Viandé

La Viandé es una empresa de servicios profesionales, que coordinan, supervisan y montan fiestas o eventos, con decoraciones al gusto del cliente.

## CAPÍTULO II

### 2.1 MODELO VISTA CONTROLADOR

#### 2.1.1 INTRODUCCIÓN AL PATRÓN MVC

MVC: corresponde a las siglas Modelo-Vista-Controlador, fue introducido como parte de la versión Smalltalk-80 del lenguaje de programación Smalltalk<sup>1</sup>. Fue diseñado con el propósito de reducir el esfuerzo de programación y su principal característica es que el Modelo, las Vistas y los Controladores se tratan como entidades separadas, permitiendo que cualquier cambio que se produzca en el Modelo se refleje automáticamente en cada una de las Vistas.

El patrón MVC es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

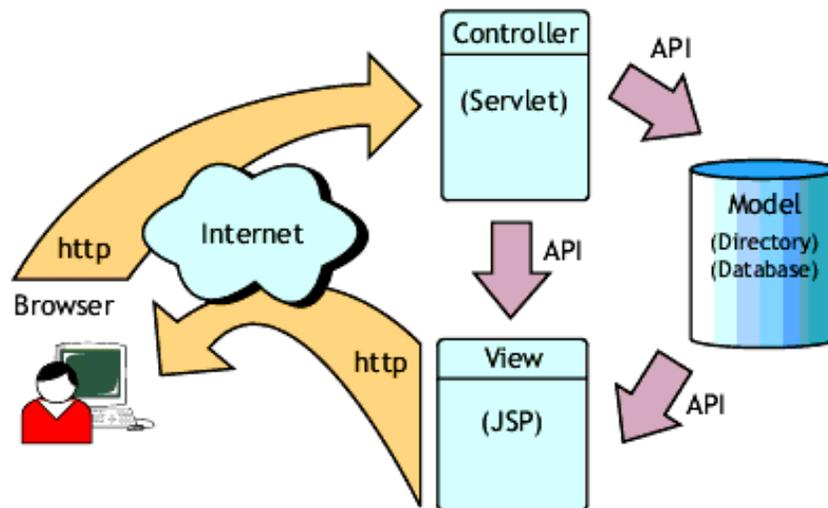


Figura N° 2. 1. Arquitectura del Patrón MVC (Modelo – Vista – Controlador).

Fuente: <http://www.rincondeloajeno.com/model-view-controler-jsf/>

<sup>1</sup> Permite realizar tareas de computación mediante la interacción con un entorno de objetos virtuales. Es un mundo virtual donde viven objetos que se comunican mediante el envío de mensajes.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Este modelo de arquitectura presenta varias ventajas:

- Hay una clara separación entre los componentes de un programa; lo cual nos permite implementarlos por separado.
- Hay un API<sup>2</sup> muy bien definido; cualquiera que use el API, podrá reemplazar el Modelo, la Vista o el Controlador, sin aparente dificultad.
- La conexión entre el Modelo y sus Vistas es dinámica; se produce en tiempo de ejecución, no en tiempo de compilación.
- Incrementa la reutilización y flexibilidad.
- Decrementa la duplicación de código.
- Centraliza el control de datos.
- Facilita el mantenimiento y modificación de la aplicación.
- Ayuda a equipos de desarrollo en centrarse en áreas específicas.
- Facilita la escalabilidad del sistema.
- Permite no mezclar lenguajes de programación en el mismo código.
- La conexión entre el Modelo y sus Vistas es dinámica.

Como desventajas tenemos:

- Exige una mayor complejidad y esfuerzo en el diseño, pues es necesario identificar bien las partes: Modelo, Vistas y Controlador, en el proceso de diseño.
- Los mensajes entre las diferentes partes pueden resultar difíciles de seguir si no se conoce el mecanismo de interacción del MVC. Esto puede ser contraproducente en un proceso de depuración de código.

---

<sup>2</sup> Es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usadas generalmente en las bibliotecas

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### 2.1.1.1 MVC EN APLICACIONES WEB

- **Vista:**
  - La página HTML.
- **Controlador:**
  - Es el intermediario entre la vista y el modelo.
  - Código que obtiene datos dinámicamente y genera el contenido HTML.
- **Modelo:**
  - La información almacenada en una base de datos o en XML
  - Junto con las reglas de negocio que transforman esa información (teniendo en cuenta las acciones de los usuarios) y permite el acceso a los datos.

### 2.1.2 PATRÓN MVC

Al hablar de una arquitectura Modelo Vista Controlador hacemos referencia a un patrón de diseño que especifica cómo se debe estructurar una aplicación, las capas que van a componer la misma y la funcionalidad de cada una.

Para realizar el diseño de aplicaciones con sofisticadas interfaces se utiliza el patrón de diseño Modelo-Vista-Controlador. La lógica de un interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica de negocio. Si se realiza un diseño que mezcle los componentes de interfaz y de negocio, entonces la consecuencia será que, cuando necesitemos cambiar el interfaz, tendremos que modificar trabajosamente los componentes de negocio. Mayor trabajo y más riesgo de error.

Se trata de realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio o de datos.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Elementos del patrón:

- Modelo: datos y reglas de negocio.
- Vista: muestra la información del modelo al usuario.
- Controlador: gestiona las entradas del usuario.

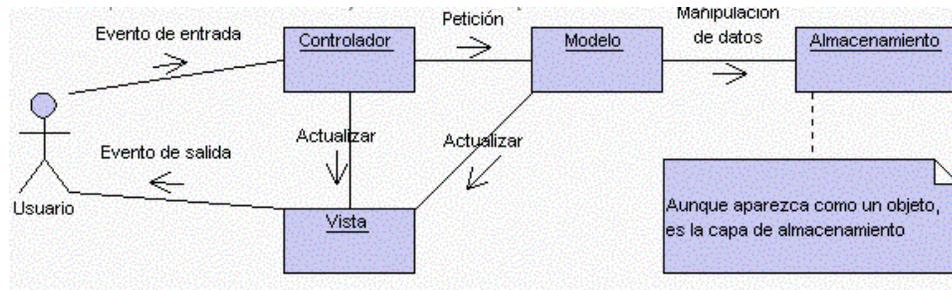


Figura N° 2. 2. Elementos del patrón MVC (Modelo – Vista – Controlador).

Fuente: <http://prestashop5estrellas.wordpress.com/2010/03/29/el-patron-mvc-modelo-vista-controlador/>

Un modelo puede tener diversas vistas, cada una con su correspondiente controlador. Un ejemplo clásico es el de la información de una base de datos, que se puede presentar de diversas formas: diagrama de tarta, de barras, tabular, etc. Veamos cada componente:

1. El **modelo** es el responsable de:

- 1.1. Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- 1.2. Define las reglas de negocio (la funcionalidad del sistema). Un ejemplo de regla puede ser: “Si la mercancía pedida no está en el almacén, consultar el tiempo de entrega estándar del proveedor”.
- 1.3. Lleva un registro de las vistas y controladores del sistema.
- 1.4. Si estamos ante un modelo activo, notificará a las vistas los cambios que en los datos pueda producir un agente externo (por ejemplo, un fichero bath que actualiza los datos, un temporizador que desencadena una inserción, etc.).

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

2. El **controlador** es responsable de:

- 2.1. Es quien recibe los eventos de entrada (como por ejemplo: un clic, un cambio en un campo de texto, etc.).
- 2.2. Contiene reglas de gestión de eventos, del tipo “Si Evento X, entonces Acción Y”. Estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada al método “Actualizar ()”. Una petición al modelo puede ser “Obtener\_tiempo\_de\_entrega (nueva\_orden\_de\_venta)”.

Ventajas:

- Hace que el desarrollo sea más sencillo y limpio.
- Facilita el posterior mantenimiento de la aplicación haciéndola más escalable.
- Facilita la detección de errores en el código.

El Controlador es implementado mediante un Servlet central que, dependiendo de la cantidad de tipos de peticiones que debe gestionar, puede apoyarse de otros servlets auxiliares para procesar cada petición.

3. Las **vistas** son responsables de:

- 3.1. Recibir datos del modelo y los muestra al usuario.
- 3.2. Tienen un registro de su controlador asociado (normalmente porque además lo instancia).
- 3.3. Pueden dar el servicio de “Actualización ()”, para que sea invocado por el controlador o por el modelo (cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes).

Un ejemplo de MVC con un modelo pasivo (aquel que no notifica cambios en los datos) es la navegación web, que responde a las entradas del usuario, pero no detecta los cambios en datos del servidor.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

El diagrama de secuencia:

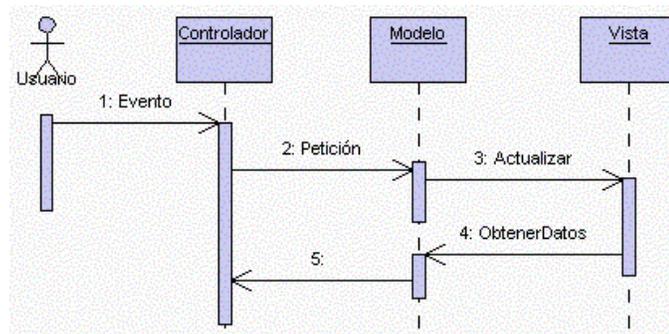


Figura N° 2. 3. Diagrama de secuencia patrón MVC (Modelo - Vista – Controlador).

Fuente: <http://prestashop5estrellas.wordpress.com/2010/03/29/el-patron-mvc-modelo-vista-controlador/>

Pasos:

1. El usuario introduce el evento.
2. El Controlador recibe el evento y lo traduce en una petición al Modelo (aunque también puede llamar directamente a la vista).
3. El modelo (si es necesario) llama a la vista para su actualización.
4. Para cumplir con la actualización la Vista puede solicitar datos al Modelo.
5. El Controlador recibe el control.

### 2.1.3 FRAMEWORKS MVC

#### PHP

- **Agavi**

Framework PHP basado en el patrón de diseño (Modelo Vista Controlador).

- **CakePHP**

CakePHP es un framework o marco de trabajo que facilita el desarrollo de aplicaciones web, utilizando el patrón de diseño MVC. Es de código abierto

y se distribuye bajo licencia MIT<sup>3</sup>. Cakephp está desarrollado en PHP con programación orientada a objetos.

- **Codelgniter**

Codelgniter es un potente framework PHP con un tamaño muy pequeño, construido para programadores PHP que necesitan un conjunto de herramientas sencillas y elegantes para crear todas las funciones de las aplicaciones web.

- **Qcodo**

Es un framework de desarrollo de código abierto para PHP 5, que construye un mapeo de objetos y una interfaz de ABMS (Alta, baja y modificación), utilizando AJAX desde un modelo existente de una base de datos. Es un framework de desarrollo liviano que puede ser utilizado tanto en pequeñas como en grandes aplicaciones. Incorpora el patrón MVC.

- **Seagull**

Este framework permite realizar una programación modular, posee un Sistema de Gestión de Contenidos (CMS).

Como características podemos observar: su compatibilidad con PHP 4 y PHP 5, ORM<sup>4</sup> integrado, incorpora el patrón MVC, uso de templates, soporte para múltiples bases de datos, validación de datos, alto nivel de configuración, autenticación, integración de librerías PEAR.

- **Symfony**

Optimiza la creación de las aplicaciones web, posee una librería de clases que permiten reducir el tiempo de desarrollo. Está desarrollado en PHP5, se

---

<sup>3</sup> Es una de tantas licencias de software. El texto de la licencia no tiene copyright, lo que permite su modificación.

<sup>4</sup> El mapeo objeto-relacionales una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, utilizando un motor de persistencia.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

lo puede utilizar en plataformas Unix, Linux y Windows. Incorpora el patrón MVC, soporta AJAX, plantillas y un gran número de bases de datos.

- **Tlalokes**

Tlalokes es un framework libre y moderno escrito en PHP, es orientado a objetos, permite generar rápidamente aplicaciones web de una manera sencilla y rápida, cuenta con una interface web por lo que no requiere línea de comandos para generar código, soporta múltiples bases de datos.

- **Zend Framework**

Es un framework de código abierto para desarrollar aplicaciones web y servicios web con PHP 5. ZF es una implementación que usa código 100% orientado a objetos. La estructura de los componentes de ZF es algo único; cada componente está construido con una baja dependencia de otros componentes. Esta arquitectura débilmente acoplada permite a los desarrolladores utilizar los componentes por separado. A menudo se refiere a este tipo de diseño como "use-at-will" (uso a voluntad).

Aunque se pueden utilizar de forma individual, los componentes de la biblioteca estándar de Zend Framework conforman un potente y extensible framework de aplicaciones web al combinarse. ZF ofrece un gran rendimiento y una robusta implementación MVC, una abstracción de base de datos fácil de usar, y un componente de formularios que implementa la prestación de formularios HTML, validación y filtrado para que los desarrolladores puedan consolidar todas las operaciones usando de una manera sencilla la interfaz orientada a objetos.

## **ASP.Net**

- **ASP.NET MVC**

El ASP.NET MVC Framework es un framework de aplicaciones web que implementa el modelo – vista - controlador del patrón. Sobre la base de

ASP.NET, que permite a los desarrolladores de software para construir una aplicación web como una composición de tres funciones: Modelo, Vista y Controlador.

- **Castle**

Es un proyecto de código abierto para .Net, que aspira a simplificar el desarrollo de aplicaciones empresariales y web. Ofreciendo un conjunto de herramientas (trabajo en conjunto o por separado) y la integración con otros proyectos de código abierto, ayuda a hacer más con menos código y en menos tiempo.

## **JAVA**

- **Apache Cocoon**

Usualmente llamado simplemente Cocoon, es un framework de desarrollo web construido en los conceptos de desarrollo web separando componentes y preocupaciones. El framework se enfoca en la publicación de XML y XSLT y está construido usando el lenguaje de programación Java.

- **Aurora**

Aurora es un marco de código abierto MVC que está destinada a la escritura configurable, totalmente orientado a objetos.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- **Grails**

Es un framework para aplicaciones web libre desarrollado sobre el lenguaje de programación Groovy (el cual a su vez se basa en la plataforma Java).

- **Jaffa**

Marco de Aplicaciones Java Para Todos es un marco empresarial centrado en Java para el desarrollo rápido de aplicaciones. Proporciona una completa aplicación Java Web pila de desarrollo, ampliando los proyectos existente como Tomcat y Struts en la interfaz con widgets web basado en MVC, todo el camino a un alto rendimiento ligero O / R motor de persistencia, con un montón de características ricas en el medio incluyendo los motores de reglas, aplicación declarativa, y seguridad de los datos.

- **Java Server Faces**

Java Server Faces es un framework o ambiente de trabajo, destinado a facilitar la construcción y desarrollo de interfaces para aplicaciones basadas en entornos Web. JSF tiene la función de proveer un grupo muy extenso de herramientas predesarrolladas para facilitar la creación de interfaces; herramientas tales como: botones, cajas de texto, tablas de datos.

- **Maverik**

Es un modelo-vista-controlador (también conocido como "Modelo 2") marco para la publicación en la Web utilizando Java y J2EE. Es un marco minimalista que se centra únicamente en la lógica de MVC, que le permitirá generar la presentación usando una variedad de plantillas y de las tecnologías de transformación.

- **Spring**

Framework de código abierto para el desarrollo de aplicaciones (empresariales) Java. Spring dispone de muchas opciones para configurar nuestras aplicaciones. La más popular de ellas es la utilización de ficheros XML. Esta es una forma tradicional que es soportada desde las primeras versiones de Spring.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- **Struts**

Es un framework de la capa de presentación que implementa el patrón MVC en Java. Evidentemente, como todo framework intenta simplificar notablemente la implementación de una arquitectura según el patrón MVC. El mismo separa muy bien lo que es la gestión del workflow<sup>5</sup> de la aplicación, del modelo de objetos de negocio y de la generación de interfaz.

- **Tapestry**

Es un framework de código abierto para la creación de aplicaciones web de forma dinámica, robusta y altamente escalable en Java. Complementa y construye desde el estándar Java Servlet API, funcionando también en cualquier servidor contenedor de servlets o contenedor de aplicaciones. Tapestry divide una aplicación web en un conjunto de páginas, cada una compuesta de componentes. Esto le otorga una estructura consistente, permitiendo a Tapestry asumir responsabilidades clave como la construcción y envío de URLs, almacenamiento del estado persistente en el cliente o en el servidor, validación de entradas de usuario, localización/internacionalización, y reporte de excepciones.

## RUBY

- **Anvil**

Marco para el desarrollo de aplicaciones basadas en interfaz gráfica de usuario de Ruby.

- **Ramaze**

Ramaze utiliza el Modelo Vista Controlador (MVC) para organizar la arquitectura de programación de aplicaciones. Sin embargo, debido a la forma del funcionamiento de Ramaze, es muy fácil de utilizar una estructura completamente diferente para su aplicación. De hecho, la posibilidad de reconfigurar la manera de cómo trabaja el framework es una de las cosas que hace a Ramaze diferente en comparación con otros frameworks.

---

<sup>5</sup> Se refiere al flujo de trabajo a seguir para la consecución de una tarea o trabajo predeterminado. Se define como un sistema de secuencia de tareas de un proceso de negocio.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- **Ruby on Rails**

Es un framework de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma de la arquitectura Modelo Vista Controlador (MVC). Trata de combinar la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros frameworks y con un mínimo de configuración

## PERL

- **Catalyst**

Es una estructura de código libre para aplicaciones web escrito en Perl. Soporta una arquitectura MVC, así como también algunos patrones web experimentales. Está altamente inspirado en Ruby on Rails, Maypole y Spring.

Catalyst promueve el re-utilizamiento de los módulos de Perl que ya soportan bien lo que requieren las páginas Web. Catalyst provee ayudas para simplificar el control de flujo y mapeo de URLs para los métodos de Control, tiene una gran variedad de plugins.

## PYTHON

- **Django**

Es un framework de desarrollo web de código abierto, escrito en Python, que cumple en cierta medida el paradigma del Modelo Vista Controlador.

- **Turbogears**

Es un framework para el desarrollo de aplicaciones web. Permite el desarrollo de aplicaciones web de una manera rápida, sencilla y divertida. Turbogears se basa en el patrón MVC.

- **Zope 3**

Es un framework de aplicaciones web, es open source, para el desarrollo de aplicaciones de manejo de contenido, intranets, portales.

## **2.2 JAVA SERVER FACES**

### **2.2.1 DEFINICIÓN DE JSF**

JSF es una tecnología y framework que permite crear aplicaciones java J2EE, las cuales están basadas en el patrón MVC. Las características principales de JSF son:

- Uso de páginas JSP para generar las vistas, añadiendo una biblioteca de etiquetas propia para la creación de los elementos de los formularios HTML.
- Cada vista se asocia con formularios, un conjunto de objetos java que serán manejados por el controlador (managed beans) que facilitan la recolección, manipulación y visualización de los valores mostrados en los diferentes elementos de los formularios.
- Introduce una serie de etapas en el procesamiento de la petición, como por ejemplo la validación, reconstrucción de la vista, recuperación de valores de los elementos, etc.
- Utiliza un fichero de configuración para el controlador en formato XML.
- Extensible, permitiendo la creación de nuevos elementos de la interfaz o modificar los ya existentes.
- Forma parte del estándar J2EE.

### **2.2.2 COMPONENTES PRINCIPALES DE JSF**

Una aplicación JSF tiene como mínimo los siguientes componentes:

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

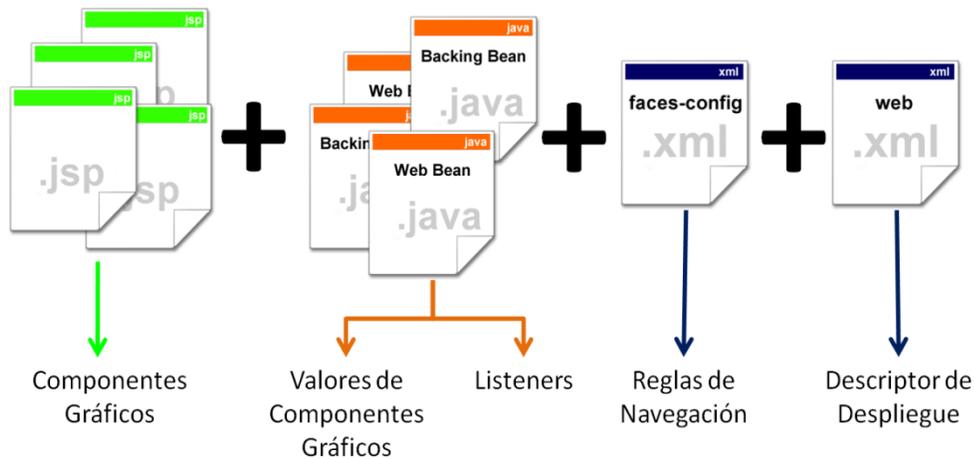
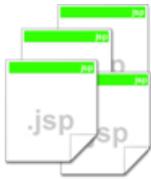
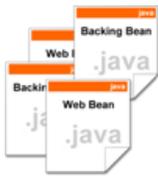


Figura N° 2. 4 Componentes de una aplicación JSF.

Fuente: <http://cupi2.uniandes.edu.co/sitio/index.php/tutoriales/jsf?start=10>



Las páginas de interacción JSPs hacen parte de la tecnología Java Server Pages que permite utilizar HTML generado dinámicamente. Esta tecnología extiende el modelo de tecnología de Servlets. El lenguaje se basa en tags y existen librerías que proporcionan la implementación de páginas JSP incluyendo las librerías JSF para poder referenciar los elementos.



Las clases de Java Beans, dentro del modelo MVC, son las clases que actuarán como controladores de las acciones. Cada uno de estos beans debe representar una entidad del mundo que quiere ser presentada de manera que el bean (o entidad web) contienen información que se puede presentar en las páginas jsp.

Todos los beans que se asocian con elementos gráficos deben tener métodos get/set de sus atributos (utilizando los prefijos en inglés) pues se enlazan

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

directamente con el nombre dado que el framework interpretará la página y buscará el método get/set correspondiente.

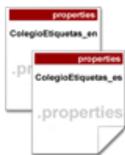


El archivo faces-config.xml define la navegación, los beans de java que utilizarán JSF y los archivos de propiedades para internacionalización. La navegación se define a partir de un punto de partida que corresponde a una acción y un punto de llegada que es una visualización del resultado de la acción y corresponde a una página jsp. La configuración de los beans establece el nombre con el que se invocan sus acciones en las páginas jsp y la ruta completa del paquete en el que se encuentra la clase que representa el backing bean.

JSF se encuentra desarrollado por arriba de los servlets, de manera que toda aplicación JSF es JSP/servlet donde cada página contiene componentes web JSF que encapsulan HTML. Es así como cada petición HTTP que requiere JSF pasa por un servlet de Faces, el cual hace parte del modelo de tecnología de este marco de trabajo.



El archivo web.xml es una modificación del web.xml estándar para indicar el tipo de solicitudes que serán procesadas por el servlet de JSF. El servlet hará la intercepción de todas las peticiones antes de invocar la página respectiva a la que le prepara un contexto en el servidor de aplicaciones para la información asociada a este.



Los archivos de propiedades para internacionalización contienen la información de configuración del texto y los mensajes en cada idioma. Las páginas se construyen utilizando etiquetas que hacen referencia a las llaves de estos

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

archivos, una vez se procesan las solicitudes y se conoce el idioma en el que se quiere construir la respuesta, se reemplazan estas etiquetas por el valor obtenido del archivo correspondiente. Cambiando la configuración de idioma del explorador de internet, se utilizará un archivo o el otro.

### **2.2.3 DEFINICIÓN DE UNA APLICACIÓN JSF**

Las aplicaciones JSF vienen del acrónimo JavaServer Faces, son como cualquier otra aplicación web Java. Estas aplicaciones se ejecutan en un contenedor de *servlets* de Java, estas aplicaciones contienen:

- Componentes *JavaBeans* (llamados también objetos del modelo en tecnología JavaServer Faces) conteniendo datos y funcionalidades específicas de la aplicación.
- Oyentes de Eventos.
- Páginas, (principalmente páginas JSP).
- Clases de utilidad del lado del servidor, como *beans* para poder acceder a las bases de datos.

Una aplicación JavaServer Faces también tiene:

- Una librería de etiquetas personalizadas para dibujar los componentes UI en una página.
- Una librería de etiquetas personalizadas para representar los manejadores de eventos, validadores y otras acciones.
- Componentes UI representados como objetos con estado en el servidor.

Debemos tener en cuenta que toda aplicación JavaServer Faces debe incluir una librería de etiquetas personalizadas, la cual define las etiquetas que representan componentes UI, así como una librería de etiquetas, las cuales controlan otras acciones importantes, como validadores y manejadores de eventos. La implementación de JavaServer Faces, de Sun provee estas dos librerías. La librería de etiquetas de componentes elimina la necesidad de codificar

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

componentes UI en HTML u otro lenguaje de marcas, lo que se traduce en el empleo de componentes completamente reutilizables. Y la librería principal (*core*) hace fácil registrar eventos, validadores y otras acciones de los componentes.

Se puede usar la librería **html\_basic** como librería de etiquetas de componentes incluida, con la implementación de referencia de la tecnología JavaServer Faces, también es posible definir una librería de etiquetas personalizadas que dibuje componentes propios o que proporcione una salida distinta a HTML.

Una de las ventajas importantes de las aplicaciones JavaServer Faces es que los componentes UI de la página están representados en el servidor como objetos con estado. A la aplicación le permite manipular el estado del componente y conectar los eventos generados por el cliente a código en el lado del servidor.

JavaServer Faces es una tecnología que permite convertir y validar datos sobre componentes individuales, así como también informar de cualquier error antes de que los datos se actualicen en el lado del servidor. El desarrollo y mantenimiento de una aplicación JavaServer Faces se la realiza de una manera rápida y fácil.

Ventajas de JavaServer Faces:

1. Una de las ventajas de utilizar la tecnología de JSF, es que permite la separación limpia entre la presentación y el comportamiento (lógica de negocio).
2. JavaServer Faces es muy flexible. Por ejemplo, nos permite crear nuestros propios componentes, crear nuestros propios renderizadores para pintar los componentes en la forma que más nos convenga.
3. JavaServer Faces provee una arquitectura rica para manejar el estado de los componentes, procesando datos de los componentes, validando entradas de usuarios y manejando eventos.
4. Permite la utilización de varias tecnologías de presentación junto a JSP.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

5. JavaServer Faces ofrece una gran cantidad de componentes open source para las funcionalidades que se necesiten.

JavaServer Faces presenta dos nuevos términos:

- Managed bean: un managed bean describe como se crea y se maneja un bean. No tiene nada que ver con las funcionalidades del bean.
- Backing bean: define las propiedades y la lógica de manejo asociadas con los componentes UI utilizados en la página. Cada propiedad del bean de respaldo está unida a un ejemplar de un componente o a su valor. Un backing bean también define un conjunto de métodos que realizan funciones para el componente, como validar los datos del componente, manejar los eventos que dispara el componente y realizar el procesamiento asociado con la navegación cuando el componente se activa.

Desventajas de JavaServer Faces:

- Interfaz poco llamativa.

#### **2.2.4 ESTRUCTURA DE UNA APLICACIÓN JSF**

- /JavaSource: Fuentes de las clases Java.
- /ant: Scripts de creación, (build.xml).
- /webContent: Archivos de la aplicación web usados por el servidor de aplicaciones.
- /Web-INF: archivos que se utilizan como parte de la rutina de la aplicación Web.
  - /clases.
  - /lib: jsf-impl.jar: contiene el paquete com.sun.faces, jsf-api.jar: contiene el paquete javax.faces.
  - Web.xml.
  - Faces-config.xml: este archivo jugará el papel de archivo de configuración de JSF.
- /pages: archivos jsp para la presentación de la interfaz.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### **2.2.5 PASOS PARA DESARROLLAR UNA APLICACIÓN JSF**

1. Desarrollar los objetos del modelo: estos son los objetos que contienen los datos.
2. Añadir las declaraciones del bean controlado al archivo de configuración de la aplicación.
3. Crear páginas.
4. Definir la navegación entre páginas, ésta se define en el archivo faces.

### **2.2.6 CICLO DE VIDA DE UNA PÁGINA JSF**

El ciclo de vida de una página JSF:

1. Restaurar vista.
2. Aplicar valores de la petición.
3. Procesar validaciones.
4. Actualizar modelo.
5. Invocar aplicación.
6. Producir respuesta.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

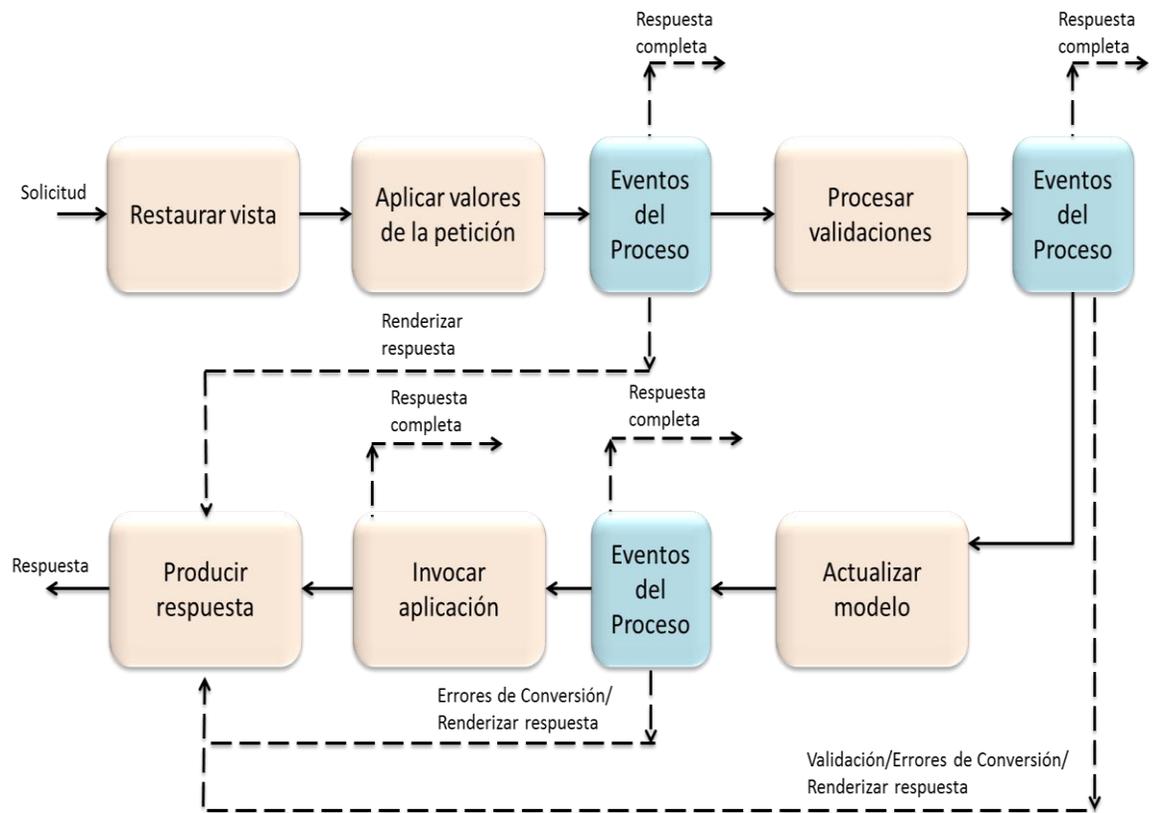


Figura N° 2. 5. Ciclo de vida de una página JSF.

Fuente: <http://www.edsongoncalves.com.br/tag/jsf-2-0/>

## 1. Fase del ciclo de vida: Restaurar vista

En esta fase del ciclo se crea un árbol de componentes. Puede crearse a partir de información existente (en sesión o en un campo hidden<sup>6</sup>, según se haya configurado) o de cero si no había información. Si no se encuentran datos POST o "query string", se pasa directamente a "Producir respuesta".

<sup>6</sup> Es un atributo que se le pone al tag. Gracias a este atributo estamos enviando al programa de gestión de datos, aparte de los datos enviados por el propio usuario, datos predefinidos por nosotros mismos invisibles para el usuario.

## **2. Fase del ciclo de vida: Aplicar valores de la petición**

En esta fase del ciclo es donde se almacenan los valores enviados por el cliente en sus respectivos componentes. Aquí se llama a `processDecodes()` recursivamente en cada componente para que actualice sus datos.

Dentro de `processDecodes()` primero se llama a `processDecodes()` en los hijos y luego a `decode()` en uno mismo. El método `decode()` invoca al convertidor asociado al componente si lo hay. Si la decodificación falla se llama a `setValid()` y a `addMessage()` para marcar el componente como erróneo

Si cualquier conversión falla o si algún componente llama a `renderResponse()` el ciclo de vida salta directamente a "Producir respuesta".

En esta fase todos los valores devueltos se guardan en los componentes (no en el bean) como String (que es lo único que sabe devolver el método `getParameter` de la `HttpServletRequest`). Después se intentan convertir mediante `Converters` a un tipo Java (las fechas de String a Date) y se vuelven a guardar en el componente. Esto se hace para que, si falla la conversión, el componente siga teniendo guardado (como String) el valor que tecleo el usuario y, en la fase de producir respuesta, se pueda repintar otra vez, junto con el error de conversión.

## **3. Fase del ciclo de vida: Procesar validaciones**

En esta fase se llama a `processValidators()` recursivamente (profundidad primero) en cada componente para que actualice sus datos.

Dentro de `processValidators()` primero se llama a `processValidators()` en los hijos y luego a `validate()` en uno mismo. El método `validate()` invoca a los validadores definidos y, si alguno falla, llamara a `setValid(false)` y a `addMessage()` y el ciclo de vida saltará a "Producir respuesta".

## **4. Fase del ciclo de vida: Actualizar modelo**

En esta fase se llama a `processUpdates()` recursivamente (profundidad primero) en cada componente para que actualice sus datos.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Dentro de processUpdates() primero se llama a processUpdates() en los hijos y luego a updateModel() en uno mismo. El método updateModel() actualiza los beans de modelo asociados al componente (normalmente mediante el atributo value del tag del componente).

En esta fase se pasan los valores almacenados en los componentes (y ya convertidos y validados) al bean.

### **5. Fase del ciclo de vida: Invocar aplicación**

En esta fase se procesan todos los ActionEvents encolados.

### **6. Fase del ciclo de vida: Producir respuesta**

En esta fase del ciclo, los componentes y sus Renderers se encargan de convertir el árbol de componentes en código visualizable (normalmente HTML).

La producción se realiza mediante la subclase de ViewHandler registrada en JSF. Todas las implementaciones de JSF deben tener un ViewHandler por defecto (normalmente un ViewHandler que procesara páginas JSP).

En esta fase se llama a encodeBegin() en cada componente para generar su contenido. Después, si el método rendersChildren() del componente devuelve true. Se llama a encodeChildren() y, finalmente a encodeEnd().

Después de producir la respuesta JSF serializa el árbol de componentes para la siguiente petición Esta serialización se puede guardar en sesión o en un campo hidden (configurable).

### **2.2.7 REQUERIMIENTOS PARA TRABAJAR CON JSF**

- Las librerías propias de JSF: jsf-api.jar y jsf-impl.jar. La primera contiene el paquete javax.faces.\* y la segunda com.sun.faces.\*.
- Las librerías comunes de Apache-Jakarta.
- Además las librerías adicionales standard.jar y jstl.jar.
- Java EE.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### **2.2.8 HERRAMIENTAS PARA LA CREACIÓN Y UTILIZACIÓN DE JSF**

**Eclipse Galileo:** es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", esta herramienta nos permite la creación de aplicaciones JSEE y generar aplicaciones JavaServer Faces.

**Java Studio Creator:** es una solución completa de desarrollo, depuración y despliegue de aplicaciones, que incorpora la primera versión de la tecnología de producción JavaServer Faces (JSF), capaz de proporcionar un mayor rendimiento y eficiencia en la codificación. El nuevo producto también incorpora:

El runtime de Java Enterprise System, que permite a los desarrolladores evaluar las aplicaciones en preproducción de forma rápida. Incluye soporte para Sun Java System Application Server, Platform Edition.

- PointBase, un servidor de bases de datos SQL (incluido en Java System Application Server).
- El kit de desarrollo (SDK) para Java 2, Standard Edition (J2SE).
- Elementos de ayuda para desarrolladores, tales como ejemplos, tutoriales, componentes visuales para la tecnología JavaServer Faces, etc.

**NetBeans:** permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados *módulos*. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándoles nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

## **CAPÍTULO III**

---

### **3.1 ANÁLISIS DE LOS ENTORNOS DE DESARROLLO ECLIPSE, JBUILDER Y NETBEANS**

#### **3.1.1 INTRODUCCIÓN**

Un entorno de desarrollo integrado o IDE (Integrated Development Environments) aporta funcionalidades al desarrollador durante todas las etapas del ciclo de vida del desarrollo de software, desde el análisis y diseño a la distribución del producto y su mantenimiento, de ahí la palabra "integrado". En la práctica, solamente los IDEs más modernos cumplen esta condición y, a menudo, de forma incompleta.

Cualquier entorno actual de desarrollo integrado ofrece, al menos, el control del editor de código, del compilador y del depurador desde una única interfaz de usuario. Su misión consiste en evitar tareas repetitivas, facilitar la escritura de código correcto, disminuir el tiempo de depuración e incrementar la productividad del desarrollador. Estas tareas pueden realizarse de muchas maneras distintas: mediante la inclusión de asistentes para las tareas más habituales y mecánicas, de editores que completen automáticamente el código y señalen los errores sintácticos, de gestores de archivos fuente, etc.

Actualmente existen muchos entornos de desarrollo para Java: Eclipse, NetBeans, JBuilder, Sun ONE Studio, Visual Age for Java, WebSphere Studio, Oracle JDeveloper, JavaStudio, Visual J++, entre otros.

A continuación se hará una descripción y comparación de los tres primeros IDEs, teniendo en cuenta que ellos son las más populares y utilizados actualmente para el desarrollo de aplicaciones Java.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### 3.1.1.1 ECLIPSE



Figura N°3.1. IDE Eclipse

Fuente: <http://flosweb.deviantart.com/art/Eclipse-Icon-10690728>

Eclipse es un proyecto de código abierto creado por IBM. Proporciona una plataforma para construir IDEs que se puedan utilizar para crear aplicaciones web, programas C++, aplicaciones basadas en EJBs, etc. Con Eclipse también vienen herramientas de terceras partes que extienden la capacidad de la plataforma que se distribuyen con la licencia propia de sus fabricantes.

WebSphere Studio Worbench es un producto de IBM que está construido sobre Eclipse, pero a diferencia de éste, no es de código abierto.

**Desarrollador:** IBM

**Licencia:** CPL Versión 1.0 (compatible con la definición de Software de Código Abierto)

**Versión actual:** 2.1.2

**Sistemas operativos soportados:** Windows 98/ME/2000/XP, Linux, Solaris 8, QNX, AIX, HP-UX, Mac OSX.

**Requerimientos hardware:**

- **Espacio en disco duro:** 125 Mbytes
- **Memoria:** 256 Mbytes
- **Procesador:** 500 MHz

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

**Requerimientos software:** JRE o JDK versión 1.3 o superior.

**Descripción:**

Eclipse es un proyecto de desarrollo de software de fuente abierta dedicado a proporcionar una plataforma robusta, funcional y de calidad comercial para el desarrollo de herramientas integradas. Está compuesto de los siguientes proyectos:

- ❖ El Proyecto Eclipse: está compuesto a su vez por tres subproyectos: Plataforma, Herramientas de Desarrollo Java (JDT-Java Development Tools) y el Entorno de Desarrollo de Plug-ins (PDE- Plug-in Development Environment).
  - Plataforma: proporciona la infraestructura núcleo y los servicios básicos bajo los cuales se crean las extensiones plug-in. También proporciona el runtime en el cual se cargan, integran y ejecutan los plug-ins.
  - Proyecto JDT: proporciona los plug-ins que implementan un IDE Java (el IDE Java soporta el desarrollo de cualquier aplicación Java incluyendo plug-ins Eclipse). Adiciona un gran número de vistas, editores, asistentes, constructores (builders) y herramientas de refactorización de código. El proyecto JDT es el que permite que Eclipse sea un entorno de desarrollo por sí mismo.
  - Proyecto PDE: proporciona un número de vistas y editores que hacen más fácil construir plug-ins para Eclipse. Permite crear archivos xml para los plug-ins, especificar el Runtime del plug-in y otros plug-ins requeridos, definir puntos de extensión, crear extensiones sobre otros puntos de extensión, etc. PDE permite que la integración de plug-ins sea fácil y rápida.
  
- ❖ El proyecto Herramientas Eclipse: busca impulsar la creación de una amplia variedad de herramientas para la plataforma Eclipse coordinando a los desarrolladores de herramientas de fuente abierta, con el fin de minimizar que

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- ❖ los trabajos se traslapen o dupliquen, asegurar la creación y utilización de componentes comunes y promover interoperabilidad entre diversos tipos de herramientas.
- ❖ El Proyecto Tecnología Eclipse: proporciona nuevos canales para que los desarrolladores de aplicaciones de fuente abierta, investigadores y educadores participen en la continua evolución de Eclipse. Los proyectos de investigación exploran problemas en dominios relevantes para Eclipse como lenguajes de programación, herramientas y entornos de desarrollo. Las incubadoras son proyectos pequeños, estructurados informalmente que adicionan nuevas capacidades a la base de software Eclipse. Los proyectos de Educación se enfocan en el desarrollo de materiales y ayudas educativas.

Según lo anterior Eclipse no es solamente un IDE para Java, sino también una plataforma de integración de herramientas de desarrollo y una infraestructura para el desarrollo de aplicaciones.

#### **3.1.1.1 Eclipse como IDE**

La plataforma Eclipse junto con el JDT forman un IDE para Java. Por sí sola, la plataforma Eclipse proporciona funcionalidades demasiado genéricas. Solamente cuando se amplía con otras herramientas (el JDT en este caso, que hace parte de la distribución estándar de Eclipse y que es un plug-in formado por otros plug-ins), permite desarrollar actividades útiles para los usuarios que la vayan a utilizar como IDE.

Como IDE de Java, Eclipse posee un editor muy visual con sintaxis coloreada, ofrece compilación incremental de código, un potente depurador (que permite establecer puntos de interrupción, modificar e inspeccionar valores de variables, etc. e incluso depurar código que resida en una máquina remota), un navegador de clases, un gestor de archivos y proyectos, entre otros. La versión estándar de Eclipse proporciona también una biblioteca de refactorización de código, soporta la integración con JUnit y proporciona soporte para Ant.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

También incluye una herramienta para completar código (el asistente de contenido, encargado de mostrar los métodos y atributos de las clase), asistentes para la creación de clases e interfaces, y proporciona una integración con CVS. Además Eclipse puede utilizarse con otros lenguajes además de Java basta con usar los plug-ins adecuados.

Java 7 Development Kit (o JDK) es el kit de desarrollo oficial del lenguaje de programación Java, uno de los más populares entre los orientados a objetos. En la actualidad, Java 7 Development Kit es el entorno de desarrollo más popular para Java. No incluye IDE gráfico (como Eclipse), así que deberás instalarlo por separado.

Además de la máquina virtual de Java, indispensable para ejecutar las clases de los programas, Java 7 Development Kit cuenta con un amplio surtido de herramientas, como javac, el compilador de bytecode de Java, javap, el desensamblador de clases, y jdb, el depurador de bugs. Las encontrarás en el subdirectorio bin del Java 7 Development Kit.

Indispensable para todo programador de Java, el Java 7 Development Kit se actualiza con frecuencia y contiene ejemplos y demostraciones para que puedas empezar en seguida con él.

Pros:

- Es el kit básico para programar en Java
- Incluye demostraciones y ejemplos
- Instalación rápida y sin complicaciones
- Abundante documentación en línea

Contras:

- No incluye ningún entorno gráfico. Hay que descargarlo por separado.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Otra característica de Eclipse, muy eficiente para reducir los tiempos de depuración y pruebas, es la compilación incremental automática del código. Eclipse no cuenta con un menú de compilación pues no es necesario: cada vez que se hacen cambios en uno o más archivos, el compilador interno de Eclipse recompila todos los archivos fuente afectados por los cambios. El usuario no tiene que preocuparse de compilar, y puede estar seguro de contar con archivos siempre compilados. En consecuencia, tampoco resulta preciso esperar a la compilación para detectar ciertos errores: Eclipse muestra indicaciones de los errores aparecidos según se van realizando o guardando los cambios.

### 3.1.1.1.2 Servidores de Aplicaciones

Nombre	Licencia	Descripción
<b>Tomcat Manager</b>	Desconocida	Plug-in Eclipse para manejar Tomcat.
<b>Easy Struts</b>	ASF	El objetivo de este proyecto es proporcionar un conjunto de herramientas de desarrollo para facilitar el desarrollo de un proyecto web basado en la infraestructura MVC2 Jakarta Struts.
<b>Glider for Eclipse</b>	Comercial	Contenedor de aplicaciones J2EE.
<b>Improve Resin Plugin</b>	CPL 0.5	Permite compilar JSPs y Servlets.
<b>Jetty Launcher</b>	MPL1.1/GPL/LGPL	Permite lanzar proyectos de aplicaciones web rápidamente en Eclipse.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

<b>JOPE</b>	Gratis	Proporciona un entorno completo para desarrollar aplicaciones JonAS sobre Eclipse.
<b>SOFIA</b>	CPL1.0	Es un conjunto de herramientas RAD para J2EE.
<b>SolarEclipse</b>	Gratis	Proporcionan un entorno de desarrollo de aplicaciones Web (XML/HTML/JSP).
<b>Sysdeo Eclipse Tomcat Launcher</b>	Gratis	Plug-in para manejar Tomcat.
<b>WASP Developer</b>	Gratis para uso no comercial	Proporciona un entorno de desarrollo completo para la creación, prueba y despliegue de Servicios Web.
<b>WAST</b>	CPL1.0	Es una infraestructura para desarrollar adaptadores de servidores de aplicaciones.
<b>Tomcat 4.1 for WAST</b>	CPL1.0	Adaptador del servidor Tomcat 4.1 para el plug-in WAST

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

<b>WebApp</b>	CPL1.0	Ayuda al desarrollo de aplicaciones web dentro de Eclipse.
---------------	--------	------------------------------------------------------------

Tabla N° 3. 1. Descripción Servidores de Aplicación.

### 3.1.1.1.3 Bases de datos

Nombre	Licencia	Descripción
<b>Persistence for Eclipse</b>	Comercial	Soporta la conversión objeto a relacional y generación de código.
<b>Clay Database Modelling</b>	Gratis	Proporciona soporte para diseño visual de bases de datos, diferentes lenguajes SQL (HSQLDB, MySQL, PostgreSQL, Firebird, ANSI SQL-92, SAP DB, McKoi), generación de código SQL y validación del modelo de bases de datos.
<b>CocoBase</b>	Comercial	Soporta un gran número de métodos para mapear entre objetos y tablas.
<b>DBCOPY</b>	CPL1.0	Es un plug-in para migrar datos desde cualquier base de datos a otra con solo seleccionar visualmente las tablas deseadas en la base de datos fuente (cualquier tipo de bases de datos soportadas por JDBC).
<b>DBExplorer</b>	Gratis	Busca y analiza cualquier base de datos relacional accesible a través de JDBC.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

<b>EasySQL</b>	LGPL	Contiene un editor y una herramienta para ejecutar sentencias SQL.
<b>OraExp</b>	GPL	Plug-in Eclipse para desarrollo de Oracle, incluyendo Java, PL/SQL, y editores para todos los objetos Oracle.
<b>QuantumDB</b>	Gratis	Permite ejecutar sentencias SQL en una base de datos.
<b>SQLDeveloper</b>	CPL1.0	Muestra información de bases de datos altamente detallada y proporciona un editor de queries con capacidad de completar código. Soporta Oracle, MySQL y DB2

Tabla N° 3. 2. Descripción Base de Datos.

#### 3.1.1.1.4 Plataforma de desarrollo J2EE

<b>Nombre</b>	<b>Licencia</b>	<b>Descripción</b>
<b>EclipseJ2EE</b>	Gratis para uso no comercial.	Permite crear, desplegar y mantener aplicaciones de alta calidad a un costo más bajo.
<b>Exadel Struts Studio</b>	Comercial	Plug-in para desarrollo con Struts y J2EE
<b>JBoss-IDE</b>	CPL1.0	Permite monitorear los servidores Jboss y controlar su ciclo de vida, así como también proporciona soporte para Xdoclet.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

<b>Lomboz</b>	Gratis	Soporta el ciclo de desarrollo completo de aplicaciones J2EE.
<b>MyEclipse Enterprise Workbench</b>	Comercial	Proporciona herramientas para los desarrolladores de Java y J2EE.

Tabla N° 3. 3. Descripción Plataforma de desarrollo J2EE.

### 3.1.1.1.5 Lenguajes

Nombre	Licencia	Descripción
<b>JavaScript Editor</b>	Gratis	Editor JavaScript.
<b>CDT</b>	LGPL	Implementa un IDE C/C++.
<b>Cobol</b>	LGPL	Implementa un IDE COBOL.
<b>CSDT</b>	Desconocida	Implementa un IDE C#
<b>EclipseME</b>	Licencia académica gratis	Permite el desarrollo de aplicaciones J2ME, incluyendo emulación y debugging.
<b>EPIC</b>	CPL1.0	Colección de plug-ins Perl para Eclipse.
<b>php Plugin</b>	Gratis	Proporciona soporte para PHP.
<b>PYDEV</b>	CPL1.0	Proporciona un entorno de desarrollo Python.

Tabla N° 3. 4. Descripción Lenguajes

### 3.1.1.1.6 Modelamiento

Nombre	Licencia	Descripción
<b>Flow4J</b>	BSD	Permite el modelamiento de los flujos del proceso en una manera drag and drop.

Tabla N° 3. 5. Descripción Modelamiento.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### 3.1.1.1.7 Investigación

Nombre	Licencia	Descripción
<b>Dsateclipse</b>	Unknown	Permite construir sistemas distribuidos. Actualmente proporciona soporte para Jini, en un futuro se soportará tecnologías como JXTA y CORBA.

Tabla N° 3. 6. Descripción Investigación.

### 3.1.1.1.8 Interfaces de usuario

Nombre	Licencia	Descripción
<b>SWT GUI Builder</b>	Comercial	Permite construir aplicaciones GUI basadas en SWT.
<b>Advanced Eclipse SWT Designer</b>	Gratis para uso no comercial	Permite construir aplicaciones GUI basadas en SWT.
<b>Eclipse V4ALL Swing &amp; SWT</b>	CPL1.0	Diseñador de GUIs basadas en Swing y SWT.
<b>EclipseVisualEditor</b>	CPL1.0	Es un proyecto para crear diseñadores de GUIs para Eclipse. Incluirá implementaciones de referencia de constructores GUI SWT y Swing/JFC, pero busca ser útil para crear diseñadores de GUIs para otros lenguajes como C/C++ y conjuntos de componentes gráficos diferentes, incluyendo los que no son soportados bajo Java.

Tabla N° 3. 7. Descripción Interfaz de Usuario.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### 3.1.1.2 NETBEANS



Figura N°3.2. IDE Netbeans

Fuente: <http://www.redeszone.net/2012/07/27/netbeans-7-2-nueva-version-de-este-popular-ide/>

El entorno de desarrollo netBeans surgió como proyecto en el año 1996, por lo que como es de suponer, es un proyecto estable y maduro. Su licencia es la Sun Public License, compatible con la definición Open Source, y que permite el uso del entorno para proyectos libres y comerciales.

Sun ONE Studio (también llamado Forte para Java) está basado en NetBeans.

- **Desarrollador:** SUN
- **Licencia:** SPL - Sun Public License (compatible con la definición de Software de Código Abierto)
- **Versión actual:** 6.8
- **Sistemas operativos soportados:** Microsoft Windows XP, Microsoft Windows Vista, Microsoft Windows 7, Ubuntu 9.04, Solaris OS, Macintosh OS
- **Requerimientos hardware:**

➤ **Configuraciones de hardware mínimas**

- Microsoft Windows XP Professional SP3:
  - Procesador: Intel Pentium III o equivalente a 800 MHz.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- Memoria: 512 MB.
- Espacio de disco: 750 MB de espacio libre en el disco.
- Microsoft Windows Vista SP1:
  - Procesador: Intel Pentium III o equivalente a 800 MHz.
  - Memoria: 512 MB.
  - Espacio de disco: 750 MB de espacio libre en el disco.
- Microsoft Windows 7:
  - Procesador: Intel Pentium III o equivalente a 800 MHz.
  - Memoria: 512 MB.
  - Espacio de disco: 750 MB de espacio libre en el disco.
  
- Ubuntu 9.04:
  - Procesador: Intel Pentium III o equivalente a 800 MHz.
  - Memoria: 512 MB.
  - Espacio de disco: 650 MB de espacio libre en el disco.
- Solaris OS versión 10 (SPARC):
  - Procesador: UltraSPARC II a 450 MHz.
  - Memoria: 512 MB.
  - Espacio de disco: 650 MB de espacio libre en el disco.
- Solaris OS versión 10 (x86/x64 Platform Edition):
  - Procesador: AMD Opteron serie 1200 a 1,8 GHz.
  - Memoria: 512 MB.
  - Espacio de disco: 650 MB de espacio libre en el disco.
- Macintosh OS X 10.5 Intel:
  - Procesador: Intel Dual Core.
  - Memoria: 512 MB.
  - Espacio de disco: 650 MB de espacio libre en el disco.
  
- Macintosh OS X 10.5 PPC:
  - Procesador: PowerPC G4.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- Memoria: 512 MB.
  - Espacio de disco: 650 MB de espacio libre en el disco.
- Configuraciones de software recomendadas
- Microsoft Windows XP Professional SP3:
    - Procesador: Intel Pentium IV o equivalente a 2,6 GHz.
    - Memoria: 2 GB.
    - Espacio de disco: 1 GB de espacio libre en el disco.
  - Microsoft Windows Vista SP1:
    - Procesador: Intel Pentium IV o equivalente a 2,6 GHz.
    - Memoria: 2 GB.
    - Espacio de disco: 1 GB de espacio libre en el disco.
  - Microsoft Windows 7:
    - Procesador: Intel Pentium IV o equivalente a 2,6 GHz.
    - Memoria: 2 GB.
    - Espacio de disco: 1 GB de espacio libre en el disco.
  - Ubuntu 9.04:
    - Procesador: Intel Pentium IV o equivalente a 2,6 GHz.
    - Memoria: 2 GB.
    - Espacio de disco: 850 MB de espacio libre en el disco.
  - Solaris OS versión 10 (SPARC):
    - Procesador: UltraSPARC IIIi a 1 GHz.
    - Memoria: 2 GB.
    - Espacio de disco: 850 MB de espacio libre en el disco.
  - Solaris OS versión 10 (x86/x64 Platform Edition):
    - Procesador: AMD Opteron serie 1200 a 2,8 GHz.
    - Memoria: 2 GB.
    - Espacio de disco: 850 MB de espacio libre en el disco.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- OpenSolaris 2009.06 (edición para plataformas x86/x64):
  - Procesador: AMD Opteron serie 1200 a 2,8 GHz.
  - Memoria: 2 GB.
  - Espacio de disco: 650 MB de espacio libre en el disco.
- Macintosh OS X 10.5 Intel:
  - Procesador: Intel Dual Core.
  - Memoria: 2 GB.
  - Espacio en disco: 850 MB de espacio libre en el disco.
- Macintosh OS X 10.5 PPC:
  - Procesador: PowerPC G5.
  - Memoria: 2 GB.
  - Espacio en disco: 850 MB de espacio libre en el disco.

- **Requerimientos software:**

NetBeans IDE funciona en el kit de desarrollo de Java SE (JDK), que consta del entorno de ejecución de Java y de herramientas para desarrolladores para la compilación, depuración y ejecución de aplicaciones escritas en lenguaje Java.

Los JDK probados con esta versión son los siguientes:

- ❖ JDK 6 Actualización 14.
- ❖ JDK 5 Actualización 19.

- **Descripción:**

El IDE NetBeans está formado por dos componentes:

- ❖ **Plataforma:** es un runtime núcleo que proporciona los servicios comunes que requieren la mayoría de aplicaciones: acciones, ventanas, almacenamiento y gestión de configuraciones, acceso a archivos entre otros. De esta forma los desarrolladores pueden concentrarse en la lógica de su aplicación y escribir

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

módulos que implementan su funcionalidad. La plataforma proporciona un conjunto bien definido de APIs y abstracciones para implementar funcionalidad rápida y fácilmente, las APIs Abiertas NetBeans (NetBeans Open APIs). Ellas son interfaces públicas y documentadas que los módulos utilizan para interactuar con la Plataforma NetBeans y otros módulos. A medida que el proyecto evolucione, la implementación subyacente de las APIs puede cambiar, pero las APIs se mantienen de tal forma que los módulos anteriores todavía trabajen en las versiones más nuevas de la Plataforma.

- ❖ Módulos: proporcionan la funcionalidad que hace de NetBeans un IDE. El soporte para el lenguaje Java, el editor de código son ejemplos de módulos plug-in. Cada pieza de funcionalidad que NetBeans ofrece es proporcionada por módulos. Ellos están distribuidos en varias categorías que son descritas más adelante.

Las principales características del IDE NetBeans son:

- ❖ Editor de código con soporte para completar código, anotaciones, macros, auto- indentación, capaz de operar en un modo mixto para documentos de múltiples lenguajes.
- ❖ Soporte para Java, C, C++, XML y HTML.
- ❖ Soporte para compiladores, debuggers y servicios de ejecución a través de la adición de plug-ins.
- ❖ Soporte para JSP, Servlets, Enterprise Java Beans, XML, RMI, CORBA, JINI y JDBC.
- ❖ Soporte para Ant, CVS y otros sistemas de control de versiones.
- ❖ Herramientas visuales de diseño (AWT/Swing).
- ❖ Asistentes y herramientas de generación y gestión de código.
- ❖ Soporte para bases de datos:

Se han realizado los siguientes cambios en la funcionalidad de la base de datos de NetBeans IDE 6.8:

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- **Controladores actualizados.** Los controladores de MySQL incluidos en NetBeans IDE se han actualizado a la versión 5.1.6.
- **Controladores probados.** NetBeans IDE 6.8 se ha probado con los siguientes controladores y bases de datos.

Controlador	Versión	Ejemplo de URL
<b>JavaDB</b>	Derby 10.4.1.3	jdbc:derby://localhost:1527/sample (Red)
<b>Oracle</b>	Base de datos Oracle 10g versión 2 (10.2.0.2)	jdbc:oracle:thin:@//localhost:1521:ora9i
<b>PostgreSQL</b>	8.3 Compilación 603	jdbc:postgresql://jsmith.mycompany.com:5432/postgres
<b>MySQL</b>	MySQL Connector/J 5.1.6	jdbc:mysql://localhost:3306/sample

Tabla N° 3. 8 Controladores de MySQL incluidos en Netbeans.

### Otros controladores

Los siguientes controladores y bases de datos no se han probado de forma oficial, pero la experiencia indica que deberían funcionar correctamente.

Controlador	Versión	Ejemplo de URL
<b>Microsoft</b>	Controlador JDBC de Microsoft SQL Server 2005 versión 1.2.2828.100	jdbc:sqlserver://localhost:1433;database Name=travel;selectMethod=cursor
<b>IBM</b>	Controlador DB2 JDBC de tipo 4 v8 fixpack 13 redistribuable	jdbc:db2://localhost:50002/sample

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

<b>jTDS</b>	jTDS 1.2.1	jdbc:jtds:sqlserver://test-mycompany.com:1433/travel
<b>DataDirect</b>	DataDirect Connect para JDBC - 3.6, DataDirect Connect para JDBC - 3.6.07	jdbc:datadirect:oracle://localhost;SID=ora9i jdbc:datadirect:sqlserver://localhost:1433;DatabaseName=travel;SelectMethod=cursor jdbc:datadirect:db2://localhost:50002;databaseName=sample
<b>Pointbase</b>	5.2 ECF compilación 294	jdbc:pointbase://localhost:29092/sample

Tabla N° 3. 9. Controladores y bases de datos que no se han probado de forma oficial.

❖ Soporte para una gran cantidad de plataformas.

### 3.1.1.2.1 Bases de datos

Nombre	Licencia	Descripción
<b>Database Explorer</b>	Gratis	Obtiene y modifica la estructura de una base de datos.
<b>SQL ToolBox</b>	Comercial	Conjunto de herramientas para la administración de una base de datos.

Tabla N° 3. 10. Descripción y la licencia de Bases de Datos.

### 3.1.1.2.2 Aplicaciones distribuidas

Nombre	Licencia	Descripción
<b>CORBA</b>	Gratis	Proporciona soporte para escribir programas que usen CORBA.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

<b>File Copy</b>	Gratis	Permite copiar archivos a y desde un servidor FTP remoto.
------------------	--------	-----------------------------------------------------------

Tabla N° 3.11. Descripción y la licencia de Aplicaciones Distribuidas.

### 3.1.1.2.3 Refactorización

Nombre	Licencia	Descripción
<b>RefactorIT</b>	Edición estándar (comercial) Edición de evaluación (comercial) Edición de comunidad (gratis)	Proporciona soporte para refactorización de código.

Tabla N° 3. 12. Descripción y la licencia de Refactorización.

### 3.1.1.2.4 Modeling Tools

Nombre	Licencia	Descripción
<b>MagicDraw UML</b>	Comercial	Herramienta de modelamiento UML (soporta UML 1.4)

Tabla N° 3. 13. Descripción y la licencia de Herramientas de Modelado.

### 3.1.1.2.5 Web

Nombre	Licencia	Descripción
<b>JSP Tag Library Editor</b>	Gratis	Soporta la creación de librerías de etiquetas personalizadas JSP.
<b>JSP/Servlet</b>	Gratis	Soporta la creación, edición, compilación y prueba de JSPs.

Tabla N° 3. 14. Descripción y la licencia Web.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### 3.1.1.2.6 Herramientas J2EE

Nombre	Licencia	Descripción
<b>Ambit/J Bean Generator - EJB 2.0 Wizard</b>	Comercial	Produce código que cumple con la especificación EJB 2.0 y proporciona soporte para Jboss 3.x y Xdoclet.

Tabla N° 3. 15 Descripción y la licencia Herramientas J2EE.

### 3.1.1.2.7 J2ME

Nombre	Licencia	Descripción
<b>J2ME Wireless</b>	Gratis	Permite escribir programas J2ME (MIDP)

Tabla N° 3. 16. Descripción y la licencia J2ME.

### 3.1.1.3 JBUILDER



Figura N°3.3. IDE J-Builder

Fuente: <http://www.kaericulture.com/2009/04/corporate-logo-design-jbuilder.html>

El entorno de desarrollo JBuilder es un elemento esencial en el conjunto de tecnologías de gestión de ciclo de vida de aplicaciones que ofrece Borland para la plataforma Java. Está diseñado para proporcionar a los grupos de desarrollo las herramientas que ellos necesitan para maximizar la productividad en el desarrollo de aplicaciones empresariales.

- **Desarrollador:** Borland
- **Licencia:** comercial. Su costo depende de la edición:

Edición	Costo (en dólares)
<b>JBuilder Enterprise</b>	3500
<b>JBuilder Developer</b>	999
<b>JBuilder WebLogic</b>	3500
<b>JBuilder Foundation</b>	Gratis

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Tabla N° 3. 17 Descripción de costos de licencia JBuilder.

- **Versión actual:** 9
- **Sistemas operativos soportados:** Windows 2000 (SP4), Windows XP y Windows NT 4.0 (SP6), Linux, Solaris.
- **Requerimientos hardware:**

Para JBuilder Enterprise o JBuilder WebLogic

- **Espacio en disco duro:** 760 Mbytes
- **Memoria:** 512 Mbytes mínimo (768 Mbytes recomendado)

Para JBuilder Developer o JBuilder Foundation

- **Espacio en disco duro:** 760 Mbytes
- **Memoria:** 256 Mbytes mínimo (512 Mbytes recomendado)

### **Descripción:**

JBuilder proporciona las siguientes características:

- ❖ Proporciona herramientas visuales para el diseño de GUIs.
- ❖ Soporte para desarrollar aplicaciones Web
- ❖ Soporte para desarrollar aplicaciones con acceso a bases de datos.
- ❖ Soporte para desarrollar aplicaciones J2EE que incluyen EJB, JSP, Servlets y componentes Struts.
- ❖ Integración con el servidor Empresarial de Borland, WebLogic, Jboss, WebSphere, Oracle, EAServer y SunONE.
- ❖ Soporte para desarrollar, consumir y desplegar Servicios Web rápida y fácilmente usando el diseñador de Servicios Web. JBuilder soporta las últimas tecnologías de Servicios Web, incluyendo Axis, SOAP, WSDL, UDDI y WSIL.
- ❖ Soporte para el desarrollo de aplicaciones para dispositivos móviles.
- ❖ Soporte para múltiples plataformas.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- ❖ Otras características incluyen diseñadores de Servicios Web, visualización de Código UML, refactorización y soporte para Ant y varios sistemas de control de versiones.

Existen varias ediciones de JBuilder: JBuilder Enterprise, JBuilder Developer, JBuilder Foundation y JBuilder WebLogic. A continuación se hace una pequeña descripción de cada una de ellas.

- ❖ JBuilder Enterprise: simplifica el desarrollo EJB y Web con dos diseñadores visuales y rápido despliegue en los servidores de aplicación J2EE, diseñadores de Servicios Web y herramientas de desempeño integrado. Su costo es de 3.500 dólares.
- ❖ JBuilder Developer: proporciona potencia y control con tecnologías para crear y desplegar rápidamente aplicaciones de bases de datos y Web, refactorización de código avanzado, entre otras. Su costo es de 999 dólares.
- ❖ JBuilder WebLogic: es una solución de desarrollo integrada para desarrollo de aplicaciones empresariales para Java con el entorno de desarrollo de JBuilder sobre la plataforma WebLogic de BEA. Su costo es de 3.500 dólares.
- ❖ Jbuilder Foundation: incluye un editor, debugger, compilador, diseñadores visuales, asistentes, refactorización, entre otros. Es un IDE Java que no tiene ningún costo.

La siguiente tabla muestra las características más sobresalientes de JBuilder y las ediciones en las que están presentes.

Característica	Edición		
	Foundation	Developer	Enterprise
Interfaz de usuario personalizable	X	X	X
Editor de código	X	X	X
Desarrollo visual de GUIs utilizando componentes JFC/Swing	X	X	X
Integración con Ant	X	X	X

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Integración con Junit	X	X	X
Debugger gráfico con capacidades de debugging remoto y funcionalidad Hot Swap		X	X
Visualización de diagramas UML y refactorización desde diagramas UML.			X
Soporte para CVS y Optimizeit Suite	X	X	X
Diseñador Web, de Struts y Servicios Web,		X	X
Soporte para JSPs, asistentes JSP/Servlet y debugging de JSPs/Servlets		X	X
Despliegue Web automatizado con integración del servidor de aplicación			X
XML y conversión de bases de datos, integración con Cocoon			X
Diseñador Visual EJB para EJB 1.1 y 2.0, BMP y CMP.			X
Integración automática de servidores de aplicaciones con servidores de aplicaciones J2EE.			X
Integración y asistentes para CORBA			X
Desarrollo, compilación y despliegue de aplicaciones Java para dispositivos móviles basados en J2ME.		X	X

Tabla N° 3. 18. Características JBuilder.

### 3.1.2 COMPARACIÓN

- ❖ Eclipse y NetBeans se benefician de la capacidad de aceptar plug-ins de fuente abierta o comerciales escritos por los propios desarrolladores Java, que pueden extender la funcionalidad de plataforma y a otros plug-ins. Esto

permite tener una plataforma flexible y potente, ya que sus capacidades se extienden hasta donde llegue la imaginación y la destreza de los desarrolladores. Además permite que los IDEs sean neutrales con respecto a la plataforma y el lenguaje, ya que es posible desarrollar plug-ins para crear un IDE para un lenguaje específico. Sin embargo esto trae como consecuencia que el proceso de desarrollo sea más lento y complicado

- ❖ debido a la búsqueda y configuración de plug-ins adecuados, e incluso la creación de uno nuevo para brindar soporte a las características requeridas. JBuilder por su parte, incorpora e integra todas las capacidades en el IDE, por lo tanto es menos flexible, porque la adición de nuevas funcionalidades implica la creación de una nueva versión del IDE, pero esto favorece el desarrollo rápido y fácil de aplicaciones.

Con NetBeans no se necesitan configurar tantos plug-ins después de la instalación porque varios de ellos son integrados como parte del sistema base. Con Eclipse es necesario buscar y configurar plug-ins, porque sin ellos Eclipse no proporciona mucha funcionalidad.

- ❖ JBuilder es el IDE con una curva de aprendizaje más corta, debido a que es bastante intuitivo y proporciona grandes facilidades para el diseño de GUIs. Eclipse se considera el entorno más difícil para aprender a manejar.
- ❖ Todos los IDEs proporcionan capacidades de refactorización de código, algo muy importante en proyectos grandes. Sin embargo, esta funcionalidad es más avanzada en JBuilder y Eclipse que en NetBeans.
- ❖ Todos los IDEs proporcionan algún tipo de asistencia para el modelado y el diseño, pero no de forma tan detallada y continua, de principio a fin, como el que puede proporcionar Eclipse mediante plug-ins. Eclipse permite realizar íntegramente el proceso de desarrollo de software tal y como se entiende en la actualidad, desde el análisis inicial de requerimientos hasta la distribución final y el mantenimiento.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- ❖ Anteriormente, una de las mayores ventajas de NetBeans con respecto a Eclipse, era que el primero proporcionaba un diseñador visual de GUIs mientras que Eclipse no, era necesario utilizar plug-ins. Sin embargo esta diferencia desapareció con el inicio del proyecto Editor Visual de Eclipse, que es una infraestructura para crear diseñadores de GUIs para Eclipse. Incluirá implementaciones de referencia de JFC/Swing y SWT, y en un futuro se
- ❖ piensa proporcionar soporte para otros lenguajes tales como C/C++ y otros conjuntos de componentes gráficos, incluyendo esos que no son soportados bajo Java.

IBM comenzará donando su Editor Visual de la línea WebSphere Studio al proyecto Eclipse que proporciona soporte para JFC/Swing, y otras compañías entre las cuales están Advanced Systems Concepts, Instantiations y Red Hat proporcionarán recursos para entregar soporte para SWT.

Según lo anterior JBuilder y NetBeans usan AWT/Swing, mientras que Eclipse proporciona soporte para Swing y SWT/Jface. SWT es una librería encargada de proporcionar los componentes gráficos (botones, listas, cuadros de texto, etc.), y permite obtener una interfaz gráfica muy integrada con la plataforma nativa bajo la cual se use Eclipse. El uso de una nueva librería en Eclipse se debe a que ella permite crear interfaces de usuario nativas, utilizando las capacidades gráficas del sistema operativo y del sistema gráfico de ventanas que se esté usando. JFace proporciona una interfaz de más alto nivel que la de SWT, basada en los componentes SWT, y lleva incorporadas muchas facilidades para usar con los plug-ins (preferencias, asistentes, etc.).

Las principales desventajas de Swing son: alto consumo de recursos del sistema, lentitud con respecto a las aplicaciones nativas, tamaño muy grande de la biblioteca, diferencias perceptibles en aspecto y sensación con respecto a las aplicaciones nativas y ausencia de soporte de características nativas. Las desventajas de SWT es que expone componentes específicos de la plataforma que permiten a los desarrolladores crear código que no es independiente de la plataforma.

Según esto podemos decir que JBuilder y NetBeans ofrecen un IDE que no depende de los componentes gráficos nativos gracias al uso de Swing, pero su desempeño y velocidad son menores en comparación a Eclipse. Sin embargo, en noviembre de 2003, NetBeans anunció que proporcionará una

nueva forma de manejar el sistema de ventanas que mejora su velocidad y le proporciona al usuario mayores facilidades, manteniendo la independencia de los componentes gráficos nativos.

Eclipse no sufre de estos problemas de desempeño porque SWT es nativa. Sin embargo, Eclipse tenía otros problemas como falta de flexibilidad para crear proyectos y la falta de una opción para deshabilitar la compilación automática.

Con el JDK 1.4 el desempeño de NetBeans ha mejorado considerablemente, pero no es lo suficientemente rápido para los desarrolladores. El JDK 1.4 también benefició Eclipse al mejorar su velocidad. Además ahora Eclipse ofrece flexibilidad para la configuración de proyectos y es posible habilitar o deshabilitar la compilación automática. Además Eclipse ha evolucionado más rápido que NetBeans. El número de plug-ins de NetBeans es mucho menor que los disponibles para Eclipse (cualquier cosa que no esté en los paquetes principales de Eclipse, se encuentran como plug-ins).

### 3.1.3 CONCLUSIONES

- ❖ Todavía no existe un IDE universal o perfecto, capaz de reunir todas las características que un desarrollador puede necesitar. Por lo general, los puntos débiles de un IDE coinciden con los puntos fuertes de otro. Por lo tanto la elección de un IDE depende de si incorpora o no las características necesarias para el proyecto y está condicionada por su precio, requerimientos previos (sistemas de gestión de bases de datos, servidores de aplicaciones, entre otros), facilidad de uso, e incluso el gusto del desarrollador.
- ❖ En general, la mayoría de características y funcionalidad proporcionada por JBuilder, NetBeans y Eclipse es la misma, a diferencia está en que la licencia

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- de JBuilder cuesta, mientras que NetBeans y Eclipse son completamente gratis y de código abierto. NetBeans y Eclipse presentan diferencias mínimas con el JBuilder comercial, además ofrecen más funcionalidad que la versión gratis de JBuilder. Por esto es importante analizar si la funcionalidad adicional que presenta JBuilder realmente justifica su alto costo.
- ❖ SWT es un conjunto de librerías para el desarrollo de interfaces gráficas que siguen un modelo muy similar al de Swing pero con rendimiento similar al de las aplicaciones nativas. Por esto Eclipse, debido al uso de SWT, proporciona GUIs con mejor tiempo de respuesta pero más dependientes de la plataforma, JBuilder y NetBeans utilizan diseñadores gráficos basados en Swing, esto disminuye el desempeño pero proporciona una verdadera independencia de la plataforma.

### 3.2 ANÁLISIS DE BASES DE DATOS MYSQL Y POSTGRESQL



Figura N°3 5 Bases de datos PostgreSQL y MySQL.

Fuente: <http://www.unocero.com/2011/08/05/apple-quita-mysql-de-su-version-server-de-mac-os-x-lion/>

#### 3.2.1 Información general

	Creador	Fecha primera versión pública	Última versión estable	Licencia de software
MySQL	MySQL AB	Noviembre de 1996	5.0	GPL o propietario
PostgreSQL	PostgreSQL Global Development Group	Junio de 1989	9.0	Licencia BSD

Tabla N° 3. 19 Información general de bases de datos.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### 3.2.2 Soporte del sistema operativo

	Windows	Mac OS X	Linux	BSD	Unix	z/OS
MySQL	✓ Sí	✓ Sí	✓ Sí	✓ Sí	✓ Sí	Quizá
PostgreSQL	✓ Sí	✓ Sí	✓ Sí	✓ Sí	✓ Sí	✗ No

Tabla N° 3. 20. Soporte del sistema operativo.

### 3.2.3 Características fundamentales

Información acerca de qué características fundamentales de las RDBMS son implementados nativamente.

	ACID	Integridad referencial	Transacciones	Unicode
MySQL	Depende (1)	Depende (1)	Depende (1)	✓ Sí
PostgreSQL	✓ Sí	✓ Sí	✓ Sí	✓ Sí

Tabla N° 3. 21. Características fundamentales.

Nota (1): Para las transacciones y la integridad referencial, el tipo de tabla InnoDB debe ser usado; el tipo de tabla por defecto, MyISAM, No soporta estas características. Sin embargo, inclusive el tipo de tabla InnoDB permite el almacenamiento de valores que excedan el rango de datos; algunas vistas violan la limitación de ACID.

### 3.2.4 Tablas y vistas

Información acerca de que tablas y vistas (unas más básicos que otras) son soportados nativamente.

	Tabla temporal	Vista materializada
<u>MySQL</u>	✓ Sí	✗ No
<u>PostgreSQL</u>	✓ Sí	✗ No (2)

Tabla N° 3. 22. Tablas y vistas que son soportados nativamente.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Nota (2): La vista materializada puede ser emulada con PL/PgSQL

### 3.2.5 Índices

Información acerca de que índices (otros como los índices básicos) son soportados nativamente.

	Árbol R-/R+	Hash	Expresión	Parcial	Reversa	Mapa de bits
<u>MySQL</u>	Tablas MyISAM solamente	Tablas HEAP solamente	✗ No	✗ No	✗ No	✗ No
<u>PostgreSQL</u>	✓ Sí	✓ Sí	✓ Sí	✓ Sí	✗ No	✗ No

Tabla N° 3. 23. Índices que son soportados nativamente.

### 3.2.6 Otros objetos

Información acerca de que otros objetos son soportados nativamente.

	Dominio	Cursor	Trigger	Funciones (4)	Procedimiento (4)	Rutina externa (4)
<u>MySQL</u>	✗ No	✓ Sí (3)	✓ Sí (3)	✓ Sí (3)	✓ Sí (3)	✓ Sí
<u>PostgreSQL</u>	✓ Sí	✓ Sí	✓ Sí	✓ Sí	✓ Sí	✓ Sí

Tabla N° 3. 24 Objetos que son soportados nativamente.

Nota (3): Estos objetos de base de datos son disponibles a partir de MySQL 5.0 disponible desde 24/12/2005.

Nota (4): Función y procedimiento se refieren a las rutinas internas escritas en SQL o lenguajes procedurales como PL/SQL. Rutina externa se refiere a la escritura en los lenguajes anfitriones como C, Java, Cobol, etc. "Procedimiento almacenado" es un término comúnmente usado para ese tipo de rutinas. Sin embargo, su definición varía entre diferentes vendedores de bases de datos.

### 3.2.7 Particionamiento

Información acerca de que métodos de particionamiento son soportados nativamente.

	Rango	Hash	Compuesto (Rango+Hash)	Lista
MySQL	✓ Sí	✓ Sí	✓ Sí	✓ Sí
PostgreSQL	✓ Sí	✗ No	✗ No	✓ Sí

Tabla N° 3. 25. Particionamiento son soportados nativamente.

### 3.2.8 Conclusiones:

#### MySQL:

- Su principal objetivo de diseño fue la VELOCIDAD. Se sacrificaron algunas características esenciales en sistemas más "serios" con este fin.
- Otra característica importante es que consume MUY POCOS RECURSOS, tanto de CPU como de memoria.
- Licencia GPL a partir de la versión 3.23.19.

#### Ventajas:

- Mayor rendimiento. Mayor velocidad tanto al conectar con el servidor como al servir selects y demás.
- Mejores utilidades de administración (backup, recuperación de errores, etc).
- Aunque se cuelgue, no suele perder información ni corromper los datos.
- Mejor integración con PHP.
- No hay límites en el tamaño de los registros.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- Mejor control de acceso, en el sentido de qué usuarios tienen acceso a qué tablas y con qué permisos.
- MySQL se comporta mejor que Postgres a la hora de modificar o añadir campos a una tabla "en caliente".

Inconvenientes:

- No soporta transacciones, "roll-backs" ni subselects.
- No considera las claves ajenas. Ignora la integridad referencial, dejándola en manos del programador de la aplicación.

PostgreSQL:

- Postgres intenta ser un sistema de bases de datos de mayor nivel que MySQL, a la altura de Oracle, Sybase o Interbase.
- Licencia BSD.

Ventajas:

- Por su arquitectura de diseño, escala muy bien al aumentar el número de CPUs y la cantidad de RAM.
- Soporta transacciones y desde la versión 7.0, claves ajenas (con comprobaciones de integridad referencial).
- Tiene mejor soporte para triggers y procedimientos en el servidor.
- Soporta un subconjunto de SQL92 MAYOR que el que soporta MySQL. Además, tiene ciertas características orientadas a objetos.

Inconvenientes:

- Consume BASTANTES más recursos y carga más el sistema.
- Límite del tamaño de cada fila de las tablas a 8k (se puede ampliar a 32k recompilando, pero con un coste añadido en el rendimiento).
- Es de 2 a 3 veces más lenta que MySQL.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

En cuanto a consideraciones de estabilidad del servidor, cada comparativa da datos contradictorios. En general parece que MySQL es más estable y que Postgres tiende a desperdiciar memoria y sobrecargar bastante el sistema.

Como conclusión a la comparación entre MySQL y Postgres, parece aceptado que MySQL junto con Apache y Eclipse forman un buen equipo para servir páginas web con contenido dinámico. En general, sistemas en los que la velocidad y el número de accesos concurrentes sea algo primordial, y la seguridad no sea muy importante (pueda bastar con hacer backups periódicos que se restaurarán tras una caída del servidor). En cambio, para sistemas más serios en las que la consistencia de la BD sea fundamental (BD con información realmente importante, bancos, etc.) PostgreSQL es una mejor opción pese a su mayor lentitud.

## **CAPÍTULO IV**

---

### **4.1 DESARROLLO DEL SISTEMA**

#### **4.1.4 DOCUMENTO DE VISIÓN**

##### **4.1.4.1 Introducción**

El propósito de este documento es coleccionar, analizar y definir las necesidades de alto nivel y características del Sistema Web para la empresa La Viandé.

Se enfoca en las capacidades necesarias de los stakeholders, y los usuarios objetivo, y porque existen estas necesidades. Los detalles de cómo el Sistema Web cubre estas necesidades son descritos en los casos de uso y especificaciones suplementarias.

##### **4.1.4.2 Propósito**

El propósito de este documento es recoger, analizar y definir las diferentes necesidades de alto nivel y las características de la empresa especialista en servicios de catering y eventos. El presente documento se centra en la funcionalidad requerida por los participantes en el proyecto y los usuarios finales. Esta funcionalidad se basa principalmente en la gestión de los procesos de logística de la empresa, que comprenden la generación de órdenes de proformas, inventario de productos y facturación. Los detalles de cómo el sistema cubre los diferentes requerimientos de la empresa especialista en servicios de catering y eventos La Viandé, se pueden observar en la especificación de los casos de uso, en la especificación de los casos de negocio y otros documentos adicionales.

##### **4.1.4.3 Alcance**

El sistema permitirá a un usuario de La Viandé tener un mejor control de los productos, clientes, y órdenes de proforma y facturas emitidas, y así optimizar el manejo de información. Este sistema cubrirá los siguientes

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

procesos: Generación de órdenes de proformas, Generación de facturas, Control de inventario de productos, logrando así una mejora en el modo en que se vienen desarrollando las actividades.

#### 4.1.4.4 Definiciones, Acrónimos, y Abreviaciones

Observar en el Documento Glosario.

RUP: Son las siglas de Rational Unified Process. Se trata de una metodología para describir el proceso de desarrollo de software.

#### 4.1.4.5 Referencias

- Glosario.
- Plan de desarrollo de software.
- RUP (Rational Unified Process).
- Diagrama de casos de uso.

#### 4.1.4.6 Posicionamiento

##### 4.1.4.6.1 Oportunidad de Negocio

El sistema permitirá a la empresa La Viandé manejarse de una manera más eficiente y rápida gestionando estos procesos, lo cual supondrá un acceso rápido y sencillo a los datos, gracias a interfaces gráficas sencillas y amigables. También podrá agilizar los procesos dado que existe un adecuado registro de productos, registros de clientes, además de una continua actualización de los mismos. Además, los datos accedidos estarán siempre actualizados, lo cual es un factor muy importante para poder llevar un control rápido y eficiente de los procesos.

##### 4.1.4.6.2 Sentencia que define el problema

El problema de	La empresa La Viandé no cuenta con un sistema informático, ya que los procesos aún no están sistematizados como Generación de órdenes de proformas, control de inventario, manejo de clientes, generación de facturas, reportes, para ello se recurre a la
----------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

	extracción de información de la base de datos para su análisis y la aplicación de Herramientas de Gestión de información como la obtención de un control de inventario de productos, reportes de ventas y de eventos.
Afecta a	Empresa La Viandé.
El impacto asociado es	Dificultad para el gerente de la empresa cuando realizan sus actividades dado que los procesos no están automatizados.
Una adecuada solución sería	Implementar los procesos que no están sistematizados para poder analizar los datos de la empresa y brindar una adecuada gestión logística para la empresa generando reportes y herramientas para la toma de decisiones, usando una red local con una base de datos accesible con interfaces amigables y sencillas.

Figura N° 4. 1 Sentencia que define el problema.

#### 4.1.4.6.3 Sentencia que define la posición del Producto

Para	Usuario Administrador y Gerente de la Viandé.
Quienes	Controlan la generación de órdenes de proformas, generación de facturas, control de inventario de productos, generación de menús de comida para su respectiva proforma, manejo de clientes.
El nombre del producto	Es un software de gestión de información y procesos.
Que	Almacena y procesa la información necesaria para gestionar los procesos de una empresa especialista en servicios de catering y eventos.
No como	El no tener automatizado los procesos de la empresa

	dedicada a servicios de catering y eventos.
Nuestro producto	Permite gestionar las distintas actividades relacionados con el servicio de catering y eventos de la empresa mediante una interfaz gráfica sencilla y amigable. Además proporciona un acceso rápido y actualizado a la información desde cualquier punto que tenga acceso a la base de datos.

Figura N° 4. 2 Sentencia que define la posición del Producto.

#### 4.1.4.7 Descripción de Stakeholders y Usuarios

Para proveer de una forma efectiva productos y servicios que se ajusten a las necesidades de los usuarios, es necesario identificar e involucrar a todos los participantes en el proyecto como parte del proceso de modelado de requerimientos. También es necesario identificar a los usuarios del sistema y asegurarse de que el conjunto de participantes en el proyecto los representa adecuadamente.

Esta sección muestra un perfil de los participantes y de los usuarios involucrados en el proyecto, así como los problemas más importantes que éstos perciben para enfocar la solución propuesta hacia ellos. No describe sus requisitos específicos ya que éstos se capturan mediante otro artefacto. En lugar de esto proporciona la justificación de por qué estos requisitos son necesarios.

##### 4.1.4.7.1 Resumen de Stakeholders

Nombre	Descripción	Responsabilidades
Paulina Obando	Gerente de la empresa y de todos los usuarios potenciales de la misma	El stakeholder realiza:  Representa a todos los usuarios posibles del sistema.  Seguimiento del desarrollo del proyecto.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

		Aprueba requisitos y funcionalidades
--	--	--------------------------------------

Tabla N° 4. 1 Resumen de Stakeholders

#### 4.1.4.7.2 Resumen de Usuarios

Nombre	Descripción	Stakeholder
Administrador del Sistema	Tiene la responsabilidad de ejecutar, mantener, operar y asegurar el correcto funcionamiento del sistema. Tiene todos los permisos, es encargado de la creación y eliminación de usuarios, eliminación de facturas, administrar la página web de la empresa.	Ana Belén Obando
Usuario Directo	Es el encargado de manipular el sistema con ciertos permisos. Como ingresar clientes, productos, realizar facturas pero no eliminarlas, generar proformas.	Paulina Obando

Tabla N° 4. 2. Resumen de Usuarios

#### 4.1.4.7.3 Entorno de usuario

El número de personas involucradas en estos procesos es fijo, ya que esta empresa es una empresa familiar, consta de pocos integrantes. Esta herramienta permite la integración de la información y la sistematización de los procedimientos de gestión para cada uno de los usuarios. Los módulos que comprenden el sistema son: El Módulo de Clientes, El Módulo de Clientes, El Módulo de Facturación, El Modulo de Contabilidad, Módulo de Menús y Platos, Módulos de Órdenes de Proformas; nuestra aplicación está pensada para desenvolverse en un entorno Web, el ambiente para los usuarios internos de la Empresa será un local físico determinado.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

El ambiente para los usuarios del sistema será el local mismo donde se encuentra la a. Los usuarios entrarán al sistema identificándose sobre un ordenador con un sistema operativo Windows, navegadores web como Mozilla e Internet Explorer (los más conocidos) y tras este paso entrarán a la aplicación. Este sistema es similar a cualquier aplicación Web cliente/servidor y por tanto los usuarios estarán familiarizados con su entorno. Por este motivo, los usuarios utilizaran las mismas PCs que vienen utilizando.

#### 4.1.4.8 Perfil de los Stakeholders

##### 4.1.4.8.1 Representante del área técnica y sistemas de información

<b>Representante</b>	Ana Belén Obando.
<b>Descripción</b>	Asistente Técnica de La Viandé.
<b>Tipo</b>	Egresada en Ingeniería en Sistemas Computacionales.
<b>Responsabilidades</b>	Encargado de mostrar las necesidades de cada usuario del sistema. Además, lleva a cabo un seguimiento del desarrollo del proyecto y aprobación de los requisitos y funcionalidades del sistema.
<b>Criterio de Éxito</b>	A definir por el cliente.
<b>Grado de participación</b>	Revisión de requerimientos, estructura del sistema.
<b>Comentarios</b>	Ninguno

Tabla N° 4. 3. Representante de Área Técnica

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

#### 4.1.4.9 Perfiles de Usuario

##### 4.1.4.9.1 *María Paulina Obando Varas.*

<b>Representante</b>	Gerente de La Viandé.
<b>Descripción</b>	Persona encargada de la Administración de la empresa.
<b>Tipo</b>	Tecnóloga en Gastronomía.
<b>Responsabilidades</b>	<p>Forma parte del grupo de Chefs de la Viandé, así como también es La Gerente de la empresa. Entre sus funciones están:</p> <ul style="list-style-type: none"> <li>• Asegurar que la empresa marche bien.</li> <li>• Encargada de la realización y coordinación de los eventos y servicios que ofrece la empresa.</li> </ul>
<b>Criterio de Éxito</b>	A definir por el cliente.
<b>Grado de participación</b>	Tipos de reportes que requiera.
<b>Comentarios</b>	Ninguno.

Tabla N° 4. 4. Perfil de Usuario – Paulina Obando

##### 4.1.4.9.2 *Guadalupe Varas Pozo.*

<b>Representante</b>	Usuario Directo
<b>Descripción</b>	Se encarga de manipular el sistema.
<b>Tipo</b>	Asistente
<b>Responsabilidades</b>	Administra el sistema con ciertos permisos, podrá ingresar clientes, realizar proformas y facturar.
<b>Criterio de Éxito</b>	A definir por el cliente.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

<b>Grado de participación</b>	A definir por el cliente.
<b>Comentarios</b>	Ninguno.

Tabla N° 4. 5. Perfil de Usuario – Guadalupe Varas

#### 4.1.4.10 Descripción Global del Producto

##### 4.1.4.10.1 Perspectiva del producto

El producto a desarrollar es un Sistema de Gestión de Información para la Empresa especialista en Servicios de Catering y Eventos LA VIANDÉ, con la intención de sistematizar sus procesos. El producto a desarrollar es una aplicación utilizando la tecnología Web, con el objetivo de automatizar el funcionamiento de las áreas de Administración para la eficiente gestión de proformas, facturaciones y productos.

El producto es una aplicación Web que será independiente y cubrirá los siguientes procesos: Generación de órdenes de proformas, Generación de Facturación, Inventario de Productos, logrando así una mejora en el modo en que se vienen desarrollando las actividades.

##### 4.1.4.11 Resumen de características

A continuación se mostrará un listado con los beneficios que obtendrá el cliente a partir del producto:

<b>Función</b>	<b>Beneficio del cliente</b>	<b>Características que lo apoyan</b>
Entorno Web	La Gerente así como La Asistente podrán acceder al sistema mediante un navegador Web.	Permitirá que La gerente y La Asistente puedan realizar sus actividades desde cualquier equipo con acceso a Internet.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Arquitectura MVC	Las funcionalidades de la aplicación estarán estructuradas de acuerdo a ese patrón.	Permitirá que se adicionen nuevas funcionalidades fácilmente sin tener que pasar por largos periodos de mantenimiento.
Seguridad	El sistema contará un manejo vía Internet, permitiendo la seguridad de los datos, estarán protegidos de manera que nadie pueda alterar los contenidos de los mismos.	Garantizará a la gerencia la fiabilidad de los datos.
Calidad	Las pruebas del sistema se realizarán de manera exhaustiva de manera que la aplicación no genere problemas.	Facilitará el uso del software y la recepción rápida por parte del cliente

Tabla N° 4. 6. Resumen de Características

#### 4.1.4.12 Suposiciones y dependencias

- Estamos siguiendo un modelo de procesos que nos permitan llegar al cliente y que el cliente perciba la calidad del servicio.
- Se contará con acceso a toda la información solicitada por el equipo del proyecto para la elaboración del sistema.
- El equipo de proyecto no se encargará de realizar integraciones con sistemas existentes desarrollados o comprados por la empresa con anterioridad.
- El equipo de proyecto será independiente de las reglas organizacionales de la empresa y rendirá informes directamente a la Gerencia General de La Viandé.
- Los documentos o información que necesite el equipo de proyecto deberán ser solicitados de manera formal a los encargados de proporcionarla.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- La Viandé se compromete a brindar el ambiente y los recursos de trabajo como equipos multimedia, PC's y software de desarrollo, cuando se efectuó algún trabajo en su empresa
- Simplicidad de entrada de la información. Con ello se evitarán errores y se logrará la rapidez deseada.
- Existe dependencia de que la computadora donde será instalada y ejecutada la aplicación deberá contar con el Sistema Operativo Microsoft Windows.
- Los riesgos del sistema están relacionados con la consistencia de las comunicaciones de los diferentes usuarios a través de la Web, pudiendo establecerse fallas de conexión que den como resultado que las operaciones no se realicen de manera adecuada.

#### 4.1.4.13 Costo y precio

El Producto posee derechos reservados, donde se establece la legalidad del software. Así, la aplicación poseerá un tiempo de soporte por parte de los que hicieron la aplicación. Esta aplicación estará disponible en un CD; a continuación se detallan los costos de nuestro producto:

##### a) Inversión

1.1	Costos Fijos Tangibles	Precio Unitario	Total
1.1.1	<b>Equipos y Maquinarias</b>	\$/.	\$/.
	1 Router	89	89
1.1.2	<b>Muebles y Enseres</b>		
	2 Escritorios	250	500
	2 Sillas giratorias	120	240
	2 Computadoras	1500	3000
1.1.3	<b>Equipos Diversos</b>		

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

1 Cable de internet (20 mts)	20	20
4 Conectores RJ45	0.5	2
<b>Total Inversión Fija Tangible</b>	<b>1979</b>	<b>3851</b>
<b>1.2 Costo Fijo Intangible</b>		
Internet (año)	79	948
<b>Total Inversión Fija Intangible</b>	<b>79</b>	<b>948</b>

Tabla N° 4. 7. Inversión

### b) Capital de Trabajo

Personal	C.U Mensual	Total
1 Diseñador Gráfico	400	400
1 Desarrollador	400	400
<b>Total</b>	<b>800</b>	<b>800</b>
<b>Resumen de las Inversiones</b>		
<b>1 Inversión</b>		
Inversión Fija Tangible	3851	
Inversión Fija Intangible	948	
<b>2 Capital de Trabajo</b>	800	
<b>Total</b>	<b>\$5599</b>	

Tabla N° 4. 8. Capital de Trabajo

#### 4.1.4.14 Descripción Global del Producto

Nombre de la Característica	Descripción	Inputs	Outputs	Usuario Responsable
Gestión de Usuarios	Registra los usuarios del sistema.	Creación de usuarios del sistema y asignación de permisos.	Listado de usuarios del sistema con sus respectivos permisos	Administrador del sistema.
Gestión de Clientes	Registra a cada uno de nuestros clientes.	Información respectiva de cada cliente de la empresa.	Reporte de clientes	Usuario Directo.
Gestión de Productos	Registra cada uno de los productos tanto como alimentos, así como materías, vajillas.	Información detallada de cada producto, así como la cantidad existente, y categoriza por tipos de productos.	Reporte de productos	Usuario Directo.
Gestión de Platos y Menús	Permite la creación y almacenamiento de platos, los cuales permiten crear menus para ofrecer al cliente en cada uno de sus eventos.	Almacenamiento de platos y menus que podrán ser reutilizados para armar diversos menus.	Reporte de platos y menús	Usuario Directo.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Gestión de Proformas	Al emitir las órdenes de proformas, se estima el precio de un evento, el cual puede cambiar al momento de facturar. Cuando se emiten las proformas pueden haber proformas pendientes.	Orden de Proforma	Generación de la orden de proforma y reportes.	Usuario Directo.
Gestión de Facturas	Al emitir facturas, se establece la fecha de entrega o realización del evento, y no se podrá facturar sin una orden de proforma.	Número de Factura	Generación de Facturas y reportes. En algunos casos eliminación de facturas.	Usuario Directo. Administrador del sistema.
Gestión de Publicidad	Tener presencia virtual es una o quizás la mejor herramienta para anunciar nuestros servicios, para cerrar negocios y comunicarnos en línea.	Información de Servicios que presta la empresa.	Descargar recetas de cocina, ver videos online.	Administrador del sistema. Usuarios indirectos.

Tabla N° 4. 9. Descripción Global del Producto

#### **4.1.4.15 Restricciones**

- Presupuesto: S/.5599 para todo el proyecto.
- Tiempo: Se calcula 72 días después de empezado el proyecto.
- El análisis efectuado para la realización del sistema abarca todos los procesos de la empresa.

#### **4.1.4.16 Precedencia y Prioridad**

- La aplicación estará implementada en un entorno Web en el cual el usuario administrador así como el usuario directo podrán utilizar el sistema mediante un navegador Web y permitirá que puedan realizar sus actividades desde cualquier equipo con acceso a Internet.
- Se empleara la arquitectura MVC, es así que las funcionalidades de la aplicación estarán estructuradas de acuerdo a ese patrón y permitirá que se adicionen nuevas funcionalidades fácilmente sin tener que pasar por largos periodos de mantenimiento.
- El sistema contará un manejo vía Internet, permitiendo la seguridad de los datos, estarán protegidos de manera que nadie pueda alterar los contenidos de los mismos, mediante encriptación de contraseñas por ejemplo.
- Las pruebas del sistema se realizarán de manera exhaustiva lo cual facilitará el uso del software y la recepción rápida por parte del cliente.

#### **4.1.4.17 Otros Requisitos del Producto**

##### **4.1.4.17.1 Estándares Aplicables**

- Lenguaje para el Diseño de aplicaciones Web: Java – Eclipse Helios.
- Lenguaje para el Diseño de páginas web: Joomla.
- Uso de una base de datos MySQL
- Protocolo de comunicación: TCP/IP
- Uso de un sistema operativo Windows.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- RUP
- UML.

#### **4.1.4.17.2 Requisitos de Sistema**

- Para instalar la página Web se necesitará un servidor.
- Plataforma: Windows 7.
- Procesador Intel Core 2 Duo 2 Ghz.
- Memoria RAM de 3 GB.
- Disco Duro de 223 GB de almacenamiento.
- La aplicación web correrá en un servidor local.

#### **4.1.4.17.3 Requisitos de Desempeño**

Se ha considerado un diseño ligero pero atractivo del sistema y que sea de fácil acceso.

#### **4.1.4.17.4 Requisitos de Entorno**

El módulo Web estará disponible al personal de la Empresa tal como lo determine la misma y en las condiciones pactadas en el contrato realizado.

#### **4.1.4.17.5 Requisitos de Documentación**

Se realizara la documentación respectiva para cada proceso que llevara a cabo el usuario final sobre la aplicación.

#### **4.1.4.17.6 Manual de Usuario**

La aplicación contendrá un archivo de ayuda, para el manejo de las interfaces.

#### **4.1.4.17.7 Ayuda en Línea**

Sobre informe de Errores que tengan los usuarios al utilizar la aplicación, podrán contactar al personal de programación de la aplicación por medio telefónico o vía Internet.

#### 4.1.4.18 Guías de Instalación, Configuración, y Fichero Léame

En la compra del Producto, el cliente recibirá una guía de instalación de la aplicación, donde estará paso a paso la descripción y los posibles errores que se encuentren durante la instalación.

#### 4.1.4.19 Atributos de Características

ID	Prio.	Descripción
1	Alta	La solución debe tener la capacidad de crear, modificar y eliminar procesos de flujo básico de información.
2	Alta	Cada proceso puede tener una o más tareas.
3	Alta	Cada Proceso puede involucrar a uno o más individuos (Ej.: Alumno y Profesor guía).
4	Alta	Las Tareas pueden ser requisito para otra tarea.
5	Baja	Las Tareas pueden ser requisito para otro proceso.
6	Alta	Posibilidad de crear automáticamente, carga y descarga de archivos (formularios, PDF´s, excel, word, etc.). Esto dependiendo del archivo asignado a una tarea específica, siempre que los permisos lo permitan.
7	Media	Los procesos y tareas, pueden tener una ventana de tiempo para ser realizadas.
8	Alta	Mantener un registro de las principales actividades realizadas.
9	Media	Cada Tarea puede manejar uno o más archivos en el flujo.
10	Alta	Las tareas automáticas podrán generar envío de emails, ventanas tipo Pop Up con alertas, o tareas en panel de tareas.

11	Alta	Manejar diversos tipos de usuarios, así como también interfaces y permisos que tendrá cada uno.
12	Alta	Sistema de mensajería externo a la herramienta de procesos.
13	Media	Seguimiento <i>curricular</i> de los alumnos de postgrado, externo a la herramienta de procesos.
14	Media	Enviar notificaciones de defensa de tesis.

Tabla N° 4. 10. Atributos de Características.

## 4.2 Plan de Desarrollo del Software

### 4.2.1 Introducción

El propósito del Plan de Desarrollo de Software es reunir toda la información necesaria para el control del proyecto. En el presente documento se describe el enfoque del desarrollo del software y es el plan de alto nivel generado y utilizado por los administradores para dirigir los esfuerzos de desarrollo.

El proyecto ha sido ofertado por Ana Belén Obando basado en una metodología de Rational Unified Process (RUP) en la que únicamente se procederá a cumplir con todas las fases de la metodología.

#### 4.2.1.1 Propósito

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto. En él se describe el enfoque de desarrollo del software.

Los usuarios del Plan de Desarrollo del Software son:

- El jefe del proyecto lo utiliza para organizar la agenda y necesidades de recursos, y para realizar su seguimiento.
- Los miembros del equipo de desarrollo lo usan para entender lo qué deben hacer, cuándo deben hacerlo y qué otras actividades dependen de ello.

#### **4.2.1.2 Alcance**

El plan de desarrollo del software describe el plan global usado para el desarrollo de “Una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé”. El detalle de las iteraciones individuales se describe en los planes de cada iteración, documentos que se aportan en forma separada. Durante el proceso de desarrollo en el artefacto “Visión” se definen las características del producto a desarrollar, lo cual constituye la base para la planificación de las iteraciones. Para esta versión 1.0 en Plan de Desarrollo de Software, basándome en las cláusulas técnicas para hacer una estimación aproximada, una vez comenzando el proyecto y durante la fase de Inicio se generará la primera versión del artefacto “Visión”, el cual se utilizará para refinar este documento. Posteriormente, el avance del proyecto y el seguimiento en cada una de las iteraciones ocasionará el ajuste de este documento produciendo nuevas versiones actualizadas.

#### **4.2.1.3 Resumen**

Después de esta introducción, el resto del documento está organizado en las siguientes secciones:

Vista General del Proyecto — proporciona una descripción del propósito, alcance y objetivos del proyecto, estableciendo los artefactos que serán producidos y utilizados durante el proyecto.

Organización del Proyecto — describe la estructura organizacional del equipo de desarrollo.

Gestión del Proceso — explica los costos y planificación estimada, define las fases e hitos del proyecto y describe cómo se realizará su seguimiento.

Planes y Guías de aplicación — proporciona una vista global del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas que serán utilizadas.

#### **4.2.1.4 Vista General del Proyecto**

##### **4.2.1.4.1 Propósito, Alcance y Objetivos**

La información que a continuación se incluye ha sido extraída de las diferentes reuniones que se han celebrado con el stakeholder de la empresa desde el inicio del proyecto, Ana Belén Obando.

La Viandé es una empresa que presta los servicios de catering y eventos dentro de la provincia de Imbabura. La entrada en un mercado competitivo como en la que se encuentra inmersa esta empresa, conllevará una previsible adaptación a los nuevos sistemas de información y a la evolución de la tecnología. Por ello, La Viandé considera necesario el desarrollo de un sistema de gestión de información, así como las bases de datos que recogen datos de clientes, productos, proformas así como también realizan facturación, por tanto los solicitantes demandan una gestión más rápida, automática y segura de las gestiones de la empresa.

El proyecto debe proporcionar una propuesta para el desarrollo de un sistema de gestión de información de servicios de catering y eventos con su respectiva base de datos, el cual constará de los siguientes módulos:

- Gestión de clientes:
  - Creación de clientes con su información respectiva.
  - Eliminación de clientes.
- Gestión de productos.
  - Creación de productos.
  - Asignación de tipos de productos para su categorización.
  - Eliminación de clientes.
  - Reserva de stock para la realización de pedidos de eventos.
  - Gestión de consultas.

- Gestión de proformas.
  - Creación de proformas en donde se detallan los menús que se ofrece al cliente.
  - Asignación de estado a las proformas como proformas pendientes y proformas realizadas.
  - Eliminación de proformas.
  - Gestión de consultas de proformas para saber si están pendientes o no.
- Gestión de facturación.
  - Creación de facturas de los eventos realizados.
  - Asignación de fecha de evento a realizar con su respectiva hora.
  - Anulación de facturas en casos que así lo amerite.
  - Gestión de consultas para saber que facturas faltan por cobrar.
- Gestión de usuarios del sistema.
  - Creación de usuarios.
  - Asignación de permisos de usuario.
  - Eliminación de usuarios.
- Gestión de publicidad.
  - Contiene información de la empresa, con los servicios que ofrecen.
  - Permite descargar recetas de cocina así como también videos referentes a todo lo que son los servicios de catering.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- Contiene un formulario de contactos para que los clientes puedan realizar sus consultas sobre los servicios que ofrecen y poder tener un contacto con ellos por medio del internet y así enviarles la información de proformas.

#### **4.2.1.4.2 Suposiciones y Restricciones**

Las suposiciones y restricciones respecto al sistema, y que se derivan directamente de las entrevistas con el stakeholder de la empresa son:

a) Debe contemplarse las implicaciones de los siguientes puntos críticos:

- Compatibilidad de la solución con protocolos IPV6.
- Caracteres multilingües.
- Sistemas seguros: protección de la información, seguridad en las transacciones de datos.
- Gestión de flujos de trabajo, seguridad de transacciones e intercambio de información.

Como es natural, la lista de suposiciones y restricciones se incrementará durante el desarrollo del proyecto, particularmente una vez establecido el artefacto “Visión”.

#### **4.2.1.4.3 Entregables del proyecto**

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de RUP<sup>7</sup> desde la perspectiva de artefactos, y que proponemos para este proyecto.

Es preciso destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al

---

<sup>7</sup> Es un proceso de desarrollo de software desarrollado por la empresa Rational Software, actualmente propiedad de IBM.

término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración.

### **1) Plan de Desarrollo del Software**

Es el presente documento.

### **2) Modelo de Casos de Uso del Negocio**

Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, solicitantes finales, otros sistemas etc.). Permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito. Este modelo se representa con un Diagrama de Casos de Uso usando estereotipos específicos para este modelo.

### **3) Modelo de Objetos del Negocio**

Es un modelo que describe la realización de cada caso de uso del negocio, estableciendo los actores internos, la información que en términos generales manipulan y los flujos de trabajo (workflows) asociados al caso de uso del negocio. Para la representación de este modelo se utilizan Diagramas de Colaboración (para mostrar actores externos, internos y las entidades (información) que manipulan, un Diagrama de Clases para mostrar gráficamente las entidades del sistema y sus relaciones, y Diagramas de Actividad para mostrar los flujos de trabajo.

### **4) Glosario**

Es un documento que define los principales términos usados en el proyecto. Permite establecer una terminología consensuada. .

### **5) Modelo de Casos de Uso**

El modelo de Casos de Uso presenta las funciones del sistema y los actores que hacen uso de ellas. Se representa mediante Diagramas de

Casos de Uso.

## **6) Visión**

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.

## **7) Especificaciones de Casos de Uso**

Para los casos de uso que lo requieran (cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa) se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, post-condiciones, flujo de eventos, requisitos no-funcionales asociados. También, para casos de uso cuyo flujo de eventos sea complejo podrá adjuntarse una representación gráfica mediante un Diagrama de Actividad.

## **8) Especificaciones Adicionales**

Este documento capturará todos los requisitos que no han sido incluidos como parte de los casos de uso y se refieren requisitos no-funcionales globales. Dichos requisitos incluyen: requisitos legales o normas, aplicación de estándares, requisitos de calidad del producto, tales como: confiabilidad, desempeño, etc., u otros requisitos de ambiente, tales como: sistema operativo, requisitos de compatibilidad, etc. 4

## **9) Prototipos de Interfaces de Usuario**

Se trata de prototipos que permiten al usuario hacerse una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Sólo los de este último tipo serán entregados al final de la fase de Elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la fase de Construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final. 4

### **10) Modelo de Análisis y Diseño**

Este modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación), de acuerdo al avance del proyecto.

### **11) Modelo de Datos**

Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se utiliza un Diagrama de Clases (donde se utiliza un profile UML<sup>8</sup> para Modelado de Datos, para conseguir la representación de tablas, claves, etc.).

### **12) Modelo de Implementación**

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema. (Este modelo es sólo una versión preliminar al final de la fase de Elaboración, posteriormente tiene bastante refinamiento).

### **13) Modelo de Despliegue**

Este modelo muestra el despliegue la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

### **14) Casos de Prueba**

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración. Cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizar la prueba, y

---

<sup>8</sup> Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

dependiendo del tipo de prueba dicho procedimiento podrá ser automatizable mediante un script de prueba.

#### **15)Solicitud de Cambio**

Los cambios propuestos para los artefactos se formalizan mediante este documento. Mediante este documento se hace un seguimiento de los defectos detectados, solicitud de mejoras o cambios en los requisitos del producto. Así se provee un registro de decisiones de cambios, de su evaluación e impacto, y se asegura que éstos sean conocidos por el equipo de desarrollo. Los cambios se establecen respecto de la última baseline (el estado del conjunto de los artefactos en un momento determinado del proyecto) establecida. En nuestro caso al final de cada iteración se establecerá una baseline.

#### **16)Plan de Iteración**

Es un conjunto de actividades y tareas ordenadas temporalmente, con recursos asignados, dependencias entre ellas. Se realiza para cada iteración, y para todas las fases.

#### **17)Evaluación de Iteración**

Este documento incluye la evaluación de los resultados de cada iteración, el grado en el cual se han conseguido los objetivos de la iteración, las lecciones aprendidas y los cambios a ser realizados.

#### **18)Lista de Riesgos**

Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación.

#### **19)Manual de Instalación**

Este documento incluye las instrucciones para realizar la instalación del producto.

#### **20)Material de Apoyo al Usuario Final**

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo: Guías del Usuario, Guías de Operación, Guías de Mantenimiento y Sistema de Ayuda en Línea

## **21)Producto**

Los ficheros del producto empaquetados y almacenadas en un CD con los mecanismos apropiados para facilitar su instalación. El producto, a partir de la primera iteración de la fase de Construcción es desarrollado incremental e iterativamente, obteniéndose una nueva release al final de cada iteración.

Los artefactos 19, 20 y 21 se generarán a partir de la fase de Construcción, con lo cual se han incluido aquí sólo para dar una visión global de todos los artefactos que se generarán en el proceso de desarrollo.

### **4.2.1.5 Evolución del Plan de Desarrollo del Software**

El Plan de Desarrollo del Software se revisará semanalmente y se refinará antes del comienzo de cada iteración.

### **4.2.1.6 Organización del Proyecto**

#### **4.2.1.6.1 Participantes en el Proyecto**

- **Jefe de Proyecto.**

Labor de Ana Belén Obando, líder de proyectos de la empresa La Viandé. Con una experiencia modesta en metodologías de desarrollo, herramientas CASE y notaciones, en particular la notación UML y el proceso de desarrollo RUP.

- **Analista de Sistemas.**

El perfil establecido es: Ingeniero en Informática con conocimientos de UML, uno de ellos al menos con experiencia en sistemas afines a la línea del proyecto.

- **Analistas - Programadores.**

Con experiencia en el entorno de desarrollo del proyecto, con el fin de que los prototipos puedan ser lo más cercanos posibles al producto final. Este trabajo ha sido encomendado a Ana Belén Obando.

- **Ingeniero de Software.**

El perfil establecido es: Ingeniero en Informática recién titulado que participará como becario en el convenio universidad-empresa, realizando labores de gestión de requisitos, gestión de configuración, documentación y diseño de datos. Encargada de las pruebas funcionales del sistema, realizará la Profesional experto en pruebas.

#### 4.2.1.6.2 Interfaces Externas

La Viandé definirá los participantes del proyecto que proporcionarán los requisitos del sistema, y entre ellos quiénes serán los encargados de evaluar los artefactos de acuerdo a cada subsistema y según el plan establecido. El equipo de desarrollo interactuará activamente con los participantes de La Viandé para especificación y validación de los artefactos generados.

#### 4.2.1.6.3 Roles y Responsabilidades

A continuación se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en RUP.

<b>Puesto</b>	<b>Responsabilidad</b>
Jefe de Proyecto	El jefe de proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El jefe de proyecto también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, el jefe de proyecto se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Analista de Sistemas	Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaboración del Modelo de Análisis y Diseño. Colaboración en la elaboración de las pruebas funcionales y el modelo de datos.
Programador	Construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario
Ingeniero de Software	Gestión de requisitos, gestión de configuración y cambios, elaboración del modelo de datos, preparación de las pruebas funcionales, elaboración de la documentación. Elaborar modelos de implementación y despliegue.

Tabla N° 4. 11. Roles y responsabilidades.

#### 4.2.1.7 Gestión del Proceso

##### 4.2.1.7.1 Estimaciones del Proyecto

El presupuesto del proyecto y los recursos involucrados se adjuntan en un documento separado.

##### 4.2.1.7.2 Plan del Proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

##### 4.2.1.7.3 Plan de las Fases

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase (para las fases de Construcción y Transición es sólo una aproximación muy preliminar).

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Fase	Nro. Iteraciones	Duración
Fase de Inicio	1	6 semanas
Fase de Elaboración	1	4 semanas
Fase de Construcción	2	14 semanas
Fase de Transición	-	-

Tabla N° 4. 12. Plan de las Fases.

Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

Descripción	Hito
Fase de Inicio	En esta fase desarrollarán los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se hará un refinamiento del Plan de Desarrollo del Proyecto. La aceptación del cliente /usuario del artefacto Visión y el Plan de Desarrollo marcan el final de esta fase.
Fase de Elaboración	En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más

	<p>relevantes y / o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera release de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. En nuestro caso particular, por no incluirse las fases siguientes, la revisión y entrega de todos los artefactos hasta este punto de desarrollo también se incluye como hito. La primera iteración tendrá como objetivo la identificación y especificación de los principales casos de uso, así como su realización preliminar en el Modelo de Análisis / Diseño, también permitirá hacer una revisión general del estado de los artefactos hasta este punto y ajustar si es necesario la planificación para asegurar el cumplimiento de los objetivos. Ambas iteraciones tendrán una duración de una semana.</p>
Fase de Construcción	<p>Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a 2 iteraciones, cada una produciendo una release a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario. El hito que marca el fin de esta fase es la versión de la release 2.0, con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para pruebas beta.</p>

Fase de Transición	En esta fase se prepararán dos releases para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto.
--------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla N° 4. 13. Hitos.

#### 4.2.1.7.4 Calendario del Proyecto

A continuación se presenta un calendario de las principales tareas del proyecto incluyendo sólo las fases de Inicio y Elaboración. Como se ha comentado, el proceso iterativo e incremental de RUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase e iteración del proyecto. La siguiente figura ilustra este enfoque, en ella lo ensombrecido marca el énfasis de cada disciplina (workflow) en un momento determinado del desarrollo.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

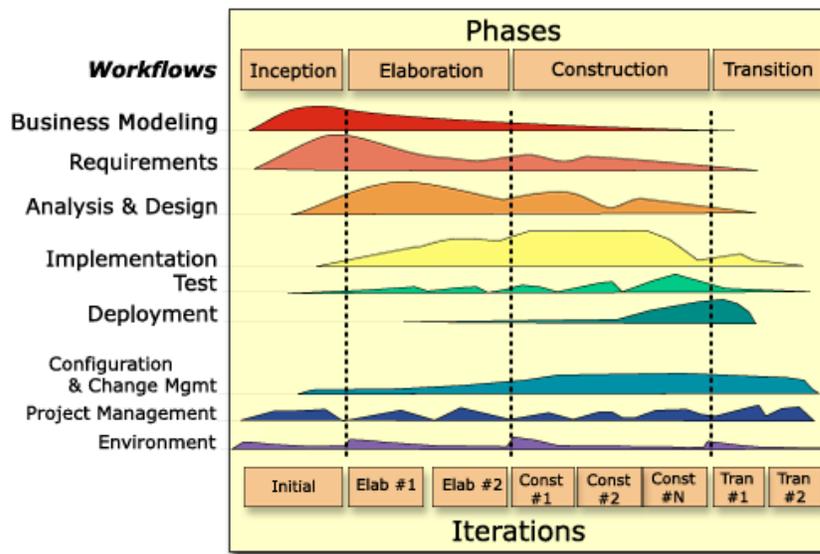


Figura N° 4. 3. Descripción del Proceso unificado.

Fuente: <http://yaqui.mxl.uabc.mx/~molguin/as/RUP.htm>

## 4.3 Especificación de caso de uso: Iniciar Sesión

### 4.3.1 Iniciar Sesión

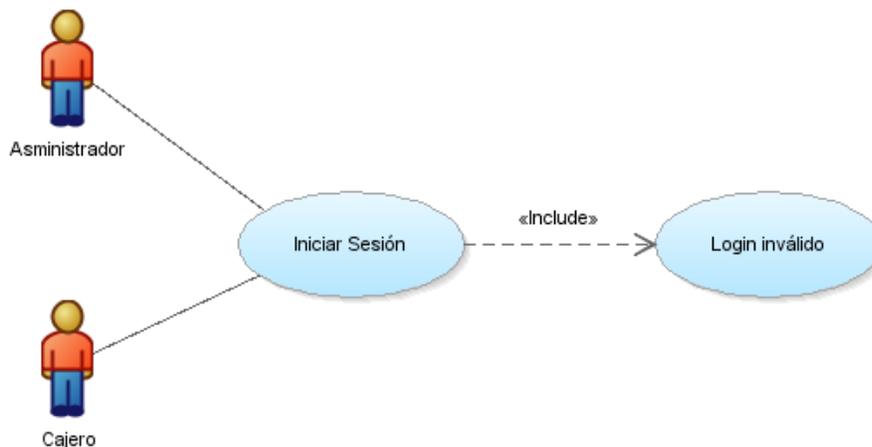


Figura N° 4. 4. Caso de Uso Iniciar Sesión

#### 4.3.1.1 Descripción

Describe cuando un usuario administrador o un cajero deben iniciar su sesión para ingresar al sistema.

### **4.3.1.2 Flujo de Eventos**

#### **4.3.1.2.1 Flujo Básico**

1. El caso de uso comienza cuando el usuario (administrador o cajero) ingresar a un navegador web y pone la dirección de la página.
2. Acceder al formulario de registro.
3. Ingresar el usuario y su contraseña.
4. El sistema valida los datos del usuario.
5. El sistema loguea al usuario.

#### **4.3.1.2.2 Flujos Alternativos**

##### **4.3.1.2.2.1 Datos Inválidos.**

El sistema muestra la ventana “Datos Inválidos” y vuelve el control al punto 2 del flujo principal.

##### **4.3.1.2.3 Precondiciones**

- Los usuarios ya deben estar registrados en el sistema.
- Los usuarios necesitan acceder al sistema y no estar logueados.
- Es necesario recordar el usuario y contraseña.

##### **4.3.1.2.4 Pos condiciones**

Usuario bloqueado.

## 4.4 Especificación de caso de uso: Crear Usuario

### 4.4.1 Crear Usuario

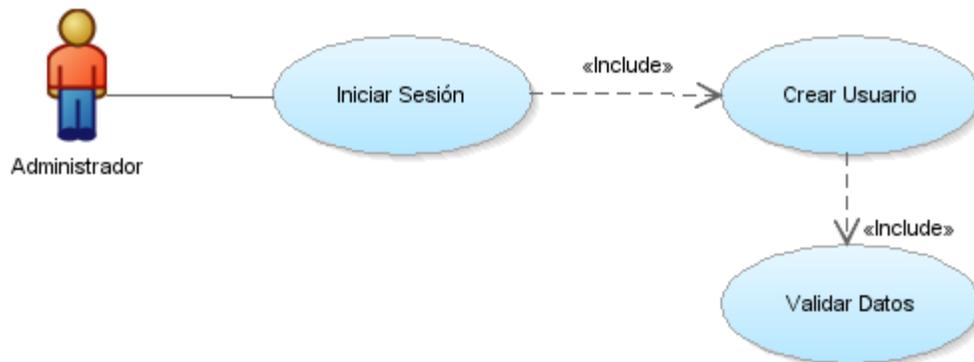


Figura N° 4. 5. Caso de Uso Crear Usuario.

#### 4.4.1.1 Descripción

Mediante este caso de uso el administrador puede crear cuentas de usuario necesarias para acceder al sistema.

#### 4.4.1.2 Flujo de Eventos

##### 4.4.1.2.1 Flujo Básico

- Hacer clic en la opción USUARIOS.
- Hacer clic en NUEVO, para crear un nuevo usuario.
- Rellenar la información de los campos.
- Hacer clic en el botón GUARDAR.

##### 4.4.1.2.2 Flujos Alternativos

- Si los datos no son válidos, el sistema mostrará un mensaje.
- Si el usuario ya existe, el sistema mostrará un mensaje.

#### 4.4.1.3 Precondiciones

El Usuario debe estar Autorizado.

## 4.5 Especificación de caso de uso: Crear Usuario

### 4.5.1 Crear Usuario

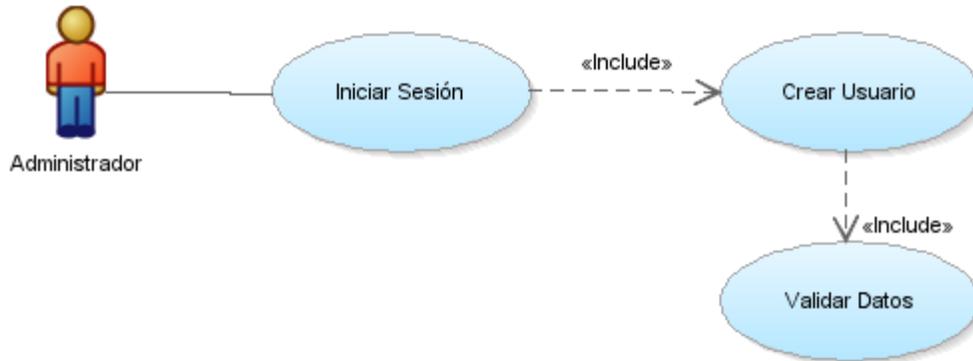


Figura N° 4. 6. Caso de Uso Crear Usuario.

#### 4.5.1.1 Descripción

Mediante este caso de uso el administrador puede crear cuentas de usuario necesarias para acceder al sistema.

#### 4.5.1.2 Flujo de Eventos

##### 4.5.1.2.1 Flujo Básico

- Hacer clic en la opción USUARIOS.
- Hacer clic en NUEVO, para crear un nuevo usuario.
- Rellenar la información de los campos.
- Hacer clic en el botón GUARDAR.

##### 4.5.1.2.2 Flujos Alternativos

- Si los datos no son válidos, el sistema mostrará un mensaje.
- Si el usuario ya existe, el sistema mostrará un mensaje.

#### 4.5.1.3 Precondiciones

El Usuario debe estar Autorizado.

#### 4.5.1.4 Pos condiciones

Usuario creado.

## 4.6 Especificación de caso de uso: Eliminar Usuario

### 4.6.1 Eliminar Usuario

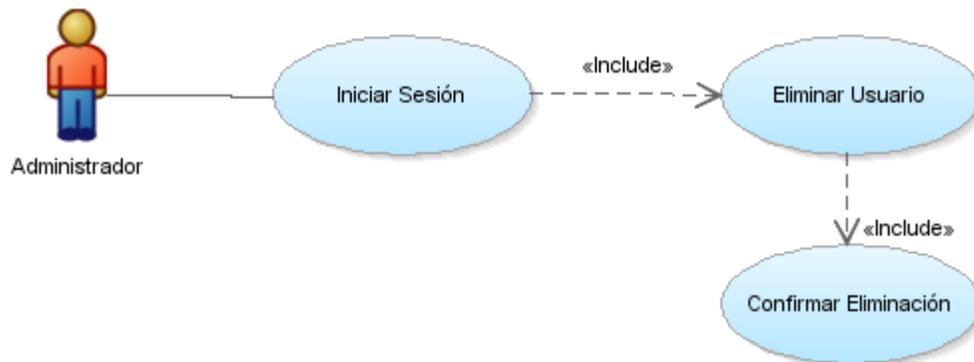


Figura N° 4. 7. Caso de Uso Eliminar Usuario.

#### 4.6.1.1 Descripción

Mediante este caso de uso el administrador puede eliminar los usuarios que se encuentren registrados en el sistema.

#### 4.6.1.2 Flujo de Eventos

##### 4.6.1.2.1 Flujo Básico

- Hacer clic en el botón ELIMINAR.
- Hacer clic en el botón ACEPTAR.

##### 4.6.1.2.2 Flujos Alternativos

Hacer clic en el botón CANCELAR en el mensaje de confirmación de eliminación.

#### 4.6.1.3 Precondiciones

El Usuario debe estar Creado.

#### 4.6.1.4 Pos condiciones

Crear Usuario, Otorgar Permisos.

## 4.7 Especificación de caso de uso: Buscar Usuario

### 4.7.1 Buscar Usuario

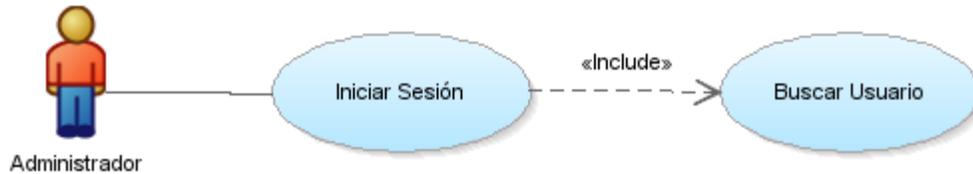


Figura N° 4. 8. Caso de Uso Buscar Usuario.

#### 4.7.1.1 Descripción

Mediante este caso de uso el administrador puede buscar los usuarios que se encuentren registrados en el sistema, permite buscar un usuario por nombre del usuario.

#### 4.7.1.2 Flujo de Eventos

##### 4.7.1.2.1 Flujo Básico

- Hacer clic en el botón BUSCAR.
- Ingresar el nombre del usuario que se desea buscar.
- Hacer clic en el botón <>.

##### 4.7.1.2.2 Flujos Alternativos

Hacer clic en CERRAR.

#### 4.7.1.3 Precondiciones

- El administrador debe estar logueado.
- El usuario debe estar Creado.
- Lista de usuarios disponible.

#### 4.7.1.4 Pos condiciones

Ninguno.

## 4.8 Especificación de caso de uso: Crear Grupo de Productos

### 4.8.1 Crear Grupo de Productos

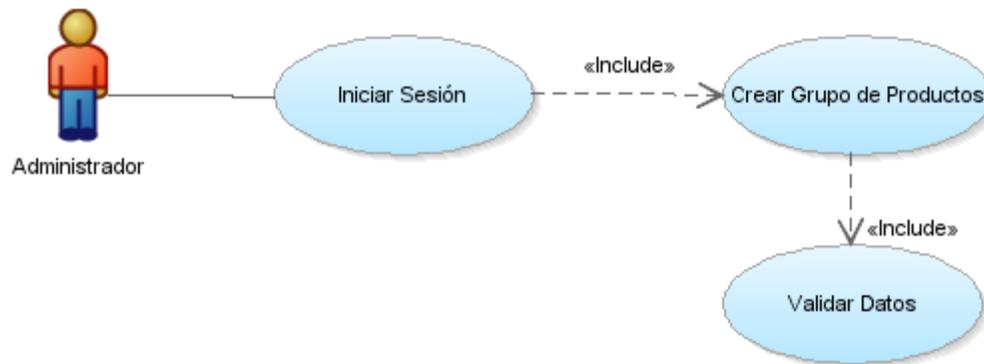


Figura N° 4. 9. Caso de Uso Crear Grupo de Productos

#### 4.8.1.1 Descripción

Mediante este caso de uso el administrador puede crear diferentes grupos de productos, los cuales serán utilizados más adelante.

#### 4.8.1.2 Flujo de Eventos

##### 4.8.1.2.1 Flujo Básico

- Hacer clic en la opción del menú principal GRUPO DE PRODUCTOS.
- Hacer clic en NUEVO, para crear un nuevo grupo de producto.
- Rellenar la información de los campos.
- Hacer clic en el botón GUARDAR.

##### 4.8.1.2.2 Flujos Alternativos

Hacer clic en CERRAR.

#### 4.8.1.3 Precondiciones

El usuario decide crear un nuevo grupo de producto.

#### 4.8.1.4 Pos condiciones

Grupo de Producto creado.

## 4.9 Especificación de caso de uso: Eliminar Grupo de Productos

### 4.9.1 Eliminar Grupo de Productos

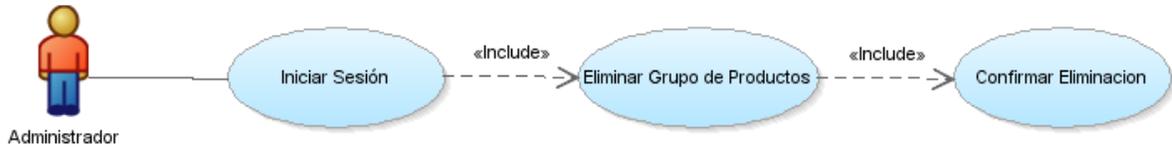


Figura N° 4. 10. Caso de Uso Eliminar Grupo de Productos.

### 4.9.2 Descripción

Mediante este caso de uso el administrador puede eliminar los grupos de productos que se encuentren registrados en el sistema.

#### 4.9.2.1 Flujo de Eventos

##### 4.9.2.1.1 Flujo Básico

- Hacer clic en el botón BUSCAR.
- Buscar el grupo de producto en el buscador.
- Seleccionar el grupo de producto que se desea eliminar presionando el botón <>, en el cual se puede editar algún campo del grupo de producto.
- Hacer clic en el botón ELIMINAR.
- Se mostrará una ventana de verificación, hacer clic en el botón ACEPTAR.

##### 4.9.2.1.2 Flujos Alternativos

Hacer clic en el botón CANCELAR que nos muestra la ventana de verificación.

##### 4.9.2.2 Precondiciones

El grupo de producto debe existir en el sistema, debe constar en la lista de grupo de productos.

##### 4.9.2.3 Pos condiciones

El usuario crea un nuevo grupo de producto.

## 4.10 Especificación de caso de uso: Crear Producto

### 4.10.1 Crear Producto

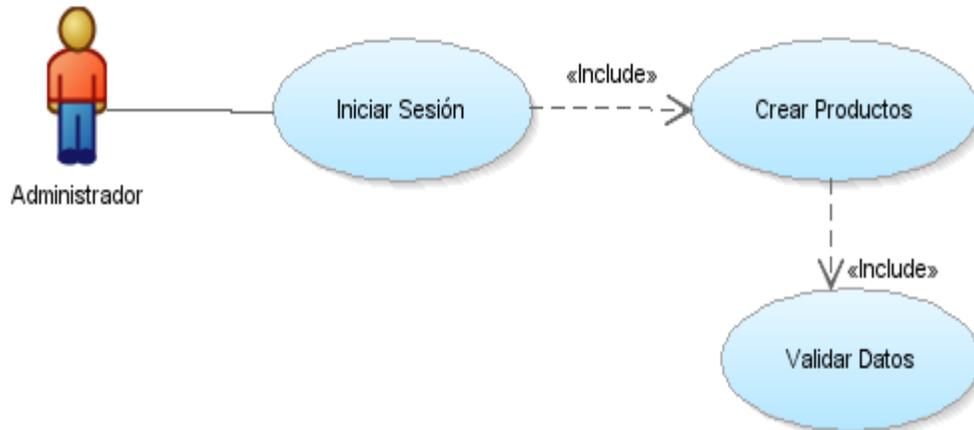


Figura N° 4. 11. Caso de Uso Crear Producto.

### 4.10.2 Descripción

Mediante este caso de uso el usuario puede crear productos en el sistema.

### 4.10.3 Flujo de Eventos

#### 4.10.3.1 Flujo Básico

- Hacer clic en la opción PRODUCTOS del menú principal
- Hacer clic en NUEVO, para crear un nuevo producto.
- Rellenar la información de los campos.
- Hacer clic en el botón GUARDAR.

#### 4.10.3.2 Flujos Alternativos

- Si los datos no son válidos, el sistema mostrará un mensaje.
- Si el producto ya existe, el sistema mostrará un mensaje.

### 4.10.4 Precondiciones

El usuario decide crear un nuevo producto.

### 4.10.5 Pos condiciones

El usuario crea un nuevo producto.

## 4.11 Especificación de caso de uso: Eliminar Producto

### 4.11.1 Eliminar Producto

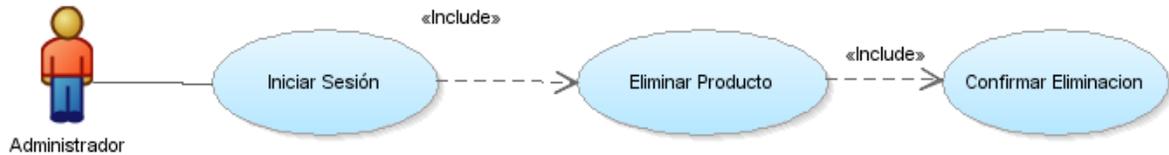


Figura N° 4. 12. Caso de Uso Eliminar Producto.

### 4.11.2 Descripción

Mediante este caso de uso el administrador puede eliminar los productos que se encuentren registrados en el sistema.

### 4.11.3 Flujo de Eventos

#### 4.11.3.1 Flujo Básico

- Hacer clic en el botón BUSCAR.
- Buscar el producto en el buscador.
- Seleccionar el producto que se desea eliminar presionando el botón <>, en el cual se puede editar algún campo del producto.
- Hacer clic en el botón ELIMINAR.
- Se mostrará una ventana de verificación, hacer clic en el botón ACEPTAR.

#### 4.11.3.2 Flujos Alternativos

Hacer clic en el botón CANCELAR que nos muestra la ventana de verificación.

### 4.11.4 Precondiciones

El producto debe estar registrado en el sistema, debe constar en la lista de productos.

#### 4.11.5 Pos condiciones

El usuario elimina un producto.

### 4.12 Especificación de caso de uso: Crear Cliente

#### 4.12.1 Crear Cliente

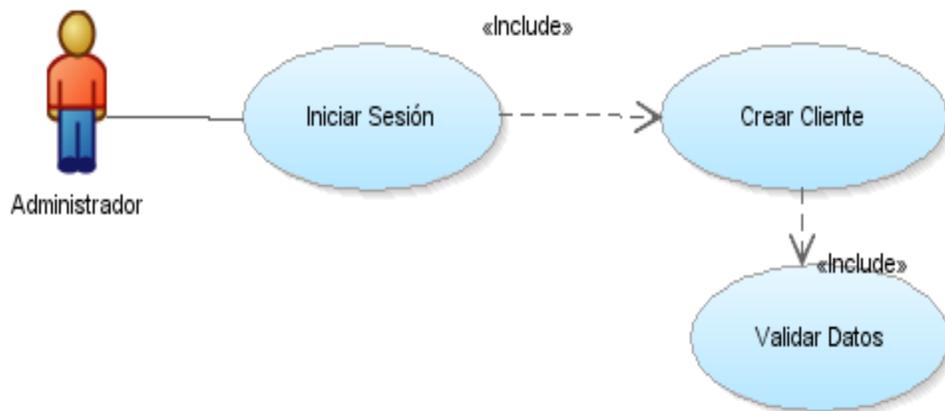


Figura N° 4. 13. Caso de Uso Crear Cliente

#### 4.12.2 Descripción

Mediante este caso de uso el cajero puede crear clientes en el sistema.

#### 4.12.3 Flujo de Eventos

##### 4.12.3.1 Flujo Básico

- Hacer clic en la opción CLIENTES, del menú principal.
- Hacer clic en NUEVO, para crear un nuevo cliente.
- Rellenar la información de los campos.
- Hacer clic en el botón GUARDAR.

#### 4.12.3.2 Flujos Alternativos

- Si los datos no son válidos, el sistema mostrará un mensaje.
- Si el cliente ya existe, el sistema mostrará un mensaje.

#### 4.12.4 Precondiciones

El cajero decide crear un nuevo cliente.

#### 4.12.5 Pos condiciones

El cajero crea un nuevo cliente.

### 4.13 Especificación de caso de uso: Eliminar Cliente

#### 4.13.1 Eliminar Cliente

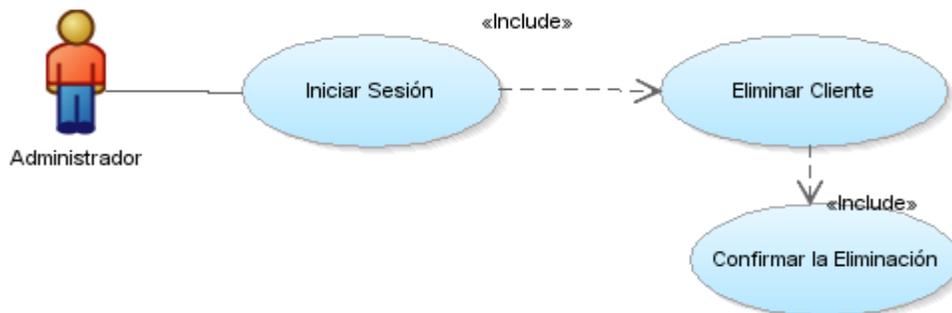


Figura N° 4. 14. Caso de Uso Eliminar Cliente.

#### 4.13.2 Descripción

Mediante este caso de uso el cajero puede eliminar clientes que se encuentren registrados en el sistema.

#### 4.13.3 Flujo de Eventos

##### 4.13.3.1 Flujo Básico

- Hacer clic en el botón BUSCAR.
- Buscar el cliente que se desea eliminar en el buscador.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- Seleccionar el cliente que se desea eliminar presionando el botón <>, en el cual se puede editar algún campo del cliente.
- Hacer clic en el botón ELIMINAR.
- Se mostrará una ventana de verificación, hacer clic en el botón ACEPTAR.

#### 4.13.3.2 Flujos Alternativos

Hacer clic en el botón CANCELAR que nos muestra la ventana de verificación.

#### 4.13.4 Precondiciones

El cliente a ser eliminado debe estar registrado en el sistema, debe constar en la lista de clientes.

#### 4.13.5 Pos condiciones

El cajero elimina un cliente.

### 4.14 Especificación de caso de uso: Crear Proforma

#### 4.14.1 Crear Proforma

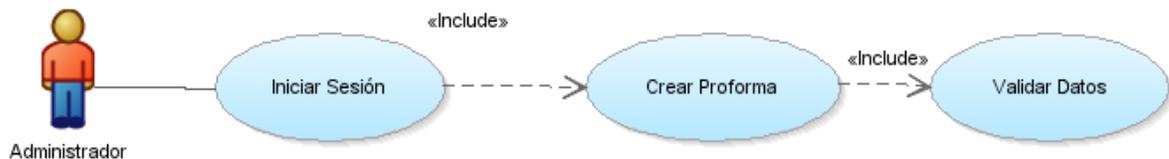


Figura N° 4. 15. Caso de Uso Crear Proforma.

##### 4.14.1.1 Descripción

Mediante este caso de uso el cajero puede crear proformas en el sistema.

## 4.15 Especificación de caso de uso: Crear Proforma

### 4.15.1 Crear Proforma

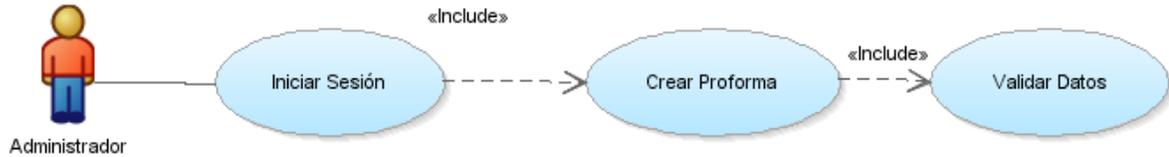


Figura N° 4. 16. Caso de Uso Crear Proforma.

#### 4.15.1.1 Descripción

Mediante este caso de uso el cajero puede crear proformas en el sistema.

#### 4.15.2 Flujo de Eventos

##### 4.15.2.1 Flujo Básico

- Hacer clic en la opción Proformas, del menú principal.
- Hacer clic en PROFORMA.
- Hacer clic en NUEVO, para crear una nueva proforma.
- Rellenar la información de los campos.
- Dar clic en el botón “?” para seleccionar el cliente al que le pertenece la proforma.
- Hacer clic en el botón “+” para agregar productos a la proforma.
- Hacer clic en el botón GUARDAR.

##### 4.15.2.2 Flujos Alternativos

- Si los datos no son válidos, el sistema mostrará un mensaje.
- Si el número de la proforma ya existe, el sistema mostrará un mensaje.

#### 4.15.3 Precondiciones

- Debe existir en el sistema el cliente perteneciente a la proforma.
- Deben existir productos para utilizarlos en la proforma.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

#### 4.15.4 Pos condiciones

La proforma fue creada.

### 4.16 Especificación de caso de uso: Crear Proforma

#### 4.16.1 Crear Proforma

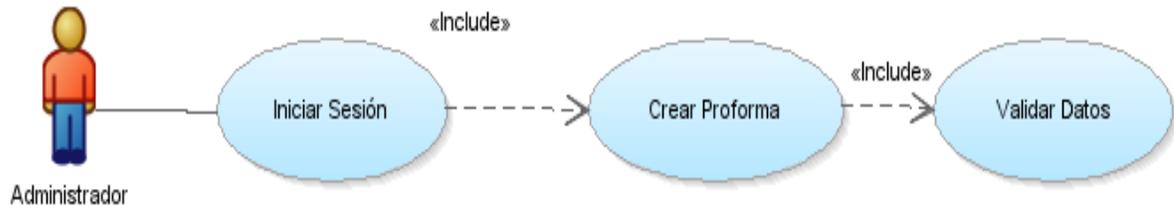


Figura N° 4. 17. Caso de Uso Crear Proforma.

##### 4.16.1.1 Descripción

Mediante este caso de uso el cajero puede crear proformas en el sistema.

#### 4.16.2 Flujo de Eventos

##### 4.16.2.1 Flujo Básico

- Hacer clic en la opción Proformas, del menú principal.
- Hacer clic en PROFORMA.
- Hacer clic en NUEVO, para crear una nueva proforma.
- Rellenar la información de los campos.
- Dar clic en el botón “?” para seleccionar el cliente al que le pertenece la proforma.
- Hacer clic en el botón “+” para agregar productos a la proforma.
- Hacer clic en el botón GUARDAR.

##### 4.16.2.2 Flujos Alternativos

- Si los datos no son válidos, el sistema mostrará un mensaje.
- Si el número de la proforma ya existe, el sistema mostrará un mensaje.

## 4.17 Especificación de caso de uso: Crear Proforma

### 4.17.1 Crear Proforma

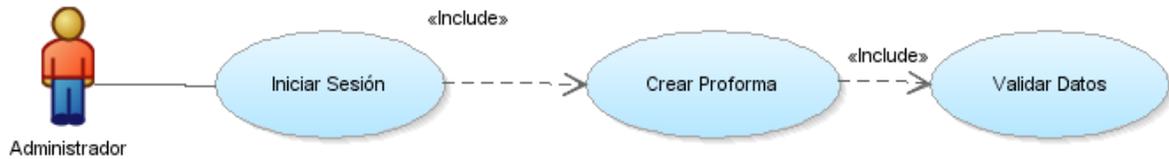


Figura N° 4. 18. Caso de Uso Crear Proforma.

#### 4.17.1.1 Descripción

Mediante este caso de uso el cajero puede crear proformas en el sistema.

#### 4.17.2 Flujo de Eventos

##### 4.17.2.1 Flujo Básico

- Hacer clic en la opción Proformas, del menú principal.
- Hacer clic en PROFORMA.
- Hacer clic en NUEVO, para crear una nueva proforma.
- Rellenar la información de los campos.
- Dar clic en el botón “?” para seleccionar el cliente al que le pertenece la proforma.
- Hacer clic en el botón “+” para agregar productos a la proforma.
- Hacer clic en el botón GUARDAR.

##### 4.17.2.2 Flujos Alternativos

- Si los datos no son válidos, el sistema mostrará un mensaje.
- Si el número de la proforma ya existe, el sistema mostrará un mensaje.

#### 4.17.3 Precondiciones

- Debe existir en el sistema el cliente perteneciente a la proforma.
- Deben existir productos para utilizarlos en la proforma.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

#### 4.17.4 Pos condiciones

La proforma fue creada.

### 4.18 Especificación de caso de uso: Eliminar Proforma

#### 4.18.1 Eliminar Proforma

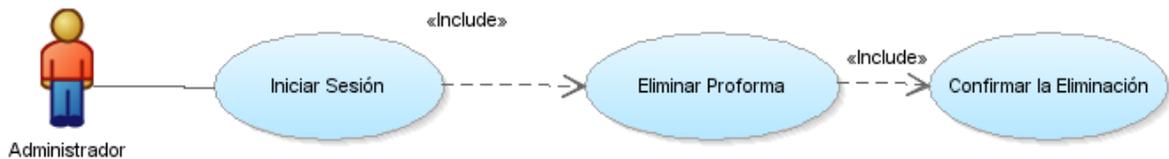


Figura N° 4. 19. Caso de Uso Eliminar Proforma.

#### 4.18.2 Descripción

Mediante este caso de uso el cajero puede eliminar proformas que se encuentren registrados en el sistema.

#### 4.18.3 Flujo de Eventos

##### 4.18.3.1 Flujo Básico

- Hacer clic en el botón ELIMINAR.
- Se mostrará una ventana de verificación, hacer clic en el botón ACEPTAR.

##### 4.18.3.2 Flujos Alternativos

Hacer clic en el botón CANCELAR que nos muestra la ventana de verificación.

#### 4.18.4 Precondiciones

Debe existir una proforma registrada en el sistema para ser eliminada.

#### 4.18.5 Pos condiciones

El cajero elimina una proforma.

## 4.19 Especificación de caso de uso: Generar Orden

### 4.19.1 Generar Orden

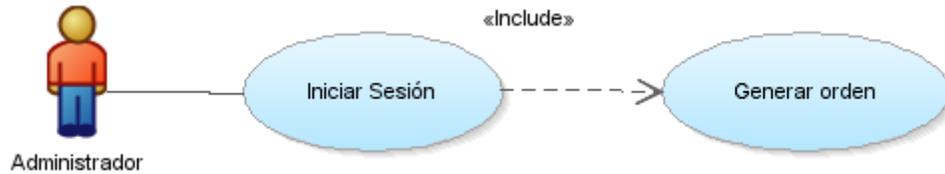


Figura N° 4. 20. Caso de Uso Generar Orden.

### 4.19.2 Descripción

Mediante este caso de uso el cajero puede generar una orden de proforma.

### 4.19.3 Flujo de Eventos

#### 4.19.3.1 Flujo Básico

- Hacer clic en GENERAR PROFORMA, para crear una orden de proforma.
- Hacer clic en el botón ACEPTAR, que se muestra en la ventana de verificación.
- Hacer clic en el botón GUARDAR.

#### 4.19.3.2 Flujos Alternativos

Hacer clic en el botón CANCELAR, en caso de que queramos salir de la ventana de verificación.

### 4.19.4 Precondiciones

Debe existir una proforma en el sistema.

### 4.19.5 Pos condiciones

La orden de proforma fue generada.

## 4.20 Especificación de caso de uso: Generar Factura

### 4.20.1 Generar Factura

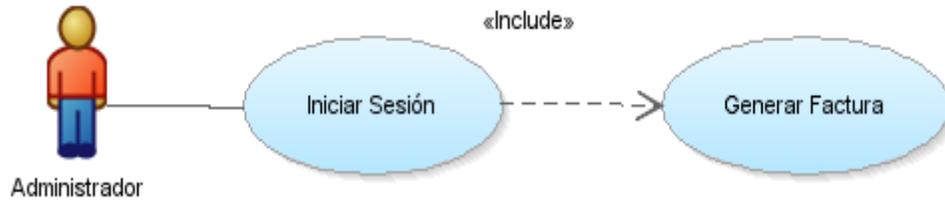


Figura N° 4. 21. Caso de Uso Generar Factura.

### 4.20.2 Descripción

Mediante este caso de uso el cajero puede generar una factura.

### 4.20.3 Flujo de Eventos

#### 4.20.3.1 Flujo Básico

- Hacer clic en la opción FACTURACIÓN, del menú principal.
- Hacer clic en FACTURA.
- Hacer clic en el botón BUSCAR ORDEN, para facturar una orden pendiente.
- Podemos buscar por número, por cliente o por comentario.
- Hacer clic en el botón <> para seleccionar la orden.
- Para agregar al cliente solo hacemos clic en el botón “?”.
  
- Para añadir mas productos a la factura hacemos clic en el botón “+”.
- Para guardar simplemente damos clic en el botón GUARDAR.

#### 4.20.3.2 Flujos Alternativos

- Hacer clic en la opción ARCHIVO.
- Hacer clic en la opción SALIR.

#### 4.20.4 Precondiciones

Debe existir una orden creada en el sistema.

#### 4.20.5 Pos condiciones

La factura fue creada.

### 4.21 Especificación de caso de uso: Eliminar Factura

#### 4.21.1 Eliminar Factura

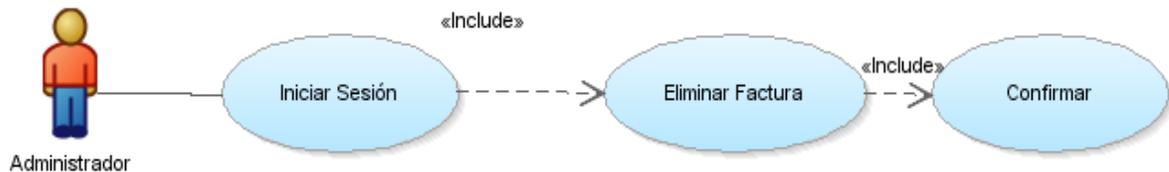


Figura N° 4. 22. Caso de Uso Eliminar Factura.

#### 4.21.2 Descripción

Mediante este caso de uso el administrador puede eliminar una factura.

#### 4.21.3 Flujo de Eventos

##### 4.21.3.1 Flujo Básico

- Hacer clic en la opción FACTURACIÓN, del menú principal.
- Hacer clic en ANULAR FACTURA.
- Hacer clic en el botón BUSCAR FACTURA, podemos realizar una búsqueda de factura por número, por cliente y por comentario.
- Seleccionamos la factura que deseamos anular con el botón <>.
- Hacer clic en ANULAR.
- Hacer clic en el botón ACEPTAR, que nos muestra la ventana de verificación.

#### **4.21.3.2 Flujos Alternativos**

Hacer clic en la opción el botón CANCELAR.

#### **4.21.4 Precondiciones**

Debe existir una factura creada en el sistema.

#### **4.21.5 Pos condiciones**

La factura fue anulada.

### **4.22 Casos de Pruebas**

#### **4.22.1 Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso Elaborar Factura

Las pruebas realizadas a este caso de uso son:

- Crear una factura y guardar.
- Crear Factura y buscar.

#### **4.22.2 Crear una factura y guardar**

##### **4.22.2.1 Descripción**

El cajero que es usuario del sistema, ingresa al sistema introduciendo su usuario y contraseña, para que pueda iniciar sesión. Luego deberá ir al menú principal y hacer clic en FACTURACIÓN y después en FACTURA. Ingresamos todos los datos necesarios para la factura así como también seleccionamos el cliente. Finalmente hacemos clic en GUARDAR.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

#### **4.22.2.2 Condiciones de ejecución**

El usuario cajero debe haber iniciado sesión en el sistema, deben existir productos almacenados en la base de datos así como también clientes.

#### **4.22.2.3 Entrada**

- Ingresamos el Id de esta factura en este caso será la N° 1.
- Ingresamos el Número de la factura para esta prueba es la N° 4.
- Ingresamos la Fecha, “12-10-12”, la seleccionamos del calendario.
- Presionamos el botón “?”, para seleccionar el Cliente, para este ejemplo el cliente será “Juan Rocha”.
- Seleccionamos del combo box la Forma de pago: efectivo.
- Ingresamos los días crédito que en este caso serán 15.
- Ingresamos un Comentario: “factura con crédito a 15 días”.
- Presionamos el botón Orden para que se cargue automáticamente la orden, en este caso será la orden 3.

#### **4.22.2.4 Resultado esperado**

El sistema almacena en la Base de datos esta factura, y nos indicará que fue guardada exitosamente.

#### **4.22.2.5 Evaluación de la Prueba**

Prueba superada con éxito.

### **4.22.3 Crear Factura y Buscar**

#### **4.22.3.1 Descripción**

El usuario del sistema cajero, ingresa al sistema introduciendo su usuario y contraseña, para que pueda iniciar sesión. Después deberá ir al menú principal y hacer clic en FACTURACIÓN y después en FACTURA. Ingresamos todos los datos necesarios para la factura así como también

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

seleccionamos el cliente, hacemos clic en GUARDAR y después hacemos clic en BUSCAR FACTURAS.

#### **4.22.3.2 Condiciones de ejecución**

El usuario cajero debe iniciar sesión en el sistema, deben existir productos almacenados en la base de datos así como también clientes, debe haber proformas pendientes existentes en el sistema.

#### **4.22.3.3 Entrada**

- Ingresamos el Id de esta factura en este caso será la N° 2.
- Ingresamos el Número de la factura para esta prueba es la N° 5.
- Ingresamos la Fecha, "14-10-12", la seleccionamos del calendario.
- Presionamos el botón "?", para seleccionar el Cliente, para este ejemplo el cliente será "Paola Mora".
- Seleccionamos del combo box la Forma de pago: efectivo.
- Ingresamos los días crédito que en este caso serán 15.
- Ingresamos un Comentario: "factura con crédito a 15 días".
- Presionamos el botón Orden para que se cargue automáticamente la orden, en este caso será la orden 4.
- Presionamos el botón BUSCAR FACTURAS y nos mostrará todas las facturas almacenadas en el sistema incluyendo la que acabamos de ingresar.
- Ingresamos el nombre Paola Mora e inmediatamente nos buscará las facturas pertenecientes a este cliente.

#### **4.22.3.4 Resultado esperado**

Búsqueda de un cliente de nombre Paola Mora.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

#### **4.22.3.5 Evaluación de la Prueba**

Búsqueda exitosa.

### **4.23 Fase de construcción**

En esta fase de construcción se realiza principalmente el diseño e implementación, aquí se evoluciona el prototipo inicial a un producto operacional con requerimientos especificados en la fase 2. Las características de la aplicación son desarrolladas e integradas al producto, además de ser extensamente probadas para su primera versión.

Los documentos generados en este documento son el modelo entidad-relación y la lista de riesgos como se muestra a continuación.

#### **4.23.1 Vista Lógica**

##### **4.23.1.1 Diagrama de Base de Datos**

Las especificaciones de cada tabla se encuentran anexadas al cd del proyecto en Diccionario de Datos.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

#### **4.23.1.2 Fase de transición**

El manual de instalación y el manual de usuario se encuentran en los anexos adjuntos al cd de este proyecto.

### **4.24 Glosario**

#### **4.24.1 Introducción**

Este documento recoge todos y cada uno de los términos manejados a lo largo de todo el proyecto de desarrollo de un sistema Web integral para La Viandé. Se trata de un diccionario informal de datos y definiciones de la nomenclatura que se maneja, de tal modo que se crea un estándar para todo el proyecto.

#### **4.24.2 Propósito**

El propósito de este glosario es definir con exactitud y sin ambigüedad la terminología manejada en el proyecto de desarrollo en un sistema Web integral para La Viandé. También sirve como guía de consulta para la clarificación de los puntos conflictivos o poco esclarecedores del proyecto.

#### **4.24.3 Alcance**

El alcance del presente documento se extiende a todos los subsistemas definidos para el proyecto de desarrollo del sistema Web integral para La Viandé, de tal modo que la terminología empleada se refleja con claridad en este documento.

#### **4.24.4 Referencias**

El presente glosario hace referencia a los siguientes documentos:

- Documento Plan de Desarrollo Software.
- Documento Visión del Proyecto.
- Documentos de Especificación de Casos de Uso del Proyecto.
- Documentos de Especificación de Casos de Pruebas del Proyecto.

#### **4.24.5 Organización del Glosario**

El presente documento está organizado por definiciones de términos ordenados de forma ascendente según la ordenación alfabética tradicional del español.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

#### **4.24.6 Definiciones**

##### **4.24.6.1 Artefacto RUP**

Producto tangible del proceso que serán expuestos durante el desarrollo de la metodología RUP, con el fin de exponer todos los puntos a tratar y detallarlos para tener una mejor visión de los objetivos, alcances y hasta posibles restricciones.

##### **4.24.6.2 Base de Datos**

Es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

##### **4.24.6.3 MVC**

Patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee datos dinámicos a la página. El modelo es el sistema de gestión de base de datos y la lógica de negocio, y el controlador es el responsable de escribir los eventos de entrada desde la vista.

##### **4.24.6.4 Navegador Web**

Programa que permite visualizar la información que contiene una página web, se encuentra alojada en un servidor dentro de la Word Wide Web o en uno local.

##### **4.24.6.5 RUP**

Proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

##### **4.24.6.6 Servidor**

Es un programa que implementa el protocolo HTTP (HyperText Transfer Protocol), se ejecuta continuamente en un ordenador, manteniéndose a

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

la espera de peticiones por parte de un cliente que responde a estas peticiones adecuadamente, mediante una página web que se exhibirá en el navegador.

#### **4.24.6.7 Sistema Web**

Es un tipo de sistema que apoya la totalidad o parte de sus procesos a través de una red de computadoras o la Word Wide Web.

#### **4.24.6.8 Validación**

Proceso de comprobación de datos según los parámetros establecidos, aplicado al registro de usuarios.

#### **4.24.6.9 Estereotipos UML**

Por ahora no se han considerado ningún estereotipo UML personalizado durante el desarrollo del sistema, sin embargo, conforme se avance en el análisis y diseño de la arquitectura, se irán incluyendo los estereotipos que aparezcan y que sean necesarios.

## CAPÍTULO V

---

### 5. CONCLUSIONES Y RECOMENDACIONES

- **CONCLUSIONES:**

1. Se cumplió satisfactoriamente con los requerimientos de la empresa, LA VIANDÉ, desarrollando una aplicación web para el registro de Clientes, productos, facturación.
2. Con esta aplicación se abre las posibilidades de automatización de los procesos de la empresa.
3. Con la sistematización de los procesos de la empresa, se consiguió mejorar la información generada, con el objeto de utilizarla para la toma de decisiones, además se simplificaron los procesos manuales que llevaba tiempo hacerlo y que algunas veces no eran exacto.
4. Este sistema se desarrolló con la ayuda de la metodología RUP, que permite desarrollar sistemas de calidad.
5. Con el uso de tecnologías open source, se pudo obtener un software eficiente y adaptable.

- **RECOMENDACIONES:**

Para asegurar el correcto funcionamiento del sistema informático de La Viandé, es necesario seguir las siguientes recomendaciones:

1. Socializar el manual de usuario con los usuarios involucrados en el uso del sistema, para que así se maneje de manera adecuada la aplicación.
2. Realizar una adecuada administración de la base de datos, con el fin de mejorar el rendimiento de la aplicación.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

3. Se recomienda a la persona encargada de la administración del sistema así como de la página web, realizar backups de la base de datos frecuentemente, con el fin de que los datos que maneje el sistema estén actualizados.
4. El ingreso de usuario al sistema sobre todo en las claves, deben ser manejadas con cuidado, de manera confidencial por parte del personal y solo personal autorizado debe trabajar con el sistema.

## CAPÍTULO VI

---

### 6. BIBLIOGRAFÍA

#### JDK

- Fundación Wikimedia (2012), Java Development Kit. Recuperado el 16 de Octubre de 2012 de:  
[http://es.wikipedia.org/wiki/Java\\_Development\\_Kit](http://es.wikipedia.org/wiki/Java_Development_Kit)
- Osmosis Latina. (2011). Menú curso básico de Java – JDK. Recuperado de:  
<http://javabasico.osmosislatina.com/curso/objetos.htm>
- Serverperformance. (2009). JAVA SERVER PERFORMANCE - Lo que un arquitecto de software sí puede hacer. Historia de versiones del JDK y conceptos de rendimiento en Java. Recuperado el 2 de Enero del 2009 de:  
<http://serverperformance.blogspot.com/2009/01/historia-de-versiones-del-jdk-y.html>
- Wikimedia Foundation (2013). Java version history.\_Recuperado el 7 de Marzo del 2013. [http://en.wikipedia.org/wiki/Java\\_version\\_history](http://en.wikipedia.org/wiki/Java_version_history)
- Wikimedia Foundation (2013). Java performance. Recuperado el 26 de Enero del 2013 de [http://en.wikipedia.org/wiki/Java\\_performance](http://en.wikipedia.org/wiki/Java_performance)
- Oracle (2013). Java con Java Development Kit. JDK. Recuperado el 7 de Marzo de 2013 de <http://www.java.com/es/download/faq/develop.xml>
- Manzanedo del Campo, M. (2011) JDK (Java Development Kit). Recuperado el 12 de Septiembre de 2011 de:  
<http://zarza.usal.es/~fgarcia/doc/tuto2/Prefacio.htm>
- Wikilibros (2012). Programación en Java. Instalación JDK. Recuperado el 21 de Noviembre de 2012 de:  
[http://es.wikibooks.org/wiki/Programaci%C3%B3n\\_en\\_Java/Instalaci%C3%B3n\\_JDK](http://es.wikibooks.org/wiki/Programaci%C3%B3n_en_Java/Instalaci%C3%B3n_JDK)

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## MVC

- Fundación Wikimedia (2013). Modelo Vista Controlador. MVC. Recuperado el 1 de Marzo de 2013. [http://es.wikipedia.org/wiki/Modelo\\_Vista\\_Controlador](http://es.wikipedia.org/wiki/Modelo_Vista_Controlador)
- García Díaz. V. (2011). Fundamentos de Ingeniería del Software. Modelo-Vista-Controlador. Recuperado el 3 de Junio de 2011 de: <http://es.scribd.com/doc/50209127/Modelo-Vista-Controlador>
- Febe A. Utilización del Patrón Modelo (MVC). Recuperado de: <http://www.monografias.com/trabajos43/patron-modelo-vista/patron-modelo-vista.shtml>
- Danto (2009). Patrón MVC. Recuperado el 28 de Mayo de 2009 de <http://www.slideshare.net/Danto/patrn-mvc>
- Rubén Mauricio. (2007). El patrón de diseño Modelo Vista Controlador o MVC. Recuperado el 1 de Noviembre de 2007 de: <http://rubenmauricio.com/programacion/el-patron-de-diseno-modelo-vista-controlador-o-mvc/>

## JOOMLA

- Morales Gómez. A (2010). Manual de Joomla. Recuperado el 01 de Junio de 2010 de <http://www.desarrolloweb.com/manuales/manual-joomla.html>
- La Webera (2010). Manual de Joomla. Recuperado el 26 de Mayo de: 2010 de <http://www.lawebera.es/manual-joomla>
- Salas V. (2009) Tutorial de Joomla Básico. Recuperado el 07 de Abril de 2009 de: <http://www.deseoaprender.com/JOOMLA/QueEs.html>

## ANEXOS

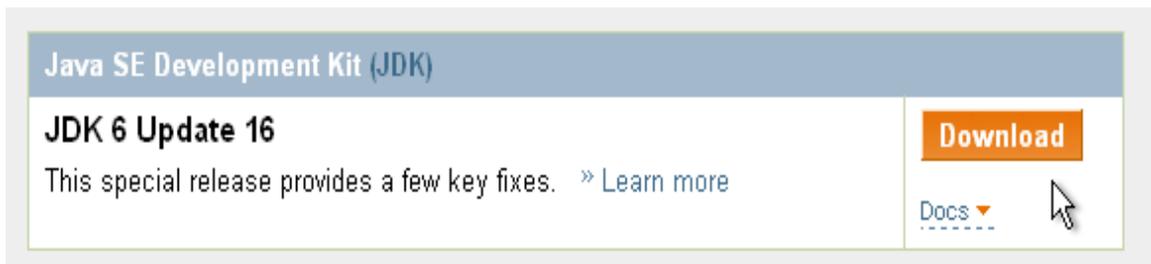
### A. INSTALACIÓN JDK

Para poder descargar JDK debemos ir a la sección de descargas del sitio web de **www.oracle.com**:

<http://www.oracle.com/technetwork/indexes/downloads/index.html>

Debemos realizar la descarga de este compilador de Java (Java Development Kit - JDK) desde la sección Java SE.

Nos aparecerán varias descargas posibles, por lo que tenemos que localizar aquella que nos muestre algo como **Java SE Development Kit (JDK)**.



A continuación debemos **seleccionar la plataforma** (sistema operativo) en el que deseamos hacer la instalación (como por ejemplo en este caso, Windows), luego tenemos que **marcar la casilla** "I agree to the Java SE Development Kit License Agreement" para indicar que si estamos de acuerdo con la licencia de Java.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## Java SE Development Kit 6u16

Provide Information, then Continue to Download

There is more information on the available files for download on the [Supported System Configurations](#) page.

Select Platform and Language for your download:

Platform:

Language:

I agree [Java SE Development Kit 6u16 License Agreement](#)

- Linux
- Linux Intel Itanium
- Linux x64
- Solaris SPARC
- Solaris x64
- Solaris x86
- Windows
- Windows Intel Itanium
- Windows x64

Se puede dar el caso en el que nos aparezca una nueva página con una serie de **posibles archivos que pueden descargarse**. En ese caso deberíamos escoger el que haga referencia a **Java SE Development Kit**. Luego hacemos **clic en el nombre del archivo** (en nuestro caso es: `jdk-6u16-windows-i586.exe`) y a continuación comenzará directamente la descarga del programa. Luego nos muestra dos opciones: la opción de **abrirlo directamente o guardarlo** para abrirlo posteriormente, lo que nos permitirá dejarlo guardado para otro momento.

Available Files

<input checked="" type="checkbox"/>	File Description and Name	Size
<input checked="" type="checkbox"/>	Java SE Development Kit 6u16 jdk-6u16-windows-i586.exe	73.54 MB

[Download Selected with Sun Download Manager »](#)

Easily manage your downloads (pause, resume, restart, verify).  
[» Learn more](#)

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

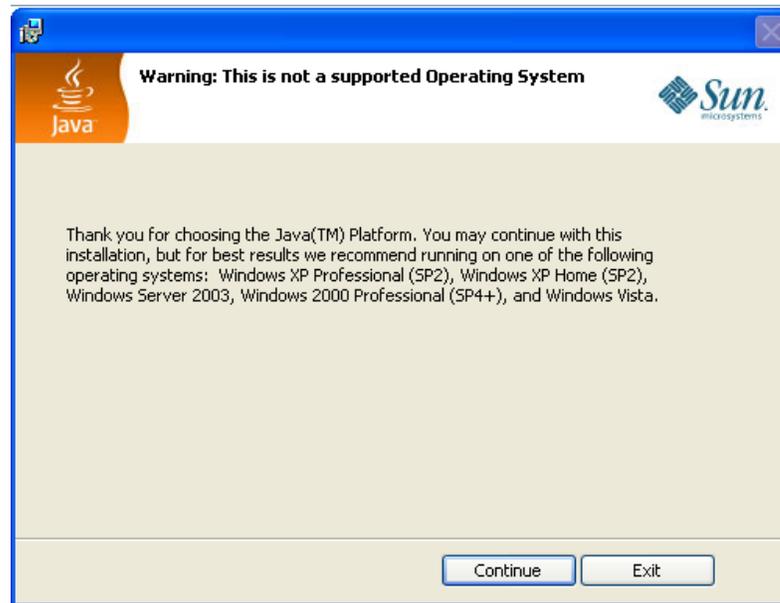


Una vez finalizada la descarga **comienza la instalación.**



Hacemos clic en "**Continue**".

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



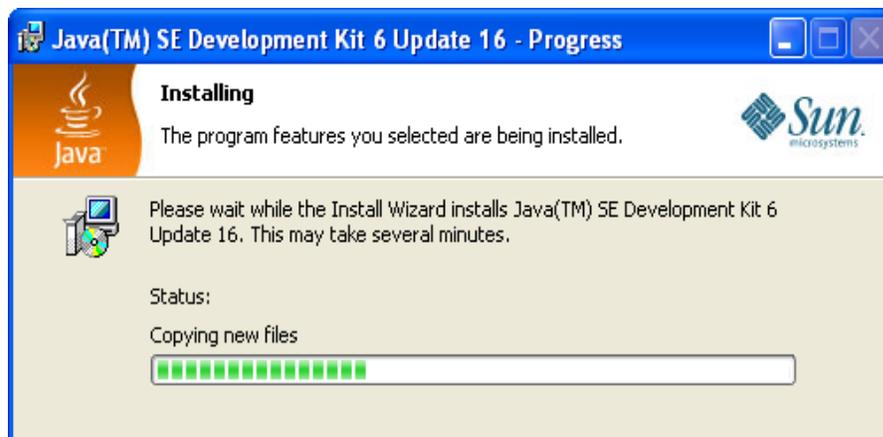
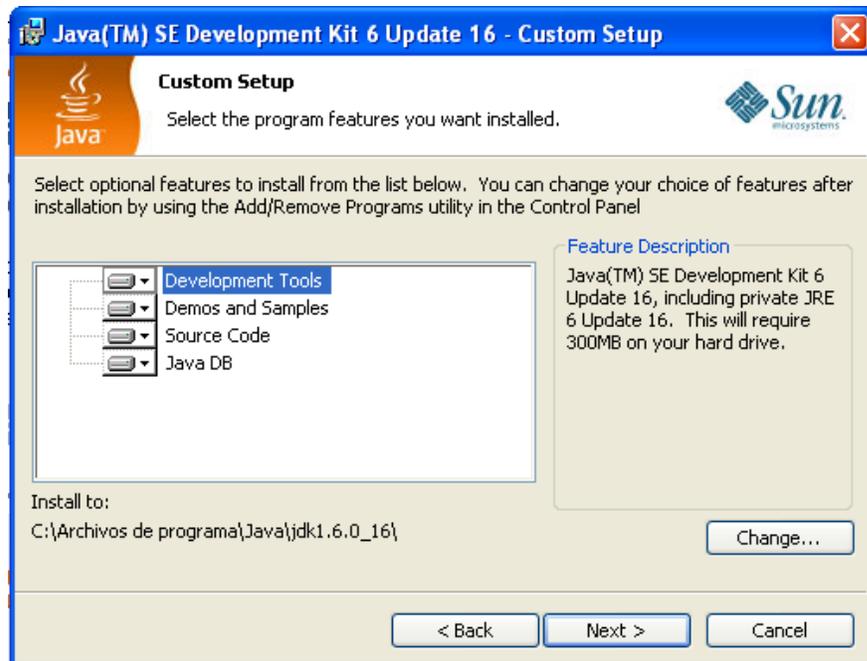
Para poder continuar con la instalación, debemos aceptar la licencia dando clic en el botón "**Accept >**".



Podemos especificar exactamente qué programas o características deseamos instalar. Si no se desea una instalación muy personalizada podemos dejar todo por defecto y hacer clic en "**Next >**" para continuar.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Es necesario **observar la carpeta** en donde se va a realizar la instalación del compilador. En este caso podemos ver que estará en: "**C:\Archivos de programa\Java\jdk1.6.0\_16\**". Según la versión que se descargue, puede variar la carpeta contenedora.



Una vez finalizado el proceso, se debe hacer clic en el botón "**Finish**".

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



Ahora una vez instalado, es el momento de **comprobar** si se realizó bien la instalación y hacer algún último ajuste.

Se debe abrir el "**Símbolo del sistema**" que se encuentra en "**Inicio > Todos los programas > Accesorios > Símbolo del sistema**".

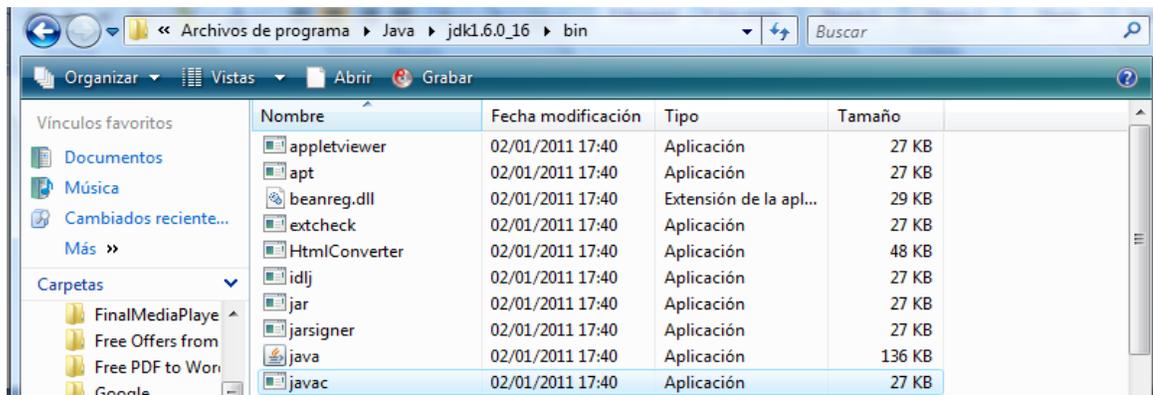
Para realizar la llamada al compilador de Java desde el Símbolo del sistema, lo hacemos con el comando **javac**.

Se debe especificar en qué carpeta se encuentra instalado el compilador, para que funcione.

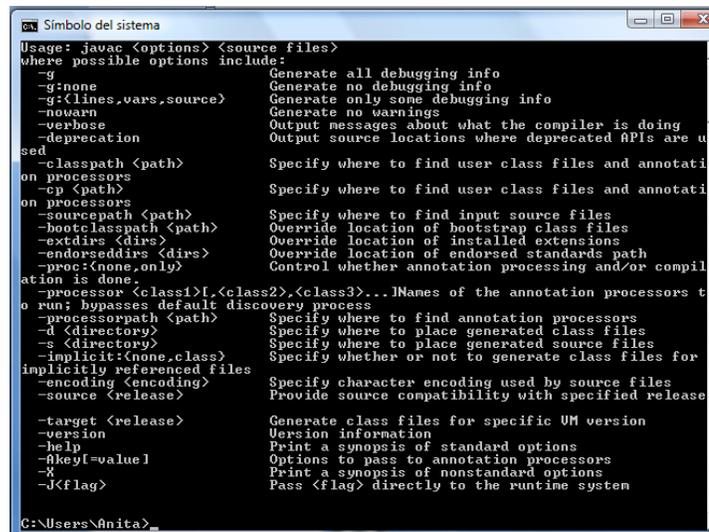
Para poder verificar cuál es la carpeta en donde se realizó la instalación, se debe abrir el ícono "Mi PC o Equipo" y buscar en la unidad C: dentro de la carpeta "Archivos de programa" y "Java". En ella aparecerá una carpeta con el nombre "**jdk**" seguido de los números correspondientes a la versión instalada, en este caso será así: "**jdk1.6.0\_16**". Esta carpeta contiene otra llamada "**bin**", al abrirla vamos a encontrar un archivo ejecutable de nombre "**javac**". En la barra de direcciones podemos observar la ruta seguida hasta encontrar el compilador. Para este caso es:

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

"C:\Archivos de programa\Java\jdk1.6.0\_16\bin".



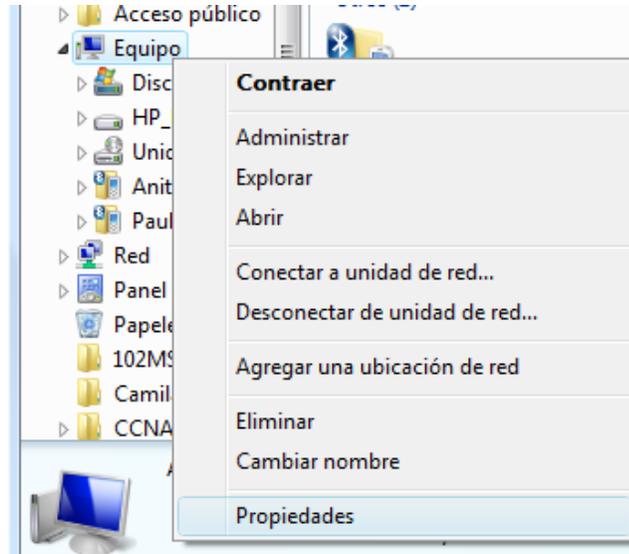
En este caso, la llamada se hace con el comando "**C:\Archivos de programa\Java\jdk1.6.0\_16\bin\javac**", dejando incluso las **comillas** para que los espacios los considere como parte de la llamada al compilador. Recuerda que dependiendo de la versión del JDK descargada, la carpeta de instalación variará.



Para usar con comodidad el compilador, lo ideal es **no tener que estar escribiendo continuamente la ruta completa**. Para esto tenemos que incluir la ruta en la variable **PATH** del sistema. Esto lo haremos siguiendo los siguientes pasos:

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Primero debemos entrar en las **"Propiedades"** de **"Mi PC o Equipo"**, seleccionamos "Propiedades" en el menú contextual abierto.



Accedemos a la pestaña de **"Opciones avanzadas o Configuración avanzada del Sistema"**.

[Ver información básica acerca del equipo](#)

Edición de Windows

Windows 7 Home Premium  
Copyright © 2009 Microsoft Corporation. Reservados todos los derechos.  
Service Pack 1  
[Obtener más características con una nueva edición de Windows 7](#)



Sistema

Fabricante: Sony Electronics Inc.  
Modelo: VAIO® Computer  
Evaluación:  [Evaluación de la experiencia en Windows](#)  
Procesador: Intel(R) Core(TM) i5-2450M CPU @ 2.50GHz 2.50 GHz  
Memoria instalada (RAM): 4,00 GB (3,90 GB utilizable)  
Tipo de sistema: Sistema operativo de 64 bits  
Lápiz y entrada táctil: La entrada táctil o manuscrita no está disponible para esta pantalla

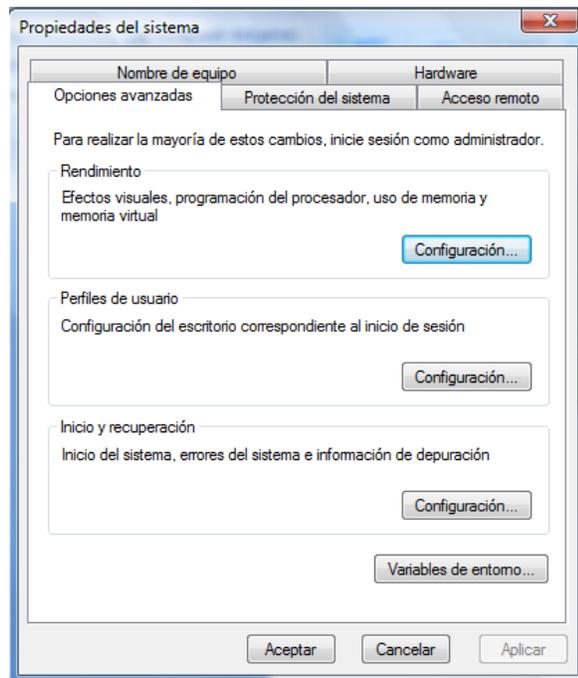
**SONY**

Compatibilidad con Sony Electronics Inc.

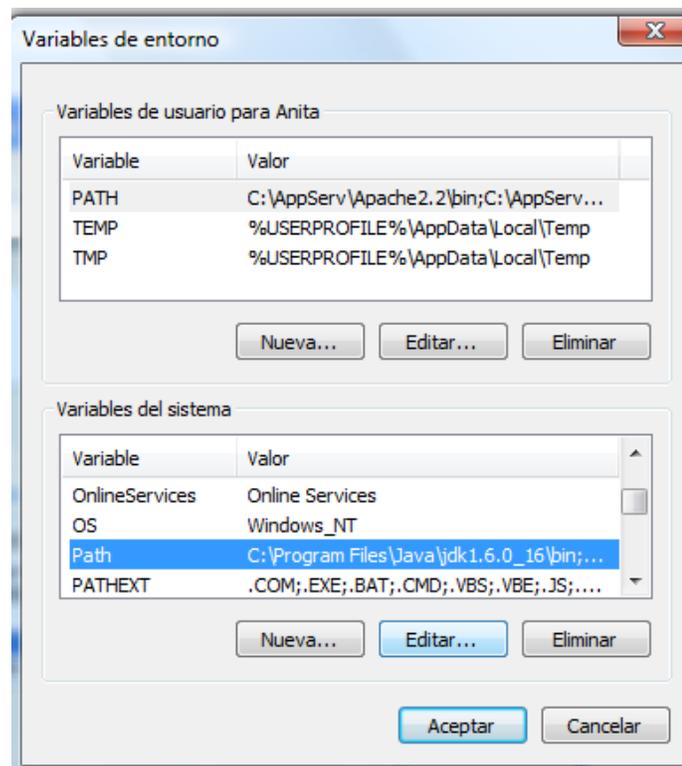
Número de teléfono: Favor de visitar nuestro [website de soporte al cliente](#)  
Horario de soporte técnico: Favor de visitar nuestro [website de soporte al cliente](#)  
Sitio web: [Consulte teléfonos en línea](#)

Luego hacemos un clic en **"Variables de entorno"**.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



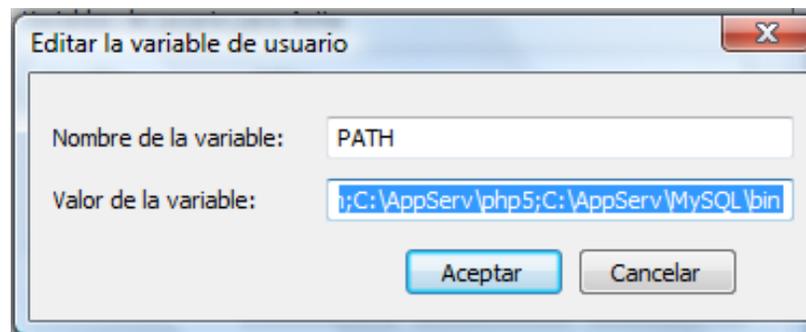
En la sección "Variables del sistema" seleccionar la variable "**Path**" y hacer clic en el botón "**Editar**".



Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

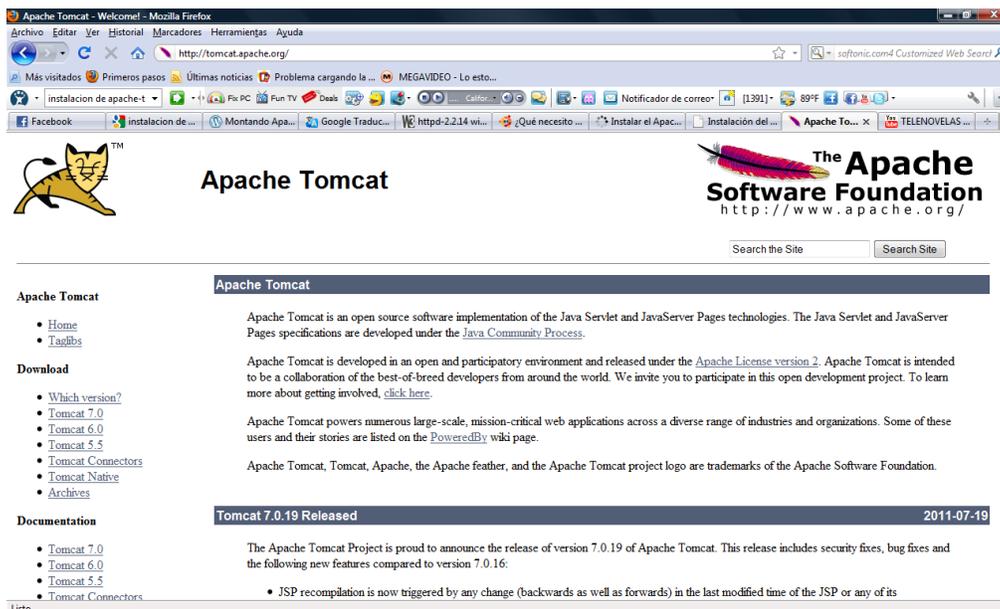
En el campo de texto "Valor de variable" debemos **escribir al final** del este contenido la **ruta al compilador** que comentamos anteriormente. Recordemos que debemos escribir al final de lo que hubiera utilizando como **separador el signo punto y coma ";"**. Como por ejemplo, en nuestro caso se añadió al final "; C:\Archivos de programa\Java\jdk1.6.0\_16\bin". No debemos utilizar espacios en la separación, sólo el punto y coma.

Por último hacemos clic en **"Aceptar"** para finalizar.



Ahora nos debe funcionar correctamente la llamada al compilador desde el "Símbolo del sistema" **sólo poniendo el comando "javac"**.

## B. INSTALACIÓN DEL CONTENEDOR APACHE TOMCAT 6.0.20



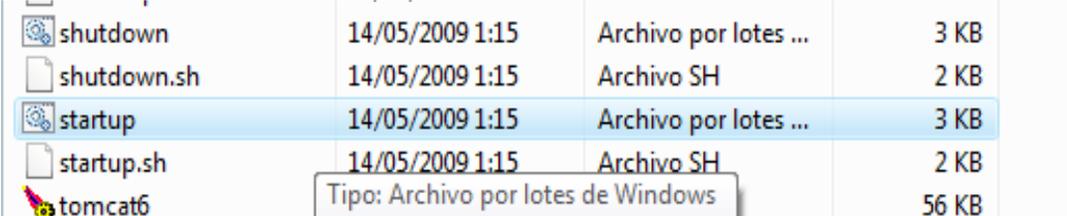
Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

El servidor Apache Tomcat es una implementación de referencia de las tecnologías Java Servlet y JavaServer Pages (JSP). Antes de comenzar la práctica de hoy, tendréis que instalar el servidor Tomcat en sus cuentas. Para ello debéis seguir los siguientes pasos:

- Descargamos el .zip apache-tomcat-6.0.20.zip desde el sitio web de Apache <http://tomcat.apache.org/download-60.cgi>
- Descomprimir el .zip que se descargó en el directorio donde deseamos tener Apache. En este caso estará en C:\Apache\apache-tomcat-6.0.20
- Luego creamos una variable de entorno CATALINA\_HOME. Para esto: damos clic derecho sobre “MiPC o Equipo” → Propiedades → Opciones Avanzadas → Variables de entorno y en “Variables del sistema” añadimos la variable CATALINA\_HOME con el siguiente valor: C:\Apache\apache-tomcat-6.0.20 (donde hayamos descomprimido la carpeta de apache-tomcat-6.0.20)

Comprobación de funcionamiento de Apache Tomcat:

- Ejecutar el comando C:\Apache\apache-tomcat-6.0.20\bin\startup.bat y si el firewall de Windows lo solicita, desbloquear este servicio.



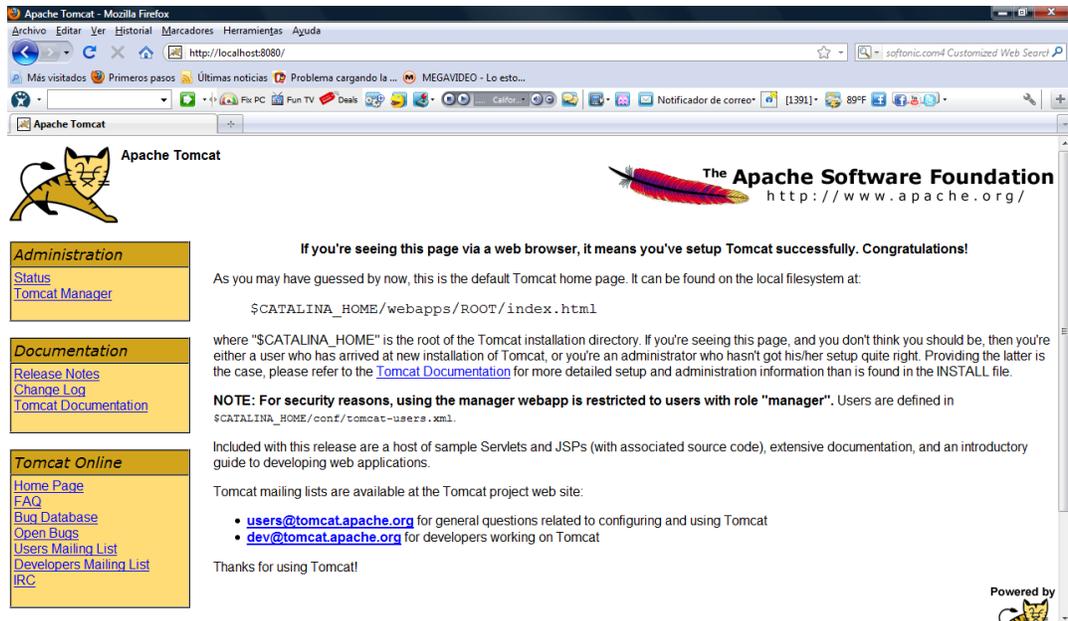
shutdown	14/05/2009 1:15	Archivo por lotes ...	3 KB
shutdown.sh	14/05/2009 1:15	Archivo SH	2 KB
startup	14/05/2009 1:15	Archivo por lotes ...	3 KB
startup.sh	14/05/2009 1:15	Archivo SH	2 KB
tomcat6		Tipo: Archivo por lotes de Windows	56 KB

- Al ejecutar nos aparecerá una pantalla como esta:

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

```
Tomcat
02/08/2011 11:51:30 PM org.apache.catalina.core.AprLifecycleListener init
INFO: La biblioteca nativa de Apache Tomcat basada en APR que permite un rendimiento
óptimo en entornos de desarrollo no ha sido hallada en java.library.path: C:\Program
Files\Java\jdk1.6.0_16\bin;. ;C:\Windows\Sun\Java\bin;C:\Windows\system32;C:\Windows;C:\Program
Files\Java\jdk1.6.0_16\bin;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\AppServ\Apache2.2\bin;C:\AppServ\php5;C:\AppServ\MySQL\bin
02/08/2011 11:51:30 PM org.apache.coyote.http11.Http11Protocol init
INFO: Inicializando Coyote HTTP/1.1 en puerto http-8080
02/08/2011 11:51:30 PM org.apache.catalina.startup.Catalina load
INFO: Initialization processed in 1520 ms
02/08/2011 11:51:30 PM org.apache.catalina.core.StandardService start
INFO: Arrancando servicio Catalina
02/08/2011 11:51:30 PM org.apache.catalina.core.StandardEngine start
INFO: Starting Servlet Engine: Apache Tomcat/6.0.20
02/08/2011 11:51:32 PM org.apache.coyote.http11.Http11Protocol start
INFO: Arrancando Coyote HTTP/1.1 en puerto http-8080
02/08/2011 11:51:32 PM org.apache.jk.common.ChannelSocket init
INFO: JK: ajp13 listening on /0.0.0.0:8009
02/08/2011 11:51:32 PM org.apache.jk.server.JkMain start
INFO: Jk running ID=0 time=0/70 config=null
02/08/2011 11:51:32 PM org.apache.catalina.startup.Catalina start
INFO: Server startup in 2048 ms
```

- Con el browser, visitar el sitio <http://localhost:8080>



**Nota:** Si el servidor no arranca, una de las causas posibles es que algunos de los puertos en que escucha conexiones por defecto (8080 y 8005) esté ocupado por otro proceso. En este caso, cambia estos puertos en el fichero de configuración. Para esto debemos entrar a la carpeta de Apache → conf → server.xml. Ponemos por ejemplo, 9090 y 9005. Recuerda que la nueva URI de acceso al servidor cambiaría en este caso a <http://localhost:9090>.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

logging.properties	14/05/2009 1:15	Archivo PROPERTI...	4 KB
server	14/05/2009 1:15	Archivo XML	7 KB

```
<!-- A "Service" is a collection of one or more "Connectors" that share
a single "Container" Note: A "Service" is not itself a "Container",
so you may not define subcomponents such as "Valves" at this level.
Documentation at /docs/config/service.html
-->
-->
- <Service name="Catalina">
<!-- The connectors can use a shared executor, you can define one or more named thread pools -->
<!--
<Executor name="tomcatThreadPool" namePrefix="catalina-exec-"
maxThreads="150" minSpareThreads="4"/>
-->
<!-- A "Connector" represents an endpoint by which requests are received
and responses are returned. Documentation at :
Java HTTP Connector: /docs/config/http.html (blocking & non-blocking)
Java AJP Connector: /docs/config/ajp.html
APR (HTTP/AJP) Connector: /docs/apr.html
Define a non-SSL HTTP/1.1 Connector on port 8080
-->
<Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000" redirectPort="8443" />
<!-- A "Connector" using the shared thread pool -->
<!--
<Connector executor="tomcatThreadPool"
```

**Nota:** Cuando hayas estado trabajando con Tomcat y vayas a cerrar tu sesión en el ordenador, recuerda comprobar que dejas Tomcat parado. Si no, supondrá una molestia para usuarios que utilicen el mismo ordenador más tarde.

### C. INSTALACIÓN DE ECLIPSE HELIOS

Para la instalación de Eclipse Galileo, nos dirigimos a la siguiente página en donde podremos descargar.

<http://www.eclipse.org/downloads/packages/release/helios/sr2>

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## Eclipse Helios Sr2 Packages



Eclipse IDE for Java EE Developers, (N/A)  
Downloaded 240,830 Times Details

Windows 32-bit 64-bit  
Mac Cocoa 32-bit 64-bit  
Linux 32-bit 64-bit

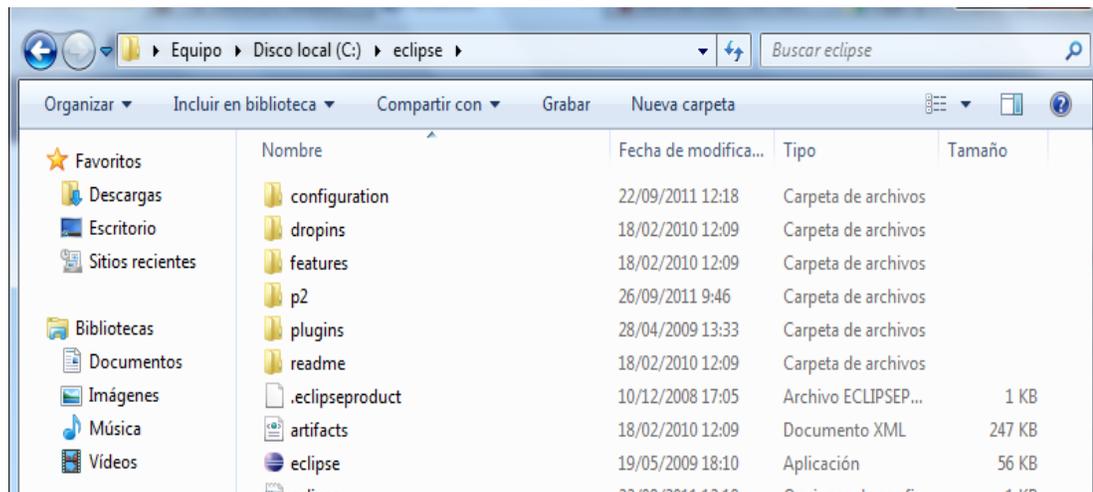
Para este caso seleccionamos “Eclipse IDE for Java EE Developers”, luego se nos abrirá una pantalla de descarga y hacemos clic en aceptar:



eclipse-jee-helios-SR2-win32.zip

Procedemos a descargar el archivo.

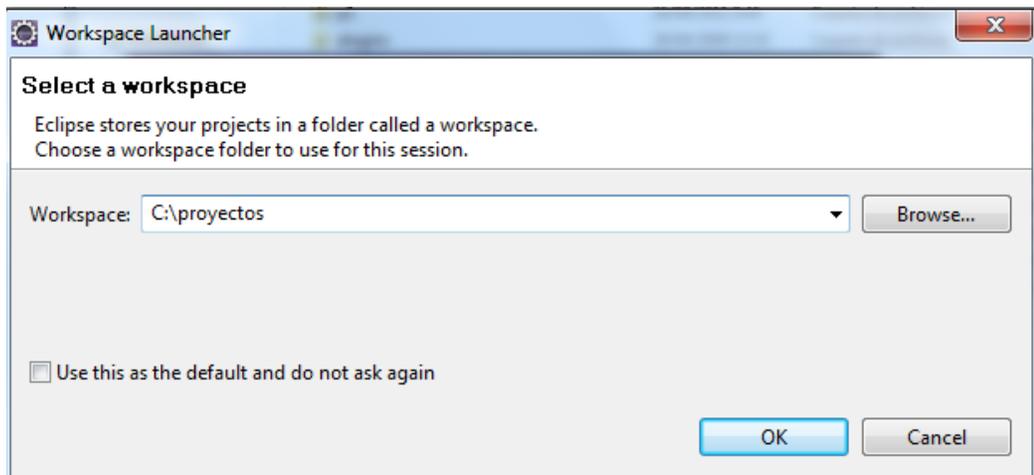
Una vez descargado el archivo, procedemos a descomprimirlo en el lugar que se desee, en este caso lo descomprimiremos en la unidad C:/



Lo único que queda es ejecutar el programa, dando doble clic en “eclipse”.

Nos pedirá una ruta en donde guardaremos los proyectos que realicemos, en este caso estará en una carpeta “proyectos” en la unidad C:/

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



## **D. MANUAL DE USUARIO SISTEMA DE GESTIÓN DE INFORMACIÓN**

### **D.1. INTRODUCCIÓN**

En este documento se describirá los objetivos e información clara y concisa de cómo utilizar el Sistema de Gestión de Información de la empresa La Viandé y su funcionamiento. El Sistema de Gestión de Información fue creado con el objetivo de brindar facilidades al usuario del sistema para realizar sus procesos cotidianos de manera sistematizada, realizar proformas, facturaciones, tener un control de productos, generación de menus, manejo de clientes así como también la administración de la página web de la empresa. Es de mucha importancia consultar este manual antes y/o durante la visualización de las páginas, ya que lo guiará paso a paso en el manejo de las funciones en él. Con el fin de facilitar la comprensión del manual, se incluye gráficos explicativos.

### **D.2. OBJETIVO DE ESTE MANUAL**

El objetivo primordial de éste Manual es ayudar y guiar al usuario a utilizar el Sistema de Gestión de Información y comprende:

- Guía para acceder al Sistema de Gestión de Información para la empresa La Viandé.

- Conocer como utilizar el sistema, mediante una descripción detallada e ilustrada de las opciones.
- Conocer el alcance de toda la información por medio de una explicación detallada e ilustrada de cada una de las páginas que lo conforman.

### **D.3. DIRIGIDO A**

Este manual está orientado a una parte de los Usuarios Finales involucrados en la etapa de Operación del Sistema de Gestión de la Información, quienes interactuarán con el sistema.

### **D.4. LO QUE DEBE CONOCER**

Los conocimientos mínimos que deben tener las personas que operarán el sistema y deberán utilizar este manual son:

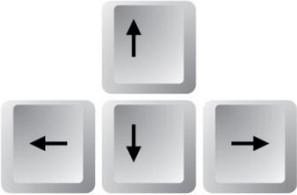
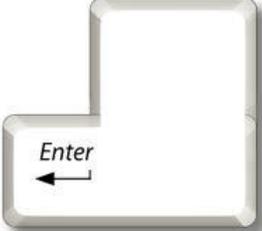
- Conocimientos básicos de Navegación en Web.
- Conocimiento básico de Internet.
- Conocimiento básico de Windows.

### **D.5. CONVENCIONES Y ESTÁNDARES A UTILIZAR**

#### **5.1. CONVENCIONES DEL USO DEL MOUSE**

<b>TÉRMINO</b>	<b>SIGNIFICADO</b>
Señalar	Colocar el extremo superior del Mouse sobre el elemento que se desea señalar.
Hacer Clic	Presionar el botón principal del Mouse (generalmente el botón izquierdo) y soltarlo inmediatamente

## 5.2. CONVENCIONES DEL USO DEL TECLADO

TECLA	SIGNIFICADO
	Puede utilizar las teclas direccionales para moverse (Izquierda, Derecha, Arriba, Abajo) en los diferentes elementos de un control de la pantalla, un elemento a la vez.
	Tecla utilizada para ejecutar un proceso. Si en el manual dice "Presionar Enter".

### D.6. INGRESO AL SISTEMA

Para ingresar al Sistema debemos tener en cuenta algunos aspectos que a continuación se detallan.

#### 6.1. PASOS PARA ENCENDER LA COMPUTADORA

- Encienda el C.P.U. presionando el botón Power.
- Encienda el Monitor presionando el botón Power.

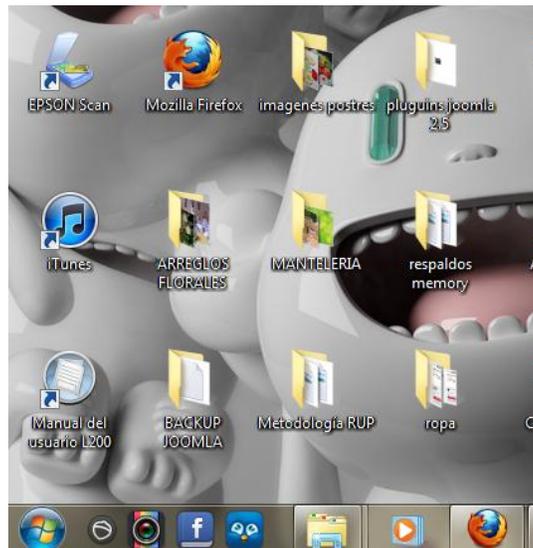
Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



- Espere mientras carga el Sistema Operativo.
- Automáticamente aparecerá la pantalla de Windows. La pantalla de Windows puede ser de varios tipos o diseños.

## 6.2. COMO ACCEDER AL SISTEMA DE GESTIÓN DE LA INFORMACIÓN.

- Ubíquese en el ícono de Mozilla Firefox  y haga doble click.



- A continuación aparecerá la pantalla de Mozilla Firefox, ubíquese en la barra de direcciones y escriba lo siguiente:  
<http://localhost:8080/sistemaViande/>.

## D.7. SISTEMA DE GESTIÓN DE INFORMACIÓN

### 7.1. LOGIN

A continuación aparecerá la pantalla de login del Sistema de Gestión de Información. En donde debemos ingresar nuestro usuario con su respectiva clave de usuario.

**SISTEMA VIANDE**

Usuario:

Clave:

Una vez ingresado al sistema podemos ver nuestra barra de módulos que conforman el sistema, las cuales son:

- Archivo.
- Administración.
- Proformas.
- Facturación.
- Cuentas por cobrar.

Archivo

Administración

Proformas

Facturación

Cuentas por Cobrar

## 7.2. ARCHIVO

Si damos clic en la pestaña ARCHIVO, nos mostrará la opción para salir del sistema.

## 7.3. ADMINISTRACIÓN

A continuación nos vamos al módulo de ADMINISTRACIÓN que consta de:

- Usuarios.
- Grupos Productos.
- Productos.
- Platos.
- Menús.
- Ajuste de Invetario.
- Rep. De Invetario.

### 7.3.1. USUARIOS

Ahora damos clic en la opción USUARIOS:



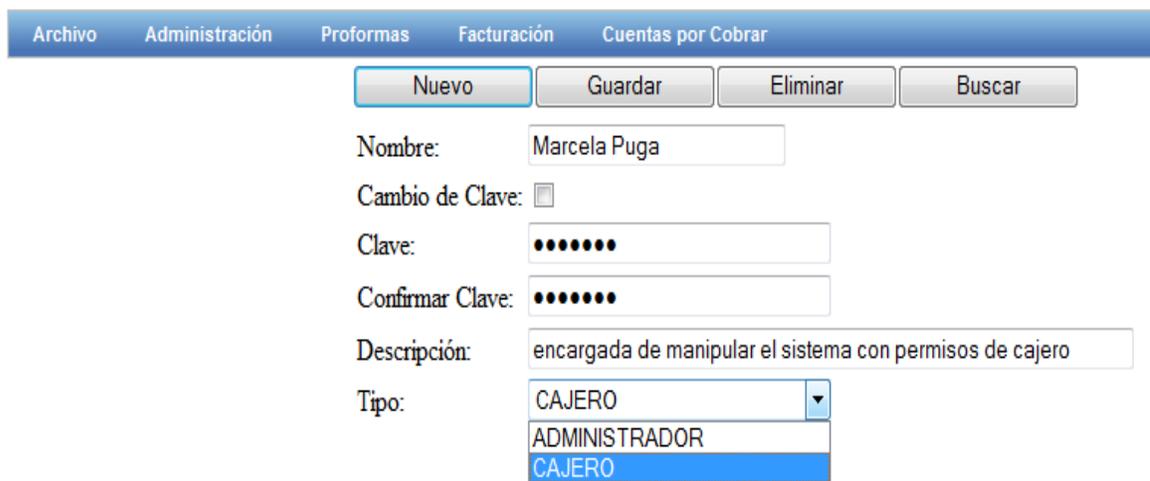
Aquí nos permitirá realizar operaciones como:

- Crear un nuevo usuario.
- Guardar un usuario.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- Eliminar un usuario.
- Buscar un usuario.

Presionamos en el botón NUEVO para crear un nuevo usuario, y llenamos los datos q nos muestra. Ingresamos un nombre, clave, descripción, tipo que puede ser de dos tipos: cajero o administrador. Una vez ingresado estos datos presionamos en el botón GUARDAR.



Archivo Administración Proformas Facturación Cuentas por Cobrar

Nuevo Guardar Eliminar Buscar

Nombre: Marcela Puga

Cambio de Clave:

Clave: ●●●●●●

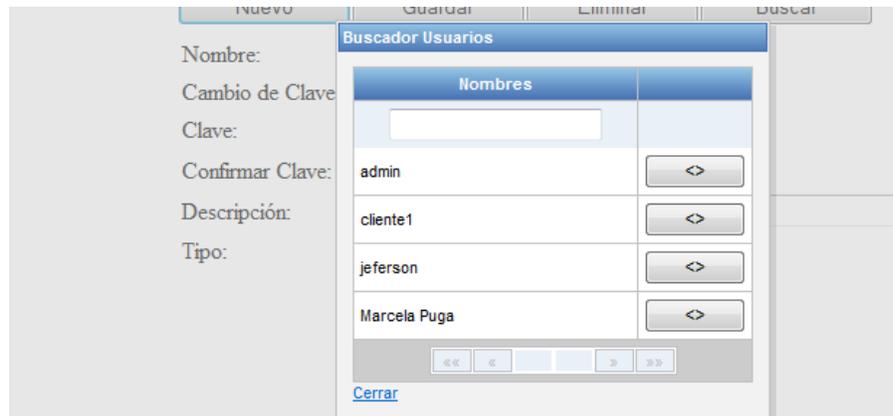
Confirmar Clave: ●●●●●●

Descripción: encargada de manipular el sistema con permisos de cajero

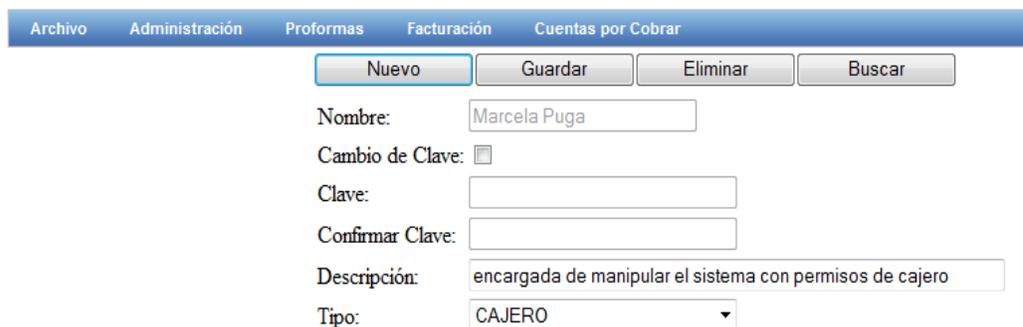
Tipo: CAJERO  
ADMINISTRADOR  
CAJERO

Presionando el botón BUSCAR podemos buscar todos los usuarios existentes en el sistema como se muestra en la imagen, este buscador nos permite buscar a nuestros usuarios del sistema por su nombre, para que sea mas fácil su búsqueda. Para salir del buscador damos clic en la opción CERRAR.

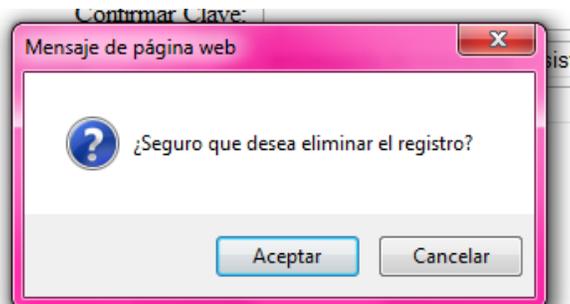
Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



En este caso damos clic en el botón <> de MARCELA PUGA que creamos anteriormente para poder editar su clave o eliminar este usuario, en este ejemplo vamos a eliminar presionando el botón ELIMINAR.



Nos mostrará una pequeña ventana, indicándonos si queremos o no eliminar este usuario. Para ello presionamos el botón ACEPTAR.



### 7.3.2. GRUPOS PRODUCTOS

Ahora nos dirigimos al menu principal y seleccionamos GRUPOS PRODUCTOS y hacemos clic.



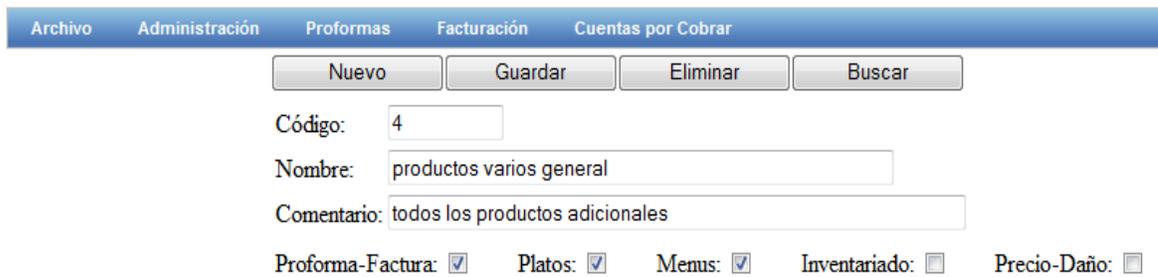
Como podemos observar nos muestra una pantalla correspondiente a grupos de productos con las opciones de: nuevo, guardar, eliminar y buscar.



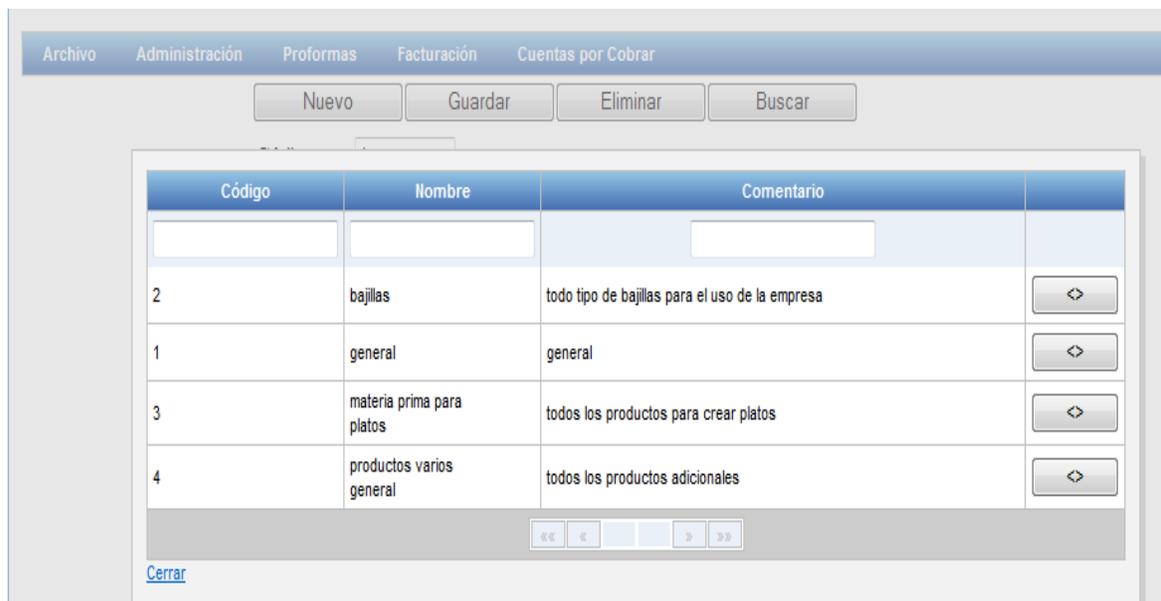
Hacemos clic en el botón NUEVO para crear un nuevo grupo de productos, nos mostrará los campos correspondientes a un grupo de productos. Debemos ingresar información como: código de grupo de producto, nombre del grupo de producto, un comentario y finalmente seleccionar los checkbox que dan las características al grupo de productos entre ellas estan:

- Es parte de una proforma o factura.
- Es parte de un plato.
- Es parte de menús.
- Es parte del inventario.
- Si el grupo de productos tiene un precio por daño.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



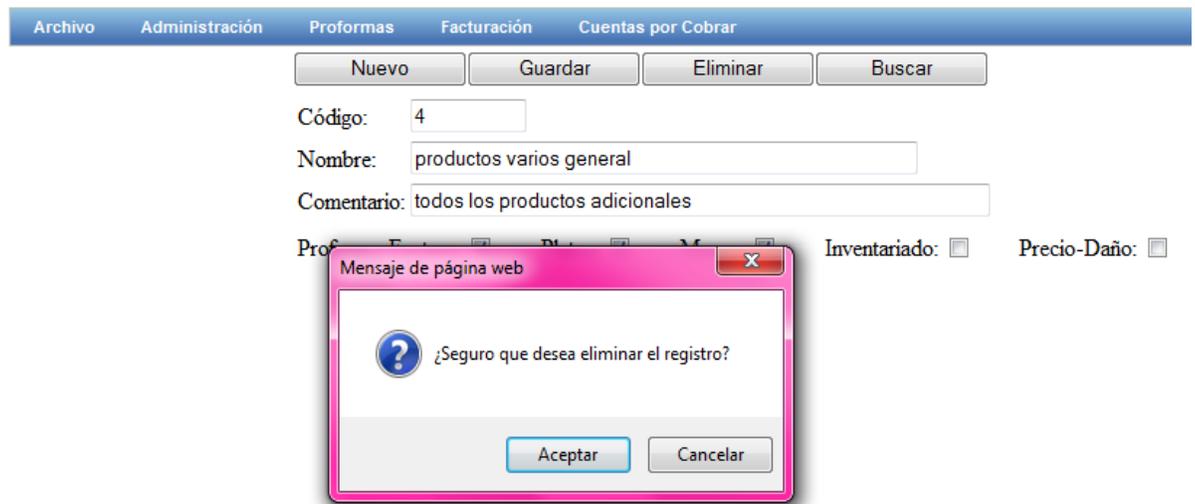
Hacemos clic en el botón GUARDAR y para comprobar que se guardaron los datos hacemos clic en el botón BUSCAR, nos muestra nuestros grupos de productos con varios tipos de buscadores: por código, por nombre y por comentario. Si queremos editar algún campo simplemente damos clic en el botón <> y si no deseamos hacer nada podemos dar clic en CERRAR.



Código	Nombre	Comentario	
2	bajillas	todo tipo de bajillas para el uso de la empresa	<>
1	general	general	<>
3	materia prima para platos	todos los productos para crear platos	<>
4	productos varios general	todos los productos adicionales	<>

En el caso que queramos eliminar un grupo de productos, hacemos clic en el botón ELIMINAR, en donde nos aparecerá una pequeña ventana de confirmación. Le damos clic en el botón ACEPTAR para realizar la eliminación.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

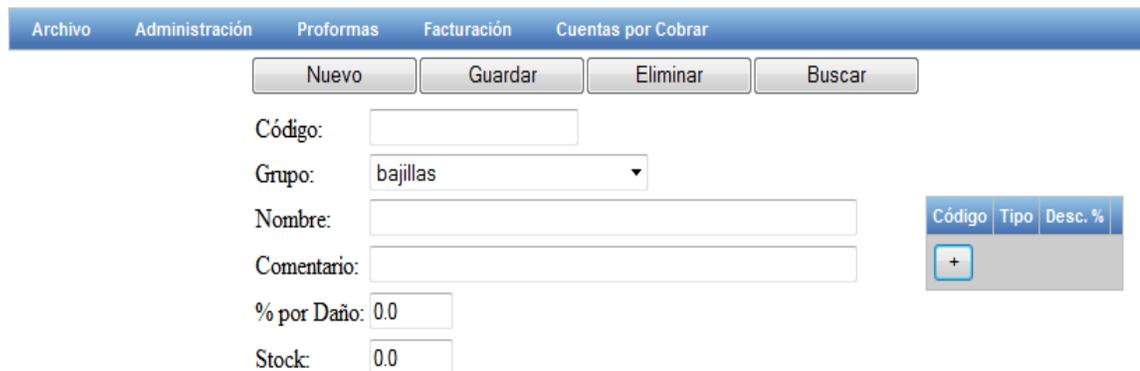


### 7.3.3. PRODUCTOS

Ahora nos dirigimos a la opción de PRODUCTOS dando clic en ella.



Nos mostrará una pantalla referente a productos, en donde daremos clic en el botón NUEVO para crear un nuevo producto.



Llenamos los campos correspondientes al producto, estos campos son: código del producto, grupo del producto en donde se

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

desplegará todos los tipos de productos existentes, nombre del producto, comentario, % por daño y el stock. Una vez ingresado estos datos presionamos el botón GUARDAR.

Código	Tipo	Desc. %	
20	A	8.8	-
3	COSTO	8.0	-

Como podemos ver existe una pequeña tabla, la cual corresponde a dos precios de cada producto:

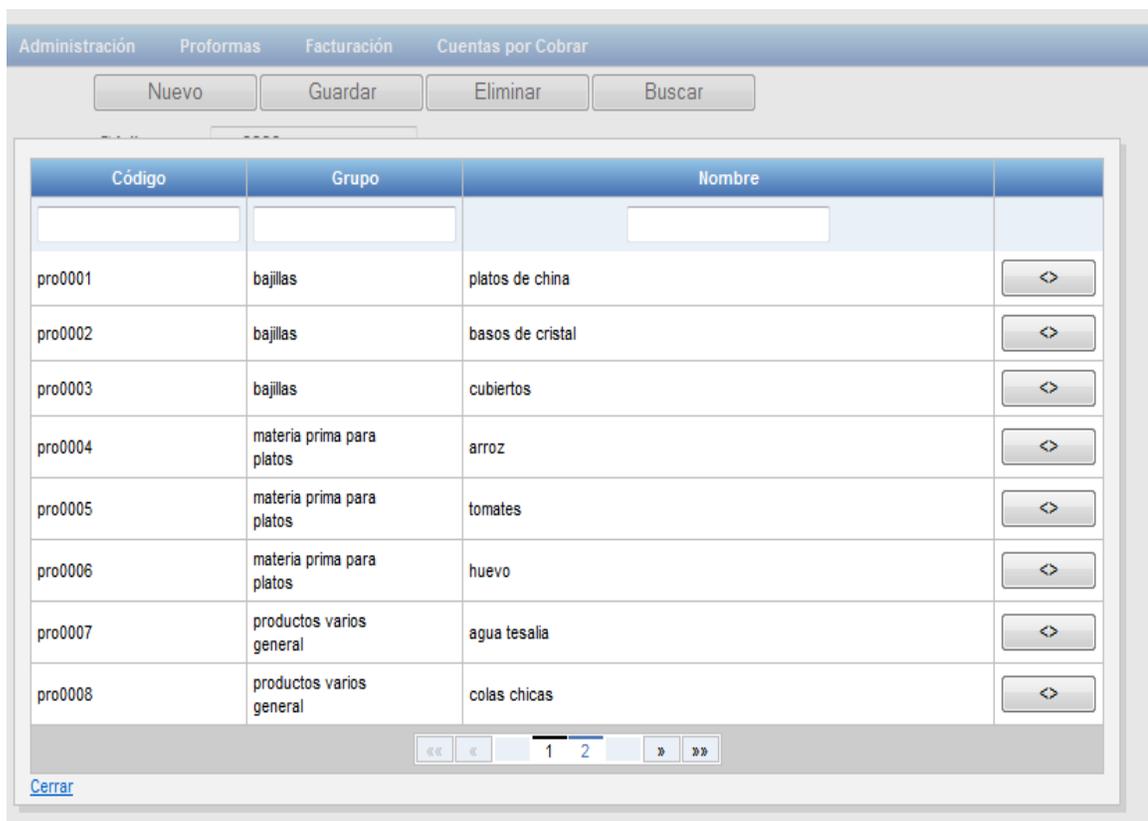
- Tipo A: que es el precio añadido el porcentaje de daño.
- Tipo COSTO: es el costo de un producto.

Código	Tipo	Desc. %	
20	A	8.8	-
3	COSTO	8.0	-

Para verificar que se haya guardado, presionamos el botón BUSCAR. Nos listará todos los productos existente. Podemos realizar la búsqueda de los productos por medio de tres buscadores que facilitarán este proceso.

Estos buscadores son:

- Buscador por Código.
- Buscador por Grupo.
- Buscador por Nombre.



Si presionamos el botón <> podemos editar ese producto, en este caso damos clic en el link CERRAR. Ahora damos clic en el botón eliminar para eliminar el producto que creamos anteriormente. Nos aparecerá una ventana de verificación, finalmente le damos clic al botón aceptar.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

The screenshot shows a web application interface with a blue header bar containing menu items: Archivo, Administración, Proformas, Facturación, and Cuentas por Cobrar. Below the header is a toolbar with buttons for Nuevo, Guardar, Eliminar, and Buscar. The main form contains the following fields:

- Código: pro0002
- Grupo: bajillas
- Nombre: basos de cristal
- Comentario: basos de todo tipo
- % por I: (empty)
- Stock: (empty)

To the right of the form is a table with the following data:

Código	Tipo	Desc. %	
20	A	8.8	-
3	COSTO	8.0	-
+			

A modal dialog box titled "Mensaje de página web" is overlaid on the form, asking: "¿Seguro que desea eliminar el registro?". It has "Aceptar" and "Cancelar" buttons.

#### 7.3.4. PLATOS

Ahora seleccionamos la opción PLATOS.

The screenshot shows the web application interface with the 'Administración' menu expanded. The menu items are: Usuarios, Grupos Productos, Productos, Platos (highlighted in orange), Menus, and Ajuste de Inventario.

Creamos un nuevo plato dando clic en el botón NUEVO, llenamos los campos que se muestran a continuación como: código de plato, nombre, comentario, porcentaje de utilidad y costo de plato.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

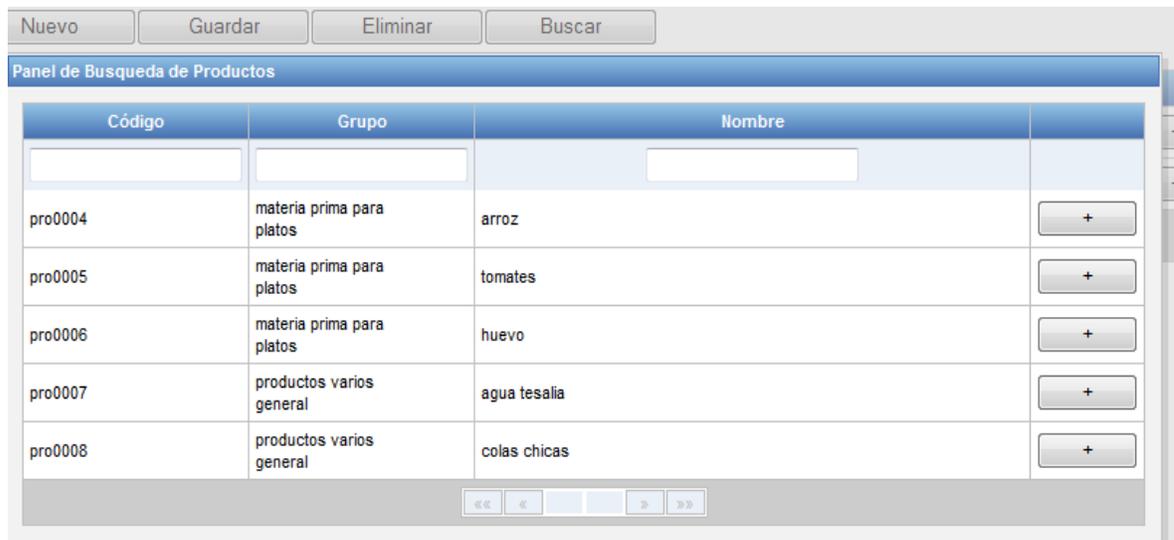
A nuestra derecha tenemos los valores de máximo y mínimo que corresponden a valores de platos en cantidad, pasado los 50 platos el cliente tiene un descuento.

Minimo	Maximo	Valor	
0.0	49.0	0.0	-
50.0	100.0	3.0	-
+			

Al precionar el botón “+” podemos agregar un nuevo producto al plato. Si presionamos el botón “-“ podemos eliminarlo.

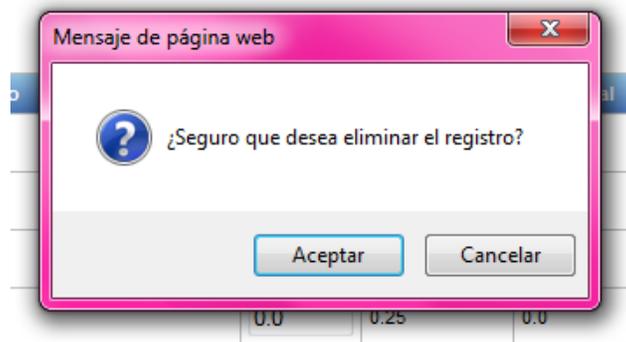
Esta es la pantalla que nos mostrará al precionar el botón .

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



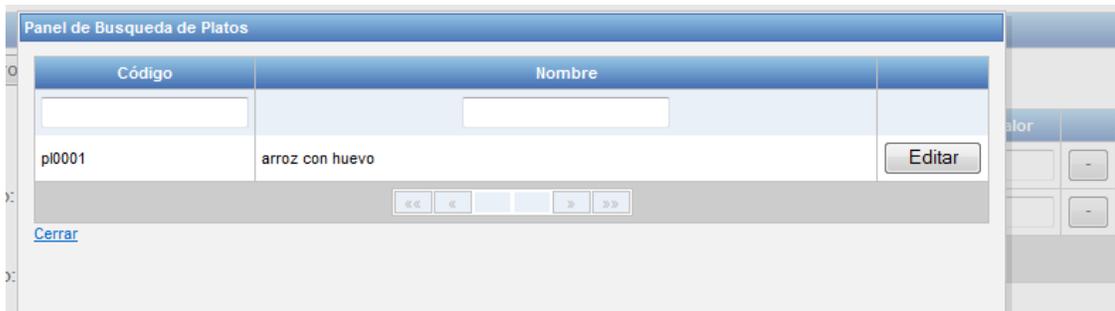
Para guardar damos clic en el botón GUARDAR. Y para ver los platos guardados precionamos en el botón BUSCAR, que consta de tres buscadores para una búsqueda eficiente. Estos buscadores son: por código, por grupo y por nombre.

Para eliminar un plato simplemente nos dirigimos al botón ELIMINAR dando un clic sobre el, nos mostrara una pequeña ventana en el que nos preguntará si deseamos eliminar ese plato. Damos clic en ACEPTAR para realizar la eliminación del plato.



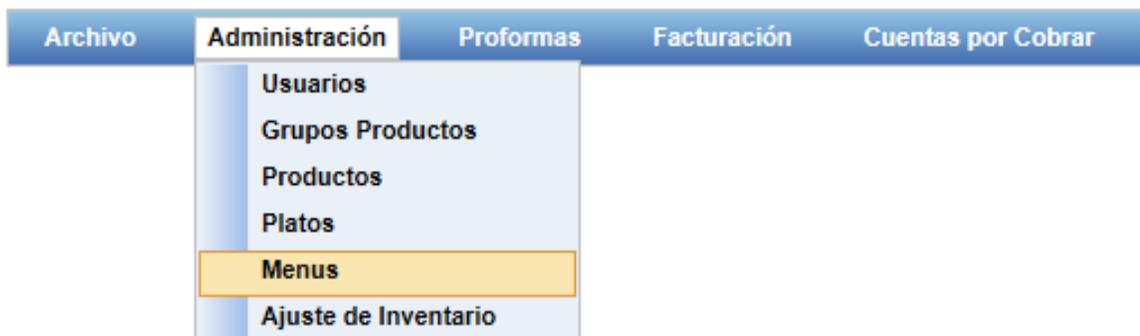
Comprobamos que se haya eliminado el plato. Y damos clic en CERRAR para salir.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

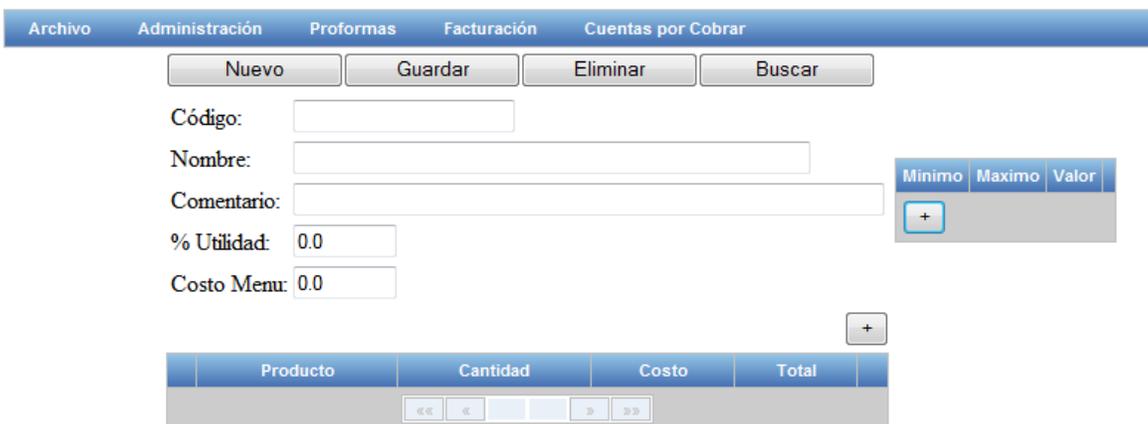


### 7.3.5. MENÚS

Damos clic en la opción MENÚS.



A continuación nos mostrará una pantalla con los campos correspondientes a un menú. Estos campos son: código del menú, nombre, comentario, utilidad y costo menú. Damos clic en el botón NUEVO para crear un nuevo menú.



Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Ingresamos los datos de cana uno de los campos.

Archivo Administración Proformas Facturación Cuentas por Cobrar

Nuevo Guardar Eliminar Buscar

Código: me0001

Nombre: menu combo 1

Comentario: menu muy simple

% Utilidad: 20.0

Costo Menu: 3.44

Minimo	Maximo	Valor	
0.0	49.0	0.0	-
50.0	99.0	2.0	-
+			

	Producto	Cantidad	Costo	Total	
+	arroz con huevo	2.0	1.32	2.64	-
+	colas chicas	2.0	0.4	0.8	-

En la parte derecha podemos observar una tabla que contiene valores máximos y mínimos, correspondientes a la cantidad de menús que el cliente desea. Lo que permite esto es tener un control del número de menús mediante un rango, de acuerdo al mismo nos proporcionará sus respectivos descuentos.

Minimo	Maximo	Valor	
0.0	49.0	0.0	-
50.0	99.0	2.0	-
+			

Si presionamos el botón “+” podemos agregar platos al menú.

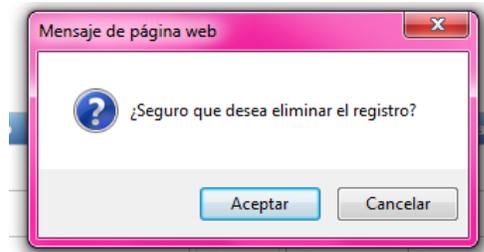
	Producto	Cantidad	Costo	Total	
+	arroz con huevo	2.0	1.32	2.64	-
+	colas chicas	2.0	0.4	0.8	-

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Esta es la ventana que nos muestra al agregar un nuevo plato al menú, para añadir un nuevo plato damos clic en el botón “+”.



Finalmente guardamos este menú dando clic en el botón GUARDAR. Nos mostrará una ventana preguntándonos si deseamos eliminar ese menú. Damos clic en ACEPTAR.



Para ver que se haya creado el menú damos clic en el botón BUSCAR, el cual nos desplegará la siguiente ventana. Podemos buscar menús por código o nombre.



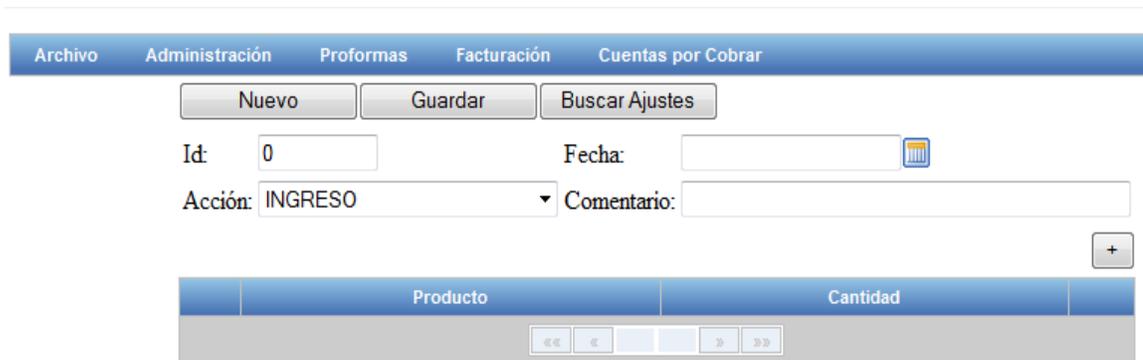
### 7.3.6. AJUSTE DE INVENTARIO

Damos clic en la opción AJUSTE DE INVENTARIO.



Nos desplegará una ventana en la que podremos crear un nuevo ajuste de inventario, guardar y buscar ajustes. Para ello le damos clic en el botón NUEVO y llenamos los campos que se muestran ahí.

Al precionar el botón “+” podemos agregar los productos que queremos realizar el ajuste de inventario.

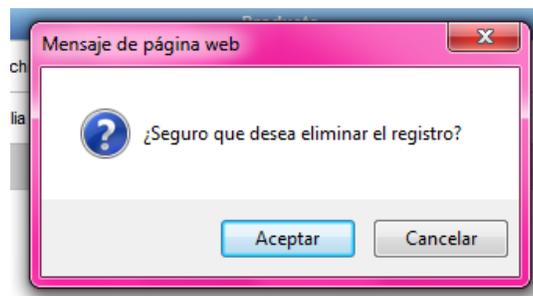
A screenshot of the 'AJUSTE DE INVENTARIO' form. At the top is a navigation bar with 'Archivo', 'Administración', 'Proformas', 'Facturación', and 'Cuentas por Cobrar'. Below the bar are three buttons: 'Nuevo', 'Guardar', and 'Buscar Ajustes'. The form contains several input fields: 'Id:' with the value '0', 'Fecha:' with a calendar icon, 'Acción:' with a dropdown menu showing 'INGRESO', and 'Comentario:' with a text input field. A small '+' button is located to the right of the 'Comentario:' field. Below the form is a table with two columns: 'Producto' and 'Cantidad'. The table has a header row and a body row. At the bottom of the table are navigation buttons: '<<', '<', '>', and '>>'.

Esta es la pantalla que se mostrará al precionar el botón “+”. Para agregar productos al ajuste de inventario solo precionamos los botones “+” del producto que se desee añadir. Contiene tres buscadores: por código, por grupo y por nombre.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Código	Grupo	Nombre	
me0001	general	menu combo 1	+
pl0001	general	arroz con huevo	+
pro0001	bajillas	platos de china	+

Luego damos clic en el botón GUARDAR. Para eliminar un ajuste simplemente le damos clic en el botón “-“, nos aparecerá un mensaje de verificación. Damos clic en ACEPTAR.



Para ver los cambios damos clic en el botón BUSCAR AJUSTES, donde nos mostrará todos los ajustes existentes, podemos buscar por tipo, fecha y por comentario. Para salir damos clic en CERRAR.

Tipo	Fecha	Comentario	
INGRESO	2012-09-10	ajuste de inventario	Editar
INGRESO	2012-09-15	ajuste de productos varios	Editar

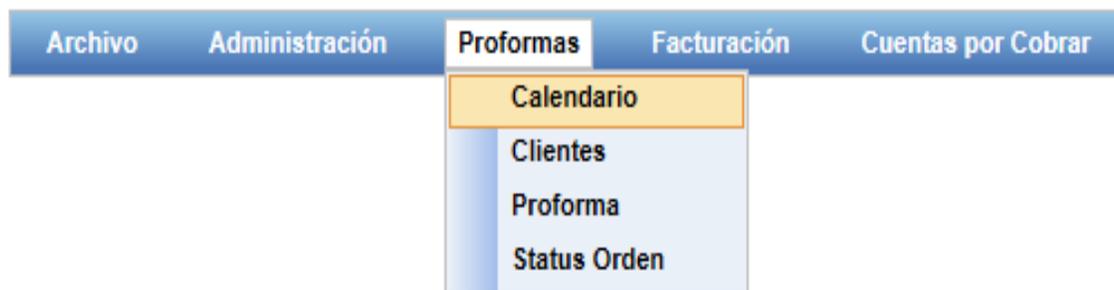
## 7.4. PROFORMAS

El módulo de proformas consta de lo siguiente:

- Calendario.
- Clientes.
- Proforma.
- Status orden.

### 7.4.1. CALENDARIO

Damos clic en CALENDARIO.



Nos mostrará las fechas de los eventos con nuestros clientes, para saber que eventos se realizaron y cuales están por realizarse, nos permite buscar por mes y año.



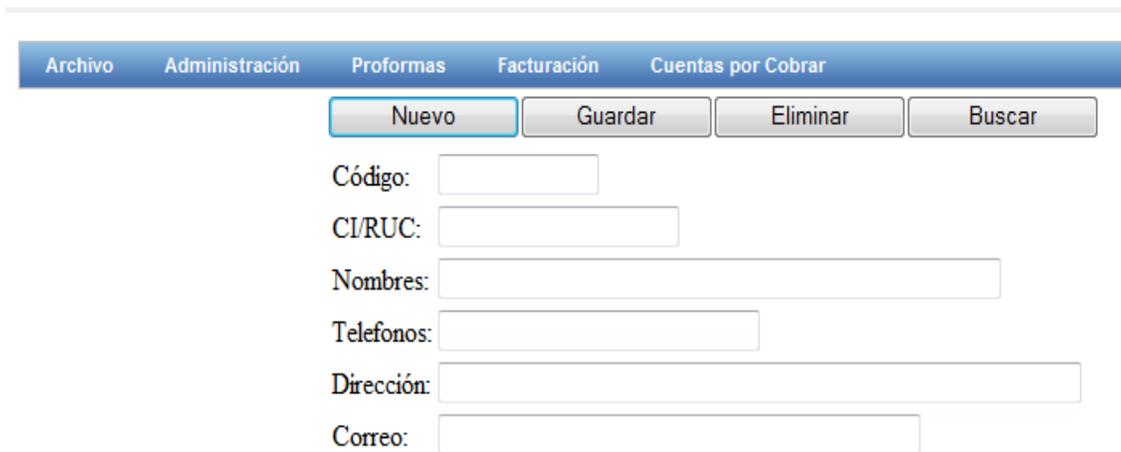
Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### 7.4.2. CLIENTES

Hacemos clic en CLIENTES.



Nos mostrará todo lo referente al cliente, hacemos clic en NUEVO.

A screenshot of the 'Nuevo' form in the web application. The form is located below the navigation menu. It has a header with the same navigation tabs as the previous screenshot. Below the tabs are four buttons: 'Nuevo', 'Guardar', 'Eliminar', and 'Buscar'. The 'Nuevo' button is highlighted. Below the buttons are several input fields with labels: 'Código:', 'CI/RUC:', 'Nombres:', 'Telefonos:', 'Dirección:', and 'Correo:'. Each field is empty.

para crear un nuevo cliente, debemos ingresar datos como código, CI o RUC, nombres, teléfonos, dirección y correo.

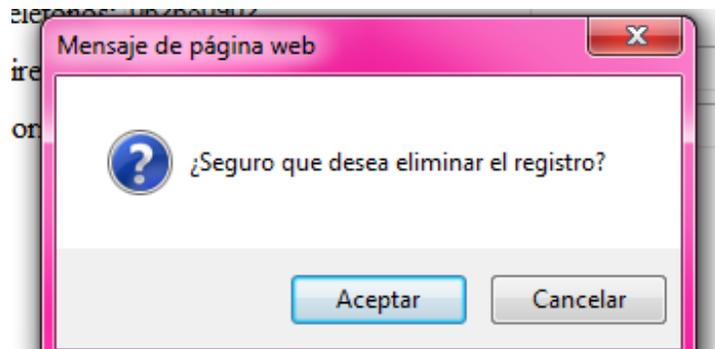
A screenshot of the 'Nuevo' form in the web application, showing data entered into the input fields. The form is located below the navigation menu. It has a header with the same navigation tabs as the previous screenshot. Below the tabs are four buttons: 'Nuevo', 'Guardar', 'Eliminar', and 'Buscar'. The 'Nuevo' button is highlighted. Below the buttons are several input fields with labels: 'Código:', 'CI/RUC:', 'Nombres:', 'Telefonos:', 'Dirección:', and 'Correo:'. The fields contain the following data: 'Código: ci0001', 'CI/RUC: 1007890125', 'Nombres: Salomé Castro', 'Telefonos: 062680902', 'Dirección: caranqui, general pintac y rumiñahui esquina', and 'Correo: salo@hotmail.com'.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Una vez ingresado los datos del cliente, damos clic en GUARDAR, para ver que se haya guardado damos clic en BUSCAR. Nos mostrara todos los clientes existentes, nos permite realizar búsquedas por código, cédula y por nombre. Si damos clic en el botón <> podemos editarlo, caso contrario damos clic en CERRAR para salir.

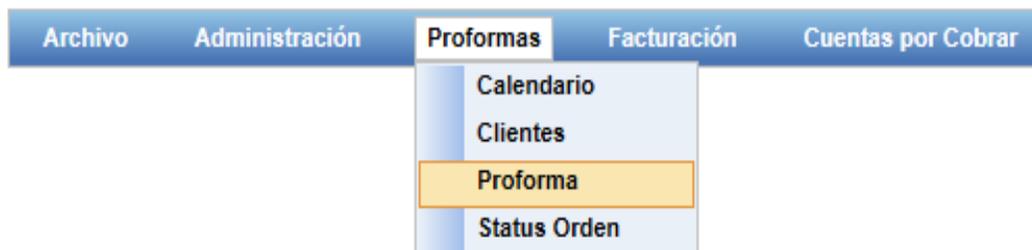


Para eliminar un cliente simplemente hacemos clic en ELIMINAR, nos preguntará si deseamos eliminar este cliente, en este caso hacemos clic en ACEPTAR.



### 7.4.3. PROFORMA

A continuación hacemos clic en PROFORMA.



Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Al hacer clic en PROFORMA nos muestra la siguiente pantalla.

Producto	Cantidad	T. Precio	Precio	Descuento %	Total
----------	----------	-----------	--------	-------------	-------

Esta pantalla contiene los datos necesarios para realizar una proforma, como: id de la proforma, numero de proforma, fecha, cliente, forma de pago, comentario.

Para crear una proforma hacemos clic en NUEVO.

Producto	Cantidad	T. Precio	Precio	Descuento %	Total
menu combo 1	50.0	A	1.85	0.0	92.5
agua tesalia	10.0	A	0.3	0.0	3.0

Para agregar un cliente simplemente hacemos clic en el botón "?". Para seleccionar el cliente que queremos para la proforma damos clic en el botón "+".

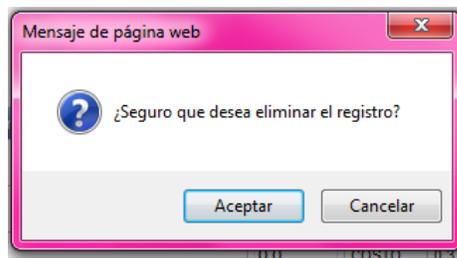
Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.



En la tabla correspondiente al detalle de la proforma necesitamos ir añadiendo menús a la proforma. Para ellos nos dirigimos a la parte derecha de la tabla y damos clic en el botón “+” para ir añadiendo.



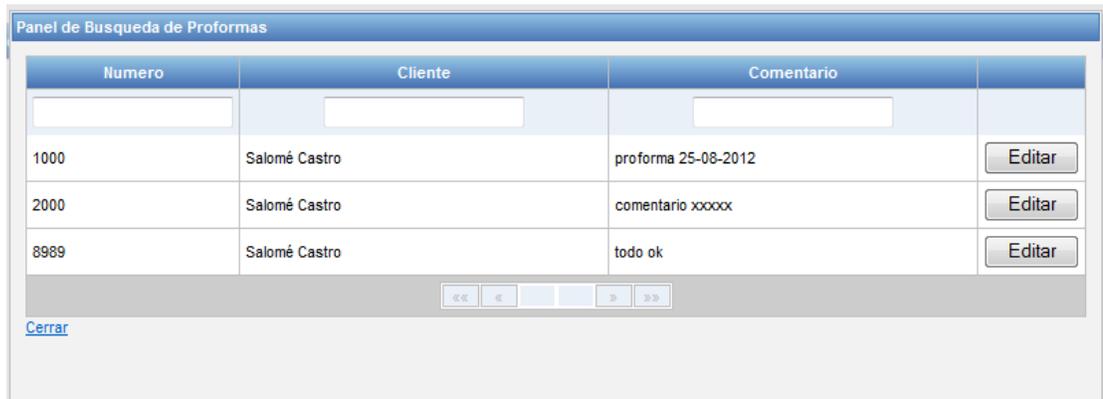
Una vez añadido los menús de la proforma hacemos clic en el botón GUARDAR. Si queremos eliminar la proforma simplemente hacemos clic en el botón ELIMINAR, donde nos preguntará si queremos eliminar, en este caso hacemos clic en ACEPTAR.



Para ver los cambios y lo que hemos guardado hacemos clic en el botón BUSCAR, donde nos mostrará todas las proformas realizadas, podemos buscar por numero, por cliente y por comentario.

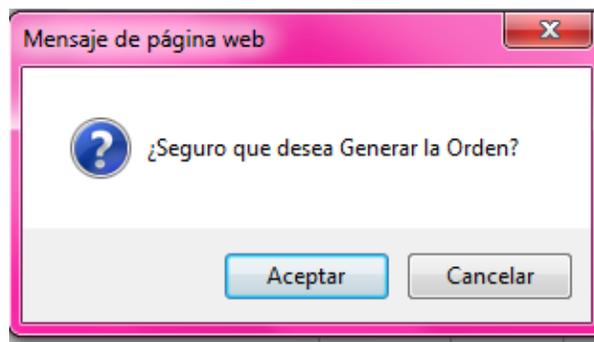
Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Para editar solo hacemos clic en el botón EDITAR.

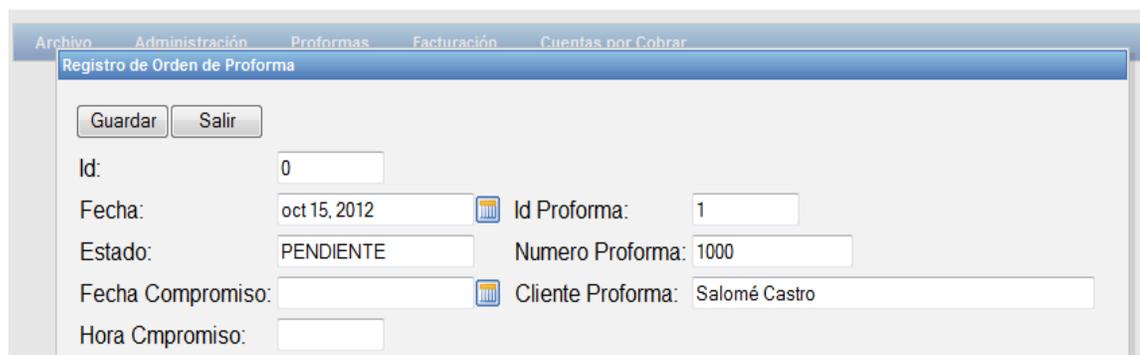


Numero	Cliente	Comentario	
1000	Salomé Castro	proforma 25-08-2012	Editar
2000	Salomé Castro	comentario xxxxx	Editar
8989	Salomé Castro	todo ok	Editar

Para generar la proforma simplemente hacemos clic en GENERAR ORDEN, donde nos preguntará si deseamos generarla. En este caso hacemos clic en ACEPTAR.



Solo damos clic en GUARDAR para que la proforma sea generada como se muestra a continuación.



Archivo Administración Proformas Facturación Cuentas por Cobrar

Registro de Orden de Proforma

Guardar Salir

Id: 0

Fecha: oct 15, 2012 Id Proforma: 1

Estado: PENDIENTE Numero Proforma: 1000

Fecha Compromiso: Cliente Proforma: Salomé Castro

Hora Compromiso:

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Registro de Orden de Proforma

La proforma N° 1000 ya esta generada una orden.

Guardar Salir

Id: 0

Fecha: oct 15, 2012

Estado: PENDIENTE

Fecha Compromiso:

Hora Cmpromiso:

Id Proforma: 1

Numero Proforma: 1000

Cliente Proforma: Salomé Castro

#### 7.4.4. STATUS ORDEN

Hacemos clic en la opción de STATUS ORDEN.



Lo que nos muestra es una lista de órdenes, para saber si estan pendientes o no. Podemos realizar búsquedas por Id de irden, por número de proforma, por cliente, por fehca o por status de orden.

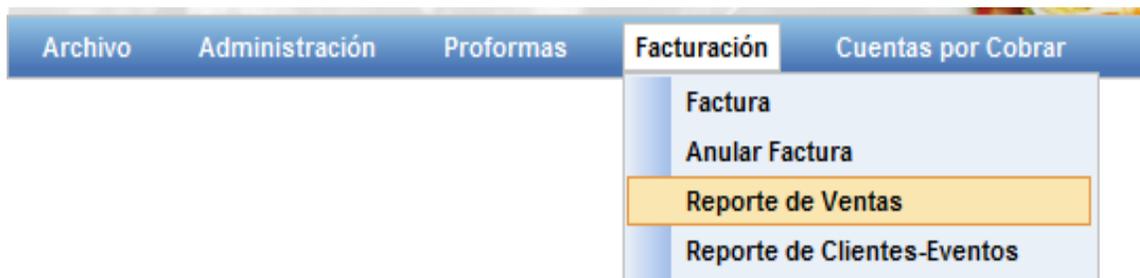
id Orden	# Proforma	Cliente	Fecha	Status Orden
1	N° 1000 (proforma 25-08-2012)	Salomé Castro	2012-08-25	PENDIENTE
2	N° 8989 (todo ok)	Salomé Castro	2012-09-13	FACTURADO

<< < > >>

## 7.5. FACTURACIÓN

Este módulo de facturación consta de lo siguiente:

- Factura.
- Anular factura.
- Reporte de Ventas.
- Resporte de Clientes-Eventos.



### 7.5.1. FACTURA

Hacemos clic en FACTURA. Y procedemos a crear una nueva factura haciendo clic en NUEVO.

The image shows the 'Factura' form in the application. The form is displayed in a window with a blue header bar containing the menu items: 'Archivo', 'Administración', 'Proformas', 'Facturación', and 'Cuentas por Cobrar'. Below the header bar, there are four buttons: 'Nuevo', 'Guardar', 'Buscar Orden', and 'Buscar Facturas'. The form fields are arranged in a grid-like structure. On the left side, there are fields for 'Id:' (value: 0), 'Numero:', 'Fecha:', 'Dias Cred.' (value: 0), and 'Orden:' (value: 0). On the right side, there are fields for 'Cliente:', 'Forma Pago:' (value: CONTADO), 'Comentario:', and 'Subtotal:' (value: 0.0). Below these fields, there are summary fields: 'Desc. \$:' (value: 0.0), 'Subtotal D.' (value: 0.0), 'IVA:' (value: 0.0), 'Servicios:' (value: 0.0), and 'Total:' (value: 0.0). At the bottom of the form, there is a table with columns: 'Producto', 'Cantidad', 'Tip Pr', 'Precio', 'Desc.', and 'Total'. The table is currently empty. Below the table, there are navigation buttons: '<<', '<', '>', and '>>'. There is also a '+' button at the bottom right of the form.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Podemos buscar por facturas, precionando el botón BUSCAR FACTURA.

Numero	Cliente	Comentario
777	Salomé Castro	Factura de proforma N° 8989

Podemos buscar por órdenes, las cuales son órdenes pendientes que deben pasar a ser facturadas. Precionando el botón BUSCAR ORDEN.

Número Prof.	Fecha	Estado
1000	2012-08-25	PENDIENTE

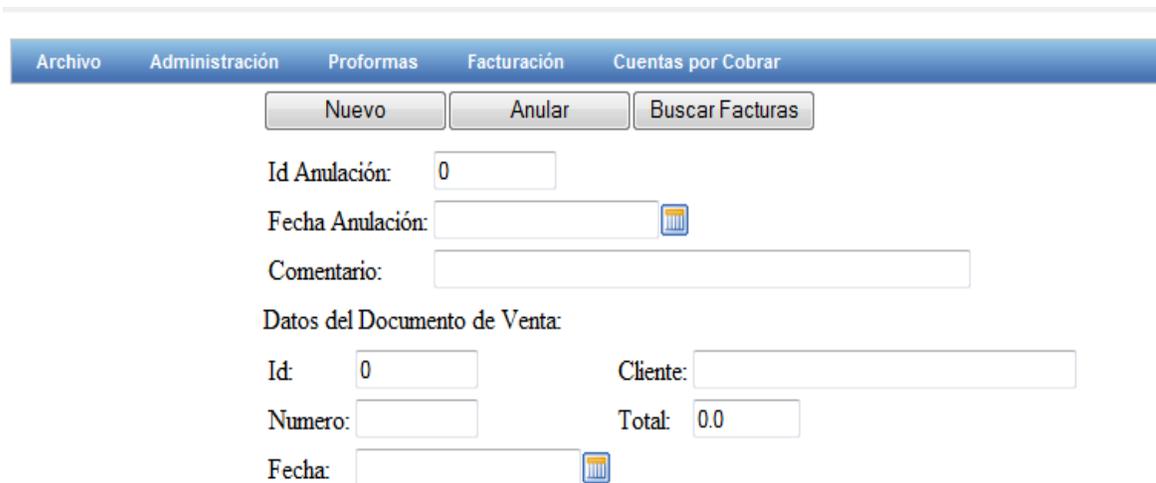
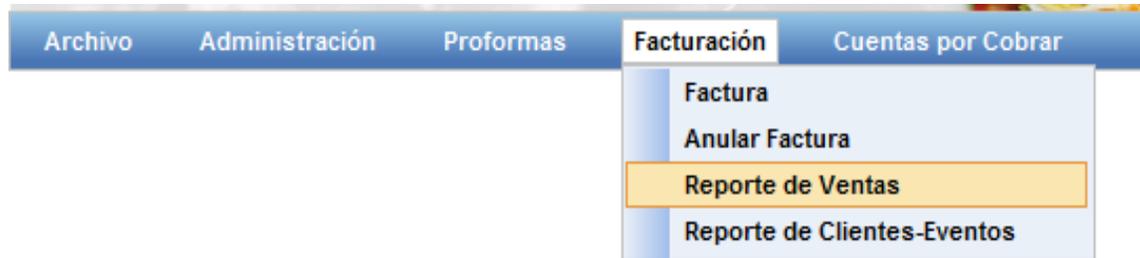
Para agregar un cliente en la factura hacemos clic en el botón “?”, nos despliega la lista de clientes. Para añadir mas productos en caso de ser necesario simplemente hacemos clic en el botón “+” para seguir añadiendo a la factura. Para guardar simplemente damos clic en el botón GUARDAR.

Producto	Cantidad	Tip Pr	Precio	Desc.	Total
menu combo 1	50.0	A	1.85	2.0	90.65
agua tesalia	10.0	A	0.3	0.0	3.0

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## 7.5.2. ANULAR FACTURA

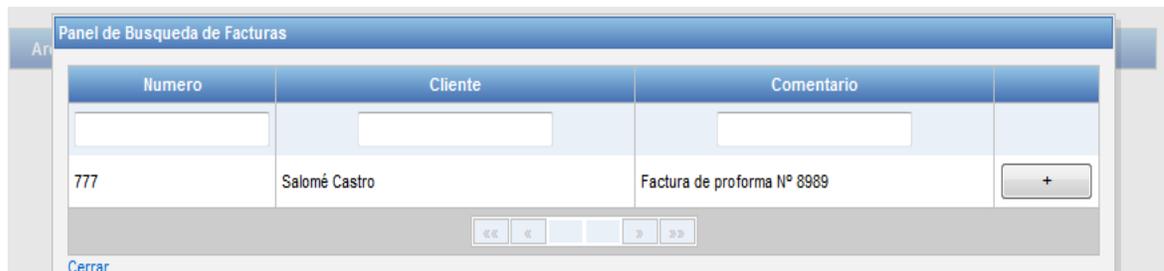
Seleccionamos la opción ANULAR FACTURA haciendo clic.



The image shows the 'Anular Factura' form. It includes a navigation bar with 'Archivo', 'Administración', 'Proformas', 'Facturación', and 'Cuentas por Cobrar'. Below the navigation bar are three buttons: 'Nuevo', 'Anular', and 'Buscar Facturas'. The form contains the following fields:

- Id Anulación:
- Fecha Anulación:  (with a calendar icon)
- Comentario:
- Datos del Documento de Venta:
  - Id:
  - Cliente:
  - Numero:
  - Total:
  - Fecha:  (with a calendar icon)

Podemos realizarla buscando por factura, en donde nos despliega todas las facturas existentes, se puede buscar por número, por cliente y por comentario.



The image shows the 'Panel de Búsqueda de Facturas' window. It contains a table with the following columns: Numero, Cliente, and Comentario. The table has one row of data:

Numero	Cliente	Comentario
777	Salomé Castro	Factura de proforma N° 8989

Below the table are navigation buttons: « «, «, », » » and a 'Cerrar' button.

En donde automáticamente se llenaran los campos correspondientes a esa factura.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Archivo Administración Proformas Facturación Cuentas por Cobrar

Nuevo Anular Buscar Facturas

Id Anulación:

Fecha Anulación:

Comentario:

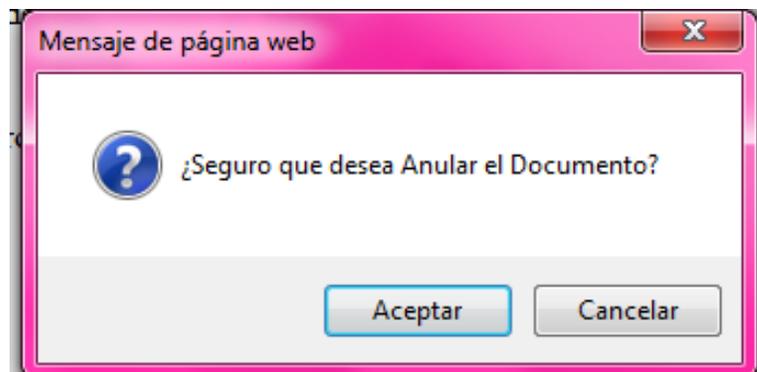
Datos del Documento de Venta:

Id:  Cliente:

Numero:  Total:

Fecha:

Para poder eliminar simplemente hacemos clic en el botón ANULAR, donde nos mostrará una ventana de verificación, luego hacemos clic en ACEPTAR.



### 7.5.3. REPORTE DE VENTAS

Seleccionamos la opción REPORTE DE VENTAS haciendo clic, y el sistema muestra el reporte que se ve a continuación.

REPORTE DE VENTAS

Forma Pago: CREDITO

RUC / CI	Cliente	Número	Fecha	Subtotal	Desc.	Subtotal	Iva	Total
1002314517	Carlos Guevara	777	13/09/2012	0,9	0	0,9	0,11	1,1
<b>Totales:</b>				0,9	0	0,9	0,11	1,1

06/02/2013 20:56

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

#### 7.5.4. REPORTE DE CLIENTE-EVENTOS

Seleccionamos la opción REPORTE DE CLIENTE-EVENTOS haciendo clic, y el sistema muestra el reporte que se ve a continuación.

REPORTE DE CLIENTES - EVENTOS			
RUC / CI	Cliente	Nº Prof.	Nº Fact.
1002314517	Carlos Guevara	3	1

06/02/2013 20:57

#### 7.6. CUENTAS POR COBRAR

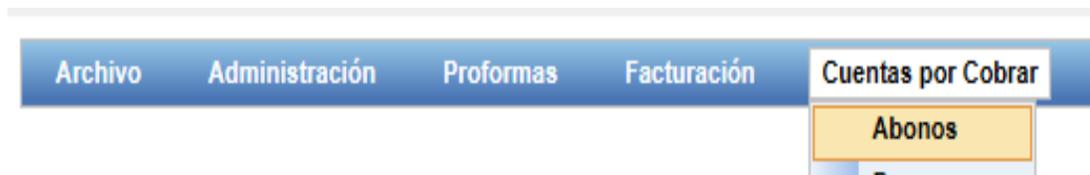
Este módulo consta de lo siguiente:

- Abonos.
- Pago.
- Pagos por Cliente



##### 7.6.1. ABONOS

Hacemos clic en ABONOS.



Nos mostrará un formulario en el que debemos llenar los campos correspondientes al abono. Para ello hacemos clic en NUEVO.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

The screenshot shows the application's main menu with options: Archivo, Administración, Proformas, Facturación, and Cuentas por Cobrar. Below the menu are buttons for 'Nuevo', 'Guardar', 'Buscar Orden', and 'Buscar Abono'. There are two data entry sections: 'Datos de la Orden' and 'Datos del Abono'. Each section contains fields for Id, Number, Date, Client, Total, Saldo Pendiente (with a calculator icon), and Form of Payment (a dropdown menu). The 'Forma de Pago' dropdown is currently set to 'EFECTIVO'. A 'Comentario' field is also present in the 'Datos del Abono' section.

Podemos buscar por ordenes, haciendo clic en BUSCAR ORDEN.

The screenshot shows the 'Panel de Búsqueda de Ordenes' (Order Search Panel). It features a table with columns: Número Prof., Fecha, Cliente, Estado, and a search button (<>). The table contains one row of data: 1000, 2012-08-25, Salomé Castro, PENDIENTE. Below the table are navigation buttons (left and right arrows) and a 'Cerrar' (Close) button.

Número Prof.	Fecha	Cliente	Estado	
1000	2012-08-25	Salomé Castro	PENDIENTE	<>

Podemos buscar por abonos, haciendo clic en BUSCAR ABONO. En cualquiera de los dos casos podemos seleccionarlos haciendo clic en el botón <>. Ofrece varios tipos de buscadores.

The screenshot shows the 'Panel de Búsqueda de Abonos' (Abono Search Panel). It features a table with columns: Id Abono, Cliente, N° Prof., Comentario, Monto, and a search button (<>). The table contains two rows of data: 1, Salomé Castro, 1000, todo ok, 20.0; and 2, Salomé Castro, 1000, ch n 1515, 40.0. Below the table are navigation buttons (left and right arrows) and a 'Cerrar' (Close) button.

Id Abono	Cliente	N° Prof.	Comentario	Monto	
1	Salomé Castro	1000	todo ok	20.0	<>
2	Salomé Castro	1000	ch n 1515	40.0	<>

Una vez ingresado todos los datos hacemos clic en el botón GUARDAR.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

The screenshot shows the 'Cuentas por Cobrar' application interface. The top navigation bar includes 'Archivo', 'Administración', 'Proformas', 'Facturación', and 'Cuentas por Cobrar'. Below the navigation bar are buttons for 'Nuevo', 'Guardar', 'Buscar Orden', and 'Buscar Abono'. The main content area is divided into two sections: 'Datos de la Orden' and 'Datos del Abono'. 'Datos de la Orden' includes fields for Id (1), Cliente (Salomé Castro), Numero (1), Total (114.25), Fecha (ago 25, 2012), and Saldo Pendiente (54.25). 'Datos del Abono' includes fields for Id (1), Forma de Pago (EFFECTIVO), Numero (1616), Comentario (todo ok), Fecha (sep 16, 2012), and Monto (20.0).

## 7.6.2. PAGO

Hacemos clic en la opción PAGO.

The screenshot shows the 'Cuentas por Cobrar' application interface. The top navigation bar includes 'Archivo', 'Administración', 'Proformas', 'Facturación', and 'Cuentas por Cobrar'. The 'Cuentas por Cobrar' menu is expanded, showing 'Abonos' and 'Pago' options. The 'Pago' option is highlighted in yellow.

Para crear un nuevo pago, hacemos clic en el botón NUEVO.

The screenshot shows the 'Cuentas por Cobrar' application interface. The top navigation bar includes 'Archivo', 'Administración', 'Proformas', 'Facturación', and 'Cuentas por Cobrar'. Below the navigation bar are buttons for 'Nuevo', 'Guardar', 'Buscar Factura', and 'Buscar Pago'. The main content area is divided into two sections: 'Datos de la Factura' and 'Datos del Pago'. 'Datos de la Factura' includes fields for Id (0), Cliente, Numero, Total (0.0), Fecha, and Saldo Pendiente (0.0). 'Datos del Pago' includes fields for Id (0), Forma de Pago (EFFECTIVO), Numero, Comentario, Fecha, and Monto (0.0).

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

Para ingresar los datos correspondientes, podemos hacerlo mediante facturas, haciendo clic en el botón BUSCAR FACTURA.

Número Fac.	Fecha	Cliente
777	2012-09-13	Salomé Castro

También podemos hacerlo mediante pagos, haciendo clic en BUSCAR PAGO. Para seleccionar solo damos clic en el botón <>.

Id Pago	Cliente	Nº Factura	Comentario	Monto
3	Salomé Castro	777	dfsdfds	0.2
4	Salomé Castro	777	tttt	0.1

Finalmente hacemos clic en el botón GUARDAR para guardar todos los cambios.

Archivo   Administración   Proformas   Facturación   Cuentas por Cobrar

Nuevo   Guardar   Buscar Factura   Buscar Pago

**Datos de la Factura:**

Id: 5   Cliente: Salomé Castro  
Numero: 777   Total: 1.1  
Fecha: sep 13, 2012   Saldo Pendiente: 0.8

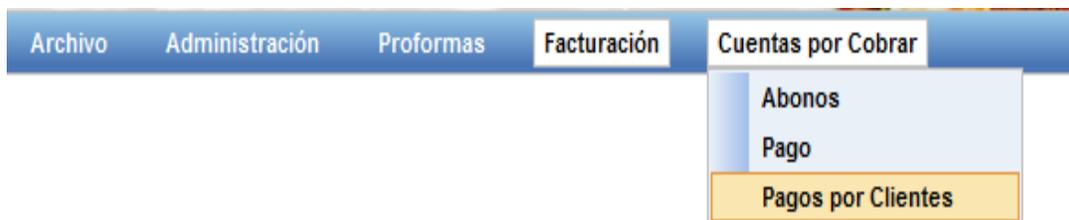
**Datos del Pago:**

Id: 3   Forma de Pago: EFECTIVO  
Numero: 1313   Comentario: dfsdfds  
Fecha: sep 24, 2012   Monto: 0.2

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

### 7.6.3. PAGOS POR CLIENTES

Hacemos clic en PAGOS POR CLIENTES.



El sistema mostrará un filtro de búsqueda de pagos de clientes por: código, RUC y nombre. Nos permite la opción IMPRIMIR.

Código	RUC	Nombre	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
cli0001	1002314517	Carlos Guevara	<a href="#">Imprimir</a>
cli11122	1002884047	Fernando Mejia	<a href="#">Imprimir</a>

Al hacer clic en IMPRIMIR, el sistema mostrará el reporte de pagos por cliente.

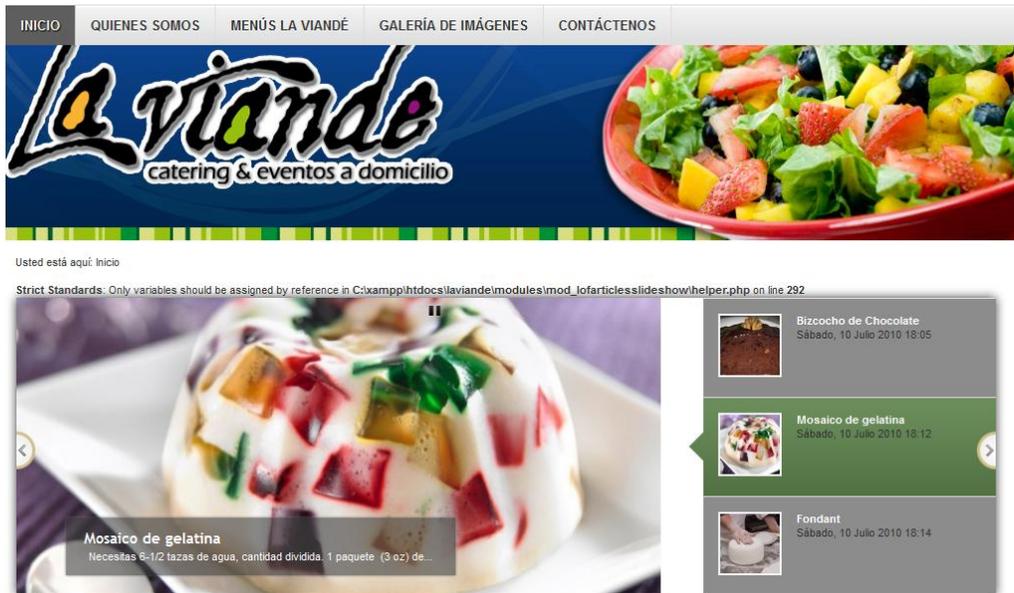
REPORTE DE PAGOS POR CLIENTE						
CLIENTE: Carlos Guevara						
Id Abono	N° Abono	N° Proforma	N° Factura	Monto	Forma Pago	Comentario
1	1616	1000		20	EFFECTIVO	pago de documento pendiente
2	1616	1000		40	CHEQUE	pago con cheque N° 1515 - banco del Pichincha
3	1313	8989	777	0,2	EFFECTIVO	Pago de cuenta pendiente
4	67676	8989	777	0,1	EFFECTIVO	Pago de cuenta pendiente

06/02/2013 21:31

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

## D.8. PÁGINA WEB LA VIANDE

La página web de La Viandé está conformada por un menú principal, que contiene las siguientes categorías:



- Inicio.
  - Esta categoría contiene información correspondiente a lo que se dedica La Viandé, describe su servicio con la sociedad, quienes son sus clientes.
- Quienes Somos.
  - Describe detalladamente el negocio en sí, los servicios que prestan y a que se dedican actualmente.
- Menús La Viandé.
  - Presenta un folleto online en el que se describen algunos menús que la empresa ofrece a sus clientes, con respecto a banquetes, cenas, etc.
- Galería de Imágenes.
  - Muestra una gran cantidad de imágenes con respecto a eventos, postres, platos fuertes, etc.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- Contáctenos.
  - Permite a los clientes que ingresan a la página web, informar sus inquietudes y preguntas para facilitar un mejor servicio.
- Redes Sociales.
  - Posee un *Síguenos*, de la red social de Facebook correspondiente a la empresa.

## E. GLOSARIO DE TÉRMINOS

- **AJAX.-** Es una técnica de desarrollo web para crear aplicaciones interactivas.
- **API.-** Interfaz de Programación de Aplicaciones. Representa una interfaz de comunicación entre componentes de software.
- **CMS.-** Es un programa que permite crear una estructura de soporte para la creación y administración de contenidos, principalmente en páginas web, por parte de los administradores, editores, participantes y demás roles.
- **FRAMEWORK.-** Es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software.
- **HIBERNATE.-** Es una potente herramienta de Mapeo Objeto – Relacional para plataforma JAVA, que facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación.
- **HTML.-** Lenguaje de marcas de hipertexto para el desarrollo de páginas Web, describe la estructura y el contenido en forma de texto, así como para complementar el texto con objetos como imágenes.
- **HTTP.-** Protocolo de transferencia de hipertexto.
- **JAVA BEAN.-** Modelo de componentes para la construcción de aplicaciones en Java. Se usan para encapsular varios objetos en un único objeto, para hacer uso de un solo objeto en lugar de varios más simples.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- **JSF.-** es una tecnología y framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE.
- **JSP.-** JavaServer Pages o Páginas de Servidor Java, es una tecnología Java que permite generar contenido dinámico para la Web.
- **LICENCIA MIT.-** El texto de la licencia no tiene copyright, lo que permite su modificación.
- **MVC.-** Es un patrón o modelo de abstracción de desarrollo de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos (modelo, vista y controlador).
- **ORM.-** El mapeo objeto-relacionales una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, utilizando un motor de persistencia.
- **PHP.-** Es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.
- **POO.-** Programación Orientada a Objetos, este tipo de programación usa objetos y sus interacciones para diseñar software. Incluye varias técnicas como herencia, polimorfismo, encapsulamiento entre otros. La mayoría de lenguajes de programación soportan este tipo de programación.
- **SERVLET.-** Son objetos que corren dentro y fuera del contexto de un contenedor de servlets y extienden su funcionalidad.

Tema: Desarrollo de una Aplicación Web utilizando la tecnología JSF con una arquitectura MVC para La Viandé.

- **STAKEHOLDER.-** Cualquier persona interesada, afectada e implicada con el funcionamiento o desarrollo del software.
- **TOMCAT.-** Funciona como un contenedor de servlets. Tomcat es un servidor web con soporte de servlets y JSPs.
- **URL.-** Localizador Uniforme de Recursos. En Internet estos recursos pueden ser imágenes, documentos de texto, portales Web, sitios FTP, archivos de audio, etc.
- **WIDGETS.-** Es una pequeña aplicación o programa, usualmente presentado en archivos o ficheros pequeños que son ejecutados por un motor de widgets o Widget Engine. Entre sus objetivos estándar fácil acceso a funciones frecuentemente usadas y proveer de información visual.
- **WORKFLOW.-** Es el estudio de los aspectos operacionales de una actividad de trabajo: cómo se estructuran las tareas, cómo se realizan, cuál es su orden correlativo, cómo se sincronizan, cómo fluye la información que soporta las tareas y cómo se le hace seguimiento al cumplimiento de las tareas.
- **XSTL.-** Es un estándar de la organización W3C que presenta una forma de transformar documentos XML en otros e incluso a formatos que no son XML. Es muy usado en la edición web, generando páginas HTML o XHTML. La unión de XML y XSLT permite separar contenido y presentación, aumentando así la productividad.
- **XML.-** Permite definir la gramática de lenguajes específicos para estructurar documentos grandes.