



**ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO  
FACULTAD DE INFORMATICA Y ELECTRONICA  
ESCUELA DE INGENIERIA EN SISTEMAS**

**“ANÁLISIS COMPARATIVO DE LA INTERNACIONALIZACIÓN DE APLICACIONES WEB  
EN ASP Y JSF APLICADO AL MÓDULO DE ADQUISICIONES DEL SISTEMA FINANCIERO  
DE LA ESPOCH”**

**TESIS DE GRADO**

**Previa a la obtención del título de:  
INGENIERIA EN SISTEMAS INFORMATICOS**

**Presentado por:  
EDWIN FERNANDO YAULEMA ROJAS**

**RIOBAMBA- ECUADOR  
2013**

## **AGRADECIMIENTO**

Agradezco a Dios quien me ha guiado y me ha dado la fortaleza para seguir adelante, a mi madre Elci María por sus consejos, por su apoyo incondicional y por todo su sacrificio para darme la educación, a mi hermano Diego por confiar en mí y motivarme a ser cada día una mejor persona, sin su apoyo no habría podido dar este paso tan importante en mi vida.

Fernando Yaulema Rojas

## **DEDICATORIA**

A mi madre por su dedicación, esfuerzo y todas sus enseñanzas, a mi hermano por motivarme, a mis sobrinos ya que llegaron a alegrar mi vida y a mis profesores por compartir su conocimiento y ser grandes referentes en toda mi etapa universitaria.

Fernando Yaulema Rojas

NOMBRE	FIRMAS	FECHA
Ing. Iván Menez <b>DECANO DE LA FACULTAD DE            INFORMÁTICA Y ELECTRÓNICA</b>	.....	.....
Ing. Jorge Huilca <b>DIRECTOR DE LA ESCUELA DE            INGENIERÍA EN SISTEMAS</b>	.....	.....
Ing. Gloria Arcos Medina <b>DIRECTORA DE TESIS</b>	.....	.....
Ing. Eduardo Villa <b>MIEMBRO DEL TRIBUNAL</b>	.....	.....
Tlgo. Carlos Rodríguez <b>DIRECTOR DEL CENTRO DE            DOCUMENTACIÓN</b>	.....	.....
<b>NOTA DE LA TESIS</b>	.....	



## **RESPONSABILIDAD DEL AUTOR**

“Yo, Edwin Fernando Yaulema Rojas, soy el responsable de las ideas, doctrinas y resultados expuestos en esta Tesis de Grado, y el patrimonio intelectual de la misma pertenece a la Escuela Superior Politécnica de Chimborazo.”

---

Fernando Yaulema Rojas.

## ÍNDICE DE ABREVIATURAS

### A

**ANSI:** Instituto Estadounidense de Estándares Nacionales.

**ASCII:** American Standard Code for Interchange (Código Estadounidense Estándar para el Intercambio de Información).

**ASP:** Application Server Providers

### C

**CLR:** Common Language Runtime.

**CORBA:** Common Object Request Broker Architecture

### E

**EJB:** Enterprise JavaBeans.

**ESPOCH:** Escuela Superior Politécnica de Chimborazo

### I

**IIS:** Internet Information Services.

### J

**JDBC:** Java database Connectivity.

**JIT:** Just –In- Time.

**JNDI:** Service Java Naming and Directory Interface.

**JSF:** Java Server Faces

### L

**LDAP:** Lightweight Directory Access Protocol

**LISA:** Asociación de estándares del sector de la localización

## **M**

**MVC:** Modelo Vista Controlador.

## **N**

**NDS:** Novell Directory Service

**NIS:** Network Information Service.

## **P**

**PAC:** Plan Anual de contrataciones

## **S**

**SPI:** Service Provider Interface.

## **U**

**UCS:** Universal Character Set.

**UNICODE:** Estándar de codificación de caracteres diseñado para facilitar el tratamiento informático.

**UTC:** Coordinate Universal Time.

**UTF:** Unicode Transformation Format

## **W**

**WWW:** World Wide Web

## INDICE GENERAL

CAPITULO I .....	- 17 -
MARCO REFERENCIAL .....	- 17 -
1.1 Antecedentes .....	- 17 -
1.2 Justificación del Proyecto de Tesis .....	- 18 -
1.2.1 Justificación Teórica .....	- 18 -
1.2.2 Justificación Aplicativa .....	- 19 -
1.3 Objetivos .....	- 20 -
1.3.1 Objetivo General .....	- 20 -
1.3.2 Objetivos Específicos .....	- 20 -
1.4 Hipótesis .....	- 20 -
CAPITULO II .....	- 22 -
MARCO TEÓRICO .....	- 22 -
2.1 Introducción .....	- 22 -
2.2 Aplicación Web .....	- 23 -
2.3 Internacionalización de Aplicaciones Web .....	- 23 -
2.3.1 ¿Qué es la Internacionalización de Aplicaciones Web? .....	- 23 -
2.3.2 Definición .....	- 23 -
2.3.2 Aspectos a tener en cuenta para Internacionalizar .....	- 24 -
2.3.3 Configuraciones Regionales y Referencias Culturales .....	- 26 -
2.3.4 Aspectos Relativos al Idioma .....	- 27 -
2.3.5 Aspectos Relativos al Formato .....	- 27 -
2.4 Localización de Aplicaciones Web .....	- 30 -
2.4.1 ¿Qué es la Localización? .....	- 30 -
2.4.2 Definición .....	- 31 -
2.4.3 Funcionalidad Técnica .....	- 31 -
2.4.4 Representación de Caracteres .....	- 31 -
2.4.5 Recomendaciones para Localizar una Aplicación: .....	- 36 -
2.5 ¿Qué es un Framework? .....	- 38 -
2.5.1 Framework Web .....	- 38 -
2.6 Patrón de Diseño Modelo Vista Controlador (MVC) .....	- 38 -
2.6.1 Introducción .....	- 38 -
2.6.2 Definición .....	- 39 -
2.6.4 Vista .....	- 40 -
2.6.5 Controlador .....	- 40 -
2.6.6 Ventajas .....	- 41 -

2.6.7 Desventajas .....	- 41 -
2.7 Application Services Providers (ASP) .....	- 41 -
2.7.1 Introducción .....	- 41 -
2.7.2 Evolución de Application Service Providers .....	- 42 -
2.7.2.1 Fase Inicial Evolución como Active Server Pages .....	- 42 -
2.7.2.2 Primera Evolución Active Server Pages .....	- 42 -
2.7.2.3 Segunda Evolución Active Server Pages .....	- 43 -
2.7.2.4 Evolución de Active Server Pages a Application Services Providers .....	- 43 -
2.7.2.5 Application Services Providers en la Actualidad .....	- 44 -
2.7.2 Características .....	- 44 -
2.7.3 Arquitectura de ASP.NET .....	- 46 -
2.7.3.1 Aplicaciones ASP.NET .....	- 47 -
2.7.3.2 Internet Information Services (IIS) .....	- 49 -
2.7.3.3 Framework .NET .....	- 49 -
2.7.3.4 Sistemas Operativos de Familia Microsoft .Net Server .....	- 55 -
2.7.4 La Internacionalización en Application Services Providers .....	- 55 -
2.8 Tecnología Java 2 Enterprise Edition (Java2EE) .....	- 57 -
2.8.1 ¿Qué es J2EE? .....	- 57 -
2.8.2 Arquitectura de J2EE .....	- 58 -
2.8.3 ¿Qué Tecnologías son Incluidas en J2EE? .....	- 59 -
2.8.3.1 Enterprise JavaBeans (EJB) .....	- 59 -
2.8.3.2 JavaServersPages (JSP) .....	- 61 -
2.8.3.3 Servlets Java Naming and Directory Interface (JNDI) .....	- 62 -
2.8.3.4 Java Transaction API (JTA) .....	- 63 -
2.8.3.5 Common Object Request Broker Architecture (CORBA) .....	- 63 -
2.8.3.6 Java Database Connectivity (JDBC) .....	- 64 -
2.8.4 JavaServer Faces (JSF) .....	- 66 -
2.8.4.1 Introducción .....	- 66 -
2.8.2.2 Características .....	- 67 -
2.8.2.3 ¿Hacia dónde está dirigido JSF? .....	- 68 -
2.8.2.4 Componentes de JSF .....	- 69 -
2.8.2.5 La Internacionalización en JSF .....	- 71 -
2.8.2.6 PrimeFaces .....	- 74 -
CAPITULO III .....	- 76 -
ANÁLISIS COMPARATIVO DE LA INTERNACIONALIZACIÓN DE APLICACIONES WEB CON ASP Y JSF .....	- 76 -
3.1 Introducción .....	- 76 -
3.2 Parámetros de Comparación .....	- 77 -

3.2.1 Definición de Parámetros: .....	- 77 -
3.2.2 Escala de Medición.....	- 78 -
3.3 Instrumentos de Medición .....	- 78 -
3.4 Prototipos de Prueba: .....	- 80 -
3.4.1 Prototipo en JSF .....	- 80 -
3.4.2 Prototipo en ASP .....	- 82 -
3.5 Procedimiento.....	- 83 -
3.6 Análisis de Resultados .....	- 84 -
3.7 Demostración de la Hipótesis:.....	- 93 -
3.8 Resultados Obtenidos de la Demostración .....	- 94 -
CAPITULO IV .....	- 95 -
IMPLEMENTACION DEL MÓDULO DE ADQUISICIONES DEL SISTEMA FINANCIERO DE LA ESPOCH .....	- 95 -
4.1 Metodología XP (eXtreme Programming) .....	- 95 -
4.1.1 Fase de Planificación .....	- 96 -
4.1.2 Fase de Diseño.....	- 105 -
4.1.3 Fase de Codificación .....	- 135 -
4.1.4 Fase de Pruebas .....	- 137 -
4.1.5 Pruebas de Stress y Rendimiento .....	- 161 -
RESUMEN.....	-166-
ABSTRACT.....	-167-
CONCLUSIONES.....	- 166 -
RECOMENDACIONES .....	- 170 -
ANEXOS .....	- 176 -
MANUAL DE USUARIO .....	- 177 -
DICCIONARIO DE DATOS .....	- 209 -
HISTORIA DE USUARIO DEL PROTOTIPO EN ASP.....	- 216 -

## INDICE DE TABLAS

Tabla II.1 Formato de Fechas .....	- 28 -
Tabla II.2 Formato Número Negativos y Monedas .....	- 29 -
Tabla III.1 Parámetros e Indicadores .....	- 77 -
Tabla III.2 Escala de Medición .....	- 78 -
Tabla III.3 Request del Servidor.....	- 85 -
Tabla III.4 Kb Procesados/Segundo .....	- 85 -
Tabla III.5 Tiempo de Respuesta .....	- 87 -
Tabla III.6 Diseño de Base de Datos .....	- 88 -
Tabla III.7 Tiempo de Desarrollo en Codificación .....	- 89 -
Tabla III.8 Valores para los Indicadores del Rendimiento .....	- 90 -
Tabla III.9 Valores Indicadores de Tiempo de Desarrollo.....	- 90 -
Tabla III.10 Valoración de Indicadores de Rendimiento .....	- 91 -
Tabla III.11 Valoración de Indicadores de Tiempo de Desarrollo.....	- 92 -
Tabla III.12 Tabla General de Valores de Indicadores .....	- 92 -
Tabla III.13 Ponderación de Indicadores .....	- 93 -
Tabla IV.1 Equipo de Trabajo .....	- 96 -
Tabla IV.2 Historias de Usuario .....	- 97 -
Tabla IV.3 Historias de Usuario - Primera Iteración.....	- 98 -
Tabla IV.4 Historias de Usuario - Segunda Iteración.....	- 99 -
Tabla IV.5 Historias de Usuario – Tercera Iteración .....	- 99 -
Tabla IV.6 Historia de Usuario Diseño e Implementación de la Base de Datos.....	- 100 -
Tabla IV.7 Historia de Usuario Gestión de Categorías de Productos.....	- 100 -
Tabla IV.8 Historia de Usuario Gestión de Subcategorías de Productos .....	- 101 -
Tabla IV.9 Historia de Usuario Gestión de Formas de Presentación de Productos .....	- 101 -
Tabla IV.10 Historia de Usuario Gestión de Tipos de Productos .....	- 101 -
Tabla IV.11 Historia de Usuario Gestión de Productos .....	- 102 -
Tabla IV.12 Historia de Usuario Gestión de Productos del PAC por Actividades .....	- 102 -
Tabla IV.13 Historia de Usuario Incremento de Precios de Productos.....	- 102 -
Tabla IV.14 Historia de Usuario Gestión del Tipo de Período del PAC.....	- 103 -
Tabla IV.15 Historia de Usuario Gestión del Encabezado del PAC .....	- 103 -
Tabla IV.16 Historia de Asignación de Perfiles de Usuario .....	- 104 -
Tabla IV.17 Historia de Usuario Asignación de Parámetros Iniciales.....	- 104 -
Tabla IV.18 Historia de Usuario Revisión PAC con el Rol Adquisiciones .....	- 104 -
Tabla IV.19 Historia de Usuario Revisión PAC con el Rol Presupuesto .....	- 105 -
Tabla IV.20 Historia de Usuario Revisión PAC con el Rol Rector .....	- 105 -
Tabla IV.21 Detalle del Caso de Uso de la Gestión del PAC .....	- 117 -
Tabla IV.22 Detalle del Caso de Uso Usuario Adquisiciones .....	- 119 -
Tabla IV.23 Detalle del Caso de Uso Revisión Presupuestaria.....	- 120 -
Tabla IV.24 Detalle del Caso de Uso Aprobación del PAC .....	- 121 -
Tabla IV.25 Tarea Diseño e Implementación de la Base de Datos .....	- 124 -
Tabla IV.26 Implementación de la Clase Categoría .....	- 124 -
Tabla IV.27 Implementación de la Clase Subcategoría.....	- 125 -
Tabla IV.28 Implementación de la Clase Unidad (Formas de Presentación) .....	- 125 -
Tabla IV.29 Implementación de la Clase Tipo (Tipos de Productos).....	- 125 -
Tabla IV.30 Implementación de la Clase ProductoUnidad .....	- 126 -
Tabla IV.31 Implementación de la Clase Producto.....	- 126 -
Tabla IV.32 Implementación de la Clase TipoPeríodo.....	- 126 -
Tabla IV.33 Implementación de la Clase TipoPeríodoDetallePac .....	- 126 -
Tabla IV.34 Implementación de la Clase EncabezadoPac.....	- 127 -

Tabla IV.35 Implementación de la Clase DetallePac.....	- 127 -
Tabla IV.36 Asignación de Perfiles de Usuario.....	- 127 -
Tabla IV.37 Implementación de la Clase Parametros.....	- 128 -
Tabla IV.38 Implementación de la Clase FCategoria .....	- 128 -
Tabla IV.39 Implementación de la Clase FSubcategoría.....	- 128 -
Tabla IV.40 Implementación de la Clase FTipo .....	- 129 -
Tabla IV.41 Implementación de la Clase FUnidades .....	- 129 -
Tabla IV.42 Implementación de la Clase FProducto.....	- 129 -
Tabla IV.43 Implementación de la Clase FProductoUnidad .....	- 130 -
Tabla IV.44 Implementación de la Clase FTipoPeríodo .....	- 130 -
Tabla IV.45 Implementación de la Clase FEncabezadoPac.....	- 130 -
Tabla IV.46 Implementación de la Clase FDetallePac.....	- 130 -
Tabla IV.47 Implementación de la Clase FTipoPeríodoDetallePac.....	- 131 -
Tabla IV.48 Implementación de la Clase FParametros .....	- 131 -
Tabla IV.49 Implementación del Controlador CategoriaController .....	- 131 -
Tabla IV.50 Implementación del Controlador SubcategoríaController.....	- 132 -
Tabla IV.51 Implementación del Controlador TipoController .....	- 132 -
Tabla IV.52 Implementación del Controlador UnidadController .....	- 132 -
Tabla IV.53 Implementación del Controlador ProductoController .....	- 132 -
Tabla IV.54 Implementación del Controlador ProductoUnidadController .....	- 133 -
Tabla IV.55 Implementación del Controlador TipoPeríodoController .....	- 133 -
Tabla IV.56 Implementación del Controlador EncabezadoPacController.....	- 133 -
Tabla IV.57 Implementación del Controlador DetallePacController .....	- 134 -
Tabla IV.58 Implementación del Controlador TipoPeríodoDetallePacController.....	- 134 -
Tabla IV.59 Implementación del Controlador LenguajeController .....	- 134 -
Tabla IV.60 Implementación del Controlador RevisionAdquisicionesController.....	- 135 -
Tabla IV.61 Implementación del Controlador RevisionPresupuestoController.....	- 135 -
Tabla IV.62 Implementación del Controlador RevisionRectorController .....	- 135 -



## INDICE DE FIGURAS

Figura II.1 ASCII Básico .....	- 33 -
Figura II.2 ASCII Extendido – Página de Códigos 437 .....	- 34 -
Figura II.3 ASCII Extendido - Página de Códigos 850.....	- 34 -
Figura II.4 ASCII Extendido – Latin-1 (ISO-8859-1) .....	- 35 -
Figura II.5 Modelo Vista Controlador .....	- 39 -
Figura II.6 Arquitectura de ASP.NET .....	- 47 -
Figura II.7 Componentes de una Aplicación Web en ASP.NET .....	- 48 -
Figura II.8 Arquitectura del Framework .NET .....	- 50 -
Figura II.9 Relación CLR y Biblioteca de Clases .....	- 52 -
Figura II.10 Arquitectura de J2EE .....	- 58 -
Figura II.11 Uso de los EJBs.....	- 60 -
Figura II.12 Funcionamiento de JSP .....	- 61 -
Figura II.13 Arquitectura JDNI.....	- 63 -
Figura II.14 JDBC ODBC .....	- 64 -
Figura II.15 JDBC Parcialmente Nativo .....	- 65 -
Figura III.1 Captura del Prototipo de JSF.....	- 81 -
Figura III.2 Captura del Prototipo de ASP .....	- 82 -
Figura III.3 Reporte de Resumen JSF .....	- 84 -
Figura III.4 Reporte de Resumen ASP .....	- 84 -
Figura III.5 Request del Servidor .....	- 85 -
Figura III.6 Kb Procesados/Segundo .....	- 86 -
Figura III.7 Valores - Figura de Resultados JSF .....	- 86 -
Figura III.8 Valores - Figura de Resultados ASP .....	- 87 -
Figura III.9 Tiempo de Respuesta.....	- 87 -
Figura III.10 Diseño de Base de Datos .....	- 88 -
Figura III.11 Tiempo de Desarrollo en Codificación .....	- 89 -
Figura III.12 Valores para los Indicadores del Rendimiento .....	- 90 -
Figura III.13 Valores Indicadores Tiempo de Desarrollo.....	- 91 -
Figura III.14 Valoración de Indicadores de Rendimiento .....	- 91 -
Figura III.15 Valoración de Indicadores del Tiempo de Desarrollo.....	- 92 -
Figura III.16 Figura General de Valores de Indicadores .....	- 93 -
Figura III.17 Ponderación de Indicadores .....	- 94 -
Figura III.18 Resultado de Comparación entre ASP y JSF.....	- 94 -
Figura IV.1 Metodología XP (eXtreme Programming) .....	- 95 -
Figura IV.2 Plan de Entrega Iteración Primera .....	- 98 -
Figura IV.3 Plan de Entrega Iteración Segunda .....	- 99 -
Figura IV.4 Plan de Entrega Iteración Tercera .....	- 100 -
Figura IV.5 Esquema de la Base de Datos del Módulo de Adquisiciones.....	- 106 -
Figura IV.6 Esquema Tablas del Esquema “master” para Autenticación .....	- 107 -
Figura IV.7 Esquema Tablas del Esquema “psto” para Asignación de Actividades.....	- 107 -
Figura IV.8 Diseño General de la Interfaz del Sistema.....	- 108 -
Figura IV.9 Gestión de Categorías.....	- 109 -
Figura IV.10 Nueva, Edición y Eliminación de Categoría .....	- 109 -
Figura IV.11 Gestión de Subcategorías .....	- 109 -
Figura IV.12 Nueva, Edición y Eliminación de Subcategoría.....	- 110 -
Figura IV.13 Gestión de Tipos de Productos .....	- 110 -
Figura IV.14 Nueva, Edición y Eliminación de Tipos de Productos.....	- 110 -
Figura IV.15 Gestión de Formas de Presentación .....	- 111 -
Figura IV.16 Nueva, Edición y Eliminación de Formas de Presentación.....	- 111 -

Figura IV.17 Gestión del Catálogo de Productos.....	- 112 -
Figura IV.18 Nuevo, Edición y Eliminación de Productos del Catálogo .....	- 112 -
Figura IV.19 Incremento Porcentual de Precios de Productos .....	- 113 -
Figura IV.20 Gestión de Tipos de Períodos.....	- 113 -
Figura IV.21 Nuevo, Edición y Eliminación de Tipos de Períodos.....	- 113 -
Figura IV.22 Gestión de Encabezados del PAC .....	- 114 -
Figura IV.23 Nuevo, Edición y Eliminación de Encabezados del PAC.....	- 114 -
Figura IV.24 Gestión de Ítems del PAC .....	- 115 -
Figura IV.25 Nuevo y Edición de Ítems del PAC .....	- 115 -
Figura IV.26 Eliminación de Ítems al PAC .....	- 115 -
Figura IV.27 Revisión del PAC por Adquisiciones .....	- 116 -
Figura IV.28 Revisión del PAC por Presupuestos .....	- 116 -
Figura IV.29 Revisión del PAC por el Rector.....	- 117 -
Figura IV.30 Caso de Uso de la Gestión del PAC .....	- 117 -
Figura IV.31 Caso de Uso Usuario Adquisiciones .....	- 118 -
Figura IV.32 Caso de Uso Revisión Presupuestaria.....	- 120 -
Figura IV.33 Caso de Uso Aprobación del PAC .....	- 121 -
Figura IV.34 Flujo de Proceso para Gestión del PAC.....	- 123 -
Figura IV.35 Arquitectura del Sistema .....	- 124 -
Figura IV.36 Diagrama de Despliegue .....	- 137 -
Figura IV.37 Diagrama de Componentes.....	- 137 -
Figura IV.38 Captura de Navegación en BadBoy .....	- 162 -
Figura IV.39 Configuración de Propiedades de Hilos - JMeter.....	- 163 -
Figura IV.40 Reporte Resumen – JMeter .....	- 164 -
Figura IV.41 Figura de Resultados - JMeter .....	- 165 -
Figura IV.42 Árbol de Resultados .....	- 165 -

## INTRODUCCIÓN

En los últimos años la tecnología ha avanzado de manera vertiginosa tanto en hardware, software e internet, lo que hace posible que cada vez se tenga más información disponible en distintas regiones geográficas. Como consecuencia de la globalización del acceso a la información se ha visto la necesidad de visualizar dicha información de sitios y aplicaciones web en varios idiomas, este proceso ha sido llamado como Internacionalización, el cual consiste en la creación o modificación de aplicaciones web de tal manera que el idioma deje ser una limitante para el usuario.

En nuestro país los sistemas informáticos en general son desarrollados de forma Internacionalizada en pocas ocasiones y de manera particular en la Escuela Superior Politécnica de Chimborazo.

Como resultado de las políticas del gobierno actual el cual garantiza el acceso a la información, recomienda crear sistemas informáticos multilinguaje o internacionalizados para superar la barrera del idioma.

Ya que el Departamento de Adquisiciones de la ESPOCH no cuenta con un módulo de Adquisiciones para el Sistema Financiero, se ha visto la necesidad de crear dicho módulo para un fácil manejo de la información de ésta área, con el adicional que dicho módulo será internacionalizado.

Para realizar este cometido existe una gran cantidad de herramientas para el desarrollo de software, sin embargo centraremos nuestro estudio en Application Server Providers (ASP) y JavaServer Faces (JSF) ya que son dos de las herramientas más utilizadas en la actualidad.

Estas dos herramientas nos brindan todos los componentes necesarios para realizar una aplicación internacionalizada con interfaces amigables para facilitar la interacción usuario-sistema.

La presente investigación tiene por objetivo realizar el análisis comparativo de la internacionalización de aplicaciones web en ASP y JSF. Para determinar que herramienta de desarrollo de software es más eficiente en rendimiento y tiempo de desarrollo se creó

prototipos en ASP y JSF los cuales a través de pruebas de stress y rendimiento nos permitieron determinar la mejor herramienta de desarrollo para las aplicaciones web internacionalizadas. Una vez obtenida la tecnología más óptima se creó el Módulo de Adquisiciones del Sistema Financiero de la ESPOCH utilizando la tecnología JSF.

Para este cometido se ha dividido en capítulos donde el primero contiene el marco referencial, el cual contiene los antecedentes, las justificaciones, objetivos y la hipótesis de la tesis el cual ha servido como referencia para iniciar en este trabajo; el segundo capítulo trata acerca de los conceptos generales de los términos que han sido de gran relevancia y poder aplicarlos en la creación de aplicaciones web internacionalizadas; el tercer capítulo trata acerca de la realización del trabajo de investigación como: identificación de variables, definición de indicadores, definición de herramientas software y hardware también trata acerca del análisis de resultados obtenidos, la comprobación de la hipótesis y finalmente el cuarto capítulo trata acerca de la metodología utilizada para el desarrollo e implementación del Módulo de Adquisiciones de la Escuela Superior Politécnica de Chimborazo.

# **CAPITULO I**

## **MARCO REFERENCIAL**

### **1.1 Antecedentes**

En el pasado las aplicaciones web tenían limitaciones de gran trascendencia, entre las principales se puede mencionar un mismo idioma para diferentes regiones y/o culturas en el mundo, además de las limitaciones en ciertas tecnologías usadas para su desarrollo.

Los problemas que genera una aplicación web monolingüística (para un solo idioma) es el entendimiento de la información desde otra región o país.

En general las aplicaciones web monolingüísticas presentan inconvenientes al momento de acceder a la información desde una región que hable un idioma distinto al de la aplicación web, es por ello que el desarrollo de una aplicación web multilingüística ayuda en la accesibilidad a la información, ya que ésta permitirá la traducción de dicha aplicación web, la cual se creará con herramientas de desarrollo de software que permitan realizar este tipo de aplicaciones.

Actualmente este tipo de aplicaciones han ido incrementándose en la web muy rápidamente, mostrando la información de manera más globalizada gracias a la gran variedad de herramientas para el desarrollo de aplicaciones web multilingües, cabe mencionar que en nuestro medio el desarrollo de aplicaciones web multilingües es todavía escaso.

Por medio de las herramientas de desarrollo como JSF (Java Server Faces) y ASP (Application Service Providers) se logra la internacionalización (varios idiomas) de las aplicaciones web.

En la actualidad tanto JSF como ASP proporcionan un conjunto de características nuevas y mejoradas para este cometido.

Desde una interfaz intuitiva se puede tener acceso a la información y proporcionar seguridad de los datos en la web de manera transparente para el usuario.

Ya que el Departamento de Adquisiciones de la ESPOCH no cuenta con un módulo de Adquisiciones para el Sistema Financiero, se ha visto la necesidad de crear dicho módulo para un fácil manejo de la información de ésta área ya que los registros se llevan de forma manual en hojas de Excel.

## **1.2 Justificación del Proyecto de Tesis**

### **1.2.1 Justificación Teórica**

La Internacionalización de las Aplicaciones Web consiste en la creación o modificación de aplicaciones que puedan ser usadas en distintas regiones geográficas y en diferentes idiomas. Este proceso se lleva a cabo durante la fase de diseño y desarrollo de una aplicación. Es decir, deberá ser culturalmente neutro, y deberá aislar elementos que sean específicos del idioma como formato de fechas, números y moneda.

Ésta necesidad nace con el fin de alcanzar una gran cantidad de usuarios, así como el comprobar que tecnología entre Application Service Providers (ASP) y Java Server Faces (JSF) es la más adecuada para el desarrollo de este tipo de aplicaciones.

La importancia de la internacionalización en las aplicaciones de software, se explica en los recursos que se deben conocer para internacionalizar adecuadamente, y profundizar en los esquemas de codificación de caracteres (Unicode), ya que todavía se discute sobre la influencia de las características culturales, geográficas y políticas de una región sobre el desarrollo del software.

Las principales ventajas de este modelo son:

- Es posible agregar rápidamente compatibilidad para nuevas referencias culturales: Para agregar compatibilidad con una nueva referencia cultural, sólo es necesario traducir los recursos de la aplicación para la referencia cultural de destino.
- Se aprovechan los recursos de manera más eficaz: al aislar desde la fase de desarrollo los recursos que serán susceptibles de cambiar dependiendo del idioma o localidad del producto.

### **1.2.2 Justificación Aplicativa**

El desarrollo de la aplicación web se realizará tanto en JSF como ASP ya que son herramientas que nos proporcionan una gran gama de características para el desarrollo de aplicaciones web multilingüe.

La aplicación internacionalizada se desarrolla para facilitar la traducción de las aplicaciones web, como resultado de las nuevas políticas de nuestro Gobierno con el fin de globalizar y garantizar el acceso a la información. Por ello recomienda crear sitios web sean multilingües.

A través de la implementación de prototipos en ASP y JSF se determinará la tecnología más apropiada para el desarrollo del Módulo de Adquisiciones del Sistema Financiero.

El Módulo de Adquisiciones del Sistema Financiero realizará los siguientes procesos:

- El módulo permitirá el ingreso, modificación y eliminación de los artículos del PAC.
- El módulo permitirá realizar búsquedas de los artículos registrados en el PAC.

- El módulo emitirá reportes del PAC por Facultad y/o Unidad Administrativa y del Plan Anual de Compras.

### **1.3 Objetivos**

#### **1.3.1 Objetivo General**

Realizar el Análisis Comparativo de la Internacionalización de Aplicaciones Web en ASP y JSF Aplicado al Módulo de Adquisiciones del Sistema Financiero de la ESPOCH.

#### **1.3.2 Objetivos Específicos**

- Analizar el concepto de la Internacionalización de las Aplicaciones Web para un mejor desarrollo y conocer los componentes que hacen posible la Internacionalización de Aplicaciones Web en ASP y JSF.
- Determinar los parámetros de comparación para realizar el análisis entre ASP y JSF.
- Crear Prototipos con las tecnologías ASP y JSF para implementar el Módulo de Adquisiciones del Sistema Financiero de la ESPOCH para determinar el rendimiento de la mejor tecnología.
- Implementar el Módulo de Adquisiciones del Sistema Financiero de la ESPOCH.

### **1.4 Hipótesis**

La implementación de una Aplicación Web Internacionalizada en JSF es más eficiente en Rendimiento y Tiempo de Desarrollo que una Aplicación Web Internacionalizada desarrollada en ASP.





# CAPITULO II

## MARCO TEÓRICO

### 2.1 Introducción

El avance continuo de la tecnología ha dado origen a que la interacción computador-humano sea cada vez más común, esto ha producido un incremento exponencial en el número de usuarios que usan la World Wide Web (WWW) en todo el mundo, como consecuencia de ello se planteó la Internacionalización (i18n) de Aplicaciones Web, lo cual facilita la traducción de las aplicaciones web al idioma nativo de los usuarios. Las aplicaciones web interactúan con los usuarios mediante los navegadores web de un modo fácil y práctico, por esta razón son muy utilizados hoy en día. El incremento del uso del internet ha obligado que sea imperativo el desarrollo de aplicaciones web ya que las personas han hecho del uso de los computadores un factor determinante en casi todas las actividades del diario convivir.

Desde los inicios de la WWW el inglés ha sido el idioma predominante en esta tecnología, a pesar que el primer idioma de muchos de los usuarios no es el inglés y aunque dicho idioma se acepte como la lengua oficial del internet, es natural que una aplicación web en el idioma

materno del usuario sea mucho más cómoda al momento de utilizarla, pudiendo de esta manera saltar las barreras del idioma y así ser mucho más atractiva y fácil de utilizar.

## **2.2 Aplicación Web**

Las aplicaciones web son soluciones informáticas que los usuarios utilizan accediendo a un servidor a través de Internet o su red interna (intranet). Como interfaz con la aplicación se utiliza un navegador de Internet. [13]

## **2.3 Internacionalización de Aplicaciones Web**

### **2.3.1 ¿Qué es la Internacionalización de Aplicaciones Web?**

Se denomina Internacionalización a desarrollar un diseño que se pueda usar en todo el mundo, mientras que la configuración regional hace referencia a la creación de una versión específica para una región o zona específica. El proceso de crear un conjunto de elementos específicos de un país o región se llama Localización. La internacionalización implica minimizar en la interfaz elementos que sean dependientes de un país o cultura. Por ejemplo, si el programa está originalmente desarrollado para usuarios de habla inglesa se intentaría usar un lenguaje sencillo para que aquellos que sólo tengan un conocimiento superficial de esta lengua no lo encontrarán excesivamente complicado. También, se aconsejaría utilizar indicaciones de tipo Figura o de disposición de tal modo que se pudiera eliminar texto a favor de otros elementos menos vinculados a una cultura determinada. [2]

### **2.3.2 Definición**

La Internacionalización de Aplicaciones Web consiste en la identificación de toda la información local que aparece en un sitio web, es decir, aquella información que viene dictada por el idioma y la cultura del país donde se diseñó originalmente. Por ejemplo fechas, moneda, etc. Estos elementos deberán aislarse y guardarse de forma independiente para que sea posible adaptarlos a las especificaciones de cualquier idioma. [14]

La internacionalización es el diseño y desarrollo de un producto, una aplicación o el contenido de un documento de modo tal que permita una fácil localización con destino a audiencias de diferentes culturas, regiones o idiomas. [14]

### **2.3.2 Aspectos a tener en cuenta para Internacionalizar**

Para poder entender la actividad de la traducción, es necesario aclarar lo que entendemos por "idioma" y "cultura" y la relación entre uno y otra. Ya no es posible concebir la traducción desde un enfoque puramente lingüístico, como tampoco es factible centrarse de manera exclusiva en la relación entre las culturas distintas. [7]

Los principales aspectos a tener en cuenta a la hora de diseñar una aplicación internacionalizada son los siguientes:

#### **Adaptación a las Referencias Culturales y Regionales:**

Mediante una planificación previa se debe considerar los siguientes aspectos:

- Identificación de los idiomas y localidades en que debe ser desarrollado el producto.
- Diseñar las características propias de estos idiomas y localidades.
- Escribir código que funcione igualmente en cualquiera de los idiomas definidos para la aplicación.

#### **Personalización de Características:**

Un buen diseño minimiza la necesidad de personalizar características para los diferentes idiomas o culturas. Sin embargo, cuando se escriben las especificaciones de un producto, puede haber ciertas características que sean necesarias incluir para un idioma determinado aunque no se utilicen en otros idiomas.

#### **Establecer el Entorno para el Desarrollo del Producto Internacionalizado:**

A la hora de crear un producto internacionalizado (multilingüe) hay que tener en cuenta que se debe tener un único código principal para todas las ediciones del producto.

Mantener un código específico para cada edición puede ser contraproducente, ya que se necesitará mucho más trabajo de mantenimiento. Una ventaja muy importante de tener un código común para todas las ediciones es la detección y manejo de errores. Se puede mantener una única base de datos donde se guarden todos los errores. Muchos errores serán comunes para todas las ediciones, de tal manera que arreglándolos una vez, se arreglarán para todas las ediciones.

A la hora de elegir las herramientas a utilizar es importante tener en cuenta que tanto los compiladores que se usen así como los editores deben permitir el uso de diferentes idiomas o culturas. Organizar los recursos del sistema de tal manera que los traductores solo tengan acceso a los archivos que necesitan ser traducidos. Los desarrolladores tendrán que, por ejemplo, implementar las actualizaciones únicamente una vez en el código fuente y propagar los cambios necesarios a los archivos de recursos por lotes. En el ejemplo práctico de este proyecto se ha utilizado como entorno de desarrollo Java Server Faces (JSF) que mantiene el código separado del texto traducible mediante archivos de propiedades, estos archivos de propiedades incluyen todo el texto traducible. De esta manera se realiza cualquier cambio necesario en el texto o su visualización en los archivos de propiedades sin que el código fuente sea modificado.

### **Los Traductores y Desarrolladores:**

El trabajo de los desarrolladores tiene un gran impacto sobre el trabajo de los traductores. Habitualmente el trabajo de traducción es costoso, así que cuanto más se les facilite el trabajo a los traductores, más se estará ahorrando en tiempo y costo en la traducción del sistema. Para ello el desarrollador deberá tener en cuenta que mientras más documentado esté el código será más fácil para el traductor entender qué es lo que el desarrollador quería decir para poder traducirlo. En estos glosarios el desarrollador puede incluir la terminología que desea que sea utilizada en el producto de tal manera que esos glosarios puedan ser utilizados por los traductores. También está muy extendido el uso de la traducción automática, aunque si bien

facilita mucho la traducción, sin embargo no puede evitar la necesidad de la intervención humana, ya que necesitará ser revisado por un traductor. Es muy conveniente crear un Kit de localización preparado por el desarrollador para que lo use el traductor. Este kit deberá incluir los archivos localizables, las herramientas necesarias para su traducción así como todo el material de referencia que se le pueda facilitar como glosarios, memorias de traducción, etc.

### **Consideraciones sobre el Control de Calidad del Producto:**

Mediante el control de calidad en el desarrollo del sistema, se supervisa los procesos que tienen que ver con la internacionalización y localización del sistema de esta manera se asegura el correcto funcionamiento.

Se puede resumir que la internacionalización consiste en el diseño y desarrollo de una aplicación de software para que pueda funcionar con diversas culturas y configuraciones regionales. [7]

### **2.3.3 Configuraciones Regionales y Referencias Culturales**

Una configuración regional está compuesta por varias categorías para las que existe un formato u otras especificaciones que dependen del país. La configuración regional de un programa define los conjuntos de códigos, las convenciones de formato de fecha y hora, moneda y decimales, y el orden de intercalación.

Una aplicación o página Web además de tener el contenido correctamente codificado, de manera que se pueda leer correctamente en los distintos idiomas en los que haya sido diseñado, además debe tener en cuenta las reglas del lenguaje y las convenciones culturales. La representación de los datos tales como fechas, hora o símbolos de moneda son algo muy importante ya que su representación final puede tener diferencias dependiendo del idioma o cultura. [11]

El desarrollador de software debe elegir un idioma predeterminado, el cual se mostrará por defecto al momento de desplegar la aplicación. También deberá crear todos los archivos necesarios para cada idioma que desee que estén disponibles en la aplicación.

### **2.3.4 Aspectos Relativos al Idioma**

#### **Bidireccionalidad:**

Hay que tener en cuenta que la aplicación o sitio Web puede que requiera ser localizado en idiomas como árabe o hebreo, cuya lectura y escritura es de derecha a izquierda. Por tanto la aplicación deberá tener capacidad bidireccional, es decir, que admita texto escrito de izquierda a derecha y de derecha a izquierda y por supuesto ser capaz de mostrar el interfaz en ambas direcciones.

#### **Saltos de palabras y líneas:**

Los idiomas occidentales siguen patrones de división de líneas mediante reglas de guionado o límites de palabras, es decir, dividen el texto basándose en espacios en blanco (espacios, tabulaciones, fin de línea, puntuación, etc.). Sin embargo las reglas para los idiomas asiáticos de doble byte son totalmente distintas. Por ejemplo, el tailandés no distingue necesariamente una palabra de la siguiente usando un espacio.

Para estos idiomas, el algoritmo de división y ajuste de línea no puede basarse en un carácter de espacio sino que deberá seguir otras directrices.

#### **Diferencias en los teclados:**

Los esquemas de teclados también cambian con la configuración regional. Algunos caracteres no existen en determinados teclados. Por tanto al asignar una combinación de teclas de un método abreviado, hay que asegurarse que pueden ser reproducidas en teclados internacionales. Una recomendación es usar números y teclas de función (F2, F3, etc.) en vez de letras en las combinaciones de teclas de método abreviado. [8]

### **2.3.5 Aspectos Relativos al Formato**

Debemos tener en cuenta que cada casi todos los idiomas tienen un formato diferente (ya sea en la fecha, medidas, etc.), por lo que se debe tener muy en cuenta éste aspecto al momento de desarrollar una aplicación internacionalizada.

### Diferencias de Formatos de Fechas

El calendario gregoriano se usa en la mayoría de países de habla inglesa, pero las aplicaciones (software) de hoy en día también deben tener en cuenta los otros sistemas de calendarios que se usan en el resto del mundo. Están, por ejemplo, el calendario japonés, el de la era budista, el islámico Hijri, el hebreo lunar y los calendarios taiwaneses. Una de las principales diferencias entre ellos es que cada uno puede tener un valor de año diferente. Por ejemplo, el año 2000 gregoriano equivale al doceavo año de la era Heisei japonesa y al año 1421 del calendario Hijri. Es posible que el año no comience el 1 de enero como el año chino por ejemplo. También puede variar la duración de los meses y de los años, así como el modo en que se administran los años bisiestos. [20]

A continuación podemos visualizar un ejemplo de los diferentes formatos de una fecha en diferentes países, esto se debe a los diferentes idiomas que adoptaron como lengua materna dichos países (Tabla II.1).

**Tabla II.1 Formato de Fechas**

FECHA PROPUESTA	PAÍS	FORMATO	INTERPRETACION DE FECHA
01/03/09	Estados Unidos	mm/dd/aa	3 de Enero del 2009
	Ecuador	dd/mm/aa	1 de Marzo del 2009
	Hungría	aa/mm/dd	9 de Marzo del 2001

Elaborado por: Fernando Yaulema Rojas

### Diferencias de Formatos de Horas

Al igual que ocurre con los formatos de fecha y calendario, los formatos de hora no son iguales en todo el mundo. Aunque cada representación de la hora muestre básicamente la hora, los minutos y los segundos, su orden de presentación y separadores pueden variar considerablemente. De hecho, puede haber muchas diferencias entre regiones dentro del mismo país. Los formatos de hora pueden diferir en función de la referencia cultural de una de las tres maneras siguientes:



- **Uso del Reloj de 12 o 24 Horas:** La mayoría de los países europeos y asiáticos usan el reloj de 24 horas en lugar del de 12 horas. El modelo A.M/P.M. se usa en Estados Unidos.
- **El Carácter Usado para Separar las Horas, Minutos y Segundos:** Aunque los dos puntos (“:”) es el separador más común para separar horas, minutos y segundos, sin embargo algunos países asiáticos usan caracteres ideográficos. Además en algunas regiones se requiere añadir “h”, “m” y “s” como parte del formato mostrado.
- **Forma de Mostrar la Franja Horaria:** la forma más común de mostrar la franja horaria a la que pertenece una hora determinada es añadir GMT (Greenwich Mean Time) o una forma más moderna UTC (Coordinated Universal Time).  
Estas siglas van seguidos de la horas que hay que añadir o restar a la hora mostrada, por ejemplo UTC +5:30. [17]

#### Diferencias de Formatos de Monedas

El formato de moneda (Tabla II.2) debe tener en cuenta los siguientes elementos:

- **Símbolo de Moneda:** puede ser un símbolo predefinido como el del euro “€” o una combinación de letras como el uso de “lei” para el Lei rumano.
- **Colocación del Símbolo de Moneda:** Puede ir colocado antes o después de los dígitos.
- **Visualización de las Cantidades Negativas:** Las cantidades negativas son se representan de la misma manera en las diferentes regiones del mundo. [18]

Tabla II.2 Formato Número Negativos y Monedas

Descripción	País	Formato
Un símbolo negativo antes del símbolo de la moneda y el número.	Reino Unido	-£127.54
	Francia	-127,54 F
Un símbolo negativo antes del número pero después del símbolo de la moneda.	Dinamarca	kr-127,54
Un signo negativo después del número.	Países Bajos	€ 127,54-
Uso de paréntesis.	EEUU	(\$127.54)

Elaborado por: Fernando Yaulema Rojas

## Diferencias de Formatos de Valores Numéricos

A la hora de tratar con números hay que tener en cuenta los siguientes factores:

1. **El carácter usado como separador de miles:** varía de unos países a otros, por ejemplo en EEUU se usa la coma (,) y en Alemania el punto (.)
2. **El carácter usado como separador decimal:** también varía de unos países a otros, por ejemplo en EEUU usan el punto (.) y en Alemania la coma (,).
3. **La manera de mostrar los números negativos:** el signo negativo puede aparecer al principio del número o al final. El signo negativo también puede aparecer entre paréntesis.
4. **Correspondencia entre dígitos y números:** no todos los números tienen que tener una correspondencia uno a uno con el sistema decimal que se conoce habitualmente, es decir, no tienen que tener una correspondencia uno a uno con dígitos del 0 al 9. Por ejemplo, mientras que en España todos los números se representan con los dígitos del 0 al 9, en Japón tienen un dígito más que se correspondería con el número 10 de España.
5. **Agrupación de dígitos:** se refiere al número de dígitos que hay entre cada separador dentro del grupo de dígitos que aparecen a la izquierda de separador de decimales. Por ejemplo, para la mayoría de culturas la agrupación se hace en grupos de tres dígitos, por ejemplo: 123.256,45. Sin embargo en Hindi se usan grupos de dos dígitos excepto para las centenas que se usan grupos de tres, por ejemplo: 23,34,123.98
6. **Colocación del símbolo de porcentaje (%):** se puede colocar de diferentes maneras: 56%, 56 %, 56 pct, &56. [19]

## 2.4 Localización de Aplicaciones Web.

### 2.4.1 ¿Qué es la Localización?

El término Localización (l10n) se emplea con mucha frecuencia al hablar de encargos de traducción de software, Webs o videojuegos. A primera vista puede parecer que localizar es lo

mismo que traducir, pero si se analiza con detalle todo el proceso de localización, observaremos que la traducción es solo una parte de él. [3]

#### **2.4.2 Definición**

La asociación de estándares del sector de la localización (LISA) define la localización (I10n) como “el proceso de modificación de productos o servicios para diferentes mercados”. Según esta definición, localizar un producto o servicio implica realizar modificaciones o adaptaciones en el original para crear un producto o servicio que pueda comercializarse en el mercado al que se pretende llegar. Para ello, no solo será necesario traducir, ya que también habrá que tener en cuenta aspectos culturales, aspectos funcionales y técnicos. [15]

#### **2.4.3 Funcionalidad Técnica**

Para localizar una aplicación se debe hacer el análisis o preparación de una aplicación para ser localizada. Una aplicación preparada para ser localizada tendrá un bloque de código y un bloque de datos o archivos de recursos. Estos archivos de recursos contienen todos los recursos de cadena de la interfaz de usuario. El bloque de código contendrá exclusivamente el código de la aplicación, válido para todas las culturas o configuraciones regionales.

En teoría se podrá obtener una versión localizada de la aplicación cambiando únicamente los archivos de recursos. La combinación del bloque de datos y el bloque de código dará como resultado una versión localizada de la aplicación. [22]

#### **2.4.4 Representación de Caracteres.**

##### **¿Qué es la Codificación?**

Con codificación de caracteres nos referimos al método que permite convertir un carácter de un lenguaje natural en un símbolo de otro sistema de representación, por ejemplo en un número, una secuencia de pulsos eléctricos en un sistema electrónico, octetos aplicando normas o reglas de codificación. La representación de caracteres es un aspecto muy importante dentro de la localización, ya que no todos los idiomas utilizan los mismos caracteres (símbolos) para representarlos. Un ejemplo de ello es que el

idioma inglés no utiliza tildes, signos de exclamación e interrogación al inicio de un texto, la letra eñe (ñ) al contrario del español que si utiliza todos los ejemplos anteriores.

Los tipos de codificación más utilizados son los siguientes:

### **ASCII**

De sus siglas en inglés American Standard Code for Information Interchange (Código Estadounidense Estándar para el Intercambio de Información), es un código de caracteres basado en el alfabeto latino tal como se usa en inglés moderno y en otras lenguas occidentales. Fue creado en 1963 por el Comité Estadounidense de Estándares (ASA, conocido desde 1969 como el Instituto Estadounidense de Estándares Nacionales, o ANSI) como una evolución de los conjuntos de códigos utilizados entonces en telegrafía. Más tarde, en 1967, se incluyeron las minúsculas, y se redefinieron algunos códigos de control para formar el código conocido como US-ASCII.

ASCII es, en sentido estricto, un código de siete bits, lo que significa que usa cadenas de bits representables con siete dígitos binarios (que van de 0 a 127 en base decimal) para representar información de caracteres. En el momento en el que se introdujo el código ASCII muchos ordenadores trabajaban con grupos de ocho bits (bytes u octetos), como la unidad mínima de información; donde el octavo bit se usaba habitualmente como bit de paridad con funciones de control de errores en líneas de comunicación u otras funciones específicas del dispositivo. El código ASCII define una relación entre caracteres específicos y secuencias de bits; además de reservar unos cuantos códigos de control para el procesador de textos, y no define ningún mecanismo para describir la estructura o la apariencia del texto en un documento; estos asuntos están especificados por otros lenguajes como los lenguajes de etiquetas.

Caracteres ASCII de control			Caracteres ASCII imprimibles					
00	NULL	(carácter nulo)	32	espacio	64	@	96	`
01	SOH	(inicio encabezado)	33	!	65	A	97	a
02	STX	(inicio texto)	34	"	66	B	98	b
03	ETX	(fin de texto)	35	#	67	C	99	c
04	EOT	(fin transmisión)	36	\$	68	D	100	d
05	ENQ	(consulta)	37	%	69	E	101	e
06	ACK	(reconocimiento)	38	&	70	F	102	f
07	BEL	(timbre)	39	'	71	G	103	g
08	BS	(retroceso)	40	(	72	H	104	h
09	HT	(tab horizontal)	41	)	73	I	105	i
10	LF	(nueva línea)	42	*	74	J	106	j
11	VT	(tab vertical)	43	+	75	K	107	k
12	FF	(nueva página)	44	,	76	L	108	l
13	CR	(retorno de carro)	45	-	77	M	109	m
14	SO	(desplaza afuera)	46	.	78	N	110	n
15	SI	(desplaza adentro)	47	/	79	O	111	o
16	DLE	(esc.vínculo datos)	48	0	80	P	112	p
17	DC1	(control disp. 1)	49	1	81	Q	113	q
18	DC2	(control disp. 2)	50	2	82	R	114	r
19	DC3	(control disp. 3)	51	3	83	S	115	s
20	DC4	(control disp. 4)	52	4	84	T	116	t
21	NAK	(conf. negativa)	53	5	85	U	117	u
22	SYN	(inactividad sinc)	54	6	86	V	118	v
23	ETB	(fin bloque trans)	55	7	87	W	119	w
24	CAN	(cancelar)	56	8	88	X	120	x
25	EM	(fin del medio)	57	9	89	Y	121	y
26	SUB	(sustitución)	58	:	90	Z	122	z
27	ESC	(escape)	59	;	91	[	123	{
28	FS	(sep. archivos)	60	<	92	\	124	
29	GS	(sep. grupos)	61	=	93	]	125	}
30	RS	(sep. registros)	62	>	94	^	126	~
31	US	(sep. unidades)	63	?	95	_		
127	DEL	(suprimir)						

Figura II.1 ASCII Básico

### ASCII Extendido

Se denomina ASCII extendido a cualquier juego de caracteres de 8 bits en el cual los códigos 32 a 126 coinciden con los caracteres imprimibles de ASCII, así como los caracteres comúnmente llamados “de espacio”, estos son los códigos de control de 8 a 13 (0x08 a 0x0D), ambos inclusive.

Las codificaciones de ASCII extendido utilizan además parte o la totalidad de los códigos superiores a 128 para codificar caracteres adicionales a los caracteres imprimibles ASCII.

Codificaciones “ASCII extendido” más comunes:

- Página de códigos 437 (usual en las versiones en inglés del IBM PC y MS-DOS/ Figura II.2)
- Página de códigos 850 (usual en las versiones de Europa occidental del IBM PC y MS-DOS/ Figura II.3)
- Latin-1 (ISO-8859-1) (típico de Unix y, con modificaciones, en Microsoft Windows y Macintosh / Figura II.4)

**Página de Codigos 437**




	·0	·1	·2	·3	·4	·5	·6	·7	·8	·9	·A	·B	·C	·D	·E	·F
B·	Ç U+00C7	ü U+00FC	é U+00E9	â U+00E2	ä U+00E4	à U+00E0	ã U+00E5	ç U+00E7	ê U+00EA	ë U+00EB	è U+00E8	ï U+00EF	î U+00EE	ì U+00EC	Ä U+00C4	Å U+00C5
9·	É U+00C9	æ U+00E6	Æ U+00C6	ô U+00F4	ö U+00F6	ò U+00F2	û U+00FB	ù U+00F9	ÿ U+00FF	Ö U+00D6	Ü U+00DC	ç U+00A2	£ U+00A3	¥ U+00A5	Pts U+20A7	f U+0192
A·	á U+00E1	í U+00ED	ó U+00F3	ú U+00FA	ñ U+00F1	Ñ U+00D1	a U+00AA	o U+00BA	¿ U+00BF	¬ U+2310	¬ U+00AC	½ U+00BD	¼ U+00BC	i U+00A1	« U+00AB	» U+00BB
B·	 U+2591	 U+2592	 U+2593	 U+2502	├ U+2524	┤ U+2521	┆ U+2522	┇ U+2556	┈ U+2555	┉ U+2553	┊ U+2551	┋ U+2557	┌ U+255D	┍ U+255C	┎ U+255B	┏ U+2510
C·	L U+2514	┐ U+2594	┑ U+252C	┒ U+251C	━ U+2500	┓ U+253C	└ U+255E	┕ U+255F	┖ U+255A	┗ U+2554	┘ U+2569	┙ U+2566	┚ U+2560	┛ U+2550	├ U+256C	┝ U+2567
D·	┞ U+2568	┟ U+2564	┠ U+2565	┡ U+2559	┢ U+2558	┣ U+2552	┤ U+2553	┥ U+255B	┦ U+256A	┧ U+2518	┨ U+250C	▀ U+2558	▁ U+2554	▂ U+255C	▃ U+2550	▄ U+2550
E·	α U+03B1	β U+00DF	Γ U+0393	π U+03C0	Σ U+03A3	σ U+03C3	μ U+00B5	τ U+03C4	Φ U+03A6	Θ U+0398	Ω U+03A9	δ U+03B4	∞ U+221E	φ U+03C6	ε U+03B5	∩ U+2229
F·	≡ U+2261	± U+00B1	≥ U+2265	≤ U+2264	∫ U+2920	∫ U+2921	÷ U+00F7	≈ U+2248	° U+00B0	· U+2219	· U+00B7	√ U+221A	n U+207F	² U+00B2	■ U+25A0	□ U+00A0

Figura II.2 ASCII Extendido – Página de Códigos 437

**Página de Códigos 850**




	·0	·1	·2	·3	·4	·5	·6	·7	·8	·9	·A	·B	·C	·D	·E	·F
B·	Ç U+00C7	ü U+00FC	é U+00E9	â U+00E2	ä U+00E4	à U+00E0	ã U+00E5	ç U+00E7	ê U+00EA	ë U+00EB	è U+00E8	ï U+00EF	î U+00EE	ì U+00EC	Ä U+00C4	Å U+00C5
9·	É U+00C9	æ U+00E6	Æ U+00C6	ô U+00F4	ö U+00F6	ò U+00F2	û U+00FB	ù U+00F9	ÿ U+00FF	Ö U+00D6	Ü U+00DC	ø U+00F8	£ U+00A3	Ø U+00D8	× U+00D7	f U+0192
A·	á U+00E1	í U+00ED	ó U+00F3	ú U+00FA	ñ U+00F1	Ñ U+00D1	a U+00AA	o U+00BA	¿ U+00BF	® U+00AE	¬ U+00AC	½ U+00BD	¼ U+00BC	i U+00A1	« U+00AB	» U+00BB
B·	 U+2591	 U+2592	 U+2593	 U+2502	├ U+2524	Á U+00C1	Â U+00C2	À U+00C0	© U+00A9	┆ U+2556	┇ U+2551	┈ U+2557	┉ U+255D	┊ U+00A2	┋ U+00A5	┌ U+2510
C·	L U+2514	┐ U+2594	┑ U+252C	┒ U+251C	━ U+2500	┓ U+253C	ã U+00E3	Ã U+00C3	ℒ U+255A	℞ U+2554	ℵ U+2569	℥ U+2566	Ω U+2560	℧ U+2550	ℨ U+256C	℩ U+00A4
D·	ð U+00F0	Ð U+00D0	Ê U+00CA	Ë U+00CB	È U+00C8	ı U+0131	Í U+00CD	Î U+00CE	Ï U+00CF	Ĵ U+2518	Ĳ U+250C	▀ U+2558	▁ U+2554	▂ U+00A6	▃ U+00CC	▄ U+2580
E·	Ó U+00D3	β U+00DF	Ô U+00D4	Ò U+00D2	õ U+00F5	Õ U+00D5	μ U+00B5	þ U+00FE	Ɔ U+00DE	Ú U+00DA	Û U+00DB	Ù U+00D9	ý U+00FD	Ý U+00DD	— U+00AF	´ U+00B4
F·	□ U+00A0	± U+00B1	≡ U+2017	¾ U+00BE	¶ U+00B6	§ U+00A7	÷ U+00F7	¸ U+00B8	° U+00B0	¨ U+00A8	· U+00B7	1 U+00B9	3 U+00B3	2 U+00B2	■ U+25A0	□ U+00A0

Figura II.3 ASCII Extendido - Página de Códigos 850

**Latin-1 (ISO-8859-1)**

	·0	·1	·2	·3	·4	·5	·6	·7	·8	·9	·A	·B	·C	·D	·E	·F
8·	€ U+20AC	...	, U+201A	f U+0192	" U+201E	... U+2026	† U+2020	‡ U+2021	^ U+02C6	‰ U+2030	Š U+0160	< U+2039	Œ U+0152	...	Ž U+017D	...
9·	...	\' U+2018	/' U+2019	" U+201C	" U+201D	• U+2022	— U+2013	— U+2014	~ U+02DC	™ U+2122	š U+0161	> U+203A	œ U+0153	...	ž U+017E	ÿ U+017F
A·		ı U+00A0	¢ U+00A2	£ U+00A3	¤ U+00A4	¥ U+00A5	¦ U+00A6	§ U+00A7	¨ U+00A8	© U+00A9	ª U+00AA	« U+00AB	¬ U+00AC	...	® U+00AE	¯ U+00AF
B·	° U+00B0	± U+00B1	² U+00B2	³ U+00B3	´ U+00B4	µ U+00B5	¶ U+00B6	· U+00B7	¸ U+00B8	¹ U+00B9	º U+00BA	» U+00BB	¼ U+00BC	½ U+00BD	¾ U+00BE	¿ U+00BF
C·	À U+00C0	Á U+00C1	Â U+00C2	Ã U+00C3	Ä U+00C4	Å U+00C5	Æ U+00C6	Ç U+00C7	È U+00C8	É U+00C9	Ê U+00CA	Ë U+00CB	Ì U+00CC	Í U+00CD	Î U+00CE	Ï U+00CF
D·	Ð U+00D0	Ñ U+00D1	Ò U+00D2	Ó U+00D3	Ô U+00D4	Õ U+00D5	Ö U+00D6	× U+00D7	Ø U+00D8	Ù U+00D9	Ú U+00DA	Û U+00DB	Ü U+00DC	Ý U+00DD	Þ U+00DE	ß U+00DF
E·	à U+00E0	á U+00E1	â U+00E2	ã U+00E3	ä U+00E4	å U+00E5	æ U+00E6	ç U+00E7	è U+00E8	é U+00E9	ê U+00EA	ë U+00EB	ì U+00EC	í U+00ED	î U+00EE	ï U+00EF
F·	ð U+00F0	ñ U+00F1	ò U+00F2	ó U+00F3	ô U+00F4	õ U+00F5	ö U+00F6	÷ U+00F7	ø U+00F8	ù U+00F9	ú U+00FA	û U+00FB	ü U+00FC	ý U+00FD	þ U+00FE	ÿ U+00FF

Figura II.4 ASCII Extendido – Latin-1 (ISO-8859-1)

**UTF (Unicode TransformationFormat)**

En la actualidad Unicode TransformationFormats (UTF) ofrece una cobertura mucho mayor y más sofisticada que es independiente del Sistema Operativo utilizado.

HTML utiliza un sistema más completo denominado Universal Character Set (UCS). Ha sido definido por la ISO10646 y es equivalente caracter por caracter a Unicode. Ambas normas son actualizadas regularmente para añadir nuevos caracteres.

Los Unicode TransformationFormats (UTFs) se crearon para subsanar los problemas de compatibilidad que presentaba el UCS con muchas aplicaciones y protocolos. Existen diferentes versiones, la más utilizada en la web es el UTF-8 que es compatible con ASCII siempre que no haya más de 127 caracteres. Un documento de HTML en UTF-8 continua siendo un documento ASCII/ISO-8859-1 (inglés, idiomas europeos occidentales, suahili y hawaiano) normal. Si contiene caracteres Unicode por encima de la posición 255 deben indicarse como unidades "&", por ejemplo "é" se representaría como "&eacute;". [30]

#### 2.4.5 Recomendaciones para Localizar una Aplicación:

- **Cuestiones sobre Cadenas:** El texto que aparece en el interfaz de usuario de una aplicación tiene que ser traducido. Este interfaz incluirá principalmente, menús, cuadros de diálogo y mensajes de alerta. A desarrollar una aplicación de uso internacional, hay que tener en cuenta las diferencias entre idiomas, en este caso, especialmente las diferencias relacionadas con las cadenas. La aplicación debe permitir que sea traducida a otros idiomas sin que por ello se vea afectada en su uso o efectividad. Para ello hay que tener en cuenta una serie de aspectos al escribir las cadenas que pertenecen a un interfaz.
- **Longitud de las Cadenas:** Habitualmente el texto traducido es más largo que el original, lo cual produce cambios en la interfaz de usuario. Es por ello por lo que el equipo de desarrollo de la interfaz de usuario debe tener en cuenta esta consideración en todo momento del diseño.
- **Concatenación de Cadenas:** Cuando un desarrollador necesita reducir el tamaño de una cadena, muchas veces recurre a la concatenación de cadenas. Sin embargo esto dificultará el proceso de localización de la aplicación ya que puede acarrear errores lingüísticos o como resultado se crearán frases carentes de significado. Por ejemplo en un código se puede encontrar fácilmente estas tres cadenas en inglés:
  - String 1: One after the other
  - String 2: The controls will be deleted
  - String 3: The variables will be deleted

Estas tres frases en español se podrían traducir fácilmente de la siguiente manera:

- String 1: Uno después de otro
- String 2: Los controles serán borrados
- String 3: Las variables serán borradas



En este caso, un desarrollador podría pensar en concatenar las cadenas String 3 y String 1. En inglés sería correcto y no tendría mayor problema.

Sin embargo se concatenan estas dos cadenas en español se obtiene una frase incorrecta, ya que variables en español es femenino y “uno después de otro” es masculino. Por tanto, se puede concluir que para evitar este problema se debe evitar la concatenación de cadenas, utilizándolas íntegramente en el archivo de recursos.

- **Variables dentro de las Cadenas:** Aunque lo ideal sería que las cadenas no incluyeran variables, hay veces que es necesario su uso.
- **Ordenación:** El orden alfabético varía según el idioma, así como las convenciones para secuenciar entradas en diccionarios o listines telefónicos. El uso de mayúsculas y minúsculas también son específicos de cada idioma.

#### **Archivos de cadenas y recursos:**

Diseñar una aplicación teniendo en cuenta el proceso de localización y separando los recursos de cadena del código desde el inicio del proyecto, trae mejores resultados que si se tiene que revisar la aplicación una vez finalizada para ajustarse a los requisitos de internacionalización. Los recursos de cadena deben estar en un archivo separado desde el que la aplicación los cargará durante la ejecución. Este archivo contendrá todas las cadenas localizadas, así como coordenadas o iconos. De esta manera, el equipo de localización trabajará exclusivamente en el archivo de recursos.

Las ventajas de este método son:

- Si se desarrolla una nueva versión de la aplicación, únicamente habrá que crear un nuevo archivo de recursos ya que el bloque de código no será necesario cambiarlo.
- Mayor seguridad, extrayendo los recursos únicamente localizables, es seguro que no se queda ninguna cadena localizable sin traducir y que no se traduce en exceso, es decir, que no se traduce alguna línea de código que provocaría que la aplicación dejase de funcionar. [29]

## **2.5 ¿Qué es un Framework?**

El concepto Framework se emplea en muchos ámbitos del desarrollo de sistemas software, no solo en el ámbito de aplicaciones Web. Podemos encontrar frameworks para el desarrollo de aplicaciones médicas, de visión por computador, para el desarrollo de juegos, y para cualquier ámbito que pueda ocurrírseos.

En general, con el término Framework, nos estamos refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. Un Framework se puede considerar como una aplicación genérica y configurable (a la que podemos añadirle componentes personalizados de ser necesario) para construir una aplicación concreta.

Los objetivos principales que persigue un Framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones.

### **2.5.1 Framework Web**

Podemos definirlo como un conjunto de componentes (por ejemplo clases, archivos de configuración XML, etc.) que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas Web.

Existen varios tipos de frameworks Web: orientados a la interfaz de usuario, como Java Server Faces, orientados a aplicaciones de publicación de documentos como Coocon, orientados a la parte de control de eventos como Struts y algunos que incluyen varios elementos como Tapestry.

## **2.6 Patrón de Diseño Modelo Vista Controlador (MVC)**

### **2.6.1 Introducción**

En ingeniería de software los patrones de diseño son una herramienta fundamental que nos permite una solución general y reutilizable a problemas comunes en el diseño de software. En un nivel superior existen los patrones de arquitectura que presentan un mayor alcance, ya que

en estos describe un patrón general de todo un sistema. Los patrones de arquitectura expresan esquemas fundamentales de la estructura organizacional de los sistemas de software.

El patrón MVC separa los datos de una aplicación, la interfaz de usuario y la lógica de control cada uno especializado para un conjunto de tareas específicas.

Entre los patrones de arquitectura, el MVC es uno de los mayormente usados en la actualidad. Hoy en día casi todas las herramientas de desarrollo de software soportan esta arquitectura para el desarrollo de aplicaciones.

### 2.6.2 Definición

Modelo Vista Controlador (MVC) es un principio de diseño arquitectónico que separa los componentes de una aplicación web. Esta separación ofrece más control sobre las partes individuales de la aplicación, lo cual permite desarrollarlas, modificarlas y probarlas más fácilmente, como podemos visualizar en la Figura II.5. [16]

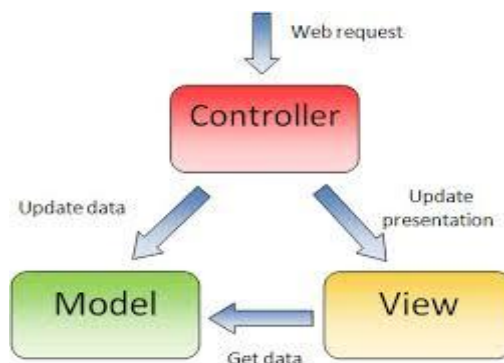


Figura II.5 Modelo Vista Controlador

### 2.6.3 Modelo

Esta es la representación específica de la información con la cual opera el sistema.

El modelo es el responsable de:

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.

- 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”.
- Lleva un registro de las vistas y controladores del sistema.
- 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.). [27]

#### **2.6.4 Vista**

Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.

- Recibir datos del modelo y mostrarlos al usuario.
- Tienen un registro de su controlador asociado (normalmente porque además lo instancia).
- 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). [27]

#### **2.6.5 Controlador**

Este responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista.

El controlador es el responsable de:

- Recibe los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
- Contiene reglas de gestión de eventos, del tipo “Si Evento A, entonces Acción B”. Estas acciones pueden suponer peticiones al modelo o a las vistas. [27]

### **2.6.6 Ventajas**

- La separación del modelo de la vista, es decir, separar los datos de la representación visual de los mismos.
- Es mucho más sencillo agregar múltiples representaciones de los mismos datos o información.
- Facilita agregar nuevos tipos de datos según sea requerido por la aplicación ya que son independientes del funcionamiento de las otras capas.
- Crea independencia de funcionamiento.
- Ofrece maneras más sencillas para probar el correcto funcionamiento del sistema.
- Permite el escalamiento de manera relativamente fácil de la aplicación en caso de ser requerido.

### **2.6.7 Desventajas**

- La separación de conceptos en capas agrega complejidad al sistema.
- La cantidad de archivos a mantener y desarrollar se incrementa considerablemente.

## **2.7 Application Services Providers (ASP)**

### **2.7.1 Introducción**

ASP fue la primera herramienta de desarrollo web de Microsoft, lanzada al mercado en 1996. Ésta primera versión de ASP se denominó Active Server Page o ASP Clásico, es la primera tecnología de Microsoft del tipo “lado de servidor” para páginas Web que se generan de manera dinámica. En enero de 2002 apareció ASP.NET (Application Services Providers) con la versión 1.0 del Framework .NET, y que es la tecnología sucesora de la tecnología Active Server Pages (ASP). ASP.NET está construido sobre el Common Language Runtime (CLR), permitiendo a los programadores escribir código ASP.NET usando cualquier lenguaje admitido por el Framework .NET.

En la actualidad el desarrollo en ASP.NET es cada vez más similar a programar una aplicación de escritorio. Es muy común hacer ésta comparación ya que se puede arrastrar componentes a

la página (WebForm) y acceder a los eventos de dichos componentes de manera muy transparente para los desarrolladores web.

## **2.7.2 Evolución de Application Service Providers**

### **2.7.2.1 Fase Inicial Evolución como Active Server Pages**

Se desarrollaba todo en uno, es decir el mismo ASP incluía todas las funcionalidades:

- Incluía código HTML y dentro se embebía el código ASP.
- Había pocos “includes” que hacían referencias a funciones muy generales o muy utilizadas.
- Se hacía muy complicado de gestionar por el tamaño que llegaban a ocupar los ficheros, en muchos casos de miles de líneas.
- Se reutilizaba poco código.

### **2.7.2.2 Primera Evolución Active Server Pages**

Se comienzan a crear clases reutilizables para centralizar la mayoría de las funciones generales:

- Conexión y operaciones con bases de datos, control de errores, control de la seguridad, validación de campos, trabajo con ficheros PDF o Excel, etc.
- Por optimización de tiempos se generaliza el uso de los “includes” con elementos de diseño comunes (cabecera, cuerpo y pie) y el código HTML restante se suele generar desde programación.

Aunque de forma muy preliminar se comienza a trabajar con un modelo modelo-controlador con aquellos elementos más generales o utilizados.

- Modelo: Clases Generales.
- Controlador: El propio ASP.

### **2.7.2.3 Segunda Evolución Active Server Pages**

Se incluye el trabajo con plantillas y con todas las clases creadas se dispone de un Framework completo de trabajo:

- Se dispone de dos tipos de plantillas las que se almacenan en memoria de cada aplicación para mejorar los tiempos de carga y las que se cargan al vuelo (nos permitía obtener plantillas de otros entornos).
- Con el framework y con las plantillas se reduce en un 90% el tamaño de los ficheros ASP. También se reducen el número de includes con elementos de diseño.
- Se compilan las clases para evitar duplicidad de includes.
- Muchos programadores comienzan a desarrollar todo con objetos, de manera que se re aprovecha mucho código y se descentraliza el papel del propio ASP a las clases que incluya.
- Al usuario acostumbrado a trabajar como en la fase inicial se le hace muy difícil centrarse en la lógica del programa y de lo que se visualiza por pantalla, porque lo primero se gestiona en las clases y lo segundo en las plantillas.

En este punto realmente se trabajaba con un sistema MVC rudimentario pero completo.

- Modelo: Clases
- Vista: Plantillas
- Controlador: El propio ASP.

### **2.7.2.4 Evolución de Active Server Pages a Application Services Providers**

Para el desarrollo de ASP.NET en su primera versión se puede decir que parte de ASP (VBScript), aunque con cambios muy significativos lo que causó que posteriormente se dejara de lado el desarrollo y mantenimiento de ASP. Esta decisión de Microsoft se debió a los buenos resultados obtenidos por ASP.NET.

- Se crea un framework básico con los objetos más utilizados en ASP, destaco algunos:
  - Trabajo con base de datos (Oracle y SQL Server)

- Gestión de errores, con avisos personalizados a los programadores
  - Gestión de Seguridad, SQLInjection, validación de campos, conversiones, etc.
  - Envíos de correos, descargas, etc.
- Se desarrolla buena parte de la lógica dentro del ASPX aunque se comienza a desarrollar con clases/objetos.

#### 2.7.2.5 Application Services Providers en la Actualidad

ASP.NET Actualmente trabaja con el modelo MVC, las diferencias están bien marcadas para cada una de las partes que componen una aplicación, como son el Modelo, la Vista y el Controlador, cuenta con soporte para tecnologías como Ajax, JQuery, JavaScript las cuales facilitan el desarrollo de Vistas con eventos.

Cada elemento de una aplicación web quedaría definido como se indica continuación:

- Modelo: Clases
- Vista: Plantillas y Masterpages
- Controlador: El propio ASPX

#### 2.7.2 Características

ASP.NET ofrece varias características importantes entre las cuales citamos las más destacadas:

- **Rendimiento.** ASP.NET es un código de Common Language Runtime compilado que se ejecuta en el servidor. A diferencia de sus predecesores, ASP.NET puede aprovechar las ventajas del enlace anticipado, la compilación just-in-time, la optimización nativa y los servicios de caché desde el primer momento. Esto supone un incremento espectacular del rendimiento antes de siquiera escribir una línea de código.
- **Compatibilidad con Herramientas de Primer Nivel.** El marco de trabajo de ASP.NET se complementa con un diseñador y una caja de herramientas muy completos en el entorno integrado de programación (Integrated Development Environment, IDE) de



Visual Studio. Los controles de servidor de arrastrar y colocar, la implementación automática son sólo algunas de las características que proporciona esta eficaz herramienta.

- **Eficacia y Flexibilidad:** Debido a que ASP.NET se basa en Common LanguageRuntime, la eficacia y la flexibilidad de toda esa plataforma se encuentra disponible para los programadores de aplicaciones Web. La biblioteca de clases de Framework .NET, la Mensajería y las soluciones de Acceso a datos se encuentran accesibles desde el Web de manera uniforme. ASP.NET es también independiente del lenguaje, por lo que puede elegir el lenguaje que mejor se adapte a la aplicación o dividir la aplicación en varios lenguajes. Además, la interoperabilidad de Common LanguageRuntime garantiza que la inversión existente en programación basada en COM se conserva al migrar a ASP.NET.
- **Simplicidad:** ASP.NET facilita la realización de tareas comunes, desde el sencillo envío de formularios y la autenticación del cliente hasta la implementación y la configuración de sitios. Por ejemplo, el marco de trabajo de página de ASP.NET permite generar interfaces de usuario, que separan claramente la lógica de aplicación del código de presentación, y controlar eventos en un sencillo modelo de procesamiento de formularios de tipo Visual Basic. Además, Common LanguageRuntime simplifica la programación, con servicios de código administrado como el recuento de referencia automático y el recolector de elementos no utilizados.
- **Facilidad de Uso:** ASP.NET emplea un sistema de configuración jerárquico, basado en texto, que simplifica la aplicación de la configuración al entorno de servidor y las aplicaciones Web. Debido a que la información de configuración se almacena como texto sin formato, se puede aplicar la nueva configuración sin la ayuda de herramientas de administración local. Esta filosofía de "administración local cero" se extiende así mismo a la implementación de las aplicaciones ASP.NET. Una aplicación ASP.NET se implementa en un servidor sencillamente mediante la copia de los archivos necesarios al servidor. No se requiere el reinicio del servidor, ni siquiera para implementar o reemplazar el código compilado en ejecución. Muestra una independencia de la

herramienta de desarrollo. Puede utilizarse para su desarrollo desde un simple editor de texto (Notepad por ejemplo), hasta las herramientas más sofisticadas y potentes (Visual Studio .NET, Web Matrix).

- **Escalabilidad y Disponibilidad.** ASP.NET se ha diseñado teniendo en cuenta la escalabilidad, con características diseñadas específicamente a medida, con el fin de mejorar el rendimiento en entornos agrupados y de múltiples procesadores. Además, el motor de tiempo de ejecución de ASP.NET controla y administra los procesos de cerca, por lo que si uno no se comporta adecuadamente (filtraciones, bloqueos), se puede crear un proceso nuevo en su lugar, lo que ayuda a mantener la aplicación disponible constantemente para controlar solicitudes.
- **Posibilidad de Personalización y Extensibilidad:** ASP.NET presenta una arquitectura bien diseñada que permite a los programadores insertar su código en el nivel adecuado.

De hecho, es posible extender o reemplazar cualquier subcomponente del motor de tiempo de ejecución de ASP.NET con su propio componente escrito personalizado. La implementación de la autenticación personalizada o de los servicios de estado nunca has ido más fácil.

- **Seguridad:** Con la autenticación de Windows integrada y la configuración por aplicación, se puede tener el control del acceso las aplicaciones. [10]

### 2.7.3 Arquitectura de ASP.NET

Como se muestra en la Figura II.6, todos los clientes Web se comunican con las aplicaciones ASP.NET a través de Servicios de Microsoft Internet Information Server (IIS). IIS descifra y, opcionalmente, realiza la autenticación de la solicitud. Si la opción *permitir anónimos* está establecida en true, no se efectúa ninguna autenticación. IIS también busca el recurso solicitado (como una aplicación ASP.NET) y, si se autoriza al cliente, devuelve el recurso correspondiente.

Además de las características de seguridad integradas de ASP.NET, una aplicación ASP.NET puede utilizar características de seguridad de bajo nivel de Framework .NET.

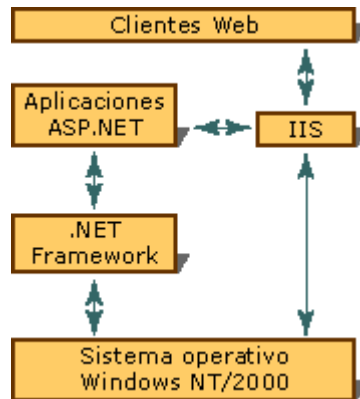


Figura II.6 Arquitectura de ASP.NET

### 2.7.3.1 Aplicaciones ASP.NET

Una aplicación ASP.NET se define como todos los archivos, páginas, controladores, módulos y código ejecutable a los que se puede llamar desde un directorio virtual y sus subdirectorios en un único servidor de aplicaciones Web. Puede utilizar cualquiera de los modelos de programación de ASP.NET dentro de las aplicaciones que defina basándose en formularios WebForms, también puede ser mediante HTML, XHTML y servicios Web. Simplemente deben coexistir en una única estructura de directorios virtuales. [5]

#### Elementos de las Aplicaciones Web ASP.NET

La creación de aplicaciones Web ASP.NET implica trabajar con muchos de los elementos que se utilizan en cualquier aplicación de escritorio o cliente-servidor. Éstos incluyen:

- **Funciones de Administración de Proyectos:** Al crear una aplicación Web ASP.NET, necesitará hacer un seguimiento de los archivos necesarios, cuáles se debe compilar y cuáles necesitan implementarse.

- **Interfaz de Usuario:** Una aplicación suele presentar información a los usuarios; en una aplicación Web ASP.NET, la interfaz de usuario se presenta en páginas de formularios *WindowsForms*, que envían los resultados a un explorador. Opcionalmente, puede crear resultado adaptado para dispositivos móviles u otros aparatos Web.
- **Componentes:** Muchas aplicaciones incluyen elementos reutilizables que contienen código para ejecutar tareas específicas. En las aplicaciones Web, puede crear estos componentes como servicios Web XML, lo que les permite aceptar llamadas de todo el Web desde una aplicación Web, otro servicio Web XML o un formulario *WindowsForms*, por ejemplo.
- **Datos:** La mayoría de las aplicaciones necesitan algún mecanismo de acceso a datos. En las aplicaciones Web ASP.NET puede utilizar cualquiera de sus proveedores de datos que forman parte del Framework.NET.
- **Seguridad, Rendimiento y otras Características de Infraestructura.** Como en cualquier aplicación, deberá implementar seguridad para evitar el uso no autorizado, probar y depurar la aplicación, ajustar su rendimiento y ejecutar otras tareas no relacionadas con la función principal de la aplicación.

La Figura II.7 proporciona información general sobre el modo en que encajan entre sí las diferentes partes de las aplicaciones Web ASP.NET y sobre cómo encajan en el contexto más amplio de .NET Framework.

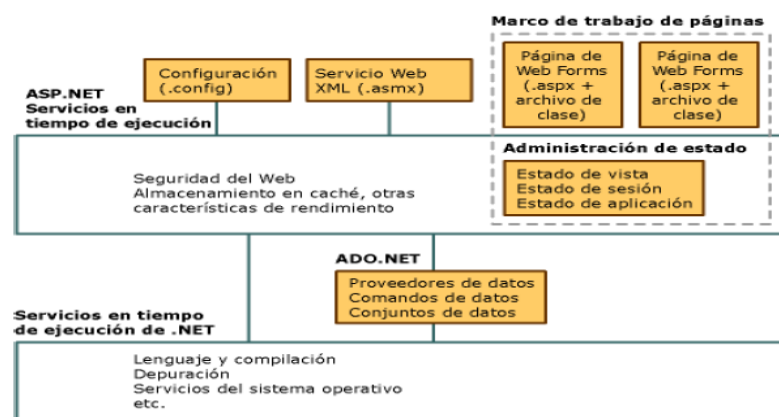


Figura II.7 Componentes de una Aplicación Web en ASP.NET

### **2.7.3.2 Internet Information Services (IIS)**

Internet Information Services (o Server), IIS, es una serie de servicios para los ordenadores que funcionan exclusivamente con Sistemas Operativos Windows ya que es de propiedad de Microsoft.

Este servicio convierte a un ordenador en un servidor de internet o Intranet, es decir que en los ordenadores que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente (servidor web).

El servidor web se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas y aplicaciones web, por ejemplo Microsoft incluye los de ASP.NET. También pueden ser incluidos los de otras herramientas de desarrollo, como PHP o Perl.

### **2.7.3.3 Framework .NET**

El Framework .NET es un entorno de ejecución de Runtime que administra aplicaciones para el Framework .NET. Incorpora Common Language Runtime, que proporciona una administración de la memoria y otros servicios del sistema y una biblioteca de clases completa, que permite a los programadores aprovechar el código sólido y confiable para todas las áreas importantes del desarrollo de aplicaciones.

El diseño del Framework .NET está enfocado a cumplir los objetivos siguientes:

- Proporcionar un entorno coherente de programación orientada a objetos, en el que el código de los objetos se pueda almacenar y ejecutar de forma local, ejecutar de forma local pero distribuida en Internet o ejecutar de forma remota.
- Proporcionar un entorno de ejecución de código que reduzca lo máximo posible la implementación de software y los conflictos de versiones.
- Ofrecer un entorno de ejecución de código que promueva la ejecución segura del mismo, incluso del creado por terceras personas desconocidas o que no son de plena confianza.

- Proporcionar un entorno de ejecución de código que elimine los problemas de rendimiento de los entornos en los que se utilizan scripts o intérpretes de comandos.
- Ofrecer al programador una experiencia coherente entre tipos de aplicaciones muy diferentes, como las basadas en Windows o en el Web.
- Basar toda la comunicación en estándares del sector para asegurar que el código de .NET Framework se puede integrar con otros tipos de código.

A continuación en la Figura II.8 se visualiza la Arquitectura del Framework .NET, posterior a ello se describirá sus principales componentes el Common Language Runtime y Class Base Library.

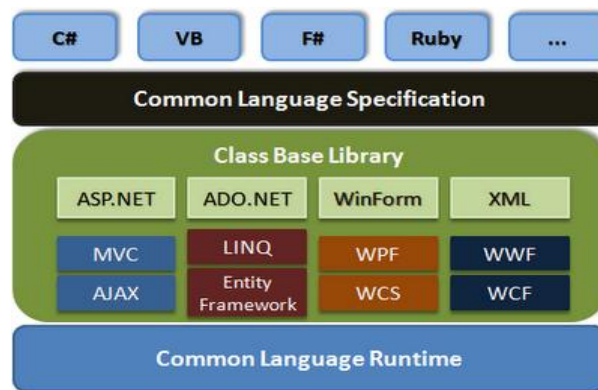


Figura II.8 Arquitectura del Framework .NET

### Common Language Runtime (CLR)

El Framework .NET está compuesto por dos componentes principales: Common Language Runtime y Class Base Library (Biblioteca de Clases) del Framework.NET. Common Language Runtime es el fundamento del Framework .NET. El motor en tiempo de ejecución se puede considerar como un agente que administra el código en tiempo de ejecución y proporciona servicios centrales, como la administración de memoria, la administración de subprocesos y la comunicación remota, al tiempo que aplica una seguridad estricta a los tipos y otras formas de especificación del código que promueven su seguridad y solidez. De hecho, el concepto de administración de código es un principio básico del motor en tiempo de ejecución. El código destinado al motor en tiempo de ejecución se denomina código administrado, a diferencia del

resto de código, que se conoce como código no administrado. La biblioteca de clases es una completa colección orientada a objetos de tipos reutilizables que se pueden emplear para desarrollar aplicaciones que abarcan desde las tradicionales herramientas de interfaz gráfica de usuario (GUI) o de línea de comandos hasta las aplicaciones basadas en las innovaciones más recientes proporcionadas por ASP.NET, como los formularios WebForms y los servicios Web XML.

El Framework .NET puede hospedarse en componentes no administrados que cargan Common Language Runtime en sus procesos e inician la ejecución de código administrado, con lo que se crea un entorno de software en el que se pueden utilizar características administradas y no administradas. En Framework .NET no sólo se ofrecen varios hosts de motor en tiempo de ejecución, sino que también se admite el desarrollo de estos hosts por parte de terceros.

Por ejemplo, ASP.NET hospeda el motor en tiempo de ejecución para proporcionar un entorno de servidor escalable para el código administrado. ASP.NET trabaja directamente con el motor en tiempo de ejecución para habilitar aplicaciones de ASP.NET y servicios Web XML, que se tratan más adelante en este tema.

Internet Explorer es un ejemplo de aplicación no administrada que hospeda el motor en tiempo de ejecución (en forma de una extensión de tipo MIME). Al usar Internet Explorer para hospedar el motor en tiempo de ejecución, puede incrustar componentes administrados o controles de WindowsForms en documentos HTML. Al hospedar el Runtime de esta manera se hace posible el uso de código móvil administrado, pero con mejoras significativas que sólo el código administrado puede ofrecer, como la ejecución con confianza parcial y el almacenamiento aislado de archivos.

En la Figura II.9 se muestra la relación de Common Language Runtime y la Biblioteca de Clases con las aplicaciones y el sistema en su conjunto. En la Figura N° se representa igualmente cómo funciona el código administrado dentro de una arquitectura mayor.

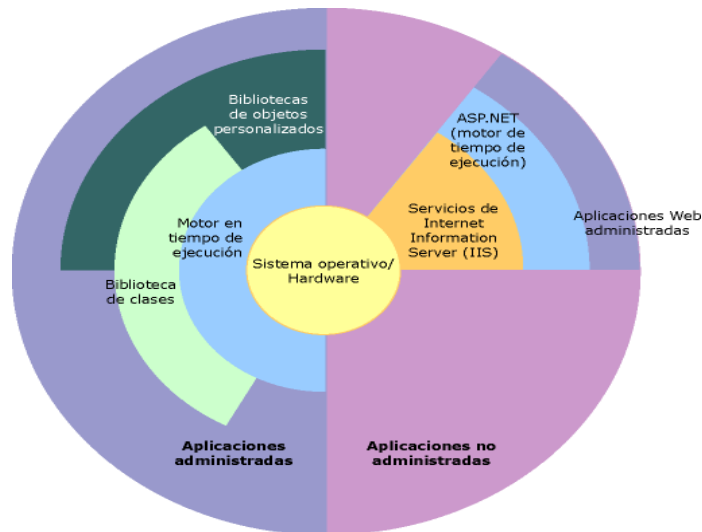


Figura II.9 Relación CLR y Biblioteca de Clases

### Características de Common Language Runtime

- Common Language Runtime administra la memoria, realiza la ejecución de subprocesos, ejecución de código, comprobación de la seguridad del código, compilación y demás servicios del sistema. Estas características son intrínsecas del código administrado que se ejecuta en Common Language Runtime.
- Con respecto a la seguridad, los componentes administrados reciben grados de confianza diferentes, en función de una serie de factores entre los que se incluye su origen (como Internet, red empresarial o equipo local). Esto significa que un componente administrado puede ser capaz o no de realizar operaciones de acceso a archivos, operaciones de acceso al Registro y otras funciones delicadas, incluso si se está utilizando en la misma aplicación activa.
- El motor en tiempo de ejecución impone la seguridad de acceso del código. Por ejemplo, los usuarios pueden confiar en que un archivo ejecutable incrustado en una página Web puede reproducir una animación en la pantalla o entonar una canción, pero no puede tener acceso a sus datos personales, sistema de archivos o red. Por ello, las características de seguridad del motor en tiempo de ejecución permiten que el software legítimo implementado en Internet sea excepcionalmente variado.



- El motor en tiempo de ejecución impone la solidez del código mediante la implementación de una infraestructura estricta de comprobación de tipos y código denominado CTS (Common Type System, Sistema de tipos común). CTS garantiza que todo el código administrado es auto descriptivo. Los diversos compiladores de lenguaje de Microsoft y de otros fabricantes generan código administrado que se atiene al CTS. Esto significa que el código administrado puede consumir otros tipos e instancias administrados, al tiempo que se exige fidelidad de tipos y seguridad de tipos estrictamente.
- El entorno administrado del motor en tiempo de ejecución elimina muchos problemas de software comunes. Por ejemplo, el motor en tiempo de ejecución controla automáticamente la disposición de los objetos, administra las referencias a éstos y los libera cuando ya no se utilizan. Esta administración automática de la memoria soluciona los dos errores más comunes de las aplicaciones: la pérdida de memoria y las referencias no válidas a la memoria.
- El motor en tiempo de ejecución aumenta la productividad del programador. Por ejemplo, los desarrolladores pueden crear aplicaciones en el lenguaje que prefieran y seguir sacando todo el provecho del motor en tiempo de ejecución, la biblioteca de clases y los componentes escritos en otros lenguajes por otros colegas. El proveedor de un compilador puede elegir destinarlo al motor en tiempo de ejecución. Los compiladores de lenguajes que se destinan a .NET Framework hacen que las características de .NET Framework estén disponibles para el código existente escrito en dicho lenguaje, lo que facilita enormemente el proceso de migración de las aplicaciones existentes.
- Aunque el motor en tiempo de ejecución está diseñado para el software del futuro, también es compatible con el software actual y el software antiguo. La interoperabilidad entre el código administrado y no administrado permite que los desarrolladores continúen utilizando los componentes COM y las DLL que necesiten.

- El motor en tiempo de ejecución está diseñado para mejorar el rendimiento. Aunque Common Language Runtime proporciona muchos servicios estándar de motor en tiempo de ejecución, el código administrado nunca se interpreta. Una característica denominada compilación JIT (Just-In-Time) permite ejecutar todo el código administrado en el lenguaje máquina nativo del sistema en el que se ejecuta. Mientras tanto, el administrador de memoria evita que la memoria se pueda fragmentar y aumenta la zona de referencia de la memoria para mejorar aún más el rendimiento.
- Por último, el motor en tiempo de ejecución se puede hospedar en aplicaciones de servidor de gran rendimiento, como Microsoft SQL Server e Internet Information Services (IIS). Esta infraestructura permite utilizar código administrado para escribir lógica empresarial, al tiempo que se disfruta del superior rendimiento de los mejores servidores empresariales del sector que puedan hospedar el motor en tiempo de ejecución.

### **Class Base Library (Biblioteca de Clases)**

La biblioteca de clases del Framework .NET es una colección de tipos reutilizables que se integran estrechamente con Common Language Runtime. La biblioteca de clases está orientada a objetos, lo que proporciona tipos de los que su propio código administrado puede derivar funciones. Esto ocasiona que los tipos de éste Framework sean sencillos de utilizar y reduce el tiempo asociado con el aprendizaje de las nuevas características de Framework .NET. Además, los componentes de terceros se pueden integrar sin dificultades con las clases de Framework.NET.

Por ejemplo, las clases de colección del Framework .NET implementan un conjunto de interfaces que puede usar para desarrollar sus propias clases de colección. Éstas se combinarán fácilmente con las clases de Framework .NET.

Como en cualquier biblioteca de clases orientada a objetos, los tipos de Framework .NET permiten realizar diversas tareas de programación comunes, como son la administración de cadenas, recolección de datos, conectividad a bases de datos y acceso a archivos. Además de

estas tareas habituales, la biblioteca de clases incluye tipos adecuados para diversos escenarios de desarrollo especializados, como en aplicaciones de escritorio, web o móvil mediante cualquiera de sus tecnologías incorporadas.

La forma de organizar la biblioteca de clases de .Net dentro del código es a través de los espacios de nombres (namespaces), donde cada clase está organizada en espacios de nombres según su funcionalidad. Por ejemplo, para manejar ficheros se utiliza el espacio de nombres System.IO y si lo que se quiere es obtener información de una fuente de datos se utilizará el espacio de nombres System.Data.

#### **2.7.3.4 Sistemas Operativos de Familia Microsoft .Net Server**

La familia Windows Server ofrece una base sólida para toda su carga de trabajo del servidor y requisitos de aplicación al tiempo que es fácil de implementar y administrar. Las nuevas herramientas de virtualización, los recursos Web, las mejoras de administración y la emocionante integración con Windows 8 ayudan a ahorrar tiempo, reducen los costos y proporcionan una plataforma para un centro de datos dinámico, en su datacenter o en la nube.[21]

#### **2.7.4 La Internacionalización en Application Services Providers**

##### **INTRODUCCIÓN**

Si su sitio web está dirigido a usuarios de diferentes partes del mundo a estos usuarios les gustaría ver el contenido de su sitio web en su propio idioma. Creación de un sitio web multilingüe no es una tarea fácil, pero sin duda permitirá a su sitio para llegar a más público. Afortunadamente, el .NET Framework ya tiene componentes que soportan diferentes idiomas y culturas.

Se puede construir una aplicación web ASP.NET MVC 3, que contiene las siguientes características:

- Se puede mostrar el contenido en diferentes idiomas.
- Se detecta automáticamente el idioma del navegador del usuario.

- Permite al usuario anular el idioma de su navegador.

La internacionalización implica la **localización** y **globalización**.

### **La Globalización en ASP**

La globalización es el proceso de diseño de aplicaciones que soportan las diferentes culturas.

Es el mecanismo mediante el cual una aplicación puede extender su uso de forma nativa con personas que vive en diferentes partes del mundo y que hablan diferentes idiomas. Por esta razón ASP.NET suministra la infraestructura para crear aplicaciones Web que se ajuste automáticamente a idiomas y formatos, conforme al usuario es privilegiado con la cultura e idioma propio. [25]

### **Localización en ASP**

La localización es el proceso mediante el cual se personaliza una aplicación para una referencia cultural y configuración regional determinada. La localización consiste principalmente en la traducción de la interfaz de usuario. Los temas en esta sección describen cómo utilizar los recursos para adaptar el contenido de páginas Web ASP.NET.

ASP.NET realiza un seguimiento de dos valores de la cultura, Culture y UICulture. El valor de la cultura determina los resultados de las funciones que dependen de la cultura, tales como la fecha, el número y formato de moneda. El UICulture determina qué recursos se van a cargar la página por el ResourceManager. El ResourceManager simplemente busca recursos específicos de la cultura que está determinada por CurrentUICulture. Cada hilo de .NET tiene los objetos CurrentCulture y CurrentUICulture. Así ASP.NET inspecciona estos valores al representar funciones que dependen de la cultura. [26]

### **ResourceManager**

Representa un administrador de recursos que proporciona un acceso más cómodo a los recursos específicos de la referencia cultural en tiempo de ejecución.

```
ResourceManager rm = newResourceManager("MyCompany.StringResources",  
typeof(Example).Assembly); [32]
```

### **CurrentUICulture**

Obtiene o establece la referencia cultural actual utilizada por el administrador de recursos para buscar recursos específicos de la cultura en tiempo de ejecución. La UICulture especifica los recursos de una aplicación necesita para apoyar la entrada del usuario y la producción, y por defecto es el mismo que la cultura del sistema operativo. [12]

### **ArchivosResources.resx**

Un archivo de recursos es un archivo XML que contiene las cadenas que desea traducir a diferentes idiomas o rutas de acceso a imágenes. El archivo de recursos contiene pares clave/valor.

ASP.NET permite crear una página que puede obtener contenido y otros datos basados en la configuración de idioma preferido para el navegador basado en la elección explícita del usuario de la lengua. Contenido y otros datos se denominan recursos y esos datos pueden ser almacenados en archivos de recursos o de otras fuentes

En la página Web ASP.NET puede configurar controles para obtener los valores de la propiedad de los recursos. Durante la ejecución, las expresiones de recursos se sustituyen por los recursos del archivo de recursos correspondiente. [6]

## **2.8 Tecnología Java 2 Enterprise Edition (Java2EE)**

### **2.8.1 ¿Qué es J2EE?**

Es una plataforma que habilita soluciones para desarrollo, uso efectivo y manejo de multicapas en aplicaciones centralizadas en el servidor.

J2EE utiliza la plataforma Java 2 SE y poder ser completa, estable, segura y la rápida plataforma Java en el ámbito de la empresa. Permite ahorrar al cliente y desarrollador de manera significativa los costos y la complejidad, porque habilita una plataforma que reduce de

desarrollo de soluciones multicapas, resultando en servicios que pueden ser desarrollados rápidamente y ampliados fácilmente.

La plataforma Java 2 Enterprise Edition es perfecta para desarrollar aplicaciones web sólidas empresariales que permiten un desarrollo acelerado.

Java es conocido por ser un lenguaje multiplataforma (su lema es: “escríbelo una vez, córralo donde sea”) por lo que hace de esta tecnología una de las utilizadas actualmente.

### 2.8.2 Arquitectura de J2EE

La especificación de J2EE define su arquitectura basándose en los conceptos de capas, containers, componentes, servicios y las características de cada uno de éstos. Las aplicaciones J2EE son divididas en cuatro capas: la capa cliente, la capa Web, la capa negocio y la capa datos. La Figura II.10 representa estas capas y las componentes relacionadas.

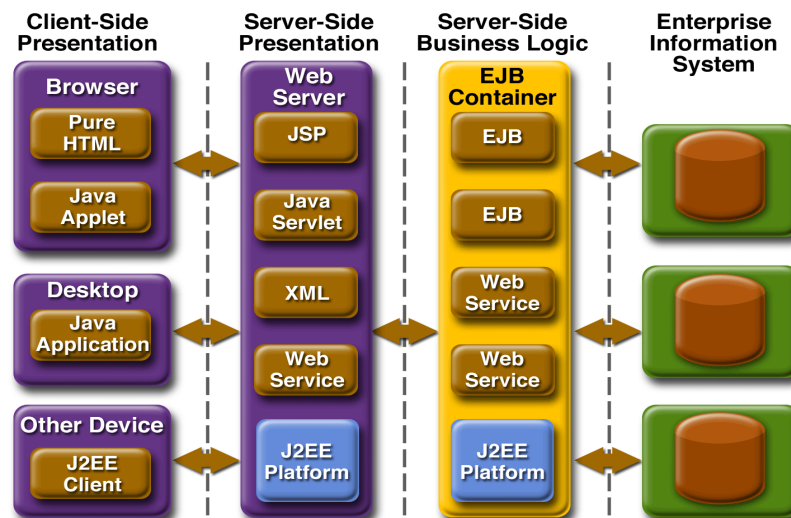


Figura II.10 Arquitectura de J2EE

#### Client-Side Presentation (Capa Cliente)

Esta capa corresponde a lo que se encuentra en el computador del cliente. Es la interfaz gráfica del sistema y se encarga de interactuar con el usuario. J2EE tiene soporte para diferentes tipos de clientes incluyendo clientes HTML, applets Java y aplicaciones Java.

### **Server-Side Presentation (Capa Web)**

Se encuentra en el servidor Web y contiene la lógica de presentación que se utiliza para generar una respuesta al cliente. Recibe los datos del usuario desde la capa cliente y basado en éstos genera una respuesta apropiada a la solicitud. J2EE utiliza en esta capa las componentes *Java Servlets* y *JavaServer Pages* para crear los datos que se enviarán al cliente.

### **Server-Side Business Logic (Capa de Negocio)**

Se encuentra en el servidor de aplicaciones y contiene el núcleo de la lógica del negocio de la aplicación. Provee las interfaces necesarias para utilizar el servicio de componentes del negocio. Las componentes del negocio interactúan con la capa de datos y son típicamente implementadas como componentes EJB.

### **Enterprise Information System (Capa de Datos)**

Esta capa es responsable del sistema de información de la empresa o Enterprise Information System (EIS) que incluye bases de datos, sistema de procesamiento datos, sistemas legados y sistemas de planificación de recursos. Esta capa es el punto donde las aplicaciones J2EE se integran con otros sistemas no J2EE o con sistemas legados.

## **2.8.3 ¿Qué Tecnologías son Incluidas en J2EE?**

### **2.8.3.1 Enterprise JavaBeans (EJB)**

Enterprise JavaBeans (EJB) es una arquitectura de componentes de servidor que simplifica el proceso de construcción de aplicaciones de componentes empresariales distribuidos en Java.

Con su utilización es posible escribir aplicaciones escalables, fiables y seguras sin escribir código de infraestructura.

La existencia de infraestructura permite un desarrollo más rápido de la parte servidora. Dado que son componentes, permiten desarrollar aplicaciones portables entre distintas plataformas (son Java) y servidores de aplicaciones (especificación estándar).

Un EJB es un componente software que se ejecuta del lado del servidor en una aplicación multicapa. Los clientes del EJB acceden a él por medio de una interfaz que esconde los detalles de implementación del componente.

Esta interfaz debe cumplir la especificación EJB:

La especificación fuerza la presencia de unos determinados métodos que permitirán al contenedor de EJBs manejar los componentes y su ciclo de vida, como se visualiza en la Figura II.11

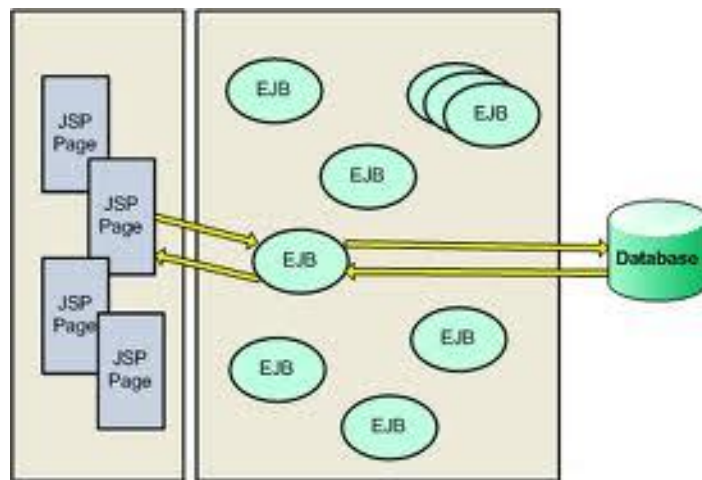


Figura II.11 Uso de los EJBs

Existen tres tipos de Enterprise JavaBeans:

- **Entity Beans:** Representan entidades de negocio y proveen acceso a datos a través de métodos. Se basan en la idea del mapeo objeto/relacional.
- **Session Beans:** Pensados para modelar procesos de negocio que son accedidos de manera síncrona.
- **Message-driven Beans:** Pensados para modelar procesos de negocio que son accedidos de manera asíncrona, permiten el uso de JMS desde EJBs



### 2.8.3.2 JavaServersPages (JSP)

JavaServer Pages es una tecnología para crear aplicaciones web, su funcionamiento se basa en scripts, que utilizan una variante del lenguaje java.

JSP es una tecnología Java que permite a los desarrolladores generar contenido dinámico para la web, en forma de documentos HTML, XML, o de otro tipo. JSP permite código Java y a algunas acciones predefinidas ser incrustadas en el contenido estático del documento web.

En JSP se escribe el texto que va a ser devuelto en la salida (normalmente código HTML) incluyendo código java dentro de él para poder modificar o generar contenido dinámicamente. La principal ventaja de JSP frente a otros lenguajes es que permite integrarse con clases Java (.class) lo que permite separar en niveles las aplicaciones web, almacenando en clases java, las partes que consumen más recursos así como las que requieren más seguridad, y dejando la parte encargada de formatear el documento html en el archivo jsp.

En la Figura II.12 siguiente se muestra el esquema de funcionamiento de JSP.

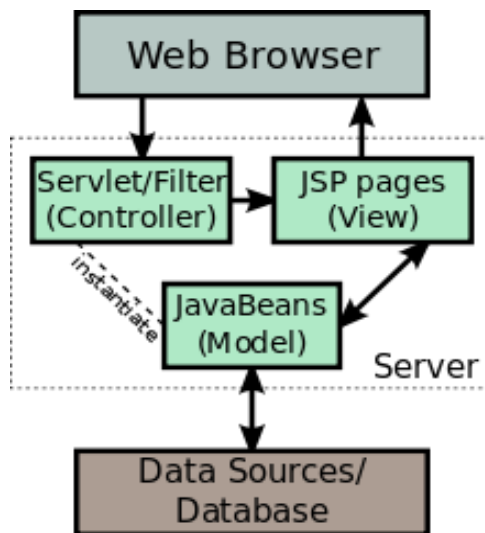


Figura II.12 Funcionamiento de JSP

### 2.8.3.3 Servlets Java Naming and Directory Interface (JNDI).

La Java Naming and Directory Interface (JNDI) es una interfaz de programación (API) que proporciona funcionalidades de nombrado y directorio a las aplicaciones escritas usando Java. Está definido para ser independiente de cualquier implementación de servicio de directorio. Así se puede acceder a una gran variedad de directorios de una forma común.

Sirve para configurar recursos compartidos para aplicaciones java organizándolos por directorio con nombres únicos. Es un servicio de nombres que permite:

- Asociar nombres con objetos.
- Permitir la búsqueda de un objeto por su nombre.

En el mercado existen diferentes servicios de directorios, por ejemplo:

- LDAP (Lightweight Directory Access Protocol)
- NDS (Novell Directory Service)
- NIS (Network Information Service)

Cada servicio tiene una forma de acceso diferente, lo que implica que si se cambia de fabricante hay que reescribir los clientes. Es por eso que se usa JNDI como capa por encima pues ofrece un acceso común a los recursos configurados.

La API JNDI permite escribir código Java que realice operaciones sobre directorios. Es una API uniforme a todos los tipos de servicios de directorios. Es similar a JDBC.

Service Provider Interface (SPI) permite acceder al servicio de directorio específico de cada fabricante. Mapea código JNDI en operaciones específicas de directorios. Es un driver JNDI. A continuación en la Figura II.13 visualizamos la arquitectura del JNDI.

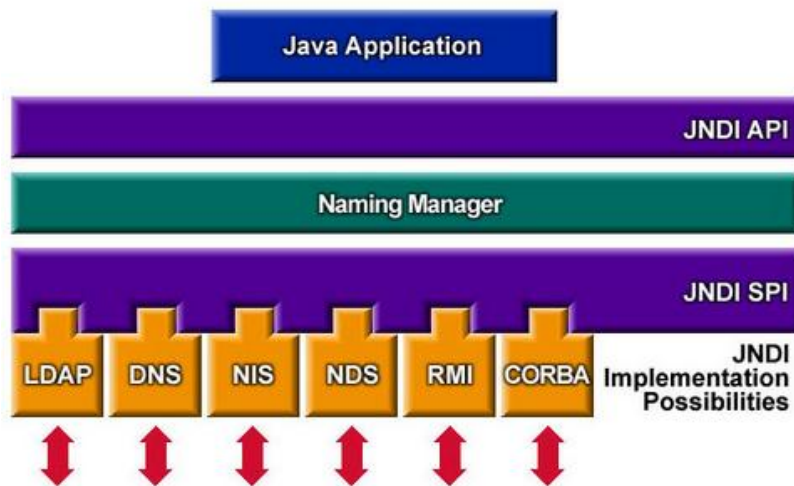


Figura II.13 Arquitectura JNDI

#### 2.8.3.4 Java Transaction API (JTA).

Java Transaction API (JTA) especifica las interfaces Java estándar entre un gestor de transacciones y las partes involucradas en un sistema de transacciones distribuidas: el gestor de recursos, el servidor de aplicaciones y las aplicaciones transaccionales.

La especificación JTA fue desarrollada por Sun Microsystems, en cooperación con los principales socios de la industria en el procesamiento de transacciones y arena sistema de base de datos.

#### 2.8.3.5 Common Object Request Broker Architecture (CORBA)

CORBA es una tecnología que oculta la programación a bajo nivel de aplicaciones distribuidas, de tal forma que el programador no se tiene que ocupar de tratar con sockets, flujos de datos, paquetes, sesiones etc. CORBA oculta todos estos detalles de bajo nivel. No obstante CORBA también brinda al programador una tecnología orientada objetos, las funciones y los datos se agrupan en objetos, estos objetos pueden estar en diferentes máquinas, pero el programador accederá a ellos a través de funciones normales dentro de su programa.

### 2.8.3.6 Java Database Connectivity (JDBC)

JDBC es el API para la ejecución de sentencias SQL. (Como punto de interés JDBCes una marca registrada y no un acrónimo. Consiste en un conjunto de clases e interfaces escritas en el lenguaje de programación Java. JDBC suministra un API estándar para los desarrolladores y hace posible escribir aplicaciones de base de datos usando un API de Java. Usando JDBC es fácil enviar sentencias SQL virtualmente a cualquier sistema de base de datos, es decir con el API JDBC, no es necesario escribir un programa que acceda a una base de datos en particular. Un único programa escrito usando el API JDBC y el programa será capaz de enviar sentencias SQL a la base de datos apropiada. Y, con una aplicación escrita en el lenguaje de programación Java, tampoco es necesario escribir diferentes aplicaciones para ejecutar en diferentes plataformas.

Existen cuatro tipos de Drivers JDBC:

- **JDBC-ODBC:**

El puente JDBC ODBC convierte las llamadas JDBC a ODBC, asignando todo el trabajo a ODBC, driver que debe ser instalado en cada cliente con sus propias librerías (APIs), siendo esto una desventaja además de ser algo lento. A continuación en el Figura II.14 visualizamos los JDBC-ODBC.

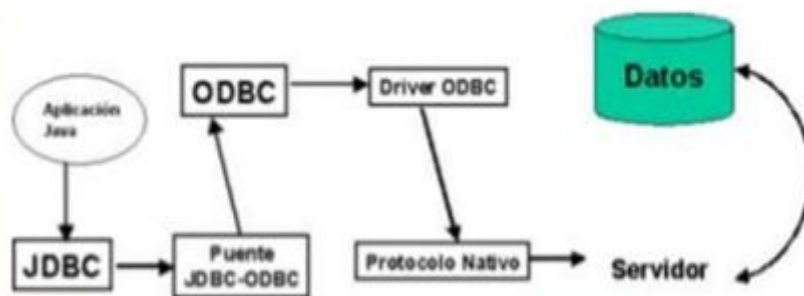


Figura II.14 JDBC ODBC

- **JDBC Parcialmente Nativo:**

Este tipo de driver convierte llamadas JDBC en llamadas del API cliente para los DBMS. Nótese que como el driver puente, este estilo de driver requiere que cierto código binario sea cargado en cada máquina cliente. A continuación en la Figura II.15 visualizamos el JDBC parcialmente nativo.



Figura II.15 JDBC Parcialmente Nativo

- **JDBC Nativo Java**

Este driver traduce llamadas JDBC al protocolo de red independiente del DBMS que después es traducido en el protocolo DBMS por el servidor. Este middleware en el servidor de red es capaz de conectar a los clientes puros (Java) a muchas bases de datos diferentes.

- **JDBC Nativo-Protocolo**

Este tipo de driver convierte llamadas JDBC en el protocolo de la red usado por DBMS directamente. Esto permite llamadas directas desde la máquina cliente al servidor DBMS y es la solución más práctica para accesos en intranets. Dado que muchos de estos protocolos son propietarios los fabricantes de bases de datos serán los principales suministradores.

## **2.8.4 JavaServer Faces (JSF)**

### **2.8.4.1 Introducción**

JavaServer Faces (JSF) es un framework, incluido dentro de la especificación J2EE, que tiene como misión facilitar la construcción y mantenimiento de aplicaciones Web en Java, siguiendo la arquitectura Modelo Vista Controlador.

JavaServer Faces fue diseñado para ser flexible, la tecnología aprovecha los conceptos de interfaz de usuario estándar y las capas web existentes, sin limitar los desarrolladores a un idioma determinado, protocolo o dispositivo cliente. Las clases de los componentes de interfaz de usuario que se incluyen en la tecnología JavaServer Faces encapsulan la funcionalidad dichos componentes, no especifica la presentación del cliente, lo que permite que los componentes de interfaz de usuario de JavaServer Faces se presenten en diversos dispositivos cliente. Al combinar la funcionalidad de los componentes personalizados de la interfaz de usuario con la renderización, se definen los atributos de representación de un componente de interfaz de usuario específica, los desarrolladores pueden construir etiquetas personalizadas para un dispositivo cliente particular. Para mayor comodidad, la tecnología JavaServer Faces proporciona un intérprete personalizado y una biblioteca de etiquetas JSP personalizada para representar a un cliente de HTML, permitiendo a los desarrolladores de aplicaciones de la Plataforma Java Enterprise Edition (Java EE) puedan utilizar la tecnología JavaServer Faces en sus aplicaciones.

La facilidad de uso es el objetivo principal, la arquitectura JavaServer Faces define claramente una separación entre la lógica de aplicación y la presentación, es fácil conectar la capa de presentación con el código de la aplicación. Este diseño permite que cada miembro de un equipo de desarrollo de aplicaciones web pueda centrarse en su tarea de desarrollo asignada, también proporciona un modelo de programación sencillo para unir todos los módulos de un sistema. Por ejemplo, desarrolladores web sin conocimientos de JavaServer Faces pueden utilizar etiquetas de componentes de interfaz de usuario para enlazar el código de la aplicación desde una página web sin necesidad de escribir ningún script.

Se desarrolló a través de la Java Community Process (comunidad dedicada a verificar las especificaciones de la tecnología JavaServer Faces), la tecnología JavaServer Faces establece el estándar para la creación de interfaces de usuario del lado del servidor. Las APIs de JavaServer Faces están siendo diseñadas para que puedan ser aprovechadas al máximo por las herramientas de desarrollo que soportan esta tecnología, lo cual hará que el desarrollo de aplicaciones web aún más fácil. [23]

### **2.8.2.2 Características**

#### **Componentes de la Interfaz de Usuario**

Habitualmente se suele decir que JSF es un framework de interfaz de usuario, lo cual es debido al amplio conjunto de componentes incluidos en el API JSF orientados a la creación de elementos gráficos. Con ellos podemos construir potentes interfaces gráficas de una manera sencilla e independiente de la tecnología utilizada para el desarrollo de la vista, es decir, no solamente podemos emplear estos componentes con páginas JSP, sino también con otras tecnologías como por ejemplo velocity

#### **Modelo de Navegación Declarativo**

Como en cualquier aplicación que siga el patrón MVC, las peticiones son dirigidas al Controlador que se encarga de analizarlas, despacharlas e invocar a la vista adecuada para la generación de la respuesta. En el caso de JSF la información referente a las posibles vistas que deben ser generadas para cada página origen de la petición se encuentra registrado en un archivo de configuración propio de JSF, llamado faces-config.xml. Esta información, conocida como reglas de navegación, es consultada por el Controlador tras el procesamiento de una petición cliente.

#### **Ejecución Basada en Eventos**

De manera similar a cómo funcionan las aplicaciones de escritorio, la programación de una aplicación JSF está basada en parte en la captura de eventos sobre la interfaz gráfica. A través de los atributos de las acciones JSF que representan a los componentes gráficos, es posible asociar estos eventos con métodos de respuesta implementados en los beans gestionados u

otras clases del Controlador, de modo que cuando se lance una petición desde la página a la aplicación (lo que se conoce como un postback) el framework será capaz de detectar los eventos producidos y ejecutar los métodos de respuesta a los mismos en el orden en que han tenido lugar.

### **2.8.2.3 ¿Hacia dónde está dirigido JSF?**

En principio JSF estaba dirigido fundamentalmente a proveer un estándar de controles para aplicaciones. Esto es, un grupo de herramientas predefinidas con comportamientos embebidos que fueran capaces de manipular datos implícitamente expresados en su implementación. Por ejemplo, una caja de texto que tuviera la capacidad de evaluar su contenido antes de que este participara en ningún otro esquema de comportamiento, o un botón que tuviera implícito cierto comportamiento en su definición.

Las tareas del manejo de procesos, comunicación de componentes o acceso a orígenes de datos se manejaba dentro de otros frameworks o ambientes de trabajo, como podían ser Struts o WebWorks. De esta manera JSF solo se encargaba de enmascarar la última porción del manejo de las entradas y salida de la capa Request/Response de la comunicación HTTP de las aplicaciones basadas en ambientes Web.

Esto no significa que JSF vaya a desplazar a otros frameworks. De hecho, el responsable del desarrollo del framework Struts ha participado activamente en el proyecto JSF desde sus inicios.

Struts y JSF pueden combinarse para dotar a la aplicación de una extensa plataforma de desarrollo en donde todas las capas de la implementación tienen su espacio definido.

Como integrar JSF con otros frameworks dirigidos para aplicaciones basadas en Web es una consideración propia de cada caso; dependerá mucho del análisis de la aplicación, de su tamaño y complejidad y de las necesidades de procesamiento que presente cada una. Podría decirse que para aplicaciones pequeñas con bajo nivel de complejidad puede usarse JSF por sí solo y mientras más complejidad presente la aplicación, se puede ir añadiendo la



capa extra de capacidad que provee el framework de Struts (u otro framework similar, actualmente se integra con Spring).

Con el paso del tiempo, JSF ha ido proveyendo de soluciones para cada una de las partes involucradas en el proceso, como por ejemplo la capacidad de establecer comunicaciones con orígenes de datos que luego pueden ser manipulados desde las interfaces desarrolladas con los componentes de la aplicación.

#### **2.8.2.4 Componentes de JSF**

Como sucede con otros frameworks, JSF está orientado a prestar soporte para la construcción de las capas Controlador y Vista de la aplicación, implementándose el Modelo con los elementos habituales para la encapsulación de la lógica de negocio: clases estándares y EJBs.

En lo que respecta a las otras dos capas de la aplicación, Controlador y Vista, la tecnología JSF aporta los siguientes elementos:

##### **Archivo de Configuración faces-config.xml**

Se utiliza para definir diversos elementos clave para el funcionamiento de la aplicación JSF, como las reglas de navegación o los beans gestionados y su ámbito de utilización.

Esta información es utilizada por el controlador FacesServlet para despachar las peticiones que llegan a la aplicación activar las vistas que deben generar las respuestas al cliente en cada caso.

##### **ServletFacesServlet**

Incluido dentro del paquete javax.faces.webapp, este servlet constituye el punto de entrada a la aplicación Web, haciendo las tareas de Front Controller. A él llegan todas las peticiones procedentes de la capa cliente, encargándose de despachar cada una de ellas en función de la información almacenada en el archivo de configuración faces-config.xml.

Como sucede con cualquier otro servlet de una aplicación, FacesServlet debe ser registrado en el archivo de configuración web.xml. Por convención suele utilizarse la expresión `"/faces/*"`

como patrón de URL asociado al servlet, lo que significa que cualquier URL relativa que comience por `"/faces/"` provocará la ejecución del servlet.

### **Beans Gestionados**

Los beans gestionados son uno de los elementos más característicos de una aplicación JSF. Se trata de simples JavaBeans que, además de ser utilizados para la encapsulación de los datos procedentes de la capa cliente, incluyen los métodos de respuesta a los eventos producidos en la capa cliente.

Buscando una comparativa con struts, un bean gestionado es una combinación de un `ActionForm` y un `Action` en una misma clase, con la particularidad añadida de que el bean gestionado es implementado mediante una simple clase POJO (Plain and Old Java Object), es decir, una clase que no necesita heredar ni implementar ninguna clase o interfaz especial.

Precisamente se les llama gestionados porque es el propio framework el que se encarga totalmente de la gestión de su ciclo de vida, es decir, de su instanciación, recuperación, rellenado de datos, invocación a los métodos de respuesta a eventos, etc. El programador simplemente tendrá que codificarlo y registrarlo en el archivo de configuración de JSF.

### **Contexto de Aplicación. La clase `facescontext`**

Las aplicaciones JSF deben ser capaces de almacenar información sobre la petición en curso. Toda esta información es accesible al programador a través de la clase `FacesContext` existente en el paquete `javax.faces.context`.

### **Acciones JSP**

Como se ha indicado con anterioridad, JSF incluye un elevado número de componentes gráficos para la construcción de potentes interfaces de usuario.

En el caso de que las vistas sean implementadas mediante páginas JSP el acceso a estos componentes se realizará a través de una serie de librerías de acciones JSP.

Concretamente, la especificación JSP incluye dos tipos de librerías de acciones:

- **La Librería html.** Esta librería proporciona además de los controles básicos html con capacidades mejoradas, una serie de componentes gráficos complejos orientados a la presentación de datos en la página. Todos los controles incluidos en esta librería se integran perfectamente con el modelo de JavaBean de Java, permitiendo vincular de manera automática las propiedades de un objeto con las de un control de la interfaz.
- **La Librería core.** Además de incluir elementos básicos para la generación de una vista JSF, esta librería proporciona unos componentes especiales conocidos como validadores y conversores que permiten realizar validaciones y conversiones de datos, respectivamente, de la información suministrada por el usuario a través de la interfaz, todo ello de manera que el programador no tenga que incluir una sola línea de código para esta tarea. [24]

#### 2.8.2.5 La Internacionalización en JSF

##### INTRODUCCION

*La internacionalización* es el proceso de diseño de una aplicación de modo que se puede adaptar a varios idiomas y regiones sin cambios de ingeniería. A veces el término internacionalización se abrevia como i18n, porque hay 18 letras entre la primera "i" y la última "n".

Un programa internacionalizado tiene las siguientes características:

- Con la adición de los datos localizados, el mismo ejecutable se puede ejecutar en todo el mundo.
- Elementos textuales, tales como mensajes de estado y las etiquetas de los componentes GUI, no están codificados en el programa. En su lugar, se almacenan fuera del código fuente y recuperan dinámicamente.
- Datos culturalmente dependientes, como las fechas y monedas, aparecen en formatos que se ajusten a la región del usuario final y el lenguaje.

- Puede ser localizada rápidamente.

La localización es el proceso de adaptación de software para una región o idioma mediante la adición de componentes específicos de la localidad y la traducción del texto. El término localización es a menudo abreviado como l10n, ya que hay 10 letras entre la "l" y la "n".

La principal tarea de la localización es la traducción de los elementos de la interfaz de usuario y la documentación. La localización implica no sólo el cambio de la interacción lingüística, sino también otros cambios relevantes tales como la visualización de los números, fechas, moneda, etc. Otros tipos de datos, tales como sonidos e imágenes, pueden requerir la localización si son culturalmente sensibles. El mejor internacionalizado una aplicación, más fácil es localizar para un idioma en particular y el esquema de codificación de caracteres.

### **¿Cómo Configurar el Idioma en JSF?**

Un programa internacionalizado puede mostrar información de manera diferente en todo el mundo. Por ejemplo, el programa mostrará diversos mensajes en París, Tokio y Nueva York. Si el proceso de localización ha sido ajustado, el programa mostrará diversos mensajes en Nueva York y Londres para dar cuenta de las diferencias entre Inglés americano y británico. ¿Cómo un programa internacionalizado puede identificar el idioma y la región apropiada de sus usuarios finales?, la respuesta es hacer referencia al objeto Locale de los objetos.

El objeto Locale es un identificador para una combinación particular de la lengua y de la región. Si una clase varía su comportamiento en función de Locale, se dice que es *sensible a la localidad*. [1]

### **Aislamiento de Datos Específicos de la Localización con JSF**

Datos específicos de entorno nacional deben adaptarse de acuerdo con las convenciones del lenguaje del usuario final y la región. El texto que muestra una interfaz de usuario es el ejemplo más obvio de los datos específicos de la localidad. Por ejemplo, una aplicación con un botón Cancelar en los EE.UU. tendrá un botón Berlín en Alemania. En otros países, este botón no tendrá otras etiquetas. Obviamente, usted no desea codificar esta etiqueta del botón. ¿No sería

agradable si usted podría conseguir automáticamente la etiqueta correcta para una determinada localización . Afortunadamente se puede aislar los objetos específicos de la localidad en un ResourceBundle. [4]

### **Formato con JSF**

Dado que los usuarios finales pueden ver estos elementos de datos, su formato debe ajustarse a varias convenciones culturales. [21]

### **Trabajar con Texto en JSF**

Casi todos los programas con interfaces de usuario manipular texto. En un mercado internacional del texto en pantalla los programas deben ajustarse a las reglas de idiomas de todo el mundo. El lenguaje de programación Java proporciona una serie de clases que le ayudan a manejar el texto de forma independiente de la Localidad. [33]

### **Componentes que hacen Posible la Internacionalización en JSF**

La internacionalización de aplicaciones web a hace posible gracias al objeto *Locale* de los objetos y a los ResourceBundle o archivos properties.

### **Resource Bundle:**

Es un paquete de recursos es un archivo filename.properties archivo que contiene datos específicos de la localización.

Es una manera de internacionalizar una aplicación Java al hacer el código independiente de la Localidad. [31]

### **Beneficios del uso de Paquetes de Recursos (Resource Bundle)**

Extracción sensible a la localidad objetos como cadenas de código (en lugar de codificar ellos) significa que la aplicación puede manejar múltiples lugares sin tener que escribir código diferente para cada localidad. También significa que los traductores pueden hacer frente a sólo el texto traducible y no el código de programación. Por lo tanto, el uso de paquetes de recursos mejora la internacionalización y la localización de proceso de un producto de software. [9]

### **Objeto Locale:**

El objeto *Locale* representa una región geográfica, política o cultural. Una operación que requiere *Locale* para llevar a cabo su tarea se llama *sensible a la localidad* y utiliza *Locale* para adaptar la información para el usuario. Por ejemplo, mostrar un número es una operación de la *sensible a la localidad* número debe tener el formato de acuerdo a las costumbres y convenciones del país del usuario nativa, región o cultura. [28]

### **2.8.2.6 PrimeFaces**

PrimeFaces es un conjunto de componentes para JSF de código abierto con varias extensiones. También se debe mencionar que PrimeFaces es una de las tecnologías más usadas en la actualidad ya que cuenta con una gran variedad de componentes, superando en el doble o triple en la cantidad de componentes a tecnologías similares como IceFaces y RichFaces. Su facilidad de uso y el tipo de licencia (open source) han hecho que PrimeFaces se constituya en fuerte tendencia en el ámbito del desarrollo web.

### **Características**

Conjunto de componentes Enriquecidos (HTML Editor, diálogos, Autocompletar, Gráficos y muchos más).

- Capacidad de Ajax basado en las APIS de Ajax en el estándar JSF 2.0.
- Ligeros, un archivo jar, sin configuración y sin dependencias requeridas.
- Kit de interfaz de usuario móvil para crear aplicaciones web móviles para dispositivos de mano.
- El Framework permite la configuración del aspecto del Marco de la Aplicación con más de 35 temas incorporados, también se cuenta con la herramienta visual de diseño temas.
- Amplia documentación.

- Cuenta con una gran comunidad de usuarios activos.
- Desarrollado con "pasión" por desarrolladores de aplicaciones para desarrolladores de aplicaciones. [23]

## **CAPITULO III**

### **ANÁLISIS COMPARATIVO DE LA INTERNACIONALIZACIÓN DE APLICACIONES WEB CON ASP Y JSF**

#### **3.1 Introducción**

Debido al auge de las Aplicaciones Web Internacionalizadas cada vez más creciente dentro del área de Desarrollo de Software, se han ido posicionando en un lugar muy importante dentro de ésta área ya que el acceso a la información en la actualidad es más globalizado gracias al internet.

Al momento de empezar con el desarrollo de un sistema informático uno de los puntos más importantes a tomar en cuenta por los desarrolladores es la tecnología a utilizar para dicho cometido, ya que ésta determinará el tiempo, costo, y todos los aspectos a tener en cuenta dentro del desarrollo del sistema.

En el presente capítulo se realizará el análisis comparativo de las Aplicaciones Web Internacionalizadas con ASP y JSF lo cual consistirá en determinar que tecnología es la más apropiada para la implementación de dichas aplicaciones web.



Las personas dedicadas al desarrollo de Aplicaciones Web deben decidir de la manera más responsable y ética al momento de seleccionar la tecnología a utilizar, esto debe realizarse a través de un análisis de criterios y/o parámetros de comparación que permitan cumplir de la mejor manera con los requerimientos del usuario así como con las especificaciones técnicas de una aplicación web.

### 3.2 Parámetros de Comparación

A continuación en la Tabla III.1 visualizamos los parámetros de comparación que se utilizará en la comparación entre ASP y JSF:

Tabla III.1 Parámetros e Indicadores

PARAMETRO	INDICADOR	UNIDAD	PONDERACIÓN	TOTAL
Rendimiento	I1: Request del Servidor	Request/Seg	25%	60%
	I2: Kb Procesados /Segundo	Kb/Seg	20%	
	I3: Tiempo de Respuesta	Segundos	15%	
Tiempo de Desarrollo	I4: Tiempo de Diseño de Base de Datos	Días	15%	40%
	I5: Tiempo de Desarrollo en Codificación	Días	25%	

Fuente: <http://www.infor.uva.es/~miquelv/eesi/mat/01.2-Introduc.pdf>, elaborado por: Fernando Yaulema Rojas.

Cabe resaltar que los indicadores I4 e I5 fueron considerados partiendo de una línea base la cual fue tomada del tiempo de desarrollo de los prototipos.

#### 3.2.1 Definición de Parámetros:

##### RENDIMIENTO:

- **Request del Servidor/Segundo:** Es la cantidad de request que el servidor procesa por segundo.
- **KB Procesados /Segundo:** Cantidad de Kb que el servidor procesa por segundo.
- **Tiempo de Respuesta:** Tiempo de respuesta para los requests.

##### TIEMPO DE DESARROLLO:

- **Tiempo de Diseño de Base de Datos:** Es la cantidad de días utilizados para el análisis, diseño e implementación de la base de datos.
- **Tiempo de Desarrollo en Codificación:** Es la cantidad de días utilizados para la codificación (Prototipos).

### 3.2.2 Escala de Medición

A continuación en la Tabla III.2 se visualiza la escala de medición a utilizar, ésta servirá de apoyo para asignarle un valor a cada indicador:

Tabla III.2 Escala de Medición

VALOR	DESCRIPCIÓN	RANGOS				
		I1	I2	I3	I4	I5
1	No Satisfactorio	>151	<50	>6500	>60	>60
2	Poco Satisfactorio	101-150	51-100	5001-6500	56-60	46-60
3	Medianamente Satisfactorio	51-100	100-150	3500-2000	46-55	31-45
4	Satisfactorio	0-50	>150	<2000	45	30

Fuente:<http://www.infor.uva.es/~miguelv/eesi/mat/01.2-Introduc.pdf>, Elaborado por: Fernando Yulema Rojas.

### 3.3 Instrumentos de Medición

Los instrumentos de medición que se utilizaron son los siguientes:

- **Badboy:** Es una herramienta de gran alcance diseñada para ayudar en la prueba y en el desarrollo de aplicaciones. Permite efectuar el testeado de la Web, con docenas de características incluyendo una interfaz simple, fácil e intuitiva, mediante los métodos de captura y repetición, siendo una gran ayuda para la prueba de carga de gran alcance, informes detallados, gráficos, etc.

Badboy trae embebido el navegador Internet Explorer de Microsoft, monitorizando y controlando las acciones que se producen.

Esto permite:

- Realizar capturas de los parámetros del CGI, las páginas y los framesets peticionados.
- Modificar dichas capturas y volver a ejecutarlas en cualquier momento automáticamente.
- Grabar las capturas como scripts y compartirlos dentro del entorno de trabajo.
- Recoger las estadísticas del funcionamiento mientras se ejecuta una prueba.
- Realizar pruebas de regresión de áreas completas de sitios Web complejos con un solo clic.

Badboy incluye una potente funcionalidad para permitir crear scripts, de una manera sencilla, con las navegaciones realizadas por sitios Web complejos sin la intervención del usuario. Una vez que estén creados los scripts, éstos se pueden compartir para conseguir un mayor aumento de la productividad.

- **JMeter:** Es un proyecto de Apache que puede ser utilizado como una herramienta de prueba de carga para analizar y medir el desempeño de una variedad de servicios, con énfasis en aplicaciones web. JMeter es una herramienta de testing cuyas funcionalidades se pueden resumir en tres:
  - Diseñar un testplan, esto es, generar un fichero .jmx
  - Ejecutar un testplan
  - Ver de distintas formas los resultados de la ejecución de un testplan (vía listeners)

Para diseñar un testplan, JMeter dispone de una interfaz GUI a modo de diseñador, en la que el tester puede ir agregando componentes de manera visual, y ejecutar los componentes agregados, viendo el resultado. Una vez finalizado el diseño del testplan, la herramienta permite grabar este como un fichero .jmx.

La propia herramienta permite ejecutar un fichero .jmx previamente generado, vía línea de comandos o vía la propia interfaz GUI. La ejecución de un fichero .jmx realiza peticiones contra la aplicación objetivo a testear (peticiones del tipo que se hayan especificado al generar el fichero .jmx, JMeter dispone de la posibilidad de generar muchos tipos de peticiones: HTTP, FTP, LDAP). Para cada petición ejecutada, JMeter recopila ciertos datos. Además, en el fichero .jmx se puede especificar qué número de usuarios de cada tipo ejecuta las peticiones contra la aplicación, es decir, el .jmx simula una o más comunidades de usuarios (roles) trabajando contra la aplicación objetivo.

Los datos generados por la herramienta para cada petición se procesan o bien con un tipo de componente que proporciona la interfaz GUI llamados listeners, o bien con herramientas externas. Los listeners permiten ver los resultados de una o más ejecuciones de múltiples maneras (cada listener de una manera).

### **3.4 Prototipos de Prueba:**

Para las pruebas se desarrolló un prototipo para el Módulo de Adquisiciones de similares características en ASP y JSF.

El prototipo permite ingresar ítems o productos al PAC, debiendo elegir la actividad (escuela, facultad, departamento, etc.), la cantidad anual, la unidad de medida del producto e ingresar los períodos en los cuales se desea ejecutar la compra. Luego de ingresar el nuevo ítem se mostrará en un listado donde se encuentran todos los productos de la actividad seleccionada.

A continuación se describe cada uno de los prototipos:

#### **3.4.1 Prototipo en JSF**

**Objetivo:** El prototipo del Módulo de Adquisiciones se desarrolló con el fin de ayudar a determinar el rendimiento y el tiempo de desarrollo de dicho prototipo en JSF.

A continuación en la Figura III.1 se visualiza la pantalla de ingreso del PAC la cual se desarrolló en el prototipo de JSF.

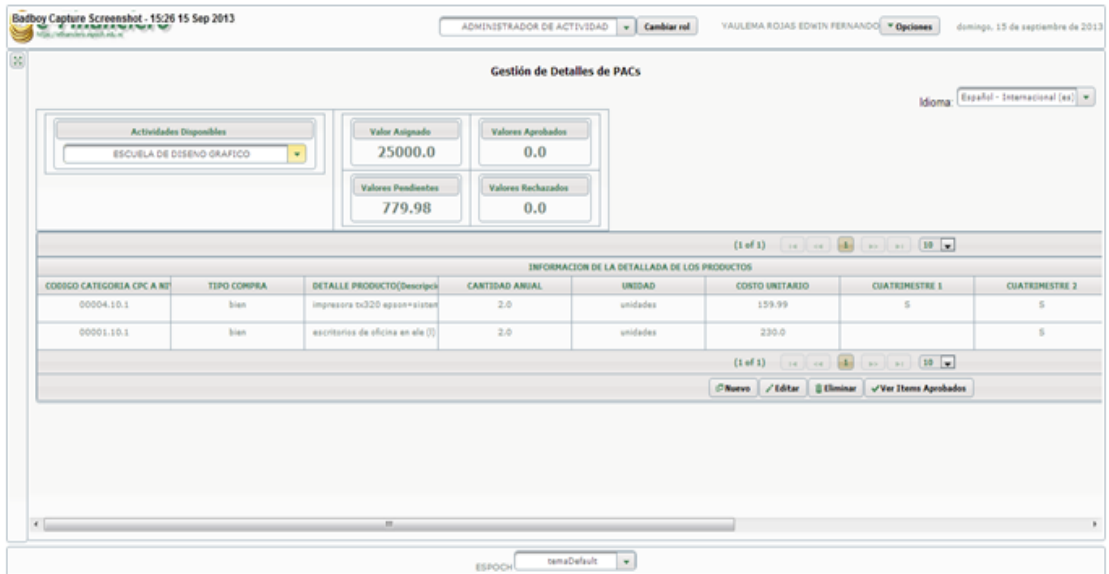


Figura III.1 Captura del Prototipo de JSF

### Hardware:

- Laptop Core I7.
- Procesador: 1.60 GHz
- RAM: 4GB
- Tarjeta de Vídeo: 1GB
- Disco Duro de 500 GB

### Software

- NetBeans 7.3.
- JavaServer Faces 2.1
- PrimeFaces 3.5.
- PgAdmin.
- PostgreSQL 9.1.
- Glassfish Server 3.1.

- Mozilla Firefox

### 3.4.2 Prototipo en ASP

**Objetivo:** El prototipo del Módulo de Adquisiciones se desarrolló con el fin de ayudar a determinar el rendimiento y el tiempo de desarrollo de dicho prototipo en ASP.

A continuación en la Figura III.2 se visualiza la pantalla de ingreso del PAC la cual se desarrolló en el prototipo de ASP.

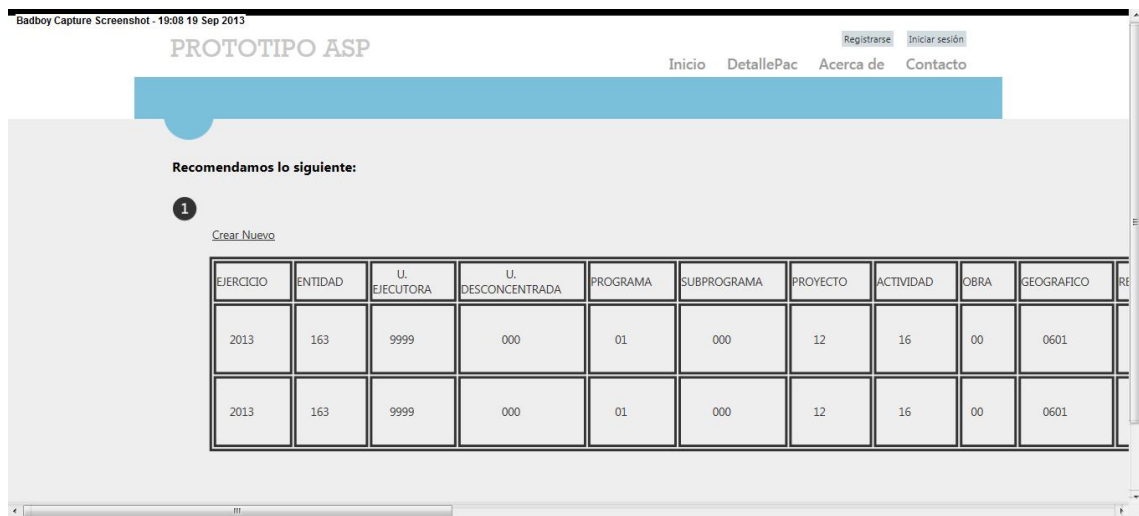


Figura III.2 Captura del Prototipo de ASP

#### Hardware:

- Laptop Core I7.
- Procesador: 1.60 GHz
- RAM: 4GB
- Tarjeta de Vídeo: 1GB
- Disco Duro de 500 GB

#### Software:

- Microsoft Visual Studio 2012.
- ASP.NET

- Visual C# 2012
- pgAdmin III
- PostgreSQL 9.1
- Internet Information Services
- Mozilla Firefox

### 3.5 Procedimiento

A continuación se describe los pasos necesarios a seguir para obtener los parámetros para la comparación:

- a) Con el software *Badboy* se captura la navegación del Prototipo y los parámetros que utiliza la aplicación en cada petición.
- b) Una vez que se inicia el proceso de captura de la secuencia de navegación se exporta a un archivo con formato "*jmx*", el cual contendrá todos los datos necesarios para utilizarlo en Software *JMeter*.
- c) Ejecutamos el software *JMeter* y luego se importamos el archivo con formato "*jmx*".
- d) Se ingresa la cantidad de usuarios (en este caso será 400) que se desea simular y el Período de Subida (en segundos) o intervalo de tiempo en que se lanzará cada hilo (en este caso será 1Seg.).
- e) Agregamos las opciones "Reporte de Resumen" , "Graficos de Resultados" y "Ver Resultado en Árbol" para observar como avanza la prueba rendimiento del Prototipo.Estas opciones nos permiten visualizar las variables utilizadas en la prueba de rendimiento del prototipo.
  - i. **Reporte de Resumen:** El reporte de resumen nos permite visualizar el resultado de la evaluación de los request de manera numérica en una matriz.

- ii. **Figura de Resultados:** La Figura de resultados nos permite visualizar el resultado de la evaluación de los request en forma gráfica.
- iii. **Árbol de Resultados:** El árbol de resultados permite observar cada uno de los requests con cada una de sus acciones o pantallas a las que accedieron con JMeter.

### 3.6 Análisis de Resultados

A continuación se visualiza los parámetros con cada uno de sus indicadores.

#### Rendimiento:

El rendimiento es una medida o cuantificación que se utiliza para evaluar el funcionamiento de un sistema ya sea tanto en hardware como en software. El rendimiento se puede evaluar a través de varios indicadores como: la capacidad de respuesta, velocidad de procesamiento, tiempo de respuesta, entre otros.

A continuación en la Figura III.3 y Figura III.4 visualizamos los indicadores con sus valores luego de realizar las pruebas de carga a los prototipos, los cuales se utilizan en los indicadores I1 e I2 del Parámetro Rendimiento:

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Media de Byt...
http://localhost/FinancieroEspoch_Web/faces/seg...	800	418	4	1123	318,54	0,00%	181,8/sec	506,37	2652,5
http://localhost/FinancieroEspoch_Web/faces/seg...	800	598	5	1171	367,88	0,00%	147,4/sec	91,09	633,0
http://localhost/FinancieroEspoch_Web/faces/seg...	2800	765	8	1583	325,72	14,29%	240,6/sec	231,44	985,1
Total	4400	672	4	1583	358,96	9,09%	340,5/sec	419,13	1260,6

Figura III.3 Reporte de Resumen JSF

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Media de Bytes
http://localhost/Vistas/DetalleP...	400	26104	818	57439	11622,47	9,25%	6,7/sec	562,56	85642,3
http://localhost/Vistas/DetalleP...	2400	4415	7	35012	8360,05	83,33%	44,4/sec	461,20	10645,2
Total	2800	7514	7	57439	11696,35	72,75%	43,9/sec	914,85	21359,1

Figura III.4 Reporte de Resumen ASP

- **I1:** Request del Servidor, Request/Segundos



Como visualizamos en la Tabla III.3 el valor de los Request/Segundo es de 340 en JSF y de 43,9 en ASP notando que el valor de JSF supera al valor de ASP. La evaluación de los valores del indicador se lo realizo en función de la Tabla III.3.

Tabla III.3 Request del Servidor

PROTOTIPOS	VALOR	EVALUACIÓN
JSF	340,5	4
ASP	43,9	1

Elaborado por: Fernando Yaulema Rojas

En la Figura III.5 visualizamos los valores de este indicador en forma gráfica:

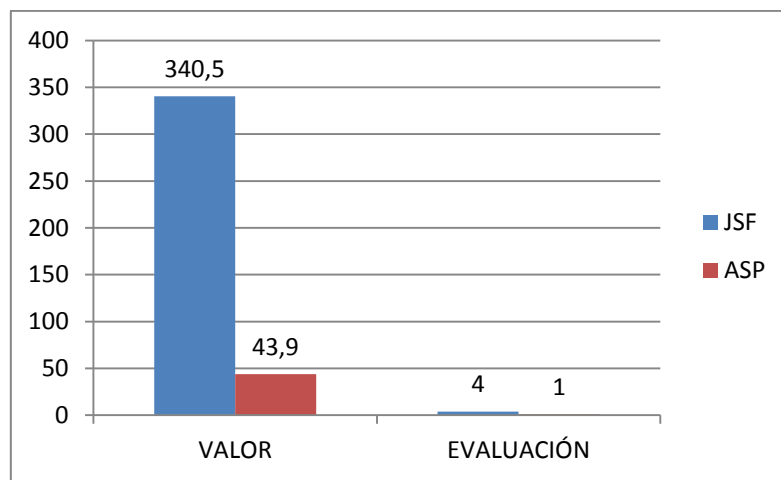


Figura III.5 Request del Servidor

- **I2: KB Procesados, Kb/Seg.** En la Tabla III.3 visualizamos los valores de este indicador:

Como visualizamos en la Tabla III.4 el valor de los Kb Procesados/Segundo es de 419,13 en JSF y de 914,85 en ASP notando que el valor de ASP supera al valor de JSF. La evaluación de los valores del indicador se lo realizo en función de la Tabla III.1.

Tabla III.4 Kb Procesados/Segundo

PROTOTIPOS	VALOR	EVALUACIÓN
JSF	419,13	3
ASP	914,85	4

Elaborado por: Fernando Yaulema Rojas

En la Figura III.6 visualizamos los valores de este indicador en forma gráfica:

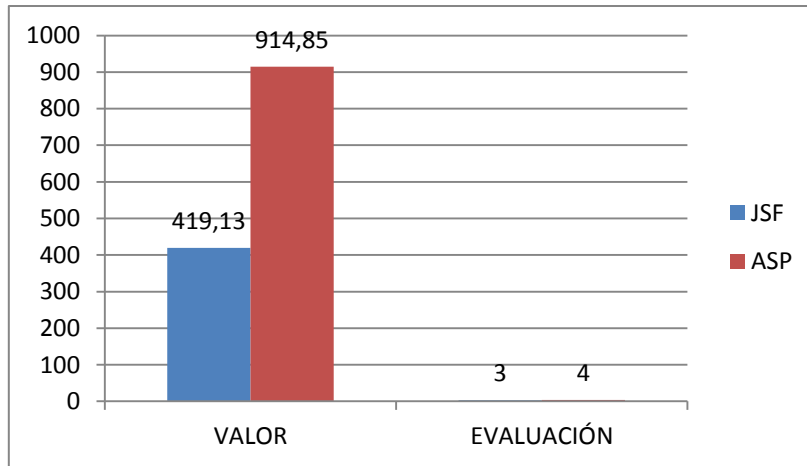


Figura III.6 Kb Procesados/Segundo

- **I3:** Tiempo de Respuesta, Milisegundos. En la Tabla III.4 visualizamos los valores de este indicador.

A continuación en la Figura III.7 y la Figura III.8 visualizamos el indicador **I3** con sus valores luego de realizar las pruebas de carga a los prototipos del Parámetro Rendimiento:

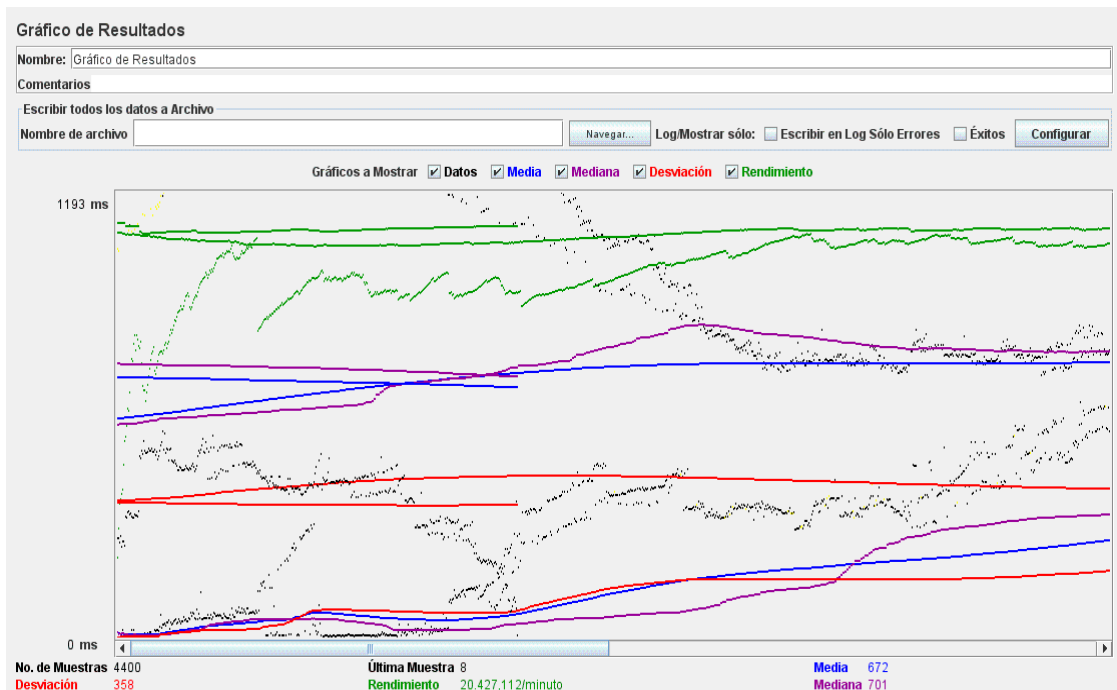


Figura III.7 Valores - Figura de Resultados JSF

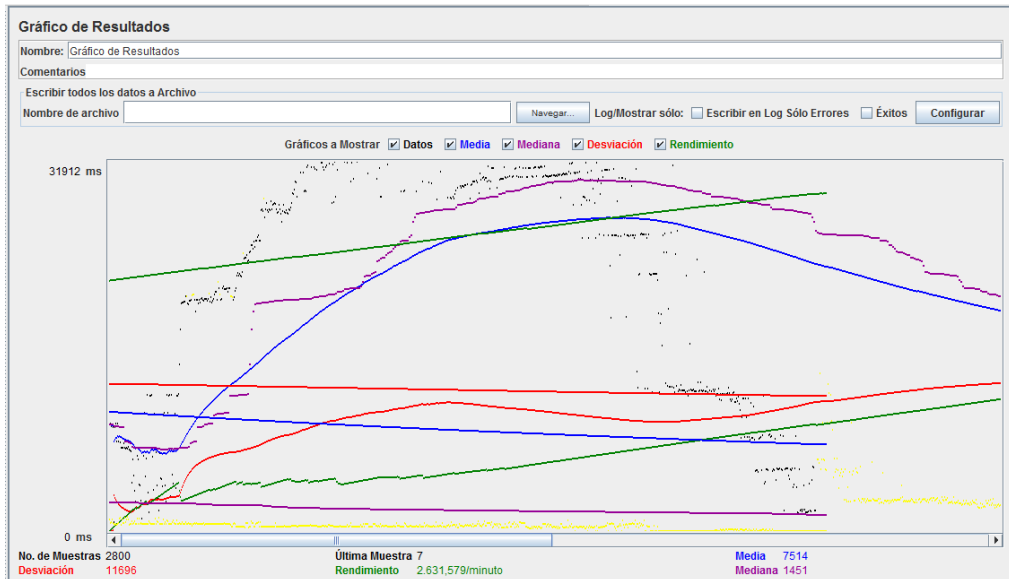


Figura III.8 Valores - Figura de Resultados ASP

Como visualizamos en la Tabla III.5 el valor del Tiempo de Respuesta, Milisegundos es de 1193 en JSF y de 31912 en ASP notando que el valor de JSF supera al valor de ASP ya que el tiempo de respuesta en JSF es mayor. La evaluación de los valores del indicador se lo realizo en función de la Tabla III.1.

Tabla III.5 Tiempo de Respuesta

PROTOTIPOS	VALOR	EVALUACIÓN
JSF	1193	4
ASP	31912	1

Elaborado por: Fernando Yaulema Rojas

En la Figura III.9 visualizamos los valores de este indicador en forma gráfica:

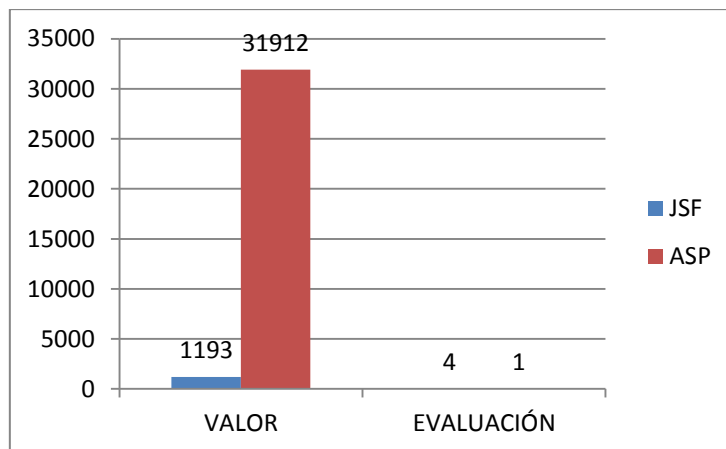


Figura III.9 Tiempo de Respuesta

**Tiempo de Desarrollo:**

El parámetro tiempo de desarrollo se refiere al tiempo empleado para realizar las tareas como que intervienen en el desarrollo de software en general.

Para medir éste parámetro se ha utilizado el indicador Diseño de Base de Datos y el indicador Tiempo de Desarrollo.

A continuación visualizamos el detalle de los indicadores con sus valores y su evaluación respectiva (de acuerdo a la Tabla III.1):

- **I4:** Tiempo de Diseño de Base de Datos, Días. En la tabla N° 6 visualizamos los valores de este indicador:

Como visualizamos en la Tabla III.6 el valor del Tiempo de Diseño de Base de Datos, días es de 45 tanto para JSF como para ASP. La evaluación de los valores del indicador se lo realizo en función de la Tabla III.1.

Tabla III.6 Diseño de Base de Datos

PROTOTIPOS	VALOR	EVALUACIÓN
JSF	45	4
ASP	45	4

Elaborado por: Fernando Yaulema Rojas.

En la Figura III.10 visualizamos los valores de este indicador en forma gráfica:

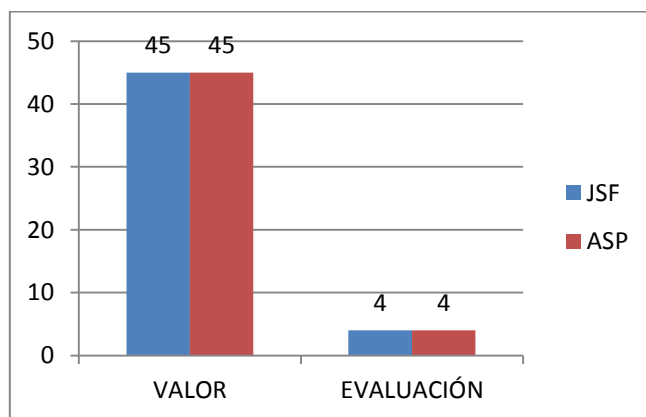


Figura III.10 Diseño de Base de Datos

- **I5:** Tiempo de Desarrollo en Codificación, Días. En la tabla N° 8 visualizamos los valores de este indicador.

Como visualizamos en la Tabla III.7 el valor del Tiempo de Desarrollo, días es de 30 en JSF y de 45 en ASP notando que el valor de JSF supera al valor de ASP ya que el número de días utilizados en JSF es menor. La evaluación de los valores del indicador se lo realizo en función de la Tabla III.1.

Tabla III.7 Tiempo de Desarrollo en Codificación

PROTOTIPOS	VALOR	EVALUACIÓN
JSF	30	4
ASP	45	3

Elaborado por: Fernando Yaulema Rojas

En la Figura III.11 visualizamos los valores de este indicador en forma gráfica:

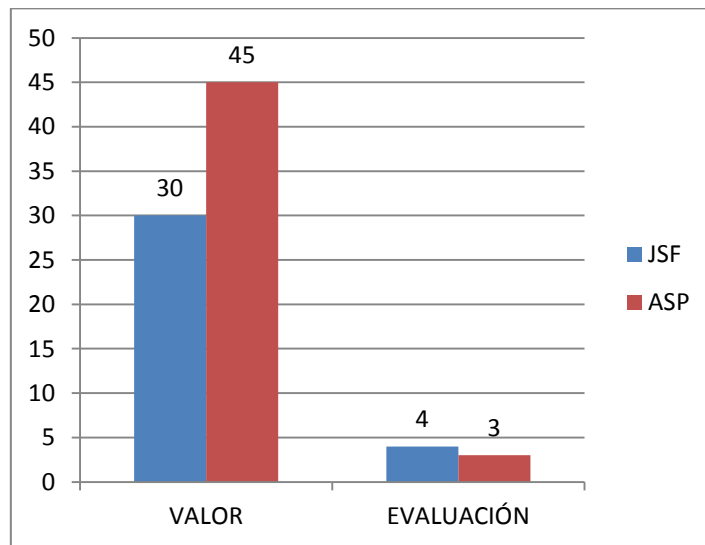


Figura III.11 Tiempo de Desarrollo en Codificación

### INTERPRETACIÓN:

#### Rendimiento:

En la Tabla III.8 visualizamos los valores para los indicadores del parámetro rendimiento, en esta tabla se visualiza los indicadores con sus valores respectivos a manera de resumen.

Tabla III.8 Valores para los Indicadores del Rendimiento

PROTOTOIPOS	I1	I2	I3
JSF	340,5	419,13	1193
ASP	43,9	914,85	31912

Elaborado por: Fernando Yaulema Rojas.

Luego de obtener los valores de todos los indicadores debemos asignarlos una valoración de acuerdo a la Tabla III.1, ya que debemos analizar dichos valores de forma cualitativa.

En la Figura III.12 visualizamos los valores de este indicador en forma gráfica:

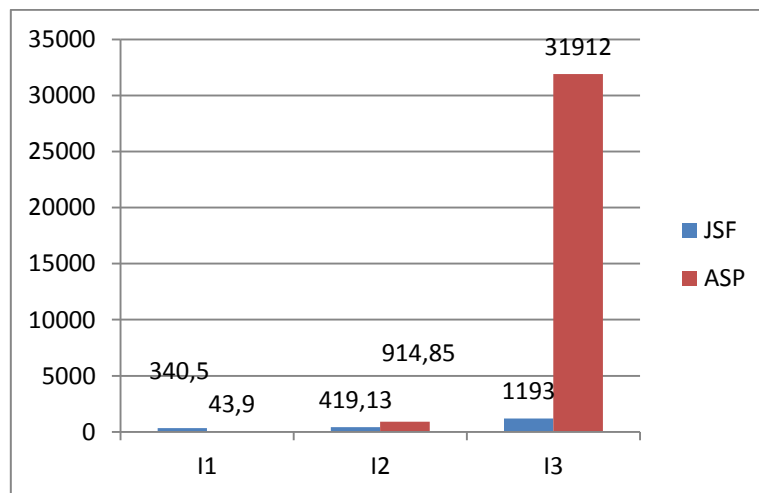


Figura III.12 Valores para los Indicadores del Rendimiento

**Tiempo de Desarrollo:**

En la Tabla III.9 visualizamos los valores para los indicadores del tiempo de desarrollo, en esta tabla se visualiza los indicadores con sus valores respectivos a manera de resumen.

Tabla III.9 Valores Indicadores de Tiempo de Desarrollo

PROTOTOIPOS	I4	I5
JSF	45	30
ASP	45	45

Elaborado por: Fernando Yaulema Rojas.

Luego de obtener los valores de todos los indicadores debemos asignarlos una valoración de acuerdo a la Tabla III.1, ya que debemos analizar dichos valores de forma cualitativa.

En la Figura III.13 visualizamos los valores de este indicador en forma gráfica:

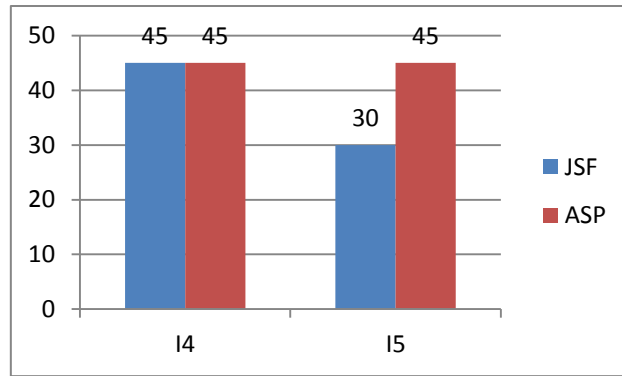


Figura III.13 Valores Indicadores Tiempo de Desarrollo

**Escala de Medición:**

**Rendimiento:**

A continuación en la Tabla III.10 se visualiza los valores asignados a los indicadores del rendimiento de acuerdo a la Escala de Medición en la Tabla III.1.

Esta valoración nos da una perspectiva más clara de los resultados, sabiendo que el número 4 representa al resultado como Satisfactorio hasta 1 como No Satisfactorio, por lo que se puede decir que los indicadores con mayor puntaje son los del prototipo de JSF.

Tabla III.10 Valoración de Indicadores de Rendimiento

PROTOTIPOS	I1	I2	I3	TOTAL
JSF	4	3	4	11/12
ASP	1	4	1	6/12

Elaborado por: Fernando Yaulema Rojas

En la Figura III.14 se visualiza la valoración de los indicadores de rendimiento en forma gráfica:

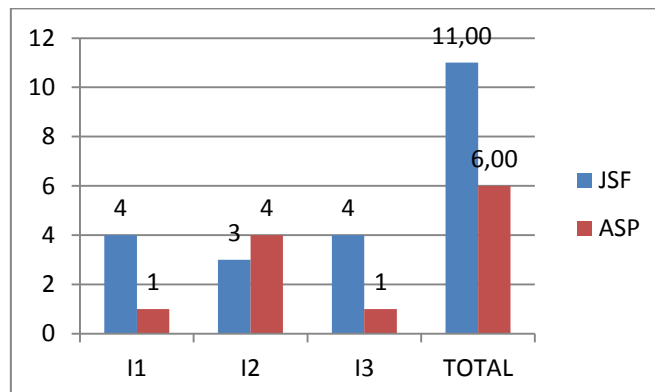


Figura III.14 Valoración de Indicadores de Rendimiento

**Tiempo de Desarrollo:**

A continuación en la Tabla III.11 se visualiza los valores asignados a los indicadores del Tiempo de Desarrollo de acuerdo a la Escala de Medición Tabla III.1.

Esta valoración nos da una perspectiva más clara de los resultados, sabiendo que el número 4 representa al resultado como Satisfactorio hasta 1 como No Satisfactorio, por lo que se puede decir que los indicadores con mayor puntaje son los del prototipo de JSF.

Tabla III.11 Valoración de Indicadores de Tiempo de Desarrollo

PROTOTOPOS	I4	I5	TOTAL
JSF	4	4	8/8
ASP	4	3	7/8

Elaborado por: Fernando Yaulema Rojas

A continuación se visualiza los indicadores de Tiempo de Desarrollo en ASP y JSF en la Figura III.15:

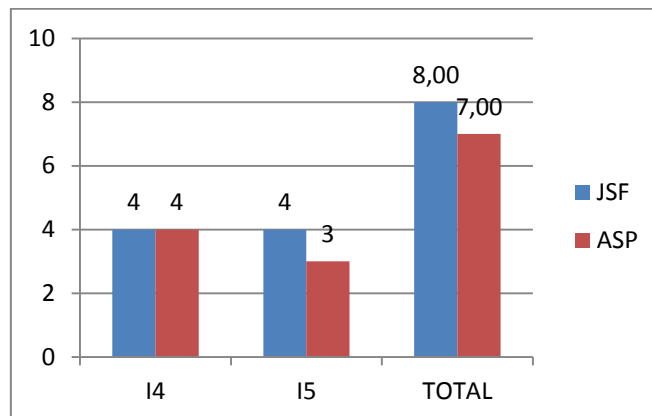


Figura III.15 Valoración de Indicadores del Tiempo de Desarrollo

A continuación en la Tabla III.12 se visualiza todos los valores asignados a los indicadores de Rendimiento y del Tiempo de Desarrollo.

El total se ha calculado como el resultado del producto de los cinco indicadores por cuatro que es el valor máximo que puede llegar a obtener cada indicador.

Tabla III.12 Tabla General de Valores de Indicadores

PROTOTOPOS	RENDIMIENTO			TIEMPO DE DESARROLLO		TOTAL
	I1	I2	I3	I4	I5	-
JSF	4	3	4	4	4	19/20



ASP	1	4	1	4	3	<b>13/20</b>
-----	---	---	---	---	---	--------------

Elaborado por: Fernando Yaulema Rojas

Luego de visualizar los valores de todos los indicadores con todos sus valores se ha llegado a obtener un total de 19/20 para el prototipo de JSF y 13/20 para el prototipo de ASP. A continuación en la Figura III.16 se visualiza los valores de cada indicador así como el total obtenido por cada prototipo.

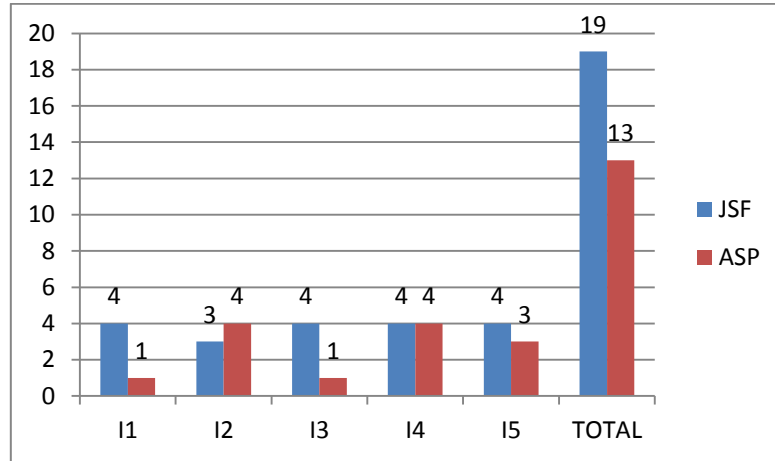


Figura III.16 Figura General de Valores de Indicadores

### 3.7 Demostración de la Hipótesis:

Una vez que se ha obtenido los resultados de las pruebas aplicadas a cada uno de los prototipos tanto en ASP y JSF se procede a realizar la ponderación en función de la valoración de los indicadores. Esto quiere decir que las ponderaciones de cada indicador se obtuvo en base a la Tabla III.1 donde se encuentra los porcentajes asignados para cada indicador y a la Tabla III.2 en donde se encuentran las valoraciones de los resultados.

Como se visualiza en la Tabla III.13 se ha obtenido las ponderaciones de los indicadores de los parámetros propuestos.

Tabla III.13 Ponderación de Indicadores

PROTOTIPOS	RENDIMIENTO			TIEMPO DE DESARROLLO		TOTAL
	I1	I2	I3	I4	I5	
JSF	0,25	0,15	0,15	0,15	0,25	0,95
ASP	0,062	0,20	0,037	0,15	0,19	0,48

Elaborado por: Fernando Yaulema Rojas

A continuación se visualiza las ponderaciones en forma gráfica (Figura III.17):

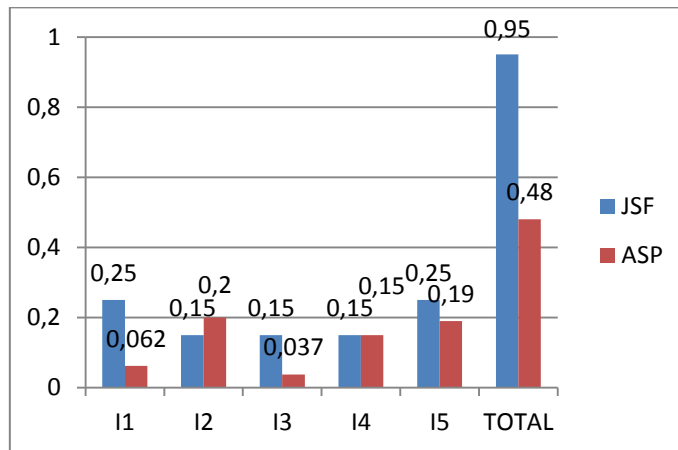


Figura III.17 Ponderación de Indicadores

### 3.8 Resultados Obtenidos de la Demostración

Al realizar la tabulación de los resultados se ha obtenido que al Utilizar JSF + PrimeFaces se obtuvo un resultado del 0,95 frente al valor obtenido con ASP del 0,48.

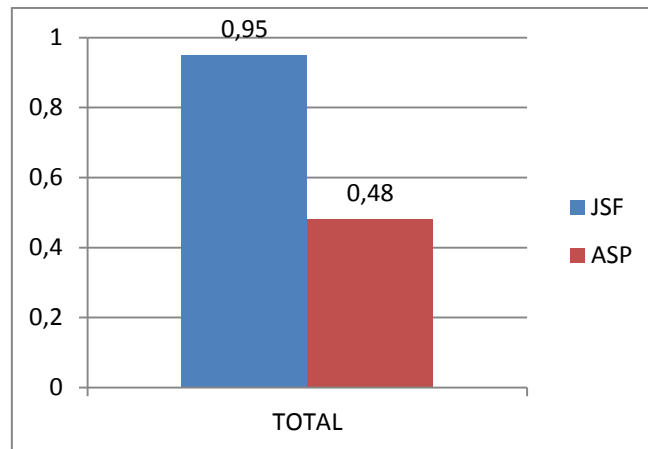


Figura III.18 Resultado de Comparación entre ASP y JSF

Al realizar el análisis de los dos prototipos se ha verificado que al utilizar JSF + PrimeFaces es más óptimo para crear aplicaciones web internacionalizadas, con lo cual se comprueba la hipótesis planteada; La implementación de una Aplicación Web Internacionalizada en JSF es más eficiente en Rendimiento y Tiempo de Desarrollo que una Aplicación Web Internacionalizada desarrollada en ASP.

## CAPITULO IV

### IMPLEMENTACION DEL MÓDULO DE ADQUISICIONES DEL SISTEMA FINANCIERO DE LA ESPOCH

#### 4.1 Metodología XP (eXtreme Programming)

Durante la etapa de ingeniería de software del proyecto se utilizó eXtreme Programming (XP) ya que es una metodología de desarrollo ágil, promueve que exista una interacción constante entre el cliente y el equipo de desarrollo, permite adaptarse a los cambios de requerimientos mediante iteraciones y permite simplificar el desarrollo. A continuación se visualiza en la Figura IV.1 la Arquitectura de la metodología XP.

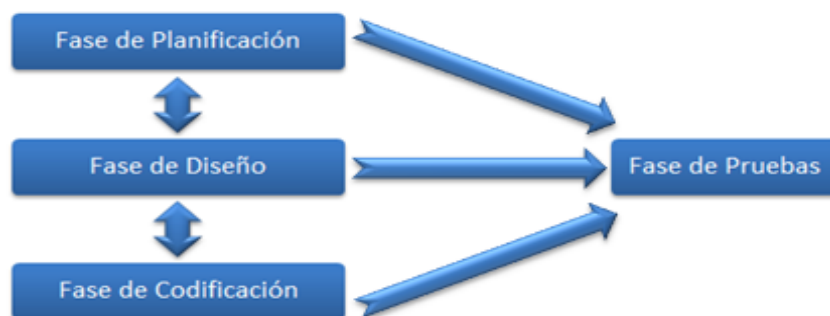


Figura IV.1 Metodología XP (eXtreme Programming)

#### 4.1.1 Fase de Planificación

##### 4.1.1.1 Planificación Inicial

Durante la planificación inicial del proyecto se realizó el análisis del problema así como la recopilación de requerimientos mediante la elaboración de historias de usuario. La planificación inicial fue variando durante la elaboración de la misma, estos cambios fueron ayudando en la mejora de las historias de usuario en base al problema planteado.

A continuación en la Tabla IV.1 se detalla el Equipo de Trabajo para el desarrollo de este proyecto.

Tabla IV.1 Equipo de Trabajo

Miembro	Grupo	Roles XP	Metodología
Fernando Yaulema R.	Tesista	Rastreador, Testeador, Programador	XP
Ing. Gloria Arco M.	Consultor	Entrenador	
Ing. Diego Palacios	Consultor	Entrenador	

Elaborado por: Fernando Yaulema Rojas

##### Recolección de Información

La recolección de la información es muy importante ya que se puede recoger documentos, e información sobre los pasos necesarios para iniciar, procesar y finalizar un proceso.

Durante la recolección de información del PAC se nos proporcionó manuales para la elaboración del mismo durante la su etapa inicial e información hasta la aprobación del PAC en su etapa final.

Todos los documentos e información recolectados fueron entregados a manera de entrevista por la Unidad de Adquisiciones de la ESPOCH, esta información ya sea mediante documentos o verbal es muy importante para el desarrollo del Módulo de Adquisiciones del Sistema Financiero de la ESPOCH.

Todas las interrogantes planteadas hacen referencia al Plan Anual de contrataciones.

- ¿Quién(es) elabora(n) el PAC?
- ¿Cuál es la fuente de donde se obtienen los productos que se registran?
- ¿Si se realiza un consolidado del PAC, quién lo elabora?
- ¿Qué Departamento/Unidad se encarga de validar los registros del PAC?
- ¿Qué Departamento/Unidad asigna las partidas presupuestarias al PAC?
- ¿Quién realiza la aprobación para la ejecución del PAC?
- ¿Qué formato (diseño del documento y extensión del programa) debe tener el documento del consolidado del PAC?

#### 4.1.1.2 Historias de Usuario

A continuación se visualizan las historias del usuario las cuales se han escrito en base a las peticiones del usuario.

Para la elaborar dichas historias de usuario se ha realizado mediante la valoración de la prioridad, riesgo y esfuerzo con los siguientes valores: Alto, Medio y Bajo, como se visualiza en la Tabla IV.2.

Tabla IV.2 Historias de Usuario

Nº	Nombre	Prioridad	Riesgo	Esfuerzo	Iteración
H1	Diseño e Implementación de la Base de Datos	Alto	Alto	Alto	1
H2	Gestión de Ítems del PAC	Alto	Alto	Alto	1
H3	Asignar Perfiles de usuario	Medio	Medio	Bajo	2
H4	Gestión del Encabezado del PAC	Alto	Medio	Medio	2
H5	Gestión del Tipo de Período del PAC	Alto	Medio	Alto	2
H6	Gestión de Categorías de Productos	Medio	Bajo	Bajo	1
H7	Gestión de Subcategorías	Medio	Bajo	Bajo	1
H8	Gestión de Formas de Presentación de Productos	Medio	Bajo	Bajo	1
H9	Gestión de Tipos de Productos	Medio	Bajo	Bajo	1
H10	Gestión del Catálogo Productos del PAC	Alto	Medio	Medio	1
H11	Revisión del PAC con el Rol Adquisiciones	Medio	Bajo	Medio	3

H12	Revisión del PAC con el Rol Presupuesto	Medio	Bajo	Medio	3
H13	Revisión del PAC con el Rol Rector	Medio	Bajo	Medio	3
H14	Asignación de Parámetros Iniciales al PAC	Medio	Bajo	Medio	3
H15	Incremento Porcentuales de Precios de Productos	Medio	Medio	Medio	2

Elaborado por: Fernando Yaulema Rojas

#### 4.1.1.3 Planificación de Publicaciones:

##### 4.1.1.3.1 Iteraciones

**Iteración Primera:** A continuación se detalla las historias de usuario que se desea realizar en esta iteración, como se visualiza en la Tabla IV.3.

Tabla IV.3 Historias de Usuario - Primera Iteración

N°	Historia de Usuario	Semanas
H1	Diseño e implementación de la Base de Datos	6
H6	Gestión de Categorías de Productos	1
H7	Gestión de Subcategorías de Productos	1,5
H8	Gestión de Formas de Presentación de Productos	1,5
H9	Gestión de Tipos de Productos	1,5
H10	Gestión del Catálogo Productos del PAC	2

Elaborado por: Fernando Yaulema Rojas

A continuación se visualiza de forma gráfica la planificación de las historia de usuario de la primera iteración, como se visualiza en la Figura IV.2.

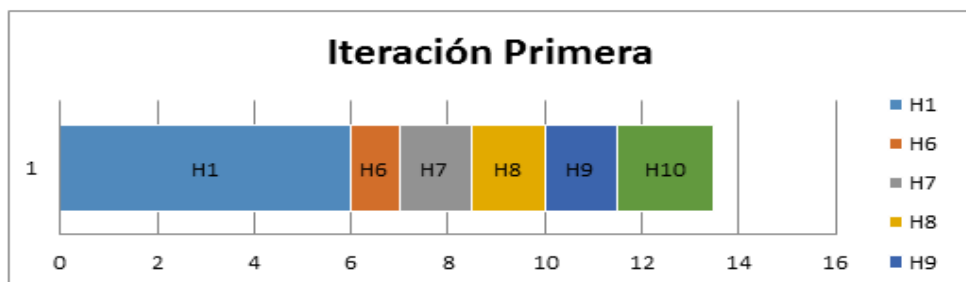


Figura IV.2 Plan de Entrega Iteración Primera

**Iteración Segunda:** A continuación se detalla las historias de usuario que se desea realizar en esta iteración, como se visualiza en la Tabla IV.4:

Tabla IV.4 Historias de Usuario - Segunda Iteración

N°	Historia de Usuario	Semanas
H15	Incremento Porcentuales de Precios de Productos	1,5
H5	Gestión del Tipo de Período del PAC	3
H4	Gestión del Encabezado del PAC	1
H2	Gestión de Ítems del PAC	4
H3	Asignar Perfiles de usuario	0,5

Elaborado por: Fernando Yaulema Rojas

A continuación se visualiza de forma gráfica la planificación de las historia de usuario de la primera iteración, como se visualiza en la Figura IV.3.

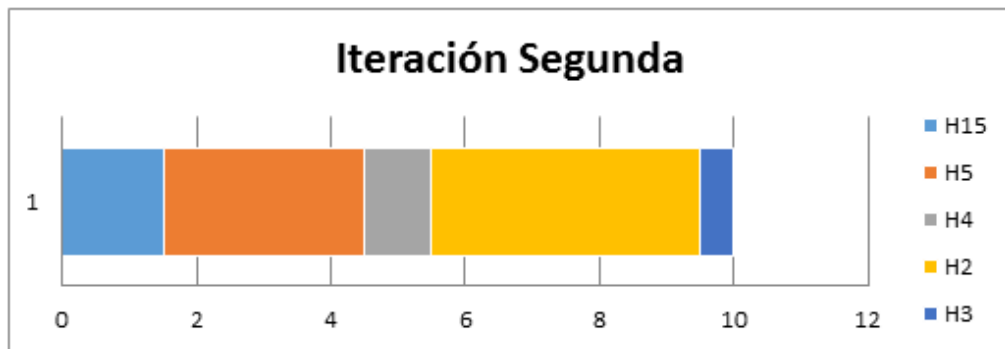


Figura IV.3 Plan de Entrega Iteración Segunda

**Iteración Tercera:** A continuación se detalla las historias de usuario que se desea realizar en esta iteración, como se visualiza en la Tabla IV.5:

Tabla IV.5 Historias de Usuario – Tercera Iteración

N°	Historia de Usuario	Semanas
H14	Asignación de Parámetros Iniciales al PAC	0,5
H11	Revisión del PAC con el Rol Adquisiciones	1,5
H12	Revisión del PAC con el Rol Presupuesto	1
H13	Revisión del PAC con el Rol Rector	2

Elaborado por: Fernando Yaulema Rojas

A continuación se visualiza de forma gráfica la planificación de las historia de usuario de la primera iteración, como se visualiza en la Figura IV.4.

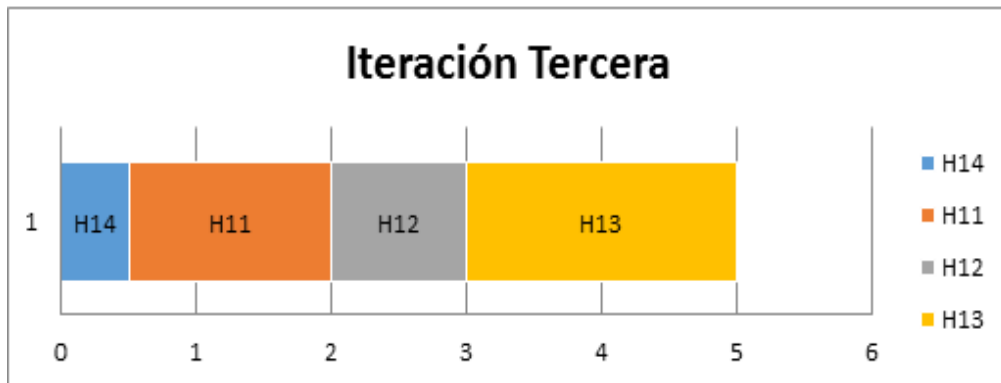


Figura IV.4 Plan de Entrega Iteración Tercera

A continuación se detalla las historias de usuario en base a la planificación de cada una de las iteraciones.

**Iteración 1:**

La historia de usuario 1 (Tabla IV.6) se refiere al Diseño e Implementación de la Base de Datos.

Tabla IV.6 Historia de Usuario Diseño e Implementación de la Base de Datos

Historia de Usuario	
<b>Número:</b> 1	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Diseño e Implementación de la Base de Datos	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Esfuerzo:</b> Alto	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b>	
<b>Descripción:</b> Se realizará el análisis, diseño e implementación de la(s) base(s) de datos a utilizar en módulo a desarrollar.	
<b>Observaciones:</b>	

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 6 (Tabla IV.7) se refiere a la Gestión de Categorías de Productos.

Tabla IV.7 Historia de Usuario Gestión de Categorías de Productos

Historia de Usuario	
<b>Número:</b> 6	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Gestión de Categorías de Productos	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Bajo
<b>Esfuerzo:</b> 4	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b>	



<b>Descripción:</b> El usuario se autentica en el sistema, selecciona la opción del menú referente a la Gestión de Categorías, desde esta opción puede ingresar, modificar, dar de baja o eliminar a los categorías.
<b>Observaciones:</b>

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 7 (Tabla IV.8) se refiere a la Gestión de Subcategorías de Productos.

Tabla IV.8 Historia de Usuario Gestión de Subcategorías de Productos

Historia de Usuario	
<b>Número:</b> 7	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Gestión de Subcategorías de Productos	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Bajo
<b>Esfuerzo:</b> Bajo	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario se autentica en el sistema, selecciona la opción del menú referente a la Gestión de Subcategorías, desde esta opción puede ingresar, modificar, dar de baja o eliminar a las Subcategorías.	
<b>Observaciones:</b>	

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 8 (Tabla IV.9) se refiere a la Gestión de Formas de Presentación (Unidades) de Productos.

Tabla IV.9 Historia de Usuario Gestión de Formas de Presentación de Productos

Historia de Usuario	
<b>Número:</b> 8	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Gestión de Formas de Presentación de Productos	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Bajo
<b>Esfuerzo:</b> Bajo	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario se autentica en el sistema, selecciona la opción del menú referente a la Gestión de Formas de Presentación, desde esta opción puede ingresar, modificar, dar de baja o eliminar a las formas de presentación (unidades).	
<b>Observaciones:</b>	

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 9 (Tabla IV.10) se refiere a la Gestión de Tipos de Productos.

Tabla IV.10 Historia de Usuario Gestión de Tipos de Productos

Historia de Usuario	
<b>Número:</b> 9	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Gestión de Tipos de Productos	

<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Bajo
<b>Esfuerzo:</b> Bajo	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario se autentica en el sistema, selecciona la opción del menú referente a la Gestión de Tipos de Productos, desde esta opción puede ingresar, modificar, dar de baja o eliminar a los Tipos de Productos	
<b>Observaciones:</b> Los Tipos de Productos pueden ser: Bienes, Servicios, Consultorías, Obras, etc.	

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 10 (Tabla IV.11) se refiere a la Gestión de Productos.

Tabla IV.11 Historia de Usuario Gestión de Productos

<b>Historia de Usuario</b>	
<b>Número:</b> 10	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Gestión del Catálogo de Productos del PAC	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Esfuerzo:</b> Medio	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario se autentica en el sistema, selecciona la opción del menú referente a la Gestión de Productos, desde esta opción puede ingresar, modificar, dar de baja o eliminar los productos.	
<b>Observaciones:</b>	

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 2 (Tabla IV.12) se refiere a la Gestión de Productos del PAC por las Actividades.

Tabla IV.12 Historia de Usuario Gestión de Productos del PAC por Actividades

<b>Historia de Usuario</b>	
<b>Número:</b> 2	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Gestión de Items del PAC	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Alto
<b>Esfuerzo:</b> Alto	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario puede ingresar, modificar o eliminar los productos del PAC por las Actividades.	
<b>Observaciones:</b>	

Elaborado por: Fernando Yaulema Rojas

### **Iteración Segunda:**

La historia de usuario 15 (Tabla IV.13) se refiere al Incremento de Precios de Productos.

Tabla IV.13 Historia de Usuario Incremento de Precios de Productos

Historia de Usuario	
<b>Número:</b> 15	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Incremento Porcentuales de Precios de Productos	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Esfuerzo:</b> Medio	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario puede incrementar los precios de los productos de forma masiva ya sea eligiendo por Categorías, Subcategorías o Productos individuales.	
<b>Observaciones:</b> El incremento de Precios de Productos será de forma porcentual.	

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 5 (Tabla IV.14) se refiere a la Gestión del Tipo de Período del PAC.

Tabla IV.14 Historia de Usuario Gestión del Tipo de Período del PAC

Historia de Usuario	
<b>Número:</b> 5	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Gestión del Tipo de Período del PAC.	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Medio
<b>Esfuerzo:</b> Alto	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario se autentica y luego de ello ingresar, modificar, eliminar o dar de baja los Tipos de Períodos del PAC.	
<b>Observaciones:</b> Los Tipos de Períodos pueden ser Mensuales, Bimensuales, Trimestrales, Cuatrimestrales, Semestrales y Anuales.	

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 4 (Tabla IV.15) se refiere a la Gestión del Encabezado del PAC.

Tabla IV.15 Historia de Usuario Gestión del Encabezado del PAC

Historia de Usuario	
<b>Número:</b> 4	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Gestión del Encabezado del PAC.	
<b>Prioridad en negocio:</b> Alto	<b>Riesgo en desarrollo:</b> Medio
<b>Esfuerzo:</b> Medio	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario se autentica y luego de ello puede ingresar y modificar los Encabezados del PAC, este proceso constituirá una de las bases para la elaboración y consolidación del PAC.	
<b>Observaciones:</b>	

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 3 (Tabla IV.16) se refiere a la asignación de Perfiles de Usuario en el Sistema.

Tabla IV.16 Historia de Asignación de Perfiles de Usuario

Historia de Usuario	
<b>Número:</b> 3	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Asignar Perfiles de usuario	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Esfuerzo:</b> Bajo	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b>	
<b>Descripción:</b> Luego de Autenticarse el usuario puede revisar los PACs de las diferentes Actividades de la ESPOCH, dependiendo del rol tendrá acceso a diferentes secciones del PAC.	
<b>Observaciones:</b>	

Elaborado por: Fernando Yaulema Rojas

**Iteración Tercera:**

La historia de usuario 14 (Tabla IV.17) se refiere a la Asignación de Parámetros Iniciales.

Tabla IV.17 Historia de Usuario Asignación de Parámetros Iniciales

Historia de Usuario	
<b>Número:</b> 14	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Asignación de Parámetros Iniciales al PAC	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Bajo
<b>Esfuerzo:</b> Medio	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario puede buscar ciertos parámetros que tienen que ver con la sección presupuestaria del PAC.	
<b>Observaciones:</b>	

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 11 (Tabla IV.18) se refiere a la Revisión del PAC con el Rol Adquisiciones.

Tabla IV.18 Historia de Usuario Revisión PAC con el Rol Adquisiciones

Historia de Usuario	
<b>Número:</b> 11	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Revisión del PAC con el Rol Adquisiciones	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Bajo
<b>Esfuerzo:</b> Medio	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario se autentica en el sistema, selecciona la opción del menú referente a Revisión con el Rol de Adquisiciones, desde esta opción puede revisar el PAC, para la verificación de los datos ingresados en el PAC.	
<b>Observaciones:</b> El usuario puede revisar la sección correspondiente a los productos del	

PAC.
------

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 12 (Tabla IV.19) se refiere a la Revisión del PAC con el Rol Presupuestos.

Tabla IV.19 Historia de Usuario Revisión PAC con el Rol Presupuesto

Historia de Usuario	
<b>Número:</b> 12	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Revisión del PAC con el Rol Presupuesto	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Bajo
<b>Esfuerzo:</b> Medio	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario se autentica en el sistema, selecciona la opción del menú referente a Revisión con el Rol de Presupuestos, desde esta opción puede revisar el PAC, para la verificación de los datos ingresados en el PAC.	
<b>Observaciones:</b> El usuario puede revisar la sección presupuestaria del PAC.	

Elaborado por: Fernando Yaulema Rojas

La historia de usuario 13 (Tabla IV.20) se refiere a la Revisión del PAC con el Rol Rector.

Tabla IV.20 Historia de Usuario Revisión PAC con el Rol Rector

Historia de Usuario	
<b>Número:</b> 13	<b>Usuario:</b> Registrado
<b>Nombre historia:</b> Revisión del PAC con el Rol Rector	
<b>Prioridad en negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Bajo
<b>Esfuerzo:</b> Medio	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b>	
<b>Descripción:</b> El usuario se autentica en el sistema, selecciona la opción del menú referente a Revisión con el Rol del Sr. Rector, desde esta opción puede revisar el PAC.	
<b>Observaciones:</b> El usuario tendrá acceso a todas las secciones del PAC.	

Elaborado por: Fernando Yaulema Rojas

#### 4.1.2 Fase de Diseño

##### 4.1.2.1 Descripción del Diseño de Base de Datos

A continuación se visualiza el diseño de la base de datos (Figura IV.5) para el desarrollo del Módulo de Adquisiciones del Sistema Financiero de la ESPOCH, ésta base de datos nos ayudará a elaborar y consolidar los PACs generados por las Actividades, además del propio

esquema “adquisiciones” del módulo también se usará las bases de datos “master” y “psto” con las cuales se trabajará para su implementación.

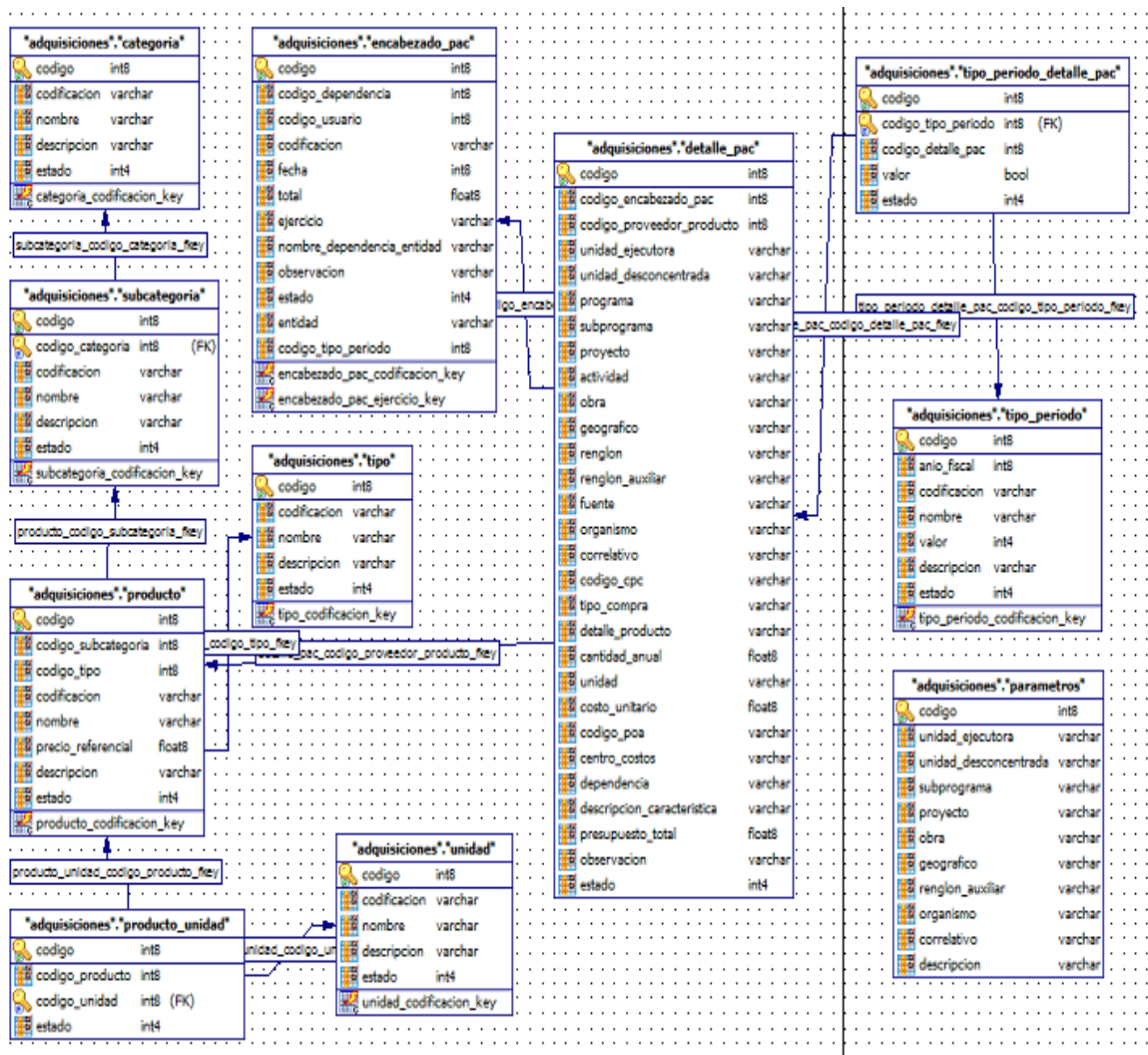


Figura IV.5 Esquema de la Base de Datos del Módulo de Adquisiciones

A continuación se visualizará las tablas usadas desde el esquema “master” las cuales nos ayudarán en la Asignación de Roles y Autenticación de Usuarios en el Módulo de Adquisiciones de la ESPOCH (Figura IV.6). La asignación de roles de usuario restringirá el acceso a usuarios indebidos así como normará el acceso a los diferentes menús disponibles.

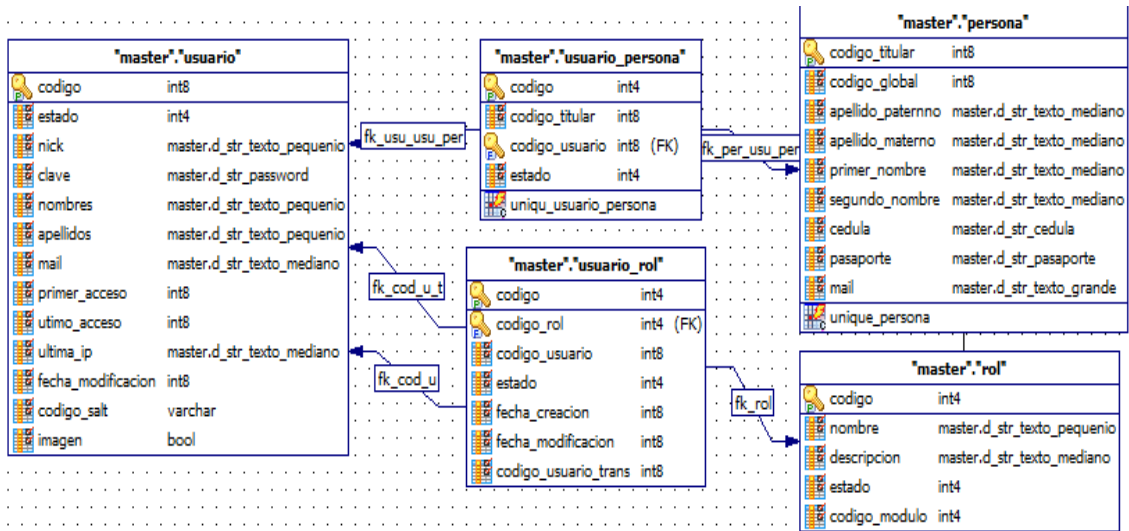


Figura IV.6 Esquema Tablas del Esquema “master” para Autenticación

En la Figura IV.7 visualizaremos las tablas usadas desde el esquema “psto” las cuales nos ayudarán en la Asignación de Actividades a los diferentes Usuarios ya que para la elaboración del PAC el Módulo de Adquisiciones de la ESPOCH se basará en Actividades, Programa y Subprograma.

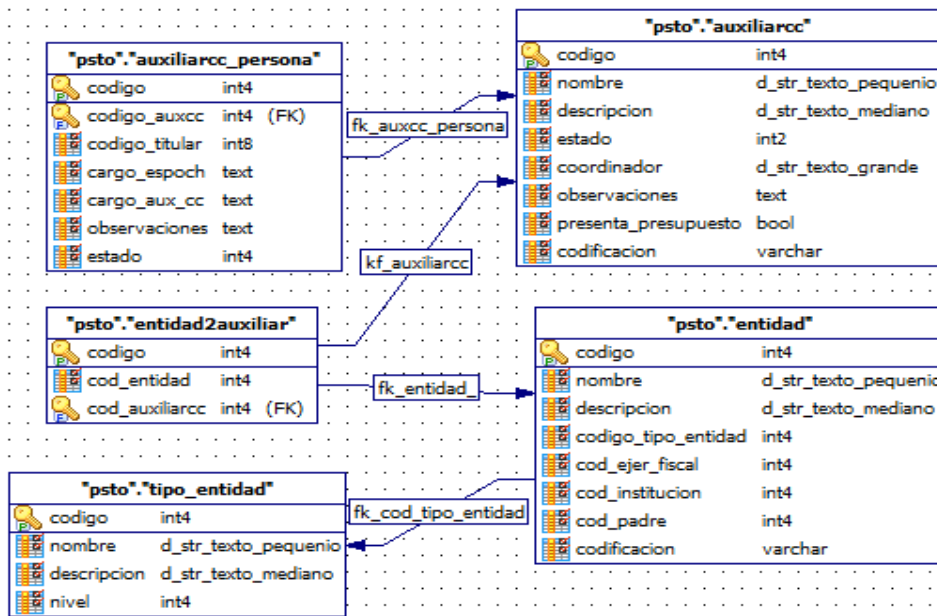


Figura IV.7 Esquema Tablas del Esquema “psto” para Asignación de Actividades

#### 4.1.2.2 Diseño de Interfaces

El diseño de interfaces es muy importante dentro del desarrollo de los sistemas informáticos ya que gracias a ello se puede tener una idea clara de las pantallas a diseñar, así también ayuda a mantener un estándar en el diseño como: la fuente del texto, la ubicación, espacios en las divisiones de la pantalla, etc.

A continuación se visualiza la interfaz (Figura IV.8) que servirá como guía para el desarrollo de todas a pantallas a utilizar en el Módulo de Adquisiciones del Sistema Financiero de la ESPOCH.

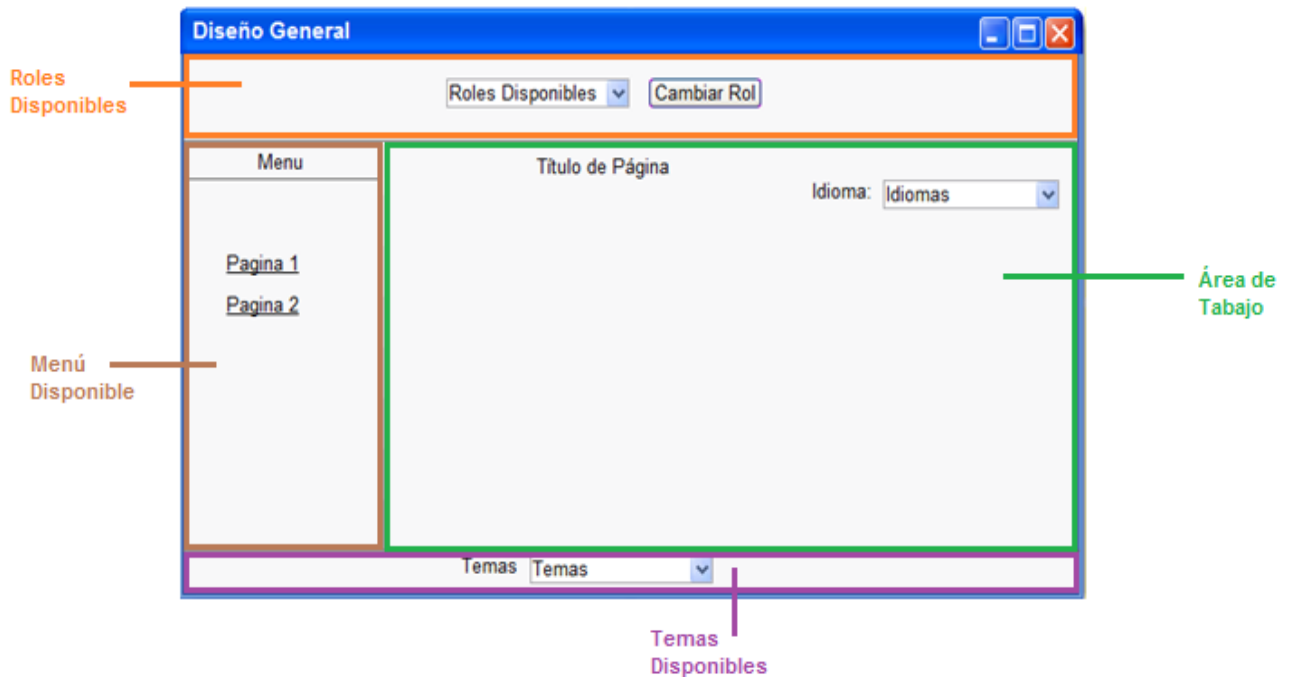


Figura IV.8 Diseño General de la Interfaz del Sistema

A continuación se visualiza la interfaz (Figura IV.9) que será el modelo a seguir para el desarrollo de la Gestión de Categorías.





Figura IV.9 Gestión de Categorías

A continuación se visualiza la interfaz (Figura IV.10) que será el modelo a seguir para el desarrollo de las pantallas de ingreso, edición y eliminación de categorías.

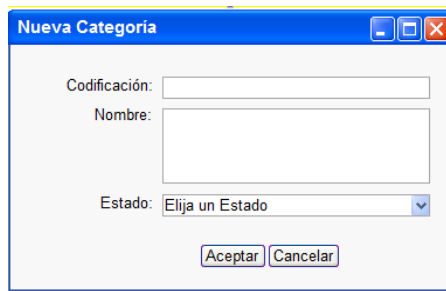


Figura IV.10 Nueva, Edición y Eliminación de Categoría

A continuación se visualiza la interfaz (Figura IV.11) que será el modelo a seguir para el desarrollo de la Gestión de Subcategorías.

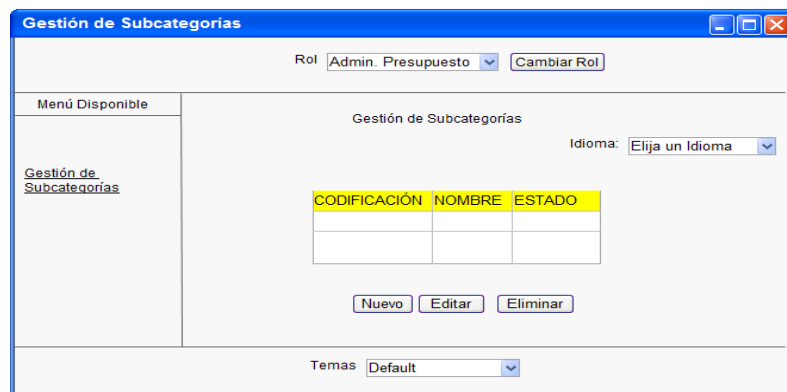


Figura IV.11 Gestión de Subcategorías

A continuación se visualiza la interfaz (Figura IV.12) que será el modelo a seguir para el desarrollo de las pantallas de ingreso, edición y eliminación de Subcategorías.

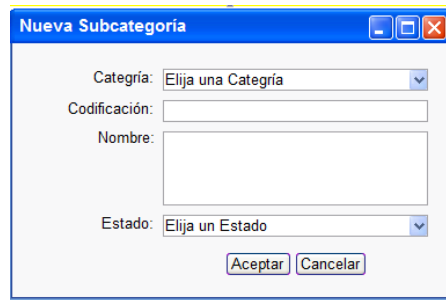


Figura IV.12 Nueva, Edición y Eliminación de Subcategoría

A continuación se visualiza la interfaz (Figura IV.13) que será el modelo a seguir para el desarrollo de la Gestión de Tipos de Productos.

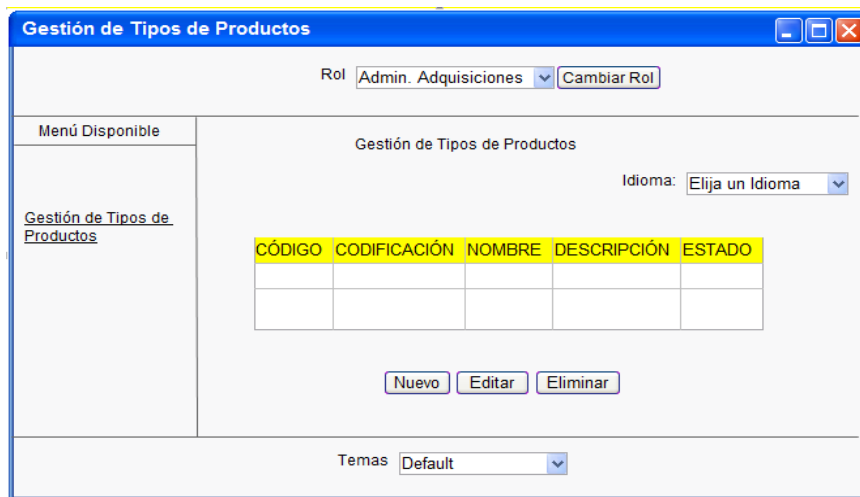


Figura IV.13 Gestión de Tipos de Productos

A continuación se visualiza la interfaz (Figura IV.14) que será el modelo a seguir para el desarrollo de las pantallas de ingreso, edición y eliminación de tipos de productos.

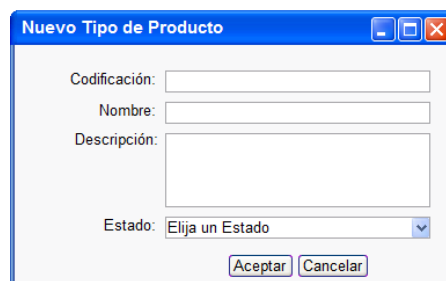


Figura IV.14 Nueva, Edición y Eliminación de Tipos de Productos

A continuación se visualiza la interfaz (Figura IV.15) que será el modelo a seguir para el desarrollo de la Gestión de Forma de Presentación.

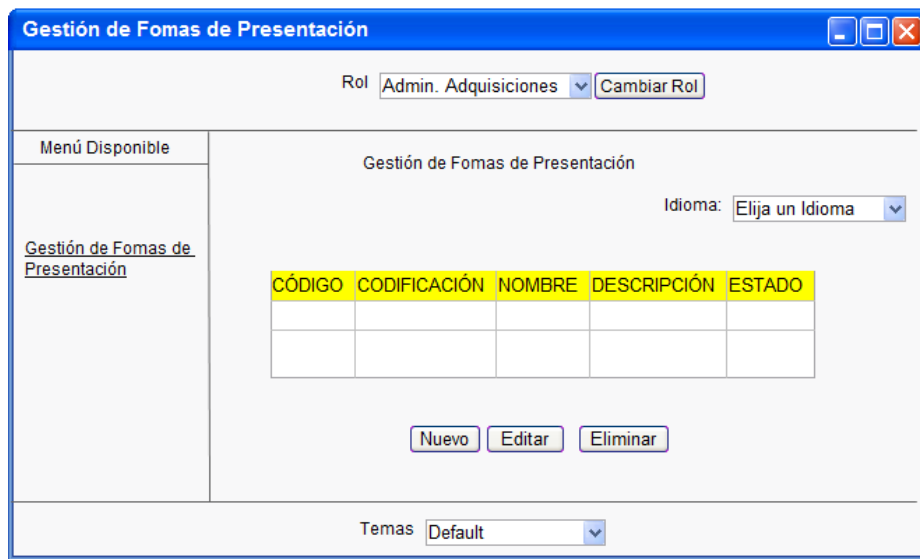


Figura IV.15 Gestión de Formas de Presentación

A continuación se visualiza la interfaz (Figura IV.16) que será el modelo a seguir para el desarrollo de las pantallas de ingreso, edición y eliminación de forma de presentación.

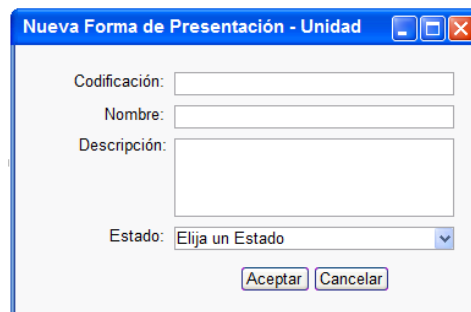


Figura IV.16 Nueva, Edición y Eliminación de Formas de Presentación

A continuación se visualiza la interfaz (Figura IV.17) que será el modelo a seguir para el desarrollo de la Gestión del Catálogo de Unidades y Productos.

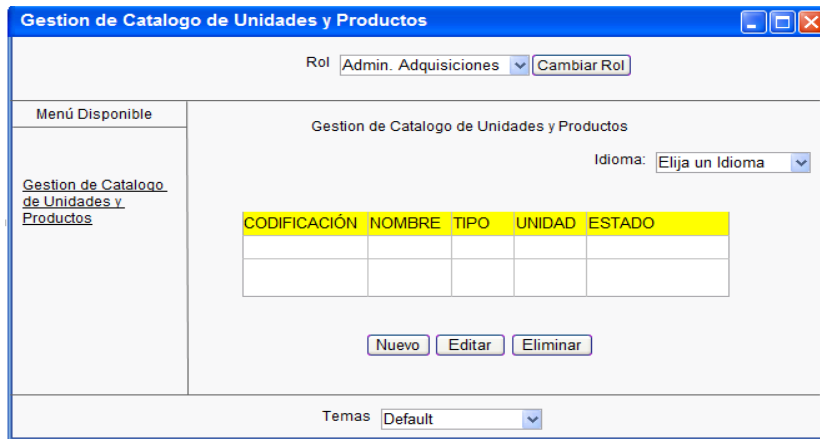


Figura IV.17 Gestión del Catálogo de Productos

A continuación se visualiza la interfaz (Figura IV.18) que será el modelo a seguir para el desarrollo de las pantallas de ingreso, edición y eliminación de productos en el catálogo.

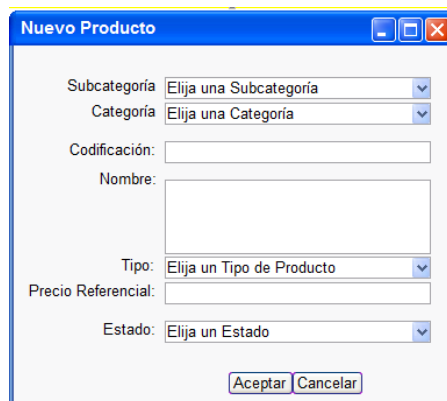


Figura IV.18 Nuevo, Edición y Eliminación de Productos del Catálogo

A continuación se visualiza la interfaz (Figura IV.19) que será el modelo a seguir para el desarrollo de Incrementos Porcentuales.

Incrementos Porcentuales de Precios

Rol: Admin. Adquisiciones

Menú Disponible: Incrementos Porcentuales de Precios

Incrementos Porcentuales de Precios Idioma: Elija un Idioma

Categoría: Elegir una Categoría

Subcategoría: Elegir una Subcategoría

Producto: Elegir una Producto

Incremento Porcentual %: 00.00

Temas: Default

Figura IV.19 Incremento Porcentual de Precios de Productos

A continuación se visualiza la interfaz (Figura IV.20) que será el modelo a seguir para el desarrollo de la Gestión de Tipos de Períodos.

Gestión de Tipos de Períodos

Rol: Admin. Adquisiciones

Menú Disponible: Gestión de Tipos de Períodos

Gestión de Tipos de Períodos Idioma: Elija un Idioma

CÓDIGO	CODIFICACIÓN	AÑO FISCAL	PERÍODO	TIPO PERÍODO	DESCRIPCIÓN	ESTADO

Temas: Default

Figura IV.20 Gestión de Tipos de Períodos

A continuación se visualiza la interfaz (Figura IV.21) que será el modelo a seguir para el desarrollo de las pantallas de ingreso, edición y eliminación de tipos de períodos.

Nuevo Tipo de Período

Nuevo Tipo de Período

Año Fiscal: aaaa

Codificación:

Período: Elegir un Período

Observación:

Estado: Elegir un Estado

Figura IV.21 Nuevo, Edición y Eliminación de Tipos de Períodos

A continuación se visualiza la interfaz (Figura IV.22) que será el modelo a seguir para el desarrollo de la Gestión de Encabezados del PAC.

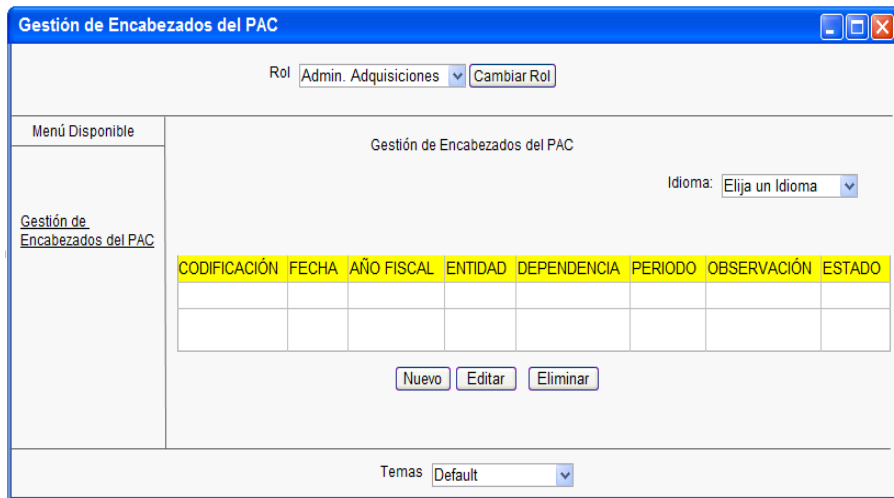


Figura IV.22 Gestión de Encabezados del PAC

A continuación se visualiza la interfaz (Figura IV.23) que será el modelo a seguir para el desarrollo de las pantallas de ingreso, edición y eliminación de encabezados del PAC.

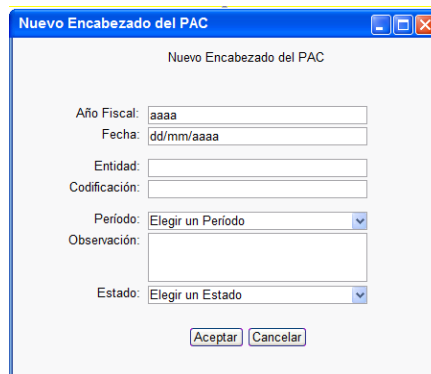


Figura IV.23 Nuevo, Edición y Eliminación de Encabezados del PAC

A continuación se visualiza la interfaz (Figura IV.24) que será el modelo a seguir para el desarrollo de la Gestión de Productos del PAC.

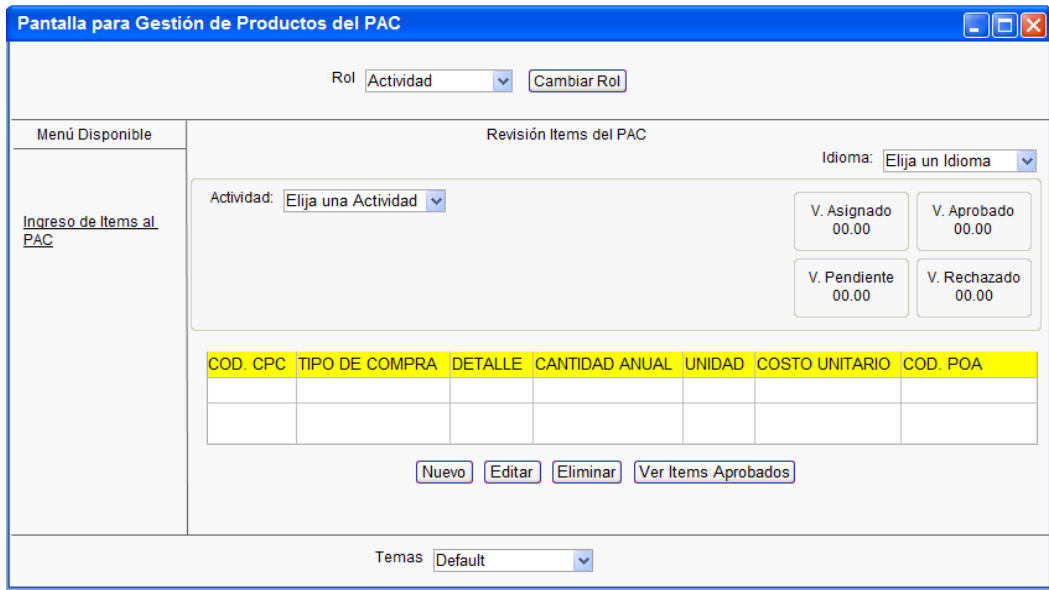


Figura IV.24 Gestión de Ítems del PAC

A continuación se visualiza la interfaz (Figura IV.25) que será el modelo a seguir para el desarrollo de las pantallas de ingreso, edición de ítems al PAC.

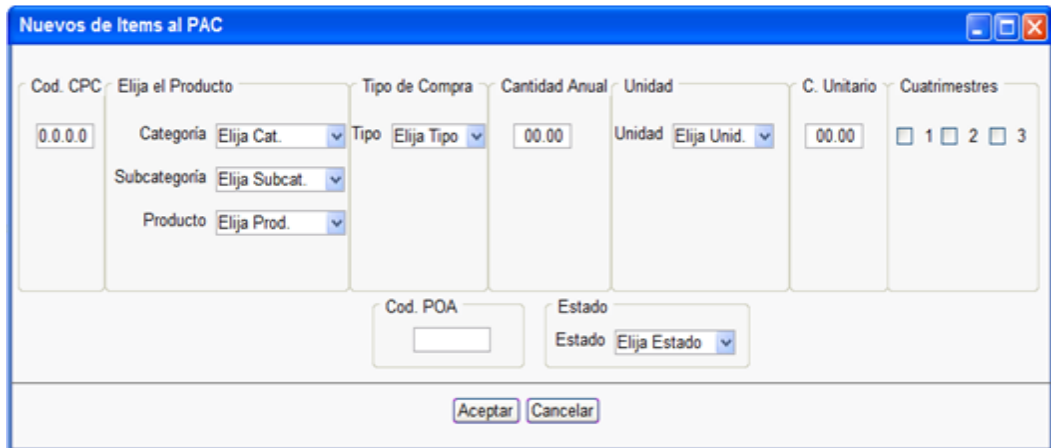


Figura IV.25 Nuevo y Edición de Ítems del PAC

A continuación se visualiza la interfaz (Figura IV.26) que será el modelo a seguir para el desarrollo de la pantalla de eliminación de ítems al PAC.

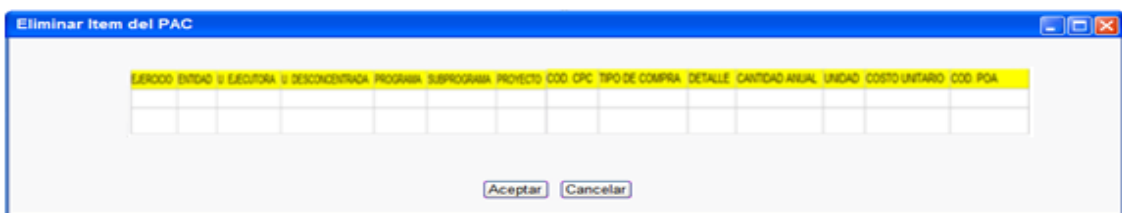


Figura IV.26 Eliminación de Ítems al PAC

A continuación se visualiza la interfaz (Figura IV.27) que será el modelo a seguir para el desarrollo de la pantalla de revisión del rol Adquisiciones.

COD. CPC	TIPO DE COMPRA	DETALLE	CANTIDAD ANUAL	UNIDAD	COSTO UNITARIO	COD. POA

Figura IV.27 Revisión del PAC por Adquisiciones

A continuación se visualiza la interfaz (Figura IV.28) que será el modelo a seguir para el desarrollo de la pantalla de revisión del rol presupuestos.

COD. CPC	TIPO DE COMPRA	DETALLE	CANTIDAD ANUAL	UNIDAD	COSTO UNITARIO	COD. POA

Figura IV.28 Revisión del PAC por Presupuestos

A continuación se visualiza la interfaz (Figura IV.29) que será el modelo a seguir para el desarrollo de la pantalla de revisión del rol rector.



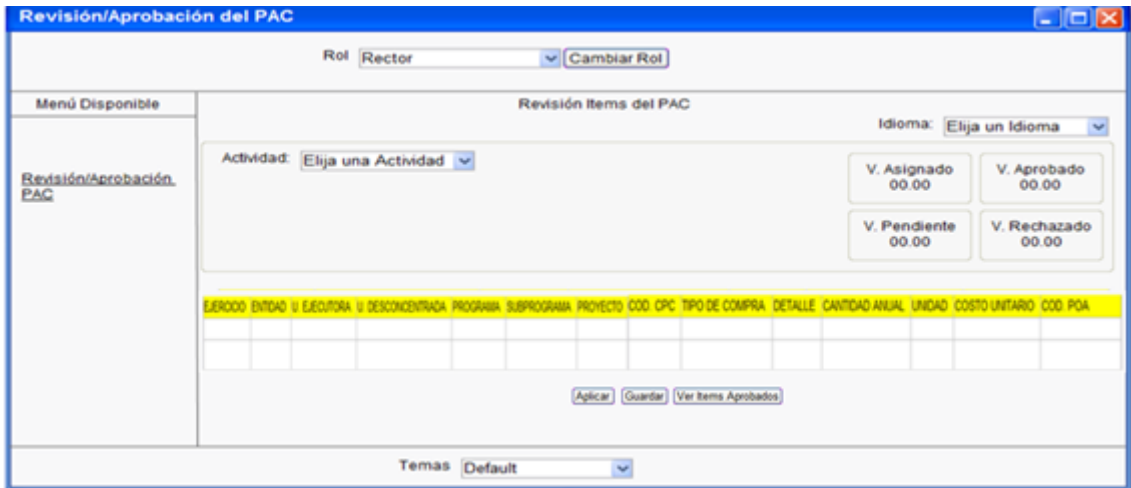


Figura IV.29 Revisión del PAC por el Rector

#### 4.1.2.3 Casos de Uso

La creación de los Casos de Uso nos ayuda a definir cada una de las funciones de los usuarios que van a interactuar en el sistema y por ende también nos ayuda a definir el alcance del sistema de una manera clara antes de iniciar con el desarrollo del mismo,

A continuación se visualiza el caso de uso de la gestión de ítems del PAC en la Figura IV.30:

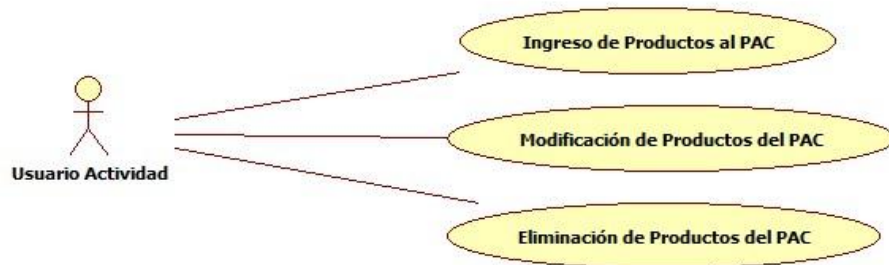


Figura IV.30 Caso de Uso de la Gestión del PAC

A continuación en la Tabla IV.21 se visualiza el detalle del caso de uso de la Gestión del PAC.

Tabla IV.21 Detalle del Caso de Uso de la Gestión del PAC

<b>Identificador de caso de uso</b>	<b>CU_GESTIÓN_DE_ITEMS_DEL_PAC</b>
<b>Nombre del caso de uso</b>	Gestión de Ítems del PAC
<b>Actores</b>	Usuario Actividad

<b>Propósito</b>	Permitir el ingreso, modificación y eliminación de ítems al PAC.	
<b>Visión general</b>	El usuario ingresa, modifica o elimina ítems del PAC de su departamento, escuela, etc.	
<b>Tipo</b>	Primario, esencial	
<b>Referencias</b>	Funciones: Historia 2	
<b>Curso típico de eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del sistema</b>	
<ul style="list-style-type: none"> <li>▪ Usuario selecciona su departamento/escuela</li> </ul>	<ul style="list-style-type: none"> <li>▪ Carga los ítems existentes (de haber)</li> </ul>	
<ul style="list-style-type: none"> <li>▪ Selecciona Nuevo/Editar/Eliminar</li> </ul>	<ul style="list-style-type: none"> <li>▪ Despliega la vista de Ingreso o Edición o Eliminación de Ítems al PAC</li> </ul>	
	<ul style="list-style-type: none"> <li>▪ Solicita datos del nuevo ítem o a editar.</li> </ul>	
<ul style="list-style-type: none"> <li>▪ Ingresa/selecciona información del nuevo ítem.</li> </ul>	<ul style="list-style-type: none"> <li>▪ Valida los datos proporcionados.</li> </ul>	
	<ul style="list-style-type: none"> <li>▪ Despliega vista de ítems existentes.</li> </ul>	
<b>Cursos alternativos</b>		
Mensaje: "Seleccione una Categoría/Subcategoría/Producto"		
Mensaje: "Ingrese la Cantidad Anual/Seleccione una Unidad"		
Mensaje: "Seleccione al menos un período. El código POA es obligatorio"		

Elaborado por: Fernando Yaulema Rojas

A continuación se visualiza el caso de uso del rol adquisiciones en la Figura IV.31:

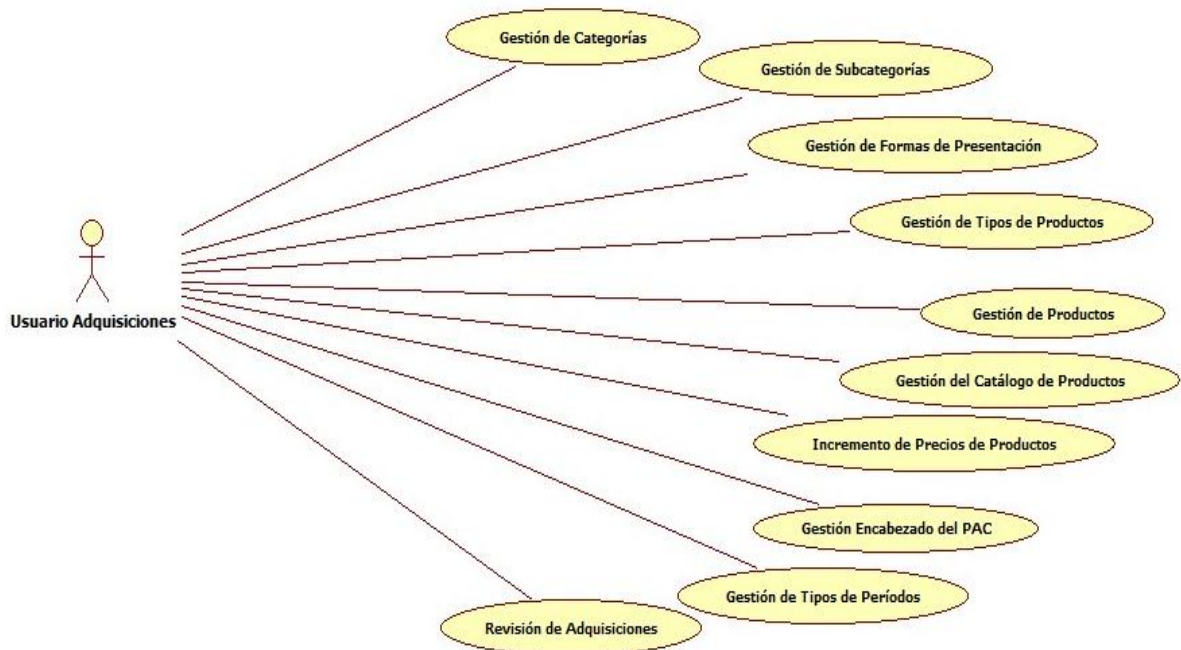


Figura IV.31 Caso de Uso Usuario Adquisiciones

A continuación en la Tabla IV.22 se visualiza el detalle del caso de uso del Usuario Adquisiciones:

Tabla IV.22 Detalle del Caso de Uso Usuario Adquisiciones

<b>Identificador de caso de uso</b>	CU_REVISION_DE_PAC_ROL_ADQUISICIONES	
<b>Nombre del caso de uso</b>	Gestión de Datos Generales y Revisión de Ítems del PAC	
<b>Actores</b>	Usuario Adquisiciones	
<b>Propósito</b>	Permitir la gestión del Catálogo de Productos, Gestión del Encabezado, Gestión de Tipos de Períodos, Incremento Porcentual de Precios y Revisión de Ítems del PAC.	
<b>Visión general</b>	El usuario luego de autenticarse, revisará los campos que componen cada ítem registrado en el PAC, así también podrá revisar (editarlos de ser necesario) la información correspondiente a productos de los ítems generados por las actividades (Escuelas, Departamentos, etc.).	
<b>Tipo</b>	Primario, esencial	
<b>Referencias</b>	Funciones: Historia 11	
<b>Curso típico de eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del sistema</b>	
<ul style="list-style-type: none"> <li>▪ Usuario selecciona Tipo de Período</li> </ul>	<ul style="list-style-type: none"> <li>▪ Carga los períodos existentes</li> </ul>	
Selecciona Nuevo/Editar/Eliminar	Despliega la vista de Ingreso o Edición o Eliminación de Ítems al PAC	
Ingresar Información requerida y guarda/edita/elimina datos.	Oculta la pantalla seleccionada.	
<ul style="list-style-type: none"> <li>▪ Usuario selecciona Encabezado del PAC</li> </ul>	<ul style="list-style-type: none"> <li>▪ Carga los encabezados del PAC existentes</li> </ul>	
Selecciona Nuevo/Editar/Eliminar	Despliega la vista de Ingreso o Edición o Eliminación de Encabezados del PAC	
Ingresar Información requerida y guarda/edita/elimina datos.	Oculta la pantalla seleccionada.	
<ul style="list-style-type: none"> <li>▪ Usuario selecciona todas o una actividad (departamento, escuela, etc.)</li> </ul>	<ul style="list-style-type: none"> <li>▪ Carga los ítems existentes (de haber)</li> </ul>	
Selecciona la celda del campo del Ítem a Revisar.	Visualiza Avisos de Cambios realizados al finalizar la acción.	
Guarda corrección de haberla		
Guardar Ítems Revisados.	Guarda Ítems (registros) revisados.	
<b>Cursos alternativos</b>		
Mensaje de Error: "Seleccione un tipo de Período Fiscal"		
Alerta: "Se pinta de color rojo las celdas que tienen datos erróneos o que no tienen datos."		

Elaborado por: Fernando Yaulema Rojas

A continuación se visualiza el caso de uso del rol revisión presupuestaria en la Figura IV.32:

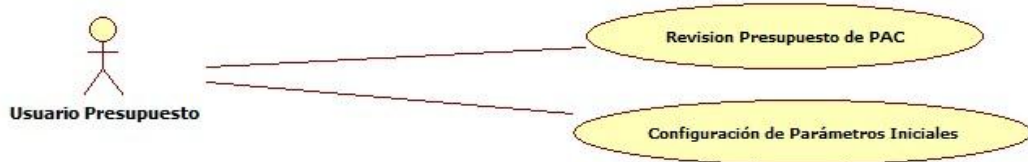


Figura IV.32 Caso de Uso Revisión Presupuestaria

A continuación en la Tabla IV.23 se visualiza el detalle del caso de uso de Revisión Presupuestaria

Tabla IV.23 Detalle del Caso de Uso Revisión Presupuestaria

<b>Identificador de caso de uso</b>	CU_REVISION_PRESUPUESTARIA_PAC	
<b>Nombre del caso de uso</b>	Gestión de Parámetros Iniciales y de la Sección Presupuestaria de los Ítems del PAC.	
<b>Actores</b>	Usuario Presupuesto	
<b>Propósito</b>	Permitir la Revisión Presupuestaria de los Ítems del PAC y Configuración de Parámetros Iniciales de la Sección Presupuestaria del PAC.	
<b>Visión general</b>	El usuario luego de autenticarse, gestionará los datos correspondientes a la sección Presupuestaria del PAC, así también podrá revisar (editarlos de ser necesario) los ítems generados por las actividades (Escuelas, Departamentos, etc.).	
<b>Tipo</b>	Primario, esencial	
<b>Referencias</b>	Funciones: Historia 12	
<b>Curso típico de eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del sistema</b>	
<ul style="list-style-type: none"> <li>▪ Usuario selecciona actividad (Escuela, Departamento, etc.)</li> </ul>	<ul style="list-style-type: none"> <li>▪ Carga los ítems del PAC existentes.</li> </ul>	
Modifica sólo la sección presupuestaria del PAC.	Visualiza Avisos de Cambios realizados al finalizar la acción.	
Guarda corrección de haberla		
Guardar Ítems Revisados.	Guarda Ítems (registros) revisados.	

<b>Cursos alternativos</b>
Alerta: "Se pinta de color rojo las celdas que tienen datos erróneos o que no tienen datos."

Elaborado por: Fernando Yaulema Rojas

A continuación se visualiza el caso de uso de la aprobación del PAC en la Figura IV.33:

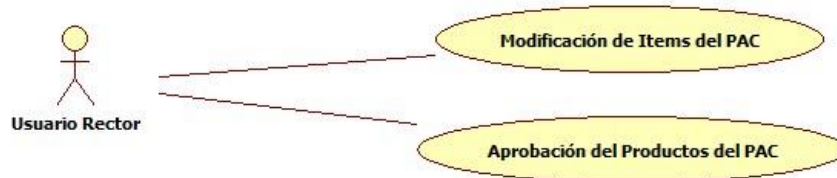


Figura IV.33 Caso de Uso Aprobación del PAC

A continuación en la Tabla IV.24 se visualiza el detalle del caso de uso de la Aprobación del PAC:

Tabla IV.24 Detalle del Caso de Uso Aprobación del PAC

<b>Identificador de caso de uso</b>	CU_APROBACIÓN_DEL_PAC	
<b>Nombre del caso de uso</b>	Aprobación de los Ítems del PAC.	
<b>Actores</b>	Usuario Rector	
<b>Propósito</b>	Permitir la Aprobación del PAC.	
<b>Visión general</b>	El usuario luego de autenticarse, tendrá acceso a toda la información del PAC, así también podrá revisar (editar de necesario) los Ítems del PAC.	
<b>Tipo</b>	Primario, esencial	
<b>Referencias</b>	Funciones: Historia 13	
<b>Curso típico de eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del sistema</b>	
<ul style="list-style-type: none"> <li>▪ Usuario selecciona actividad (Escuela, Departamento, etc.)</li> </ul>	<ul style="list-style-type: none"> <li>▪ Carga los ítems del PAC existentes.</li> </ul>	
Revisa y/o Aprueba los Ítems (Edita los ítems del PAC de ser necesario)	Visualiza Avisos de Cambios realizados al finalizar la acción.	
Guardar Ítems Revisados y/o Aprobados.	Guarda Ítems (registros) revisados.	
<b>Cursos alternativos</b>		
Alerta: "Se pinta de color rojo las celdas que tienen datos erróneos o que no tienen datos."		

Elaborado por: Fernando Yaulema Rojas

#### **4.1.2.4 Diagrama de Procesos**

En el siguiente Figura IV.34 se visualiza el diagrama de flujo de proceso que se sigue desde el inicio que es el ingreso de ítems al PAC hasta cuando se los aprueba, que es el final del proceso.

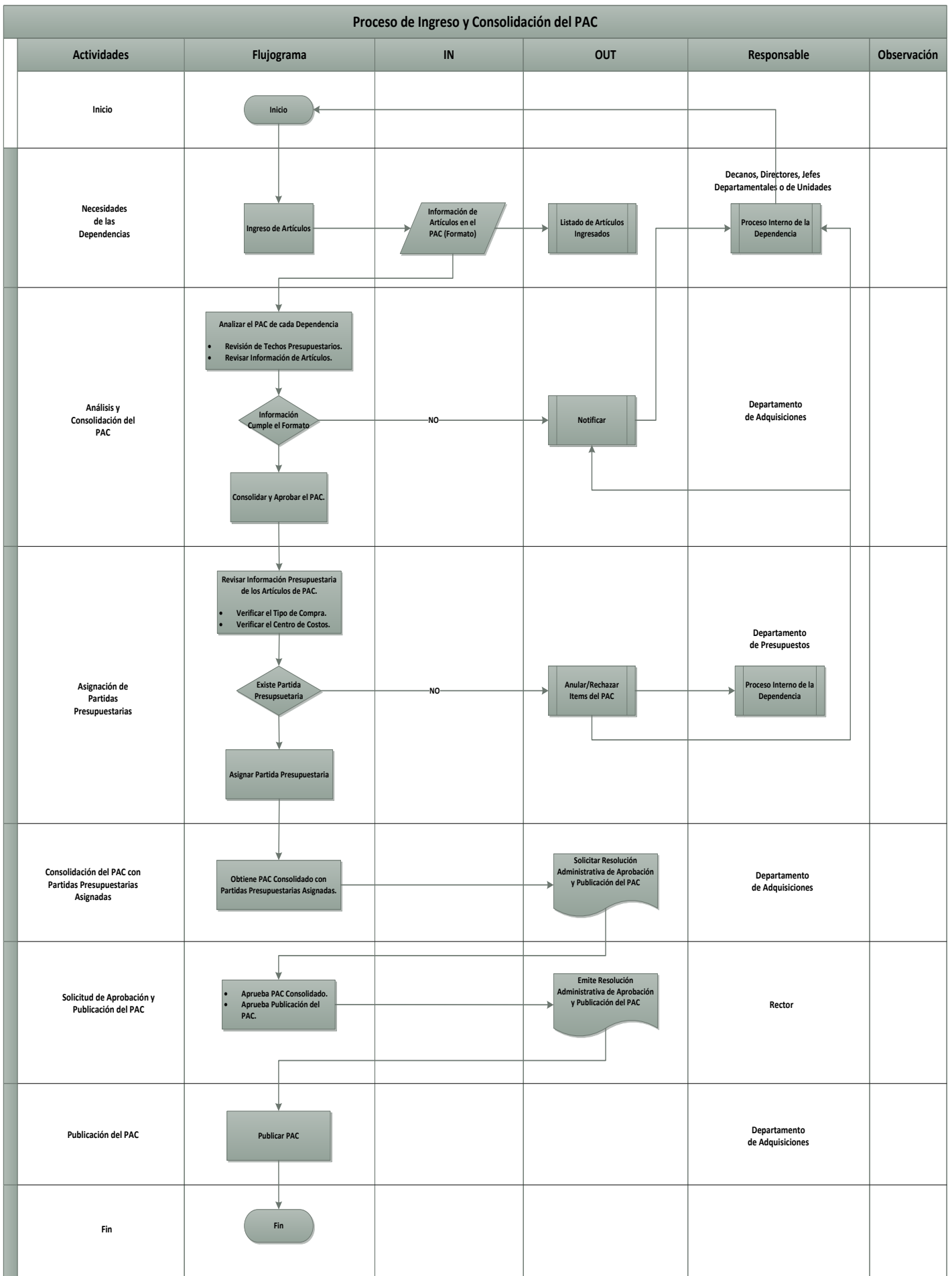


Figura IV.34 Flujo de Proceso para Gestión del PAC

#### 4.1.2.5 Arquitectura

En el siguiente Figura IV.35 se visualiza la arquitectura utilizada para el desarrollo del Módulo de Adquisiciones.

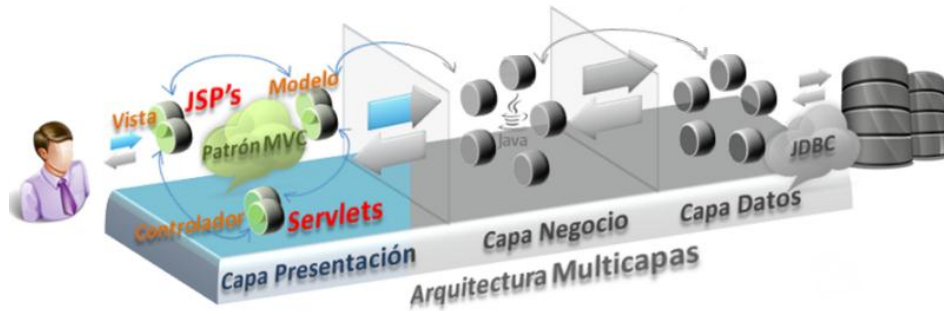


Figura IV.35 Arquitectura del Sistema

#### 4.1.2.6 Tareas de Historia de Usuario

La tarea 1 (Tabla IV.25) se refiere a las tareas necesarias para el análisis de la documentación proporcionada por el Departamento de Adquisiciones.

Tabla IV.25 Tarea Diseño e Implementación de la Base de Datos

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 1
<b>Nombre de tarea:</b> Diseño e Implementación de la Base de Datos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 23-07-2012	<b>Fecha fin:</b> 31-08-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Consistirá en analizar, diseñar e implementar la Base de Base de Datos para el Módulo de Adquisiciones así como las bases de datos necesarias para su implementación. Elaborado por: Fernando Yaulema Rojas	

La tarea 2 (Tabla IV.26) se refiere a la implementación de la clase Categoría.

Tabla IV.26 Implementación de la Clase Categoría

Tarea	
<b>Número de tarea:</b> 2	<b>Número de historia:</b> 6
<b>Nombre de tarea:</b> Implementación de clase y métodos de Categoría	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 08-10-2012	<b>Fecha fin:</b> 12-10-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>Categoría</b> y los atributos: <i>codigo, codificación, nombre, descripción, estado.</i>	



Elaborado por: Fernando Yaulema Rojas

La tarea 3 (Tabla IV.27) se refiere a la implementación de la clase Subcategoría.

Tabla IV.27 Implementación de la Clase Subcategoría

Tarea	
<b>Número de tarea:</b> 3	<b>Número de historia:</b> 7
<b>Nombre de tarea:</b> Implementación de clase y métodos de <i>SubCategoría</i>	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 15-10-2012	<b>Fecha fin:</b> 24-10-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>SubCategoría</b> y los atributos: <i>codigo, codificación, nombre, descripción, estado.</i>	

Elaborado por: Fernando Yaulema Rojas

La tarea 4 (Tabla IV.28) se refiere a la implementación de la clase Unidad (Formas de Presentación de Productos).

Tabla IV.28 Implementación de la Clase Unidad (Formas de Presentación)

Tarea	
<b>Número de tarea:</b> 4	<b>Número de historia:</b> 8
<b>Nombre de tarea:</b> Implementación de clase y métodos <i>Unidad</i>	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 25-10-2012	<b>Fecha fin:</b> 05-11-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>Unidad</b> y los atributos: <i>codigo, codificación, nombre, descripción, estado.</i>	

Elaborado por: Fernando Yaulema Rojas

La tarea 5 (Tabla IV.29) se refiere a la implementación de la clase Tipo (Tipos de Productos).

Tabla IV.29 Implementación de la Clase Tipo (Tipos de Productos)

Tarea	
<b>Número de tarea:</b> 5	<b>Número de historia:</b> 9
<b>Nombre de tarea:</b> Implementación de clase y métodos de la Clase Tipo	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 06-11-2012	<b>Fecha fin:</b> 16-11-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>Tipo</b> y los atributos: <i>codigo, codificación, nombre, descripción, estado.</i>	

Elaborado por: Fernando Yaulema Rojas

La tarea 6 (Tabla IV.30) se refiere a la implementación de la clase ProductoUnidad (Catálogo de Asignación de Unidades a Productos)

Tabla IV.30 Implementación de la Clase ProductoUnidad

Tarea	
<b>Número de tarea:</b> 6	<b>Número de historia:</b> 10
<b>Nombre de tarea:</b> Implementación de clase y métodos ProductoUnidad	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 17-11-2012	<b>Fecha fin:</b> 31-11-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>ProductoUnidad</b> y los atributos: <i>codigo, producto, unidad, estado</i> .	

Elaborado por: Fernando Yaulema Rojas

La tarea 7 (Tabla IV.31) se refiere a la implementación de la clase Producto.

Tabla IV.31 Implementación de la Clase Producto

Tarea	
<b>Número de tarea:</b> 7	<b>Número de historia:</b> 10
<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 17-11-2012	<b>Fecha fin:</b> 03-11-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>Producto</b> y los atributos: <i>codigo, sub_categoria, tipo, codificación, nombre, precio_referencial, descripción, estado</i> .	

Elaborado por: Fernando Yaulema Rojas

La tarea 8 (Tabla IV.32) se refiere a la implementación de la clase TipoPeríodo (Mensual, Bimensual, Trimestral, Cuatrimestral, etc.).

Tabla IV.32 Implementación de la Clase TipoPeríodo

Tarea	
<b>Número de tarea:</b> 8	<b>Número de historia:</b> 5
<b>Nombre de tarea:</b> Implementación de clase y métodos <i>TipoPeríodo</i>	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 04-12-2012	<b>Fecha fin:</b> 01-01-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>TipoPeríodo</b> y los atributos: <i>codigo, ejercicio_fisca, codificación, nombre, valor, descripcion, estado</i> .	

Elaborado por: Fernando Yaulema Rojas

La tarea 9 (Tabla IV.33) se refiere a la implementación de la clase TipoPeríodoDetallePac (Clase creada para los períodos que se eligen para cada ítem del PAC).

Tabla IV.33 Implementación de la Clase TipoPeríodoDetallePac

Tarea	
<b>Número de tarea:</b> 9	<b>Número de historia:</b> 5

<b>Nombre de tarea:</b> Implementación de clase y métodos <i>TipoPeríodoDetallePac</i>	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 04-12-2012	<b>Fecha fin:</b> 01-01-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>EncabezadoPAC</b> y los atributos: <b>codigo, tipo_peirodo, detalle_pac, valor, estado.</b>	

Elaborado por: Fernando Yaulema Rojas

La tarea 10 (Tabla IV.34) se refiere a la implementación de la clase EncabezadoPAC (Se deberá crear un encabezado para cada período fiscal).

Tabla IV.34 Implementación de la Clase EncabezadoPac

Tarea	
<b>Número de tarea:</b> 10	<b>Número de historia:</b> 4
<b>Nombre de tarea:</b> Implementación de clase y métodos <i>EncabezadoPAC</i>	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 03-01-2013	<b>Fecha fin:</b> 10-01-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>EncabezadoPAC</b> y los atributos: <i>codigo, usuario, usuario_rol, codificación, fecha, total, ejercicio, nombre_dependencia, observación, estado, entidad, código_tipo_período</i>	

Elaborado por: Fernando Yaulema Rojas

La tarea 11 (Tabla IV.35) se refiere a la implementación de la clase DetallePac (Clase creada para los ítems del PAC).

Tabla IV.35 Implementación de la Clase DetallePac

Tarea	
<b>Número de tarea:</b> 11	<b>Número de historia:</b> 2
<b>Nombre de tarea:</b> Implementación de clase y métodos <i>DetallePAC</i>	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 11-01-2013	<b>Fecha fin:</b> 01-02-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>DetallePAC</b> y los atributos: <i>codigo, encabezadoPac, producto, unidad_ejecutora, unidad_desconcentrada, programa, subprograma, proyecto, actividad, obra, geográfico, renglón, renglon_auxiliar, fuente, organismo, correlativo, codigo_cpc, tipo_compra, detalle_producto, cantidad_anual, unidad, costo_unitario, codigo_poa, centro_costos, dependencia, descripcion_caracteristica, presupuesto_total, observación, estado, listaltemsString, productoObject, listaltemsPeríodo.</i>	

Elaborado por: Fernando Yaulema Rojas

La tarea 12 (Tabla IV.36) se refiere a la implementación de Perfiles de Usuario

Tabla IV.36 Asignación de Perfiles de Usuario

Tarea
-------

<b>Número de tarea:</b> 12	<b>Número de historia:</b> 3
<b>Nombre de tarea:</b> Asignar Perfiles de usuario	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 04-02-2013	<b>Fecha fin:</b> 06-02-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se crear los perfiles de usuario del sistema, para posteriormente asignar a los usuarios.	

Elaborado por: Fernando Yaulema Rojas

La tarea 12 (Tabla IV.37) se refiere a la implementación de la clase Parametros (Clase creada para los parámetros iniciales del Módulo de Adquisiciones).

Tabla IV.37 Implementación de la Clase Parametros

Tarea	
<b>Número de tarea:</b> 12	<b>Número de historia:</b> 14
<b>Nombre de tarea:</b> Implementación de clase y métodos <i>Parametros</i>	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b>	<b>Fecha fin:</b>
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>Parametros</b> y los atributos: <i>codigo, unidad_ejecutora, unidad_desconcentrada, subprograma, proyecto, obra, geografico, renglon_auxiliar, organismo, correlativo, descripcion.</i>	

Elaborado por: Fernando Yaulema Rojas

## FUNCIONES

La tarea 13 (Tabla IV.38) se refiere a la implementación de la clase FCategoria.

Tabla IV.38 Implementación de la Clase FCategoria

Tarea	
<b>Número de tarea:</b> 13	<b>Número de historia:</b> 6
<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 08-10-2012	<b>Fecha fin:</b> 12-10-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>FCategoria</b> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 14 (Tabla IV.39) se refiere a la implementación de la clase FSubCategoria.

Tabla IV.39 Implementación de la Clase FSubcategoría

Tarea	
<b>Número de tarea:</b> 14	<b>Número de historia:</b> 7

<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 15-10-2012	<b>Fecha fin:</b> 24-10-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>FSubCategoria</b> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 15 (Tabla IV.40) se refiere a la implementación de la clase FTipo (Tipo de Producto).

Tabla IV.40 Implementación de la Clase FTipo

Tarea	
<b>Número de tarea:</b> 15	<b>Número de historia:</b> 5
<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 04-12-2012	<b>Fecha fin:</b> 01-01-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>FTipo</b> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 16 (Tabla IV.41) se refiere a la implementación de la clase FUnidades (Forma de Presentación de Productos).

Tabla IV.41 Implementación de la Clase FUnidades

Tarea	
<b>Número de tarea:</b> 16	<b>Número de historia:</b> 8
<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b>	<b>Fecha fin:</b>
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>FUnidades</b> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 17 (Tabla IV.42) se refiere a la implementación de la clase FProducto.

Tabla IV.42 Implementación de la Clase FProducto

Tarea	
<b>Número de tarea:</b> 17	<b>Número de historia:</b> 10
<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 17-11-2012	<b>Fecha fin:</b> 10-10-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>FProducto</b> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 18 (Tabla IV.43) se refiere a la implementación de la clase FProductoUnidad.

Tabla IV.43 Implementación de la Clase FProductoUnidad

Tarea	
<b>Número de tarea:</b> 18	<b>Número de historia:</b> 10
<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 10-10-2012	<b>Fecha fin:</b> 31-11-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>FProductoUnidad</b> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 19 (Tabla IV.44) se refiere a la implementación de la clase FTipoPeríodo.

Tabla IV.44 Implementación de la Clase FTipoPeríodo

Tarea	
<b>Número de tarea:</b> 19	<b>Número de historia:</b> 5
<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 04-12-2012	<b>Fecha fin:</b> 01-01-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>FTipoPeríodo</b> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 20 (Tabla IV.45) se refiere a la implementación de la clase FEncabezadoPac.

Tabla IV.45 Implementación de la Clase FEncabezadoPac

Tarea	
<b>Número de tarea:</b> 20	<b>Número de historia:</b> 4
<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 03-01-2013	<b>Fecha fin:</b> 10-01-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <b>FEncabezadoPac</b> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 21 (Tabla IV.46) se refiere a la implementación de la clase FDetallePac.

Tabla IV.46 Implementación de la Clase FDetallePac

Tarea
-------

<b>Número de tarea:</b> 21	<b>Número de historia:</b> 2
<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 11-01-2013	<b>Fecha fin:</b> 01-02-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <i>FDetallePac</i> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 22 (Tabla IV.47) se refiere a la implementación de la clase *FTipoPeríodoDetallePac*.

Tabla IV.47 Implementación de la Clase *FTipoPeríodoDetallePac*

Tarea	
<b>Número de tarea:</b> 22	<b>Número de historia:</b> 5
<b>Nombre de tarea:</b> Implementación de clase y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 10-10-2012	<b>Fecha fin:</b> 10-10-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <i>FTipoPeríodoDetallePac</i> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 23 (Tabla IV.48) se refiere a la implementación de la clase *FParametros*.

Tabla IV.48 Implementación de la Clase *FParametros*

Tarea	
<b>Número de tarea:</b> 23	<b>Número de historia:</b>
<b>Nombre de tarea:</b> Implementación de clase y métodos de Parámetros	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b>	<b>Fecha fin:</b>
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa la clase <i>FParametros</i> y los métodos para interactuar con las bases de datos utilizadas.	

Elaborado por: Fernando Yaulema Rojas

La tarea 24 (Tabla IV.49) se refiere a la implementación del Controlador *CategoriaController*

Tabla IV.49 Implementación del Controlador *CategoriaController*

Tarea	
<b>Número de tarea:</b> 24	<b>Número de historia:</b> 6
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 08-10-2012	<b>Fecha fin:</b> 12-10-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>CategoriaController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 25 (Tabla IV.50) se refiere a la implementación del Controlador SubcategoriaController

Tabla IV.50 Implementación del Controlador SubcategoriaController

Tarea	
<b>Número de tarea:</b> 25	<b>Número de historia:</b> 7
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 15-10-2012	<b>Fecha fin:</b> 24-10-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>SubCategoriaController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 26 (Tabla IV.51) se refiere a la implementación del Controlador TipoController

Tabla IV.51 Implementación del Controlador TipoController

Tarea	
<b>Número de tarea:</b> 26	<b>Número de historia:</b> 9
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 06-11-2012	<b>Fecha fin:</b> 16-11-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>TipoController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 27 (Tabla IV.52) se refiere a la implementación del Controlador UnidadController

Tabla IV.52 Implementación del Controlador UnidadController

Tarea	
<b>Número de tarea:</b> 27	<b>Número de historia:</b> 8
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Bajo
<b>Fecha inicio:</b> 25-10-2012	<b>Fecha fin:</b> 05-11-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>UnidadController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 28 (Tabla IV.53) se refiere a la implementación del Controlador ProductoController.

Tabla IV.53 Implementación del Controlador ProductoController

Tarea	
<b>Número de tarea:</b> 28	<b>Número de historia:</b> 10
<b>Nombre de tarea:</b> Implementación del controlador y métodos	



<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 17-11-2012	<b>Fecha fin:</b> 31-11-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>ProductoController</i> y los métodos para interactuar con las vistas (interfaces).	
Elaborado por: Fernando Yaulema Rojas	

La tarea 29 (Tabla IV.54) se refiere a la implementación del Controlador ProductoUnidadController.

Tabla IV.54 Implementación del Controlador ProductoUnidadController

<b>Tarea</b>	
<b>Número de tarea:</b> 29	<b>Número de historia:</b> 10
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 17-11-2012	<b>Fecha fin:</b> 31-11-2012
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>ProductoUnidadController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 30 (Tabla IV.55) se refiere a la implementación del Controlador TipoPeríodoController

Tabla IV.55 Implementación del Controlador TipoPeríodoController

<b>Tarea</b>	
<b>Número de tarea:</b> 30	<b>Número de historia:</b> 5
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 04-12-2012	<b>Fecha fin:</b> 01-01-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>TipoPeríodoController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 31 (Tabla IV.56) se refiere a la implementación del Controlador EncabezadoPacController

Tabla IV.56 Implementación del Controlador EncabezadoPacController

<b>Tarea</b>	
<b>Número de tarea:</b> 31	<b>Número de historia:</b> 4
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 03-01-2013	<b>Fecha fin:</b> 10-01-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>EncabezadoPacController</i> y los métodos para	

interactuar con las vistas (interfaces).
--

Elaborado por: Fernando Yaulema Rojas

La tarea 32 (Tabla IV.57) se refiere a la implementación del Controlador *DetallePacController*.

Tabla IV.57 Implementación del Controlador *DetallePacController*

Tarea	
<b>Número de tarea:</b> 32	<b>Número de historia:</b> 2
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 11-01-2013	<b>Fecha fin:</b> 01-02-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>DetallePacController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 33 (Tabla IV.58) se refiere a la implementación del Controlador *TipoPeriodoDetallePacController*.

Tabla IV.58 Implementación del Controlador *TipoPeriodoDetallePacController*

Tarea	
<b>Número de tarea:</b> 33	<b>Número de historia:</b> 2
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 11-01-2013	<b>Fecha fin:</b> 01-02-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>TipoPeriodoDetallePacController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 34 (Tabla IV.59) se refiere a la implementación del Controlador *LenguajeController*

Tabla IV.59 Implementación del Controlador *LenguajeController*

Tarea	
<b>Número de tarea:</b> 34	<b>Número de historia:</b>
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b>	<b>Fecha fin:</b>
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>LenguajeController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 35 (Tabla IV.60) se refiere a la implementación del Controlador RevisionAdquisicionesController.

Tabla IV.60 Implementación del Controlador RevisionAdquisicionesController

Tarea	
<b>Número de tarea:</b> 35	<b>Número de historia:</b> 11
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 07-02-2013	<b>Fecha fin:</b> 19-02-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>RevisionAdquisicionesController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 36 (Tabla IV.61) se refiere a la implementación del Controlador RevisionPresupuestoController.

Tabla IV.61 Implementación del Controlador RevisionPresupuestoController

Tarea	
<b>Número de tarea:</b> 36	<b>Número de historia:</b> 12
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 20-02-2013	<b>Fecha fin:</b> 27-02-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>RevisionPresupuestoController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

La tarea 37 (Tabla IV.62) se refiere a la implementación del Controlador RevisionRectorController.

Tabla IV.62 Implementación del Controlador RevisionRectorController

Tarea	
<b>Número de tarea:</b> 37	<b>Número de historia:</b> 13
<b>Nombre de tarea:</b> Implementación del controlador y métodos	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Medio
<b>Fecha inicio:</b> 28-02-2013	<b>Fecha fin:</b> 07-04-2013
<b>Programador responsable:</b> Edwin Yaulema	
<b>Descripción:</b> Se implementa el Controlador <i>RevisionRectorController</i> y los métodos para interactuar con las vistas (interfaces).	

Elaborado por: Fernando Yaulema Rojas

#### 4.1.3 Fase de Codificación

#### 4.1.3.1 Paquetes

- **accesodatos:** El paquete “accesodatos” contiene todas las configuraciones necesarias para acceder a las bases de datos a usar en el sistema.
- **adquisiciones.logica.clases:** El paquete “adquisiciones.logica.clases” contiene las clases que reflejan todas las tablas existentes en la base de datos, esto ayuda en la manipulación de los objetos que se utiliza en el Módulo de Adquisiciones. Los nombres de las clases empiezan con letra mayúscula, los atributos de la clase deben coincidir con los nombres de las columnas de la tabla a la que hacen referencia, cada atributo de cada clase contiene los métodos set() y get() para su manipulación externa.
- Ejemplo: “*publicclassDetallePac*”
- **adquisiciones.logica.funciones:** El paquete “adquisiciones.logica.funciones” contiene las clases que a su vez contienen todos los métodos necesarios para interactuar con la base de datos. Los nombres de estas clases empiezan con la letra F mayúscula seguida del nombre del objeto con que el que se interactuará.
- Ejemplo: “*publicclassFDetallePac*”
- **adquisiciones.presentacion.beans:** El paquete “adquisiciones.presentacion.beans” contiene todos los controladores, los cuales se utiliza para interactuar las pantallas del sistema con la base de datos. El nombre de los controladores o Beans tiene el nombre del objeto a utilizar seguido de la palabra Controller.
- Ejemplo: “*public class DetallePacController*”

Para conocer en detalle los paquetes y las clases desarrolladas para el Módulo de Adquisiciones diríjase al código fuente de la aplicación que se encuentra en cd.

#### 4.1.3.2 Diagrama de Despliegue

A continuación en la Figura IV.36 se visualiza el diagrama de despliegue del Módulo de Adquisiciones.

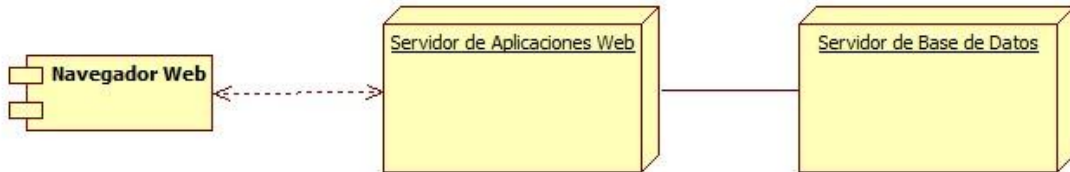


Figura IV.36 Diagrama de Despliegue

#### 4.1.3.3 Diagrama de Componentes

A continuación en la Figura IV.37 se visualiza el diagrama de componentes del Módulo de Adquisiciones.

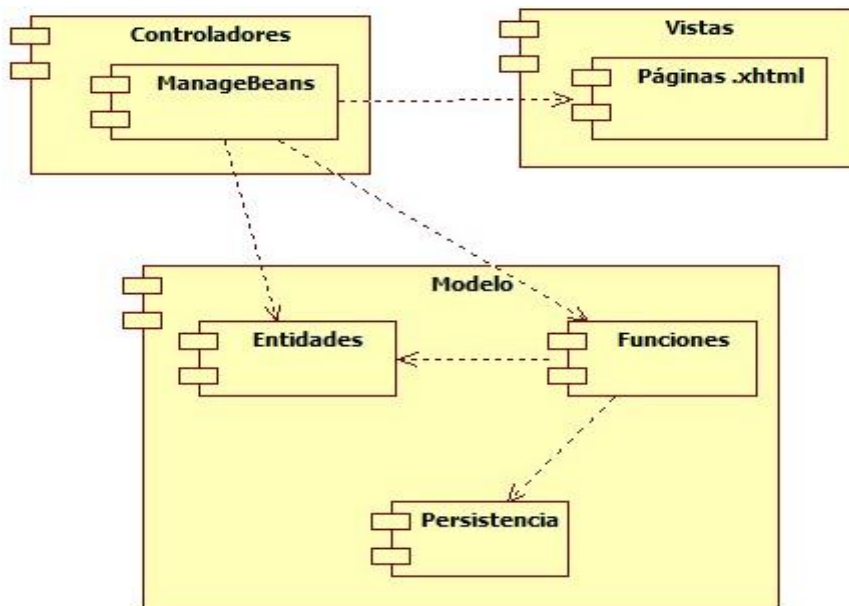


Figura IV.37 Diagrama de Componentes

#### 4.1.4 Fase de Pruebas

La fase de pruebas comprende la comprobación del correcto funcionamiento del sistema por lo que se realizan pruebas sobre la codificación del sistema. A continuación se describe las pruebas realizadas sobre la historia de Gestión de Ítems del PAC.

#### 4.1.4.1 Pruebas Funcionales

A continuación se describe las pruebas funcionales realizadas a la historia de Usuario Gestión de Ítems del PAC.

- **Historia de Usuario:** Gestión de Categorías de Productos - Correcto

- **Descripción:** Una vez que el usuario selecciona la opción Gestión de Categorías seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá ingresar la codificación, el nombre de la categoría y el estado. Internamente los datos serán validados antes de registrarlos en la base de datos.
- **Condiciones de Ejecución:** Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.

- ✓ **Entrada:**

Después de la selección se mostrará la opción “Nuevo” para ingresar una Categoría.

- El usuario deberá ingresar la codificación, el nombre de la categoría y seleccionar el estado del registro.
- El sistema validará internamente la información ingresada para posteriormente registrarla.
- El proceso de ingreso dentro de la Gestión de Categorías se considera como finalizado.

- ✓ **Resultado Esperado:**

- Luego del ingreso de la nueva categoría, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.

- ✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

- **Historia de Usuario:** Gestión de Categorías de Productos - Incorrecto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión de Categorías seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario

deberá ingresar la codificación, el nombre de la categoría y el estado. Internamente los datos en el caso de ser incorrectos se emitirá un mensaje de error por cada campo.

✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.

✓ **Entrada:**

- Después de la selección se mostrará la opción “Nuevo” para ingresar una Categoría.
- El usuario deberá ingresar la codificación, nombre de la categoría y seleccionar el estado del registro
- El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
- El proceso de ingreso dentro de la Gestión de Categorías se considera como finalizado.

✓ **Resultado Esperado:**

- Si la codificación o el nombre o el estado de la categoría son incorrectos no se registrará en el sistema.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

• **Historia de Usuario:** Gestión de Subcategorías de Productos - Correcto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión de Subcategorías seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá seleccionar la categoría, ingresar la codificación, el nombre de la categoría y seleccionar el estado. Internamente los datos serán validados antes de registrarlos en la base de datos.

✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.

✓ **Entrada:**

- Después de la selección se mostrará la opción “Nuevo” para ingresar una Subcategoría.
- El usuario deberá seleccionar la categoría, ingresar la codificación, el nombre de la subcategoría y seleccionar el estado.
- El sistema validará internamente la información ingresada para posteriormente registrarlo.
- El proceso de ingreso dentro de la Gestión de Subcategorías se considera como finalizado.

✓ **Resultado Esperado:**

- Luego del ingreso de la nueva subcategoría, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

• **Historia de Usuario:** Gestión de Subcategorías de Productos - Incorrecto

- **Descripción:** Una vez que el usuario selecciona la opción Gestión de Subcategorías seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá El usuario deberá seleccionar la categoría, ingresar la codificación, el nombre de la subcategoría y seleccionar el estado.

- ✓ Internamente los datos en el caso de ser incorrectos se emitirá un mensaje de error por cada campo.

✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.

✓ **Entrada:**



- Después de la selección se mostrará la opción “Nuevo” para ingresar una Subcategoría.
  - El usuario deberá seleccionar la categoría, ingresar la codificación, el nombre de la subcategoría y seleccionar el estado.
  - El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
  - El proceso de ingreso dentro de la Gestión de Subcategorías se considera como finalizado.
- ✓ **Resultado Esperado:**
- Si la codificación o el nombre o el estado de la subcategoría son incorrectos no se registrará en el sistema.
- ✓ **Evaluación de la Prueba**
- Prueba satisfactoria.
- **Historia de Usuario:** Gestión de Formas de Presentación de Productos - Correcto
- ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión de Formas de Presentación seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá ingresar la codificación, el nombre de la forma de presentación, descripción y el estado. Internamente los datos serán validados antes de registrarlos en la base de datos.
- ✓ **Condiciones de Ejecución:**
- Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.
- ✓ **Entrada:**
- Después de la selección se mostrará la opción “Nuevo” para ingresar una forma de presentación.
  - El usuario deberá ingresar la codificación, el nombre de la forma de presentación, descripción y seleccionar el estado del registro.

- El sistema validará internamente la información ingresada para posteriormente registrarla.
- El proceso de ingreso dentro de la Gestión de Formas de Presentación se considera como finalizado.
- ✓ **Resultado Esperado:**
  - Luego del ingreso de la nueva forma de presentación, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.
- ✓ **Evaluación de la Prueba**
  - Prueba satisfactoria.
- **Historia de Usuario:** Gestión de Formas de Presentación de Productos - Incorrecto
  - ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión de Formas de Presentación seleccionará la opción "Nuevo" y aparece una nueva ventana en la cual el usuario deberá ingresar la codificación, el nombre de la forma de presentación, descripción y el estado. Internamente los datos en el caso de ser incorrectos se emitirá un mensaje de error por cada campo.
  - ✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.
  - ✓ **Entrada:**
    - Después de la selección se mostrará la opción "Nuevo" para ingresar una forma de presentación.
    - El usuario deberá ingresar la codificación, nombre de la forma de presentación y seleccionar el estado del registro
    - El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
    - El proceso de ingreso dentro de la Gestión de Formas de Presentación se considera como finalizado.

- ✓ **Resultado Esperado:**
  - Si la codificación o el nombre o el estado de la forma de presentación son incorrectos no se registrará en el sistema.
- ✓ **Evaluación de la Prueba**
  - Prueba satisfactoria.
- **Historia de Usuario:** Gestión de Tipos de Productos - Correcto
  - ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión de Tipos de Productos seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá ingresar la codificación, el nombre del tipo de producto, descripción y el estado. Internamente los datos serán validados antes de registrarlos en la base de datos.
  - ✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.
  - ✓ **Entrada:**
    - Después de la selección se mostrará la opción “Nuevo” para ingresar un tipo de producto.
    - El usuario deberá ingresar la codificación, el nombre del tipo de producto, descripción y seleccionar el estado del registro.
    - El sistema validará internamente la información ingresada para posteriormente registrarla.
    - El proceso de ingreso dentro de la Gestión de Tipos se considera como finalizado.
  - ✓ **Resultado Esperado:**
    - Luego del ingreso de la nueva forma de presentación, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.
  - ✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

- **Historia de Usuario:** Gestión de Tipos de Productos - Incorrecto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión de Tipos de Productos seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá ingresar la codificación, el nombre del tipo de producto, descripción y el estado. Internamente los datos en el caso de ser incorrectos se emitirá un mensaje de error por cada campo.

- ✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.

- ✓ **Entrada:**

- Después de la selección se mostrará la opción “Nuevo” para ingresar un tipo de producto.
- El usuario deberá ingresar la codificación, nombre del tipo de producto y seleccionar el estado del registro
- El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
- El proceso de ingreso dentro de la Gestión de Tipos se considera como finalizado.

- ✓ **Resultado Esperado:**

- Si la codificación o el nombre o el estado del tipo de producto son incorrectos no se registrará en el sistema.

- ✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

- **Historia de Usuario:** Gestión del Catálogo de Productos de Productos - Correcto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión del Catálogo de Productos seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá ingresar la categoría, subcategoría, nombre, el tipo de producto, el precio y el estado. Internamente los datos serán validados antes de registrarlos en la base de datos.

- ✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.

- ✓ **Entrada:**

- Después de la selección se mostrará la opción “Nuevo” para ingresar un producto al catálogo.
- El usuario deberá ingresar la categoría, subcategoría, nombre, el tipo de producto, el precio y el estado.
- El sistema validará internamente la información ingresada para posteriormente registrarlo.
- El proceso de ingreso dentro de la Gestión del Catálogo de Productos se considera como finalizado.

- ✓ **Resultado Esperado:**

- Luego del ingreso de la nueva forma de presentación, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.

- ✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

- **Historia de Usuario:** Gestión del Catálogo de Productos de Productos - Incorrecto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión del Catálogo de Productos seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá ingresar la categoría, subcategoría, nombre, el tipo de producto,

el precio y el estado. Internamente los datos en el caso de ser incorrectos se emitirá un mensaje de error por cada campo.

✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.

✓ **Entrada:**

- Después de la selección se mostrará la opción “Nuevo” para ingresar un producto al catálogo.
- El usuario deberá ingresar la categoría, subcategoría, nombre, el tipo de producto, el precio y el estado El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
- El proceso de ingreso dentro de la Gestión del Catálogo de Productos se considera como finalizado.

✓ **Resultado Esperado:**

- Si la categoría o la subcategoría o el nombre o el tipo de producto o el precio o el estado son incorrectos no se registrará en el sistema.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

● **Historia de Usuario:** Incremento Porcentual de Precios - Correcto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Incremento Porcentual de Precios seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá seleccionar la categoría, la subcategoría, el producto e ingresar el porcentaje de incremento. Internamente los datos serán validados antes de registrarlos en la base de datos.

✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.

✓ **Entrada:**

- Después de la selección se mostrará la opción "Nuevo" para realizar un nuevo incremento porcentual de precios.
- El usuario deberá seleccionar la categoría, la subcategoría, el producto e ingresar el porcentaje de incremento del registro.
- El sistema validará internamente la información ingresada para posteriormente registrarlo.
- El proceso de ingreso dentro de la Incremento Porcentual de Precios se considera como finalizado.

✓ **Resultado Esperado:**

- Luego del ingreso del nuevo incremento porcentual, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

• **Historia de Usuario:** Incremento Porcentual de Precios - Incorrecto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Incremento Porcentual de Precios seleccionará la opción "Nuevo" y aparece una nueva ventana en la cual el usuario deberá seleccionar la categoría, la subcategoría, el producto e ingresar el porcentaje de incremento. Internamente los datos en el caso de ser incorrectos se emitirá un mensaje de error por cada campo.

✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.

✓ **Entrada:**

- Después de la selección se mostrará la opción "Nuevo" para ingresar un incremento porcentual.
- El usuario deberá seleccionar la categoría, la subcategoría, el producto e ingresar el porcentaje de incremento del registro

- El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
  - El proceso de ingreso dentro de la Incremento Porcentual de Precios se considera como finalizado.
- ✓ **Resultado Esperado:**
  - Si la categoría o subcategoría o el producto o el incremento son incorrectos no se registrará en el sistema.
- ✓ **Evaluación de la Prueba**
  - Prueba satisfactoria.
- **Historia de Usuario:** Gestión de Tipos de Períodos - Correcto
  - ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión de Tipos de Períodos seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá la codificación, nombre, seleccionar el período, descripción y el estado. Internamente los datos serán validados antes de registrarlos en la base de datos.
  - ✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.
  - ✓ **Entrada:**
    - Después de la selección se mostrará la opción “Nuevo” para realizar un nuevo de Tipo de Período.
    - El usuario deberá la codificación, nombre, seleccionar el período, descripción y el estado del registro.
    - El sistema validará internamente la información ingresada para posteriormente registrarlo.
    - El proceso de ingreso dentro de la Gestión de Tipos de Períodos se considera como finalizado.



- ✓ **Resultado Esperado:**
  - Luego del ingreso de la nuevo tipo de período, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.
- ✓ **Evaluación de la Prueba**
  - Prueba satisfactoria.
- **Historia de Usuario:** Gestión de Tipos de Períodos - Incorrecto
  - ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión de Tipos de Períodos seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá la codificación, nombre, seleccionar el período, descripción y el estado. Internamente los datos en el caso de ser incorrectos se emitirá un mensaje de error por cada campo.
  - ✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.
  - ✓ **Entrada:**
    - Después de la selección se mostrará la opción “Nuevo” para ingresar un nuevo tipo de período
    - El usuario deberá la codificación, nombre, seleccionar el período, descripción y el estado del registro
    - El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
    - El proceso de ingreso dentro de la Gestión de Tipos de Períodos se considera como finalizado.
  - ✓ **Resultado Esperado:**
    - Si la codificación o el nombre o el estado de la categoría son incorrectos no se registrará en el sistema.
  - ✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

- **Historia de Usuario:** Gestión de Encabezados del PAC - Correcto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión de Encabezados del PAC seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá la entidad, la codificación, seleccionar el tipo de período fiscal, observación y el estado. Internamente los datos serán validados antes de registrarlos en la base de datos.

- ✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.

- ✓ **Entrada:**

- Después de la selección se mostrará la opción “Nuevo” para realizar un nuevo de Encabezado del PAC.
- El usuario deberá la entidad, la codificación, seleccionar el tipo de período fiscal, observación y el estado del registro.
- El sistema validará internamente la información ingresada para posteriormente registrarla.
- El proceso de ingreso dentro de la Gestión de Encabezados del PAC se considera como finalizado.

- ✓ **Resultado Esperado:**

- Luego del ingreso del nuevo encabezado del PAC, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.

- ✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

- **Historia de Usuario:** Gestión de Encabezados del PAC - Incorrecto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Gestión de Encabezados del PAC seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá la entidad, la codificación, seleccionar el tipo de período fiscal, observación y el estado. Internamente los datos en el caso de ser incorrectos se emitirá un mensaje de error por cada campo.
- ✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.
- ✓ **Entrada:**
  - Después de la selección se mostrará la opción “Nuevo” para ingresar un nuevo encabezado del PAC
  - El usuario deberá la entidad, la codificación, seleccionar el tipo de período fiscal, observación y el estado del registro
  - El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
  - El proceso de ingreso dentro de la Gestión de Encabezados del PAC se considera como finalizado.
- ✓ **Resultado Esperado:**
  - Si la codificación o el nombre o el estado del encabezado del PAC son incorrectos no se registrará en el sistema.
- ✓ **Evaluación de la Prueba**
  - Prueba satisfactoria.
- **Historia de Usuario:** Gestión de Ítems del PAC - Correcto
  - ✓ **Descripción:** El usuario seleccionará una actividad y luego la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá elegir la categoría, subcategoría, producto, unidades, cantidad, seleccionar el período fiscal e ingresar el código POA. Internamente los datos serán validados antes de registrarlos en la base de datos.

✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar con una actividad asignada a su cuenta.

✓ **Entrada:**

- El usuario deberá seleccionar una actividad asignada.
- Después de la selección se mostrará la opción “Nuevo” para ingresar un ítem al PAC.
- Seleccionará la opción Nuevo y se desplegará una nueva ventana.
- Debe ingresar la categoría, subcategoría, producto, unidades, cantidad, seleccionar el período fiscal e ingresar el código POA.
- El sistema validará internamente la información ingresada para posteriormente registrarlo.
- El proceso de ingreso dentro de la Gestión de Ítems del PAC se considera como finalizado.

✓ **Resultado Esperado:**

- Luego del ingreso del nuevo ítem, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

• **Historia de Usuario:** Gestión de Ítems del PAC - Incorrecto

- **Descripción:** El usuario seleccionará una actividad y luego la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá elegir la categoría, subcategoría, producto, unidades, cantidad, seleccionar el período fiscal e ingresar el código POA. Internamente los datos serán validados y se emitirá mensajes de error indicando los campos incorrectos.

✓ **Condiciones de Ejecución:**

El usuario deberá proporcionar datos incorrectos.

✓ **Entrada:**

- El usuario deberá seleccionar una actividad asignada.
- Debe ingresar la categoría, subcategoría, producto, unidades, cantidad, seleccionar el período fiscal e ingresar el código POA.
- El sistema validará internamente la información ingresada y en el caso de detectar algún dato incorrecto emitirá mensajes de error indicando los campos incorrectos.
- El proceso de ingreso dentro de la Gestión de Ítems del PAC se considera como finalizado.

✓ **Resultado Esperado:**

- Si los datos del ítem son incorrectos no se registrará en el sistema.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

• **Historia de Usuario:** Asignar Perfiles a Usuarios - Correcto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Asignar Perfiles a Usuarios seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá seleccionar el módulo, el rol, el usuario y el estado. Internamente los datos serán validados antes de registrarlos en la base de datos.

✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.

✓ **Entrada:**

- Después de la selección se mostrará la opción “Nuevo” para realizar un perfil al usuario.
- El usuario deberá seleccionar el módulo, el rol, el usuario y el estado del registro.
- El sistema validará internamente la información ingresada para posteriormente registrarlo.

- El proceso de ingreso dentro de la Asignar Perfiles a Usuarios se considera como finalizado.
- ✓ **Resultado Esperado:**
  - Luego del ingreso del nuevo perfil de usuario, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.
- ✓ **Evaluación de la Prueba**
  - Prueba satisfactoria.
- **Historia de Usuario:** Asignar Perfiles a Usuarios - Incorrecto
  - ✓ **Descripción:** Una vez que el usuario selecciona la opción Asignar Perfiles a Usuarios seleccionará la opción “Nuevo” y aparece una nueva ventana en la cual el usuario deberá seleccionar el módulo, el rol, el usuario y el estado. Internamente los datos en el caso de ser incorrectos se emitirá un mensaje de error por cada campo.
  - ✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.
  - ✓ **Entrada:**
    - Después de la selección se mostrará la opción “Nuevo” para ingresar un nuevo perfil de usuario
    - El usuario deberá la entidad, la codificación, seleccionar el tipo de período fiscal, observación y el estado del registro
    - El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
    - El proceso de ingreso dentro de la Asignar Perfiles a Usuarios se considera como finalizado.
  - ✓ **Resultado Esperado:**
    - Si el módulo o el rol o el usuario o el estado del perfil de usuario son incorrectos no se registrará en el sistema.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

• **Historia de Usuario:** Configuración de Parámetros Iniciales - Correcto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Configuración de Parámetros Iniciales seleccionará la opción "Nuevo" y aparece una nueva ventana en la cual el usuario deberá la unidad ejecutora, unidad desconcentrada, subprograma, obra, geográfico, renglón auxiliar, organismo y el correlativo. Internamente los datos serán validados antes de registrarlos en la base de datos.

✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.

✓ **Entrada:**

- Después de la selección se mostrará la opción "Nuevo" para realizar el ingreso de los parámetros iniciales.
- El usuario deberá ingresar la unidad ejecutora, unidad desconcentrada, subprograma, obra, geográfico, renglón auxiliar, organismo y el correlativo del registro.
- El sistema validará internamente la información ingresada para posteriormente registrarlos.
- El proceso de ingreso dentro de la Configuración de Parámetros Iniciales se considera como finalizado.

✓ **Resultado Esperado:**

- Luego del ingreso de la configuración de parámetros iniciales, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

- **Historia de Usuario:** Configuración de Parámetros Iniciales - Incorrecto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Configuración de Parámetros Iniciales seleccionará la opción "Nuevo" y aparece una nueva ventana en la cual el usuario deberá la unidad ejecutora, unidad desconcentrada, subprograma, obra, geográfico, renglón auxiliar, organismo y el correlativo. Internamente los datos en el caso de ser incorrectos se emitirá un mensaje de error por cada campo.

- ✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.

- ✓ **Entrada:**

- Después de la selección se mostrará la opción "Nuevo" para registrar la configuración inicial.
- El usuario deberá ingresar la unidad ejecutora, unidad desconcentrada, subprograma, obra, geográfico, renglón auxiliar, organismo y el correlativo.
- El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
- El proceso de ingreso dentro de la Configuración de Parámetros Iniciales se considera como finalizado.

- ✓ **Resultado Esperado:**

- Si alguno de los campos ingresados son incorrectos no se registrará en el sistema.

- ✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

- **Historia de Usuario:** Revisión del PAC con el Rol Adquisiciones - Correcto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Revisión Adquisiciones se cargarán los ítems existentes (de haberlos) y seleccionará la celda a modificar en



el registro. Internamente los datos serán validados antes de registrarlos en la base de datos.

✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.

✓ **Entrada:**

- Seleccionará la actividad deseada.
- Seleccionará la celda a modificar en el registro.
- El usuario podrá modificar el tipo de compra, el producto, la unidad, del registro y editar el período cuando se va a ejecutar la compra.
- El sistema validará internamente la información ingresada para posteriormente registrarla.
- El proceso de ingreso dentro de la Revisión del PAC con el Rol Adquisiciones se considera como finalizado.

✓ **Resultado Esperado:**

- Luego del ingreso a la revisión de adquisiciones, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

• **Historia de Usuario:** Revisión del PAC con el Rol Adquisiciones - Incorrecto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Revisión Adquisiciones se cargarán los ítems existentes (de haberlos) y seleccionará la celda a modificar en el registro. Internamente los datos serán validados antes de registrarlos en la base de datos.

✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.

✓ **Entrada:**

- Seleccionará la actividad deseada.
- Seleccionará la celda a modificar en el registro.
- El usuario podrá modificar el tipo de compra, el producto, la unidad, del registro y editar el período cuando se va a ejecutar la compra.
- El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
- El proceso de ingreso dentro de la Revisión del PAC con el Rol Adquisiciones se considera como finalizado.

✓ **Resultado Esperado:**

- Si alguno de los campos modificados son incorrectos no se registrará en el sistema.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

• **Historia de Usuario:** Revisión del PAC con el Rol Presupuesto - Correcto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Revisión Presupuesto se cargarán los ítems existentes (de haberlos) y podrá modificar las celdas de la sección presupuestaria del PAC, seleccionará la celda a modificar en el registro. Internamente los datos serán validados antes de registrarlos en la base de datos.

✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.

✓ **Entrada:**

- Seleccionará la actividad deseada.
- Seleccionará la celda a modificar en el registro.

- El usuario podrá modificar unidad ejecutora, unidad desconcentrada, programa, subprograma, obra, geográfico, renglón auxiliar, fuente, organismo y el correlativo.
- El sistema validará internamente la información ingresada para posteriormente registrarla.
- El proceso de ingreso dentro de la Revisión del PAC con el Rol Presupuesto se considera como finalizado.

✓ **Resultado Esperado:**

- Luego del ingreso a la revisión de presupuesto, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

• **Historia de Usuario:** Revisión del PAC con el Rol Presupuesto - Incorrecto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Revisión Presupuesto se cargarán los ítems existentes (de haberlos) y podrá modificar las celdas de la sección presupuestaria del PAC, seleccionará la celda a modificar en el registro. Internamente los datos serán validados antes de registrarlos en la base de datos.

✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.

✓ **Entrada:**

- Seleccionará la actividad deseada.
- Seleccionará la celda a modificar en el registro.
- El usuario podrá modificar unidad ejecutora, unidad desconcentrada, programa, subprograma, obra, geográfico, renglón auxiliar, fuente, organismo y el correlativo.
- El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.

- El proceso de ingreso dentro de la Revisión del PAC con el Rol Presupuesto se considera como finalizado.
- ✓ **Resultado Esperado:**
  - Si alguno de los campos modificados son incorrectos no se registrará en el sistema.
- ✓ **Evaluación de la Prueba**
  - Prueba satisfactoria.
- **Historia de Usuario:** Revisión del PAC con el Rol Rector - Correcto
  - ✓ **Descripción:** Una vez que el usuario selecciona la opción Revisión Rector se cargarán los ítems existentes (de haberlos) y podrá modificar cualquiera de las celdas de los ítems del PAC, seleccionará la celda a modificar en el registro. Internamente los datos serán validados antes de registrarlos en la base de datos.
  - ✓ **Condiciones de Ejecución:**

Para que el usuario pueda acceder a esta pantalla deberá estar registrado como usuario y contar el perfil adecuado asignado a su cuenta.
  - ✓ **Entrada:**
    - Seleccionará la actividad deseada.
    - Seleccionará la celda a modificar en el registro.
    - El usuario podrá modificar cualquiera de las celdas que conforman los ítems en el PAC.
    - El sistema validará internamente la información ingresada para posteriormente registrarla.
    - El proceso de ingreso dentro de la Revisión del PAC con el Rol Rector se considera como finalizado.
  - ✓ **Resultado Esperado:**
    - Luego del ingreso a la revisión de adquisiciones, si el proceso de validación es correcto se registrará en la base de datos para posterior a ello seguir con el proceso.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

• **Historia de Usuario:** Revisión del PAC con el Rol Rector - Incorrecto

- ✓ **Descripción:** Una vez que el usuario selecciona la opción Revisión Rector se cargarán los ítems existentes (de haberlos) y podrá modificar cualquiera de las celdas de los ítems del PAC, seleccionará la celda a modificar en el registro. Internamente los datos serán validados antes de registrarlos en la base de datos.

✓ **Condiciones de Ejecución:**

El usuario debe proporcionar datos incorrectos.

✓ **Entrada:**

- Seleccionará la actividad deseada.
- Seleccionará la celda a modificar en el registro.
- El usuario podrá modificar cualquiera de las celdas que conforman los ítems en el PAC.
- El sistema validará internamente la información ingresada, en el caso de que ocurra un error se emitirá mensajes de error indicando los campos incorrectos.
- El proceso de ingreso dentro de la Revisión del PAC con el Rol Rector se considera como finalizado.

✓ **Resultado Esperado:**

- Si alguno de los campos modificados son incorrectos no se registrará en el sistema.

✓ **Evaluación de la Prueba**

- Prueba satisfactoria.

#### 4.1.5 Pruebas de Stress y Rendimiento

Para las pruebas de stress y rendimiento se utiliza las herramientas BadBoy y JMeter.

- La herramienta *BadBoy* permite capturar la navegación de la aplicación web y exportar a un archivo con formato *.jmx* para posteriormente utilizarlo en JMeter.
- La herramienta *JMeter* es una herramienta que permite simular carga pesada en servidores y pruebas de stress y rendimiento.

Para aplicar las pruebas en el Módulo de Adquisiciones se realiza lo siguiente:

- a) Con el software *BadBoy* se captura la navegación del Módulo de Adquisiciones y los parámetros que utiliza la aplicación en cada petición.
- b) Una vez que se inicia el proceso de captura de la secuencia de navegación se visualiza lo siguiente (Figura IV.38):

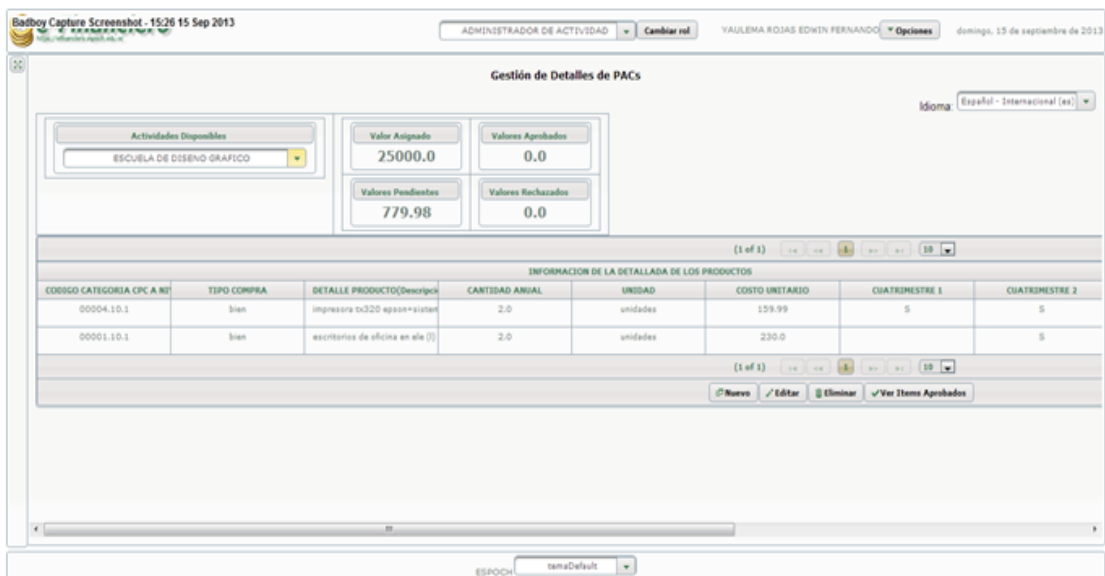
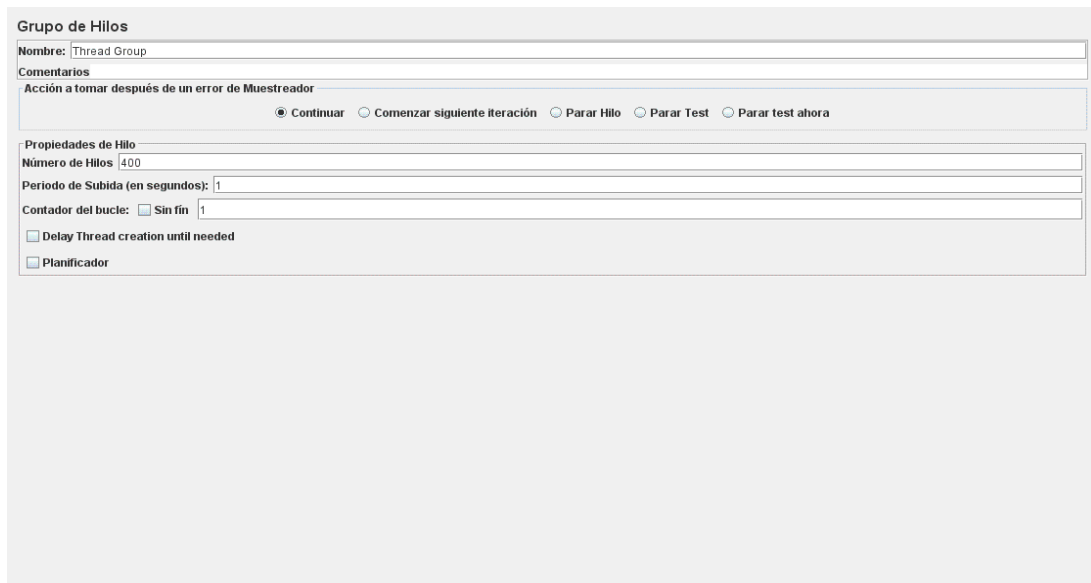


Figura IV.38 Captura de Navegación en BadBoy

- Una vez que se completado la navegación se exporta al archivo “*Script Test PAC-JSF.jmx*”, el cual contendrá todos los datos necesarios para después poder aplicar las pruebas de stress y rendimiento con JMeter.
- Para las pruebas de stress y rendimiento ejecutamos el software JMeter y luego se abre el archivo “*Script Test PAC-JSF.jmx*”. En Propiedades de Hilos en las opciones Número de Hilos se ingresará la cantidad de usuarios (en este caso será 400) que se desea simular y el Período de Subida (en segundos) se ingresa el período de demora en que se lanzará cada hilo (en este caso será 1).

- A continuación se visualiza la Figura de la configuración de Propiedades de Hilo (Figura IV.39):



Grupo de Hilos

Nombre: Thread Group

Comentarios

Acción a tomar después de un error de Muestreador

Continuar  Comenzar siguiente iteración  Parar Hilo  Parar Test  Parar test ahora

Propiedades de Hilo

Número de Hilos: 400

Periodo de Subida (en segundos): 1

Contador del bucle:  Sin fin 1

Delay Thread creation until needed

Planificador

Figura IV.39 Configuración de Propiedades de Hilos - JMeter

Al seleccionar los items “Reporte de Resumen” , “Graficos de Resultados” y “Ver Resultado en Árbol” se puede observar como avanza la prueba de stress y rendimiento del Módulo de Adquisiciones.

- **Reporte de Resumen:** En esta opción se muestra un conjunto de columnas las cuales se describen a continuación (Figura IV.40):
  - **Etiqueta:** El nombre de la muestra (conjunto de muestras).
  - **Muestras:** Cantidad total de veces que se realiza un request.
  - **Media:** Media aritmética de los tiempos de respuesta (response time) de la aplicación.
  - **Mediana:** Mediana aritmética de los tiempos de respuesta (response time) de la aplicación.
  - **Línea de Tiempo:** Tiempo de respuesta para los requests.
  - **Min:** Mínimo tiempo de respuesta para el request.
  - **%Error:** porcentaje de request con errores
  - **Rendimiento:** Es la cantidad de request que el servidor procesa por hora.
  - **Kb/Sec:** Cantidad de Kb que el servidor procesa por segundo

- **Max:** Máximo tiempo de respuesta para el request.

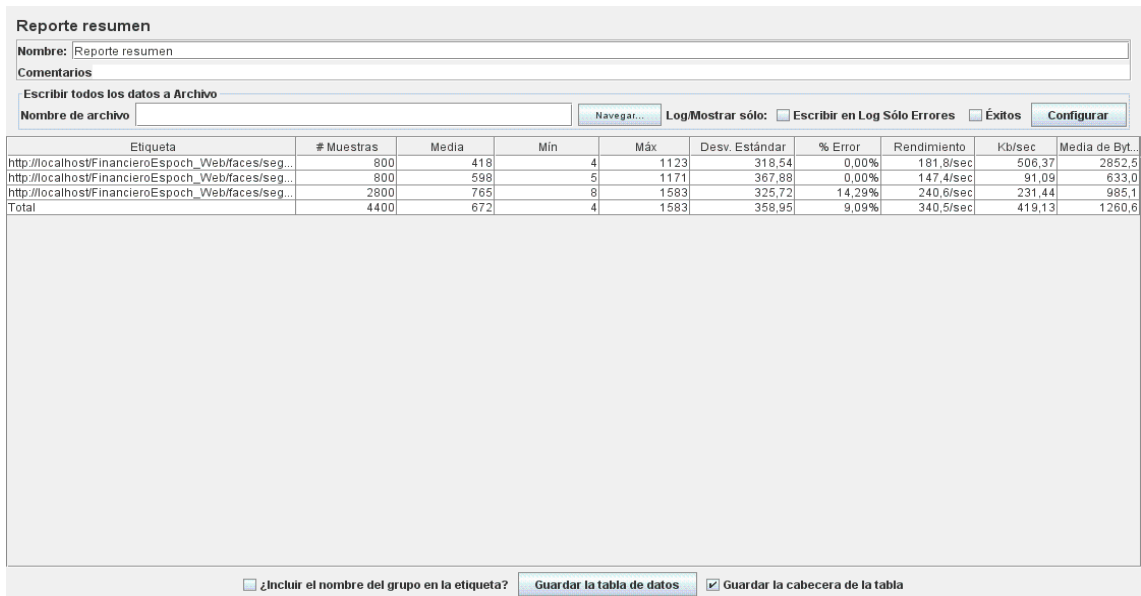


Figura IV.40 Reporte Resumen – JMeter

- **Figura de Resultados:**

Como visualizamos las pruebas se ha realizado con cierto porcentaje de error como se observa en la columna “%Error” el cual se encuentra 9,09%. La columna que indica el rendimiento indica que para la simulación con 400 usuarios relacionado en un tiempo de subida de un segundo el servidor es capaz de aceptar 340,5 peticiones por segundo. La latencia para cada conjunto de pruebas no supera el valor de 1193 milisegundos, esta se visualiza en el eje “y”, a continuación se describe la IV.41:

- **Latencia (eje y):** entendida como el tiempo de espera para la renderización de la página, el tiempo en obtener respuesta del servidor
- **Datos:** muestra los valores actuales de los datos.
- **Media:** representa la Media.
- **Mediana:** dibuja la Mediana.
- **Desviación:** muestra la Desviación Estándar (una medida de la Variación).
- **Rendimiento:** representa el número de muestras por unidad de tiempo.
- **Última muestra:** representa el tiempo transcurrido para la muestra en uso



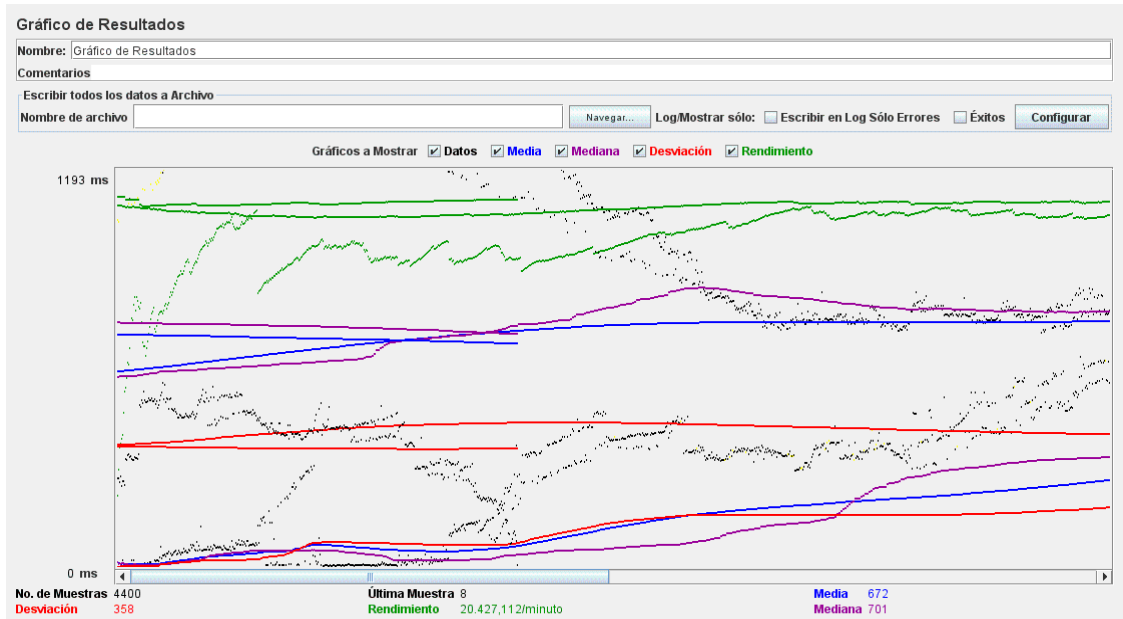


Figura IV.41 Figura de Resultados - JMeter

- **Árbol de Resultados:**

El Árbol de Resultados permite observar cada uno de los requests con cada una de sus acciones o pantallas a las que accedieron con JMeter. La columna “Estado” se simboliza de dos maneras, el símbolo verde indica que la petición se realizó con éxito y las que tienen el símbolo rojo indica que ocurrió algún error durante el request, como se visualiza en la Figura IV.42:

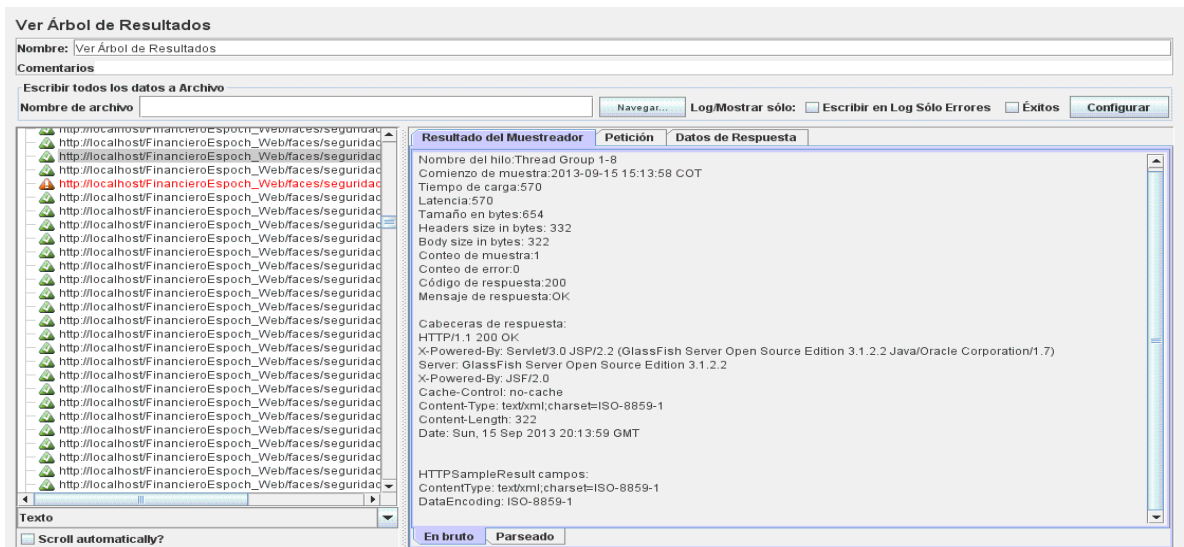


Figura IV.42 Árbol de Resultados

## RESUMEN

La presente investigación realiza el Análisis Comparativo de la Internacionalización de Aplicaciones Web en Application Services Providers (ASP) y Java Server Faces (JSF) aplicado al Módulo de Adquisiciones del Sistema Financiero de la Escuela Superior Politécnica de Chimborazo.

Se utilizó el método científico de la medición a partir del cual surge todo el complejo empírico estático por lo cual se desarrolló prototipos en ASP y JSF, los materiales utilizados en hardware: 1 computador; en software para JSF: GlassFish Server, Primefaces, Java, para ASP Visual Studio Net, Visual C#, Internet Information Servie. Las técnicas aplicadas son: pruebas de estrés y rendimiento, resultados estadísticos de las pruebas y la observación. Mediante las pruebas se analizaron los parámetros Rendimiento con sus indicadores: request del servidor/segundo 25% Kb procesados/segundo 15% y tiempo de respuesta 15%; el segundo parámetro el Tiempo de Desarrollo con sus indicadores: el tiempo de diseño de base de datos 15% y el tiempo de desarrollo en codificación 25%. Para JSF se obtuvo el 95% de aceptación frente al 48% obtenido en ASP, por lo que podemos concluir que una aplicación web internacionalizada en JSF es más eficiente en rendimiento y tiempo de desarrollo que aplicaciones internacionalizadas en ASP.

Se concluye que las aplicaciones web internacionalizadas permiten que el acceso a la información sea global porque el idioma deja de ser una limitante para el usuario.

Se recomienda que durante el desarrollo de sistemas informáticos se considere durante la fase de planificación y diseño si el sistema debe ser internacionalizado, así será más escalable.

## ABSTRACT

This research takes Comparative Analysis of Web Applications Internationalization Application Services Providers (ASP) and Java Server Faces (JSF) applied to the Acquisitions module Financial System of the Polytechnic School of Chimborazo.

We use the scientific gotten measurement, from which arises the complex empirical-statistical, so JSF prototypes developed and ASP, the materials used in the hardware elements: 1 computer and software to JSF : GlassFish Server, Primefaces , Java, ASP for: Visual Studio.Net, Visual C # , Internet Information Services .

The results of stress testing and performance of the prototypes were made based on the following parameters with performance indicators: server request/second 25 % processed Kb/second 15 % and 15 % response time, the second, and response time 15 %, the second parameter development Time with its indicators: design time database time 15% and 25% coding development. After calculating the total for each tool JSF was obtained for 95% acceptance compared to 48% obtained in ASP

It is recommended to programmers for developing internationalized computer system if the system is considered to be internationalized in the planning and design, and the system to better adapt to all requirements making it more scalable.

## CONCLUSIONES

- Después de analizar los conceptos de la internacionalización de aplicaciones web se debe tener en cuenta la gramática y la cultura de la región donde se desea implementarlo. Los componentes que hacen posible la internacionalización son los archivos de *Propiedades* ya que es donde se almacena la traducción de cada idioma, dichos archivos se gestionan a través de una clase dedicada a esa tarea y el objeto *Locale* de los objetos utilizados.
- Los parámetros analizados para realizar la comparación entre ASP y JSF fueron el Rendimiento y el Tiempo de Desarrollo con sus respectivos indicadores, los cuales sirvieron para determinar que tecnología es más eficiente para el desarrollo de aplicaciones web internacionalizadas.
- Se pudo evidenciar que el indicador Kb Procesados /Segundo del parámetro Rendimiento es más eficiente en ASP ya que se obtuvo el 20% frente al 15% obtenido en JSF.
- Para poder determinar la eficiencia de ASP y JSF en la internacionalización de aplicaciones web se desarrolló prototipos en cada tecnología, obteniendo el 95% de aceptación para JSF y el 48% para ASP, teniendo como resultado que JSF es más eficiente que ASP para desarrollar aplicaciones web internacionalizadas.
- El Módulo de Adquisiciones del Sistema Financiero de la ESPOCH fue desarrollado en JSF+PrimeFaces lo cual permitió un desarrollo más ágil del módulo.

- Se concluye que las aplicaciones web internacionalizadas permiten que el acceso a la información sea global porque el idioma deja de ser una limitante para el usuario.

## RECOMENDACIONES

- Conocer y analizar los conceptos de la Internacionalización de Aplicaciones Informáticas así como los componentes que hacen posible la internacionalización de la herramienta en la cual se desea realizar el desarrollo.
- Se recomienda desarrollar aplicaciones web con JSF+PrimeFaces, ya que JSF permite un desarrollo ágil y PrimeFaces facilita la creación de interfaces enriquecidas de manera sencilla.
- Durante las fases de planificación y diseño de los sistemas informáticos se debe considerar si el sistema a desarrollar debe o no ser internacionalizado ya que actualmente la demanda de acceso a la información exige romper las limitantes del idioma y así el sistema será más escalable.
- Para una completa internacionalización de aplicaciones web se recomienda realizar el diseño e implementación de una base de datos internacionalizada ya que para sistemas existentes no siempre es suficiente con los archivos de propiedades que contienen la traducción de los diferentes idiomas a utilizarse, esto puede ser revisado en una futura investigación.

## BIBLIOGRAFÍA

### 1.- ¿CÓMO CONFIGURAR EL IDIOMA EN JSF?

[http://www.programacion.com/articulo/internacionalizacion\\_de\\_programas\\_java\\_140/7](http://www.programacion.com/articulo/internacionalizacion_de_programas_java_140/7)

2013/04/10

### 2.- ¿QUÉ ES LA INTERNACIONALIZACIÓN DE APLICACIONES WEB?

[www.aipo.es/libro/doc/08Intern.doc](http://www.aipo.es/libro/doc/08Intern.doc)

2013/01/08

### 3.- ¿QUÉ ES LA LOCALIZACIÓN?

<http://e-sanchez.com/blog/2009/03/31/%C2%BFque-es-la-localizacion-l10n/>

2013/02/27

### 4.- AISLAMIENTO DE DATOS ESPECÍFICOS DE LA LOCALIZACIÓN CON JSF

<http://docs.oracle.com/javase/tutorial/i18n/resbundle/index.html>

2013/04/29

### 5.- APLICACIONES ASP.NET

<http://zarza.usal.es/~fgarcia/docencia/poo/04-05/Trabajos/ASP.NET.pdf>

2013/03/25

### 6.- ARCHIVOSRESOURCES.REX

<http://msdn.microsoft.com/en-us/library/ms227427.aspx>

2013/04/27

### 7.- ASPECTOS A TENER EN CUENTA PARA INTERNACIONALIZAR

- [http://courses.logos.it/pls/dictionary/linguistic\\_resources.cap\\_1\\_24\\_es?lang=es](http://courses.logos.it/pls/dictionary/linguistic_resources.cap_1_24_es?lang=es)
- [http://oa.upm.es/1802/1/PFC\\_EVA\\_JUAREZ\\_BARRENA.pdf](http://oa.upm.es/1802/1/PFC_EVA_JUAREZ_BARRENA.pdf)

2013/01/21

## **8.- ASPECTOS RELATIVOS AL IDIOMA**

[http://oa.upm.es/1802/1/PFC\\_EVA\\_JUAREZ\\_BARRENA.pdf](http://oa.upm.es/1802/1/PFC_EVA_JUAREZ_BARRENA.pdf)

2013/02/05

## **9.- BENEFICIOS DEL USO DE PAQUETES DE RECURSOS (RESOURCE BUNDLE)**

[https://en.wikipedia.org/wiki/Java\\_resource\\_bundle](https://en.wikipedia.org/wiki/Java_resource_bundle)

2013/05/03

## **10.- CARACTERÍSTICAS DE ASP**

[http://msdn.microsoft.com/es-ec/library/aa719552\(v=vs.71\).aspx](http://msdn.microsoft.com/es-ec/library/aa719552(v=vs.71).aspx)

2013/03/21

## **11.- CONFIGURACIONES REGIONALES Y REFERENCIAS CULTURALES**

[http://oa.upm.es/1802/1/PFC\\_EVA\\_JUAREZ\\_BARRENA.pdf](http://oa.upm.es/1802/1/PFC_EVA_JUAREZ_BARRENA.pdf)

2013/02/01

## **12.- CURRENTUICULTURE**

<http://msdn.microsoft.com/en-us/library/system.threading.thread.currentuiculture.aspx>

2013/04/25

## **13.- DEFINICIÓN DE APLICACIÓN WEB**

[http://www.prontosistemas.com/?page\\_id=30](http://www.prontosistemas.com/?page_id=30)

2013/01/04

## **14.- DEFINICIÓN DE INTERNACIONALIZACIÓN**

- <http://www.w3.org/International/questions/qa-i18n>
- <http://si.ua.es/va/documentacion/aplicacion-net/documentos/pdf/tema-6-globalizacion.pdf>

2013/02/14

## **15.- DEFINICIÓN DE LOCALIZACIÓN**

<http://e-sanchez.com/blog/2009/03/31/%C2%BFque-es-la-localizacion-l10n/>



2013/03/04

#### **16.- DEFINICIÓN DEL MODELO VISTA CONTROLADOR**

[http://msdn.microsoft.com/es-es/library/dd394709\(v=vs.100\).aspx](http://msdn.microsoft.com/es-es/library/dd394709(v=vs.100).aspx)

2013/03/15

#### **17.- DIFERENCIA ENTRE HORAS**

<http://msdn.microsoft.com/es-es/goglobal/bb688125.aspx>

2013/02/14

#### **18.- DIFERENCIAS DE FORMATOS DE MONEDAS**

<http://msdn.microsoft.com/es-es/goglobal/bb688126.aspx>

2013/02/18

#### **19.- DIFERENCIAS DE FORMATOS NUMÉRICOS**

<http://msdn.microsoft.com/es-es/goglobal/bb688127.aspx>

2013/02/22

#### **20.- DIFERENCIAS ENTRE FECHAS**

<http://msdn.microsoft.com/es-es/goglobal/bb688123.aspx>

2013/02/11

#### **21.- FORMATO CON JSF**

<http://docs.oracle.com/javase/tutorial/i18n/format/index.html>

2013/05/01

#### **22.- FUNCIONALIDAD TÉCNICA**

[http://oa.upm.es/1802/1/PFC\\_EVA\\_JUAREZ\\_BARRENA.pdf](http://oa.upm.es/1802/1/PFC_EVA_JUAREZ_BARRENA.pdf)

2013/03/04

#### **23.- INTRODUCCIÓN A JAVASERVER FACES**

<http://www.oracle.com/technetwork/java/javaee/overview-140548.html>

2013/03/04

#### **24.- JAVASERVER FACES**

[http://primefaces.googlecode.com/files/indexed\\_primefaces\\_users\\_guide\\_3\\_5.pdf](http://primefaces.googlecode.com/files/indexed_primefaces_users_guide_3_5.pdf)

2013/03/29

#### **25.- LA GLOBALIZACIÓN EN ASP**

<http://afana.me/post/aspnet-mvc-internationalization.aspx>

2013/04/15

#### **26.- LOCALIZACIÓN EN ASP**

<http://afana.me/post/aspnet-mvc-internationalization.aspx>

2013/04/19

#### **27.- MODELO, VISTA, CONTROLADOR**

<http://www.comusoft.com/modelo-vista-controlador-definicion-y-caracteristicas>

2013/03/19

#### **28.- OBJETO LOCALE**

<http://docs.oracle.com/javase/7/docs/api/java/util/Locale.html>

2013/05/03

#### **29.- RECOMENDACIONES PARA LOCALIZAR UNA APLICACIÓN**

[http://oa.upm.es/1802/1/PFC\\_EVA\\_JUAREZ\\_BARRENA.pdf](http://oa.upm.es/1802/1/PFC_EVA_JUAREZ_BARRENA.pdf)

2013/03/12

#### **30.- REPRESENTACIÓN DE CARACTERES**

<http://techtastico.com/post/tipos-de-codificacion-de-caracteres/>

2013/03/07

#### **31.- RESOURCE BUNDLE**

[http://en.wikipedia.org/wiki/Java\\_resource\\_bundle](http://en.wikipedia.org/wiki/Java_resource_bundle)

2013/05/01

### **32.- RESOURCEMANAGER**

[\[es/library/system.resources.resourcemanager\\(v=vs.110\\).aspx\]\(http://msdn.microsoft.com/es/library/system.resources.resourcemanager\(v=vs.110\).aspx\)](http://msdn.microsoft.com/es-</a></u></p></div><div data-bbox=)

2013/04/21

### **33.- TRABAJAR CON TEXTO CON JSF**

<http://docs.oracle.com/javase/tutorial/i18n/text/index.html>

2013/05/01

**ANEXOS**

# **MANUAL DE USUARIO**

## **1. GENERALIDADES**

### **INTRODUCCIÓN:**

El Módulo de Adquisiciones del Sistema Financiero de la ESPOCH surgió de la necesidad de automatizar el proceso de elaboración y consolidación del PAC (Plan Anual de Contrataciones) Institucional. El proceso involucra a las Actividades (Departamentos, Escuelas, Facultades, etc.) ya que cada Actividad debe elaborar su PAC. Cada PAC generado por las Actividades debe ser revisado por el Departamento de Adquisiciones, Departamento Financiero y por el Sr. Rector de la ESPOCH por lo que se ha implementado un rol para cada departamento anteriormente nombrado, de esta manera normar el acceso a las diferentes secciones del PAC lo cual facilita y organiza de mejor manera el proceso de revisión.

El Módulo de Adquisiciones del Sistema Financiero de la ESPOCH tiene la capacidad de emitir reportes en formato de Microsoft Excel los cuales pueden ser con el Formato Institucional (de manejo interno) así como con el Formato Solicitado por el Portal de Compras Públicas.

El presente manual pretende transmitir la estructura, conceptos e información básica de las operaciones del Módulo de Adquisiciones, con la finalidad de que conociendo su funcionamiento los usuarios puedan utilizarlo de manera efectiva.

Por lo mencionado anteriormente, es de suma importancia, leer el manual que se detalla a continuación antes y durante el manejo del sistema. El manual comienza explicando el ingreso al sistema, para luego ir explicando uno a uno los submódulos que conforman el Módulo y todas sus funciones.

### **OBJETIVO:**

El principal objetivo del presente manual es ayudar a los usuarios que utilizan éste Módulo ya sea de las Actividades, Departamento de Adquisiciones, Departamento Financiero y al Sr. Rector sobre el uso y funcionamiento, dar a conocer cómo opera el Módulo y así mismo una idea clara del alcance que tiene, a través de la explicación ilustrada de cada una de las opciones que lo conforman.

### **¿A QUIÉN VA DIRIGIDO EL MANUAL?**

Este manual está dirigido a los usuarios involucrados en el proceso de elaboración y consolidación del PAC, que hacen uso del Módulo de Adquisiciones del Sistema Financiero de la ESPOCH.

## **2. INTRODUCCIÓN AL MÓDULOS DE ADQUISICIONES**

En este manual se detallará los requerimientos recomendados de hardware y software para el correcto funcionamiento del Sistema. Además, se detallan los elementos del entorno con el que el usuario debe relacionarse para el manejo del Sistema.

## 2.1 REQUERIMIENTOS DE HARDWARE

Los requisitos recomendados son:

- Usuario:
  - Tener conocimientos básicos de Computación
    - Sistemas Operativos: Windows, Linux o MAC.
    - Microsoft Excel o Similares.
  - Conocimientos del Sistema que se va utilizar para procesar la información.
  
- Hardware (Mínimo Recomendado):
  - Procesador Pentium IV 1Ghz o superior.
  - Memoria RAM de 1 Gb o Superior.
  - Espacio en disco 4Gb o Superior.
  
- Software (Cualquiera de los mencionados):
  - Navegador Web (Internet Explorer, Google Chrome, Mozilla Firefox, Opera o Safari).

### 3. USO DEL MÓDULO DE ADQUISICIONES

#### 3.1 INGRESO AL MÓDULO

En ésta sección se detallará brevemente la forma en que el usuario debe acceder al Sistema:

**3.1.1 Abrir el Navegador:** El módulo puede funcionar en la mayoría de navegadores, para ello se debe abrir un navegador disponible en su computador (Figura N° 1):



Figura N° 1 Ejecutar Navegador Web

**3.1.2 Digitar la Dirección en la Barra de Direcciones del Navegador:** Debemos digitar la siguiente dirección: [adquisiciones.esPOCH.edu.ec](http://adquisiciones.esPOCH.edu.ec) (Figura N° 2):

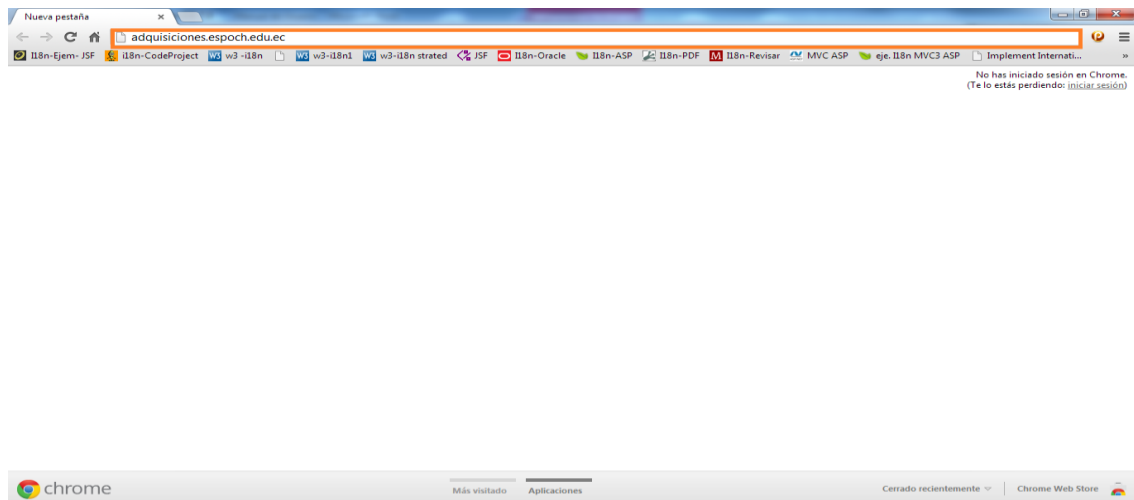


Figura N° 2 Ingresar Dirección en la Barra de direcciones

**3.1.3 La Pantalla de Bienvenida:** Luego de autenticarse se mostrará la pantalla de bienvenida, en la cual debe presionarse el hipervínculo **Iniciar Sesión** (Figura N° 3):



Figura N° 3 Enlace para ir a Pantalla de Autenticación

**3.1.4 Autenticación de Usuarios:** Nos pedirá el nombre de **Usuario** y **Contraseña**. Para acceder debemos presionar **Ingresar** (Figura N° 4):



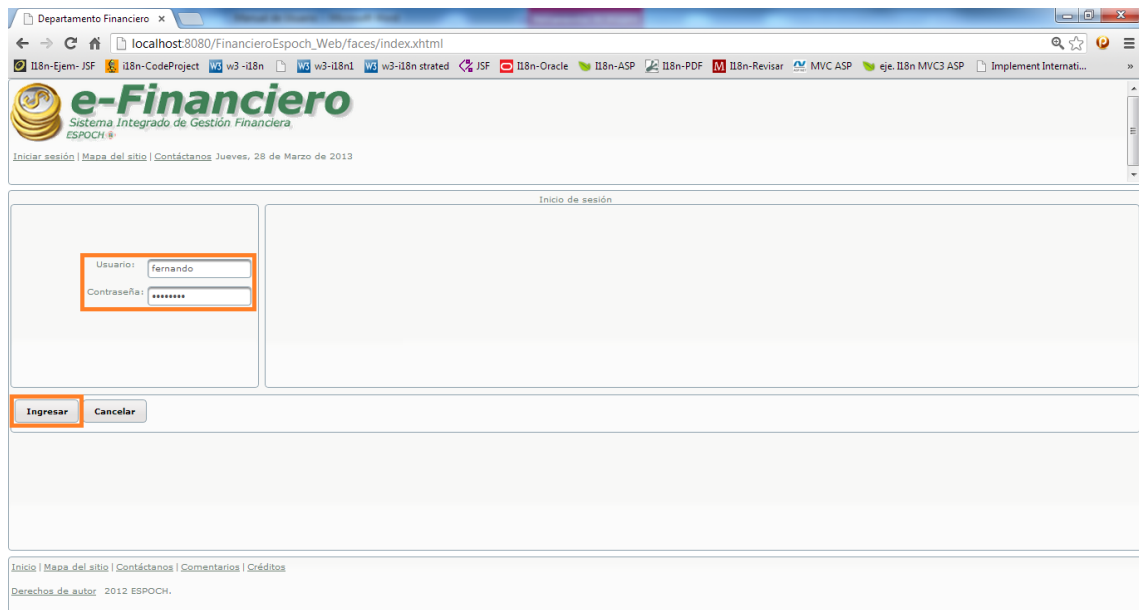


Figura N° 4 Pantalla de Autenticación

## 3.2 GESTIÓN DEL CATÁLOGO DE PRODUCTOS – ROL ADQUISICIONES

Para poder gestionar toda la información del Catálogo se debe acceder con el rol de Administrador de Adquisiciones como se muestra en la Figura N° 5:

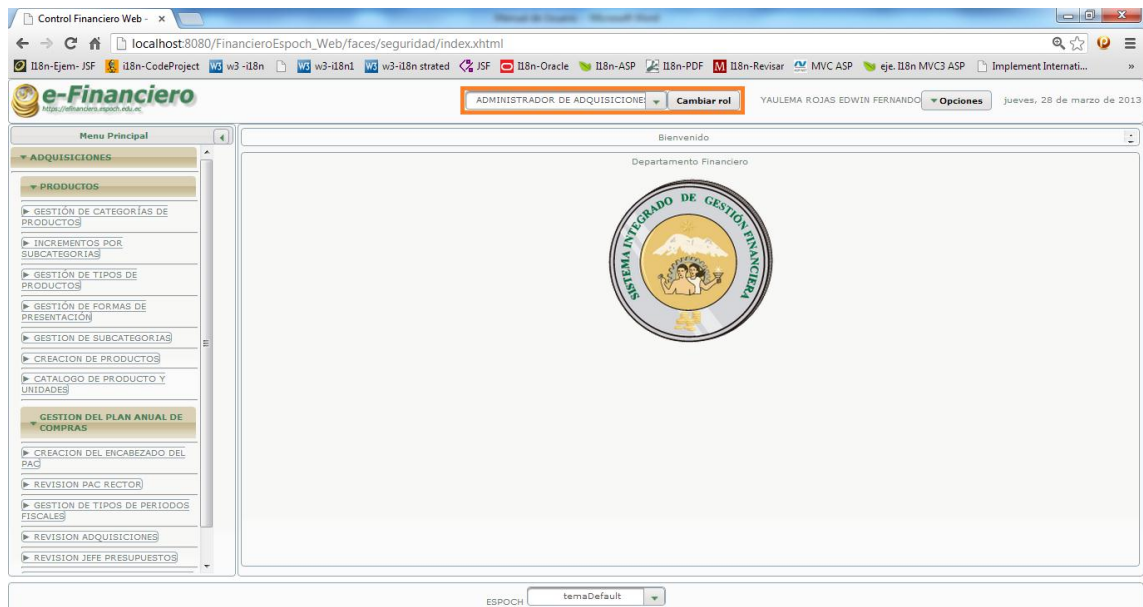


Figura N° 5 Pantalla de Selección Rol Adquisiciones

### 3.2.1 GESTIÓN DE CATEGORÍAS

La clasificación de productos en el catálogo es de suma importancia ya que en base a ello se obtendrá toda la información requerida para la elaboración del PAC Institucional. Debemos elegir la opción **GESTIÓN DE CATEGORÍAS** (Figura N° 6).

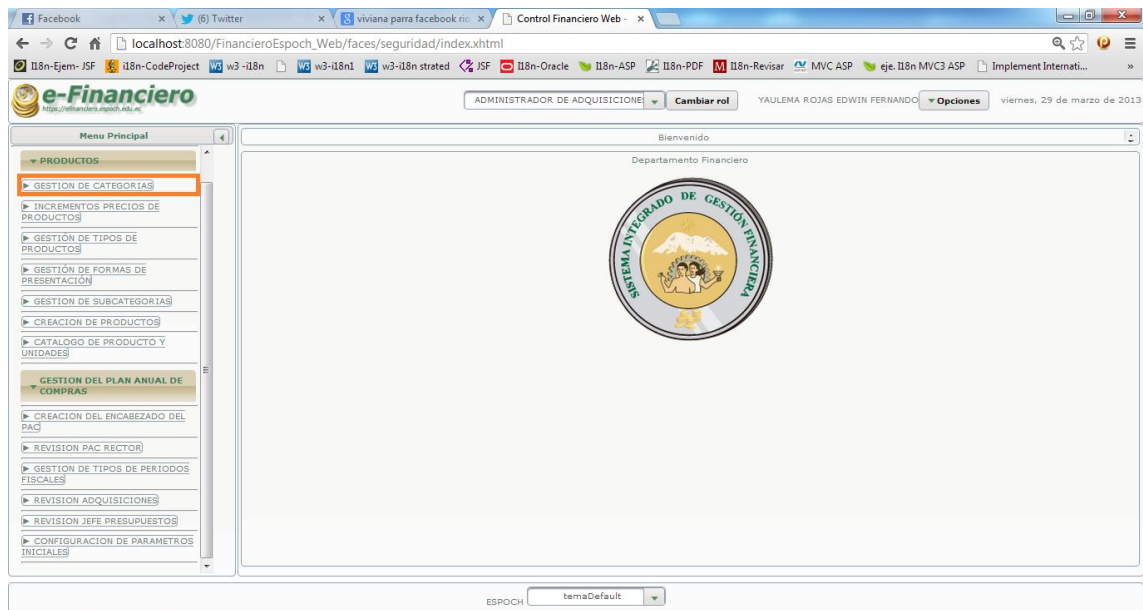


Figura N° 6 Selección Opción Gestión de Categoría de Productos.

Ahora se mostrará la pantalla principal para de la Gestión de Categorías en la que se podrá realizar búsquedas de las categorías almacenadas (Figura N° 7):

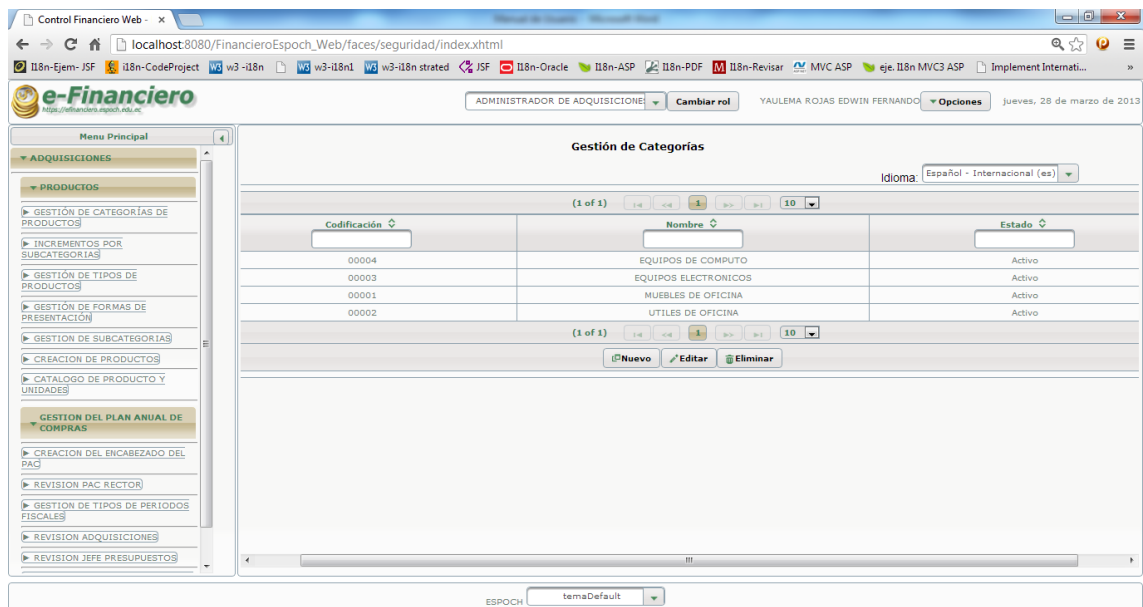


Figura N° 7 Pantalla Principal de Gestión de Categorías

### 3.2.1.1 Nueva Categoría:

Esta opción nos permite registrar una nueva Categoría, la cual estará disponible en el Catálogo (Figura N° 8).

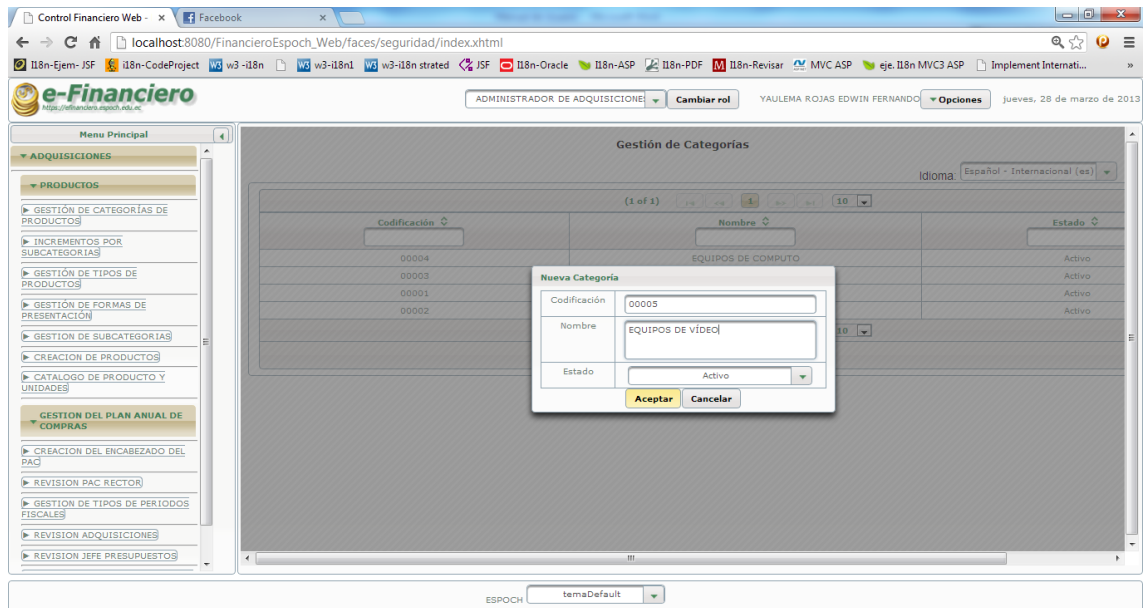


Figura Nº 8 Nueva Categoría

### 3.2.1.2 Editar/Dar de Baja Categoría:

Esta opción nos permite registrar una editar los datos de una Categoría, la cual estará disponible en el Catálogo (Figura Nº 9).

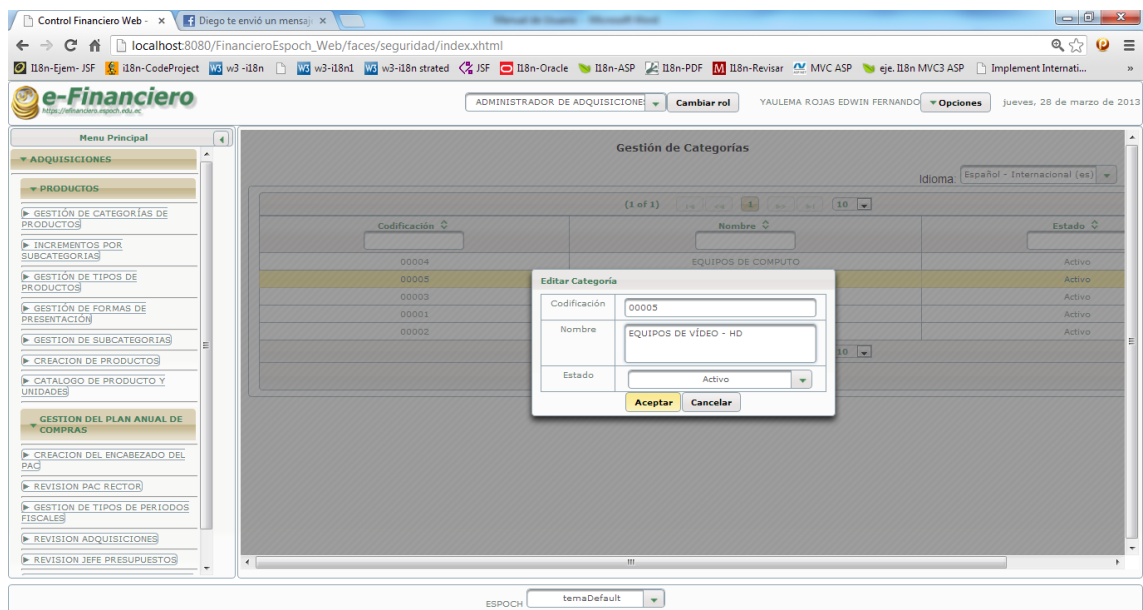


Figura Nº 9 Editar Datos Categoría

### 3.2.1.3 Eliminar Categoría:

Esta opción nos permite eliminar los datos de una Categoría, ésta acción se realizará siempre que la categoría no forme parte en otro registro (Figura Nº 10).

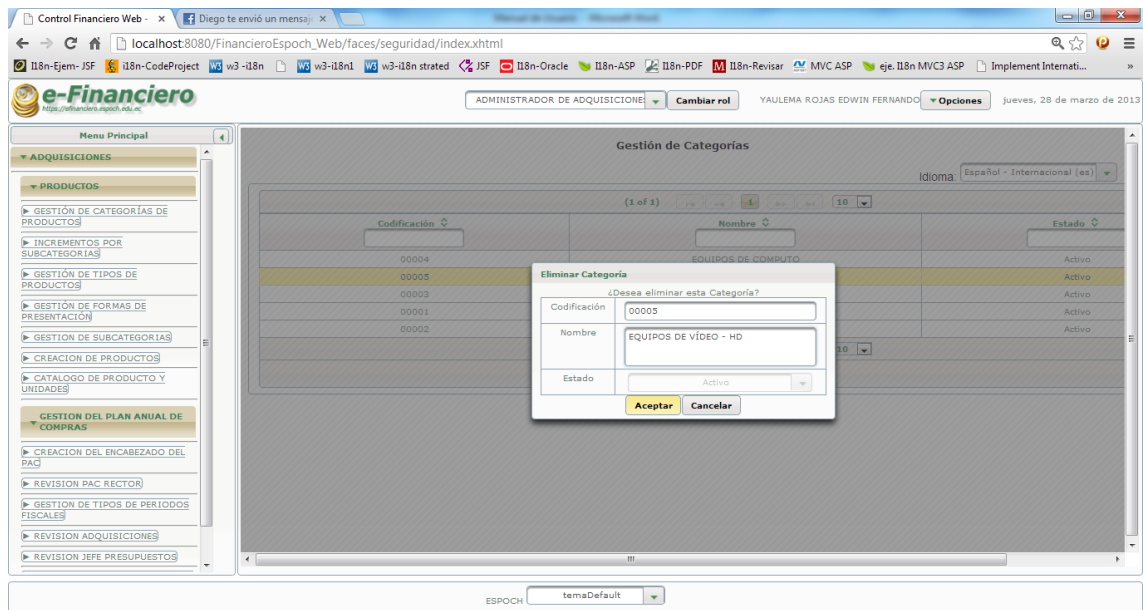


Figura N° 10 Eliminar Datos de Categoría

### 3.2.1.4 Notificaciones de Error:

Al ingresar una nueva categoría o al editarla si los campos ingresados no cumplen con el formato o son obligatorios el Módulo de Adquisiciones emitirá notificaciones (Figura N° 11):

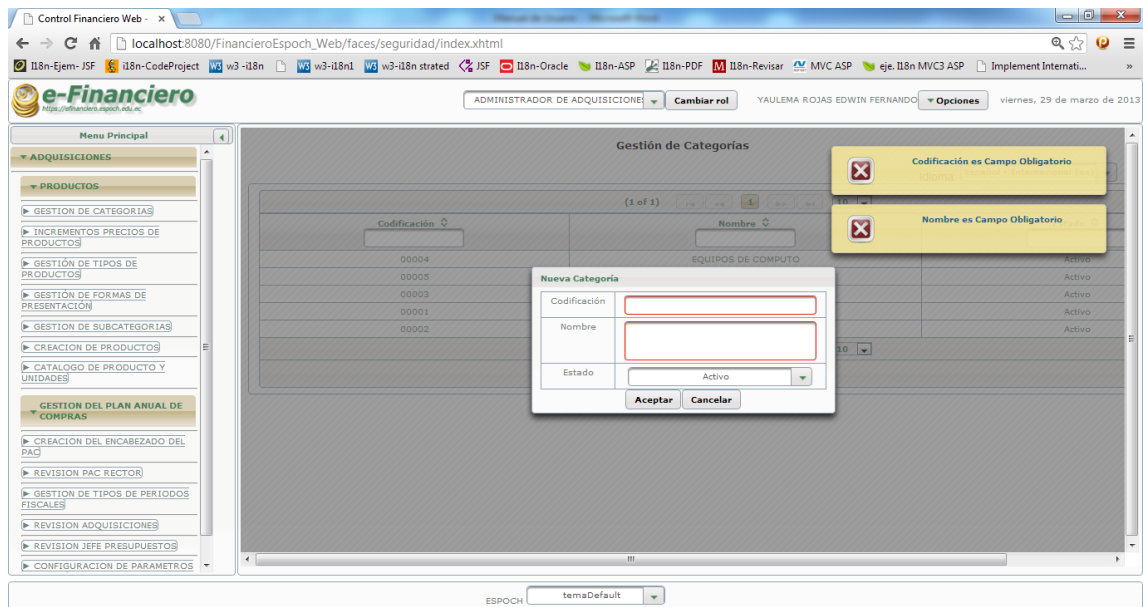


Figura N° 11 Notificaciones de Error de Categoría

### 3.2.2 GESTIÓN DE SUBCATEGORÍAS

La clasificación de productos en el catálogo es de suma importancia ya que en base a ello se obtendrá toda la información requerida para la elaboración del PAC Institucional. Debemos elegir la opción **GESTIÓN DE SUBCATEGORÍAS** (Figura N° 12).

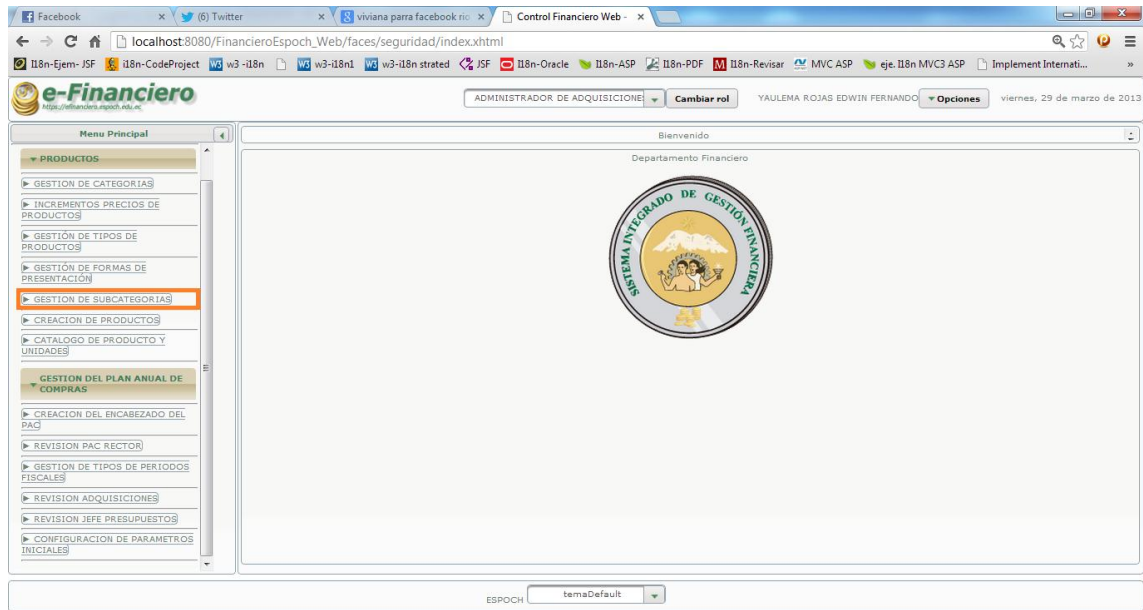


Figura N° 12 Selección Opción Gestión de Subcategorías.

Ahora se mostrará la pantalla principal para de la Gestión de Subcategorías en la que se podrá realizar búsquedas de las categorías almacenadas (Figura N° 13):

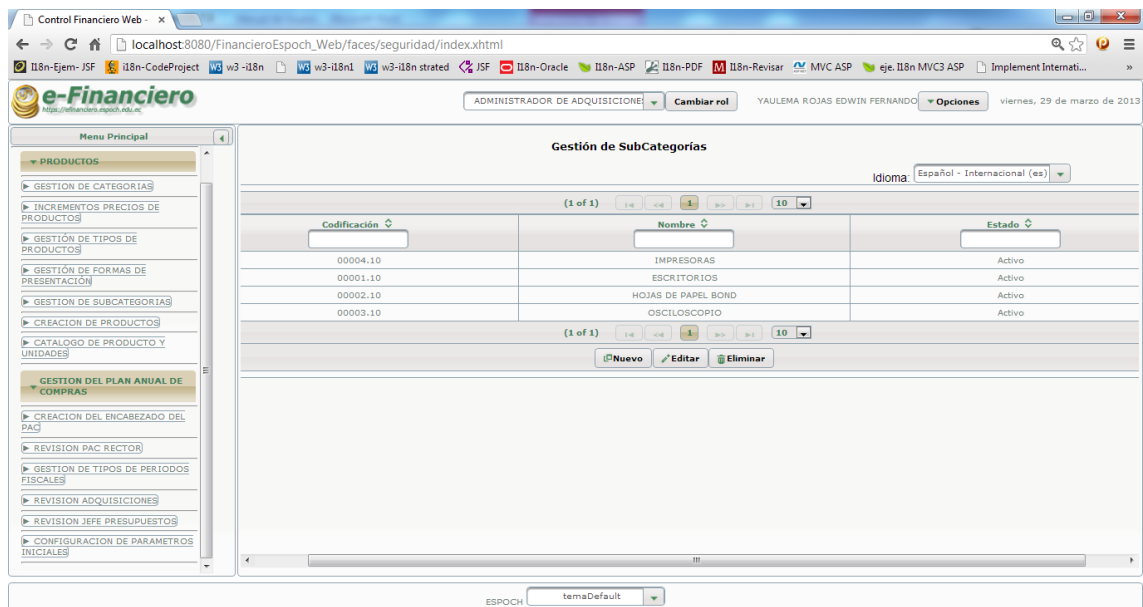


Figura N° 13 Pantalla Principal de Gestión de Subcategorías

### 3.2.2.1 Nueva Subcategoría:

Esta opción nos permite registrar una nueva Subcategoría, la cual estará disponible en el Catálogo (Figura N° 14).

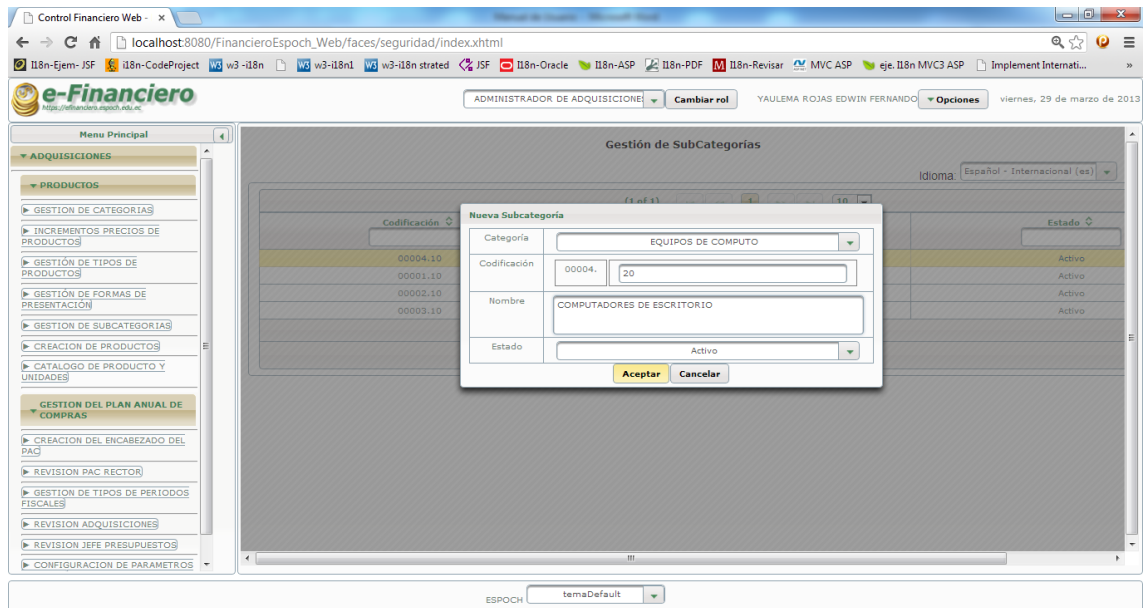


Figura Nº 14 Nueva Subcategoría

### 3.2.2.2 Editar/Dar de Baja Subcategoría:

Esta opción nos permite registrar una editar los datos de una Subcategoría, la cual estará disponible en el Catálogo (Figura Nº 15).

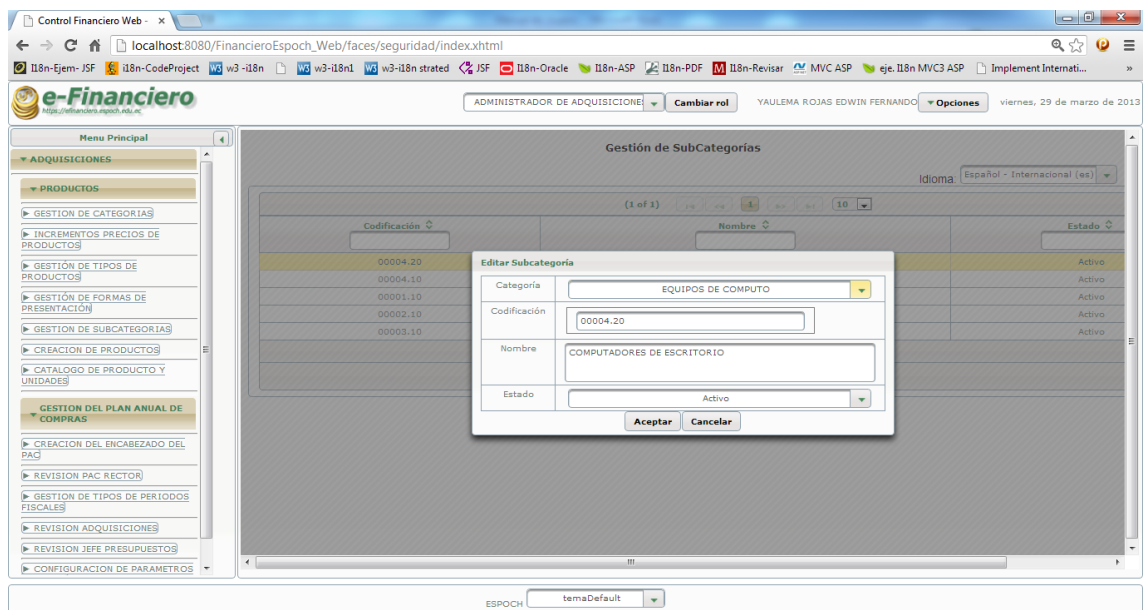


Figura Nº 15 Editar Datos Subcategoría

### 3.2.2.3 Eliminar Subcategoría:

Esta opción nos permite eliminar los datos de una Categoría, ésta acción se realizará siempre que la categoría no forme parte en otro registro (Figura Nº 10).



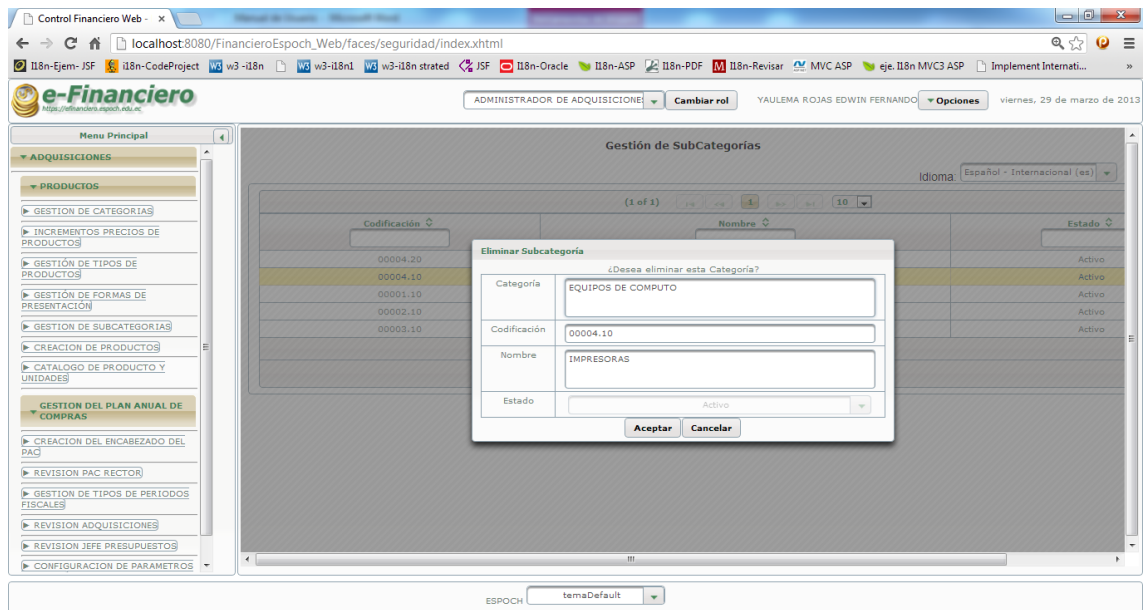


Figura N° 16 Eliminar Datos de Subcategoría

### 3.2.2.4 Notificaciones de Error:

Al ingresar una nueva categoría o al editarla si los campos ingresados no cumplen con el formato o son obligatorios el Módulo de Adquisiciones emitirá notificaciones (Figura N° 17):

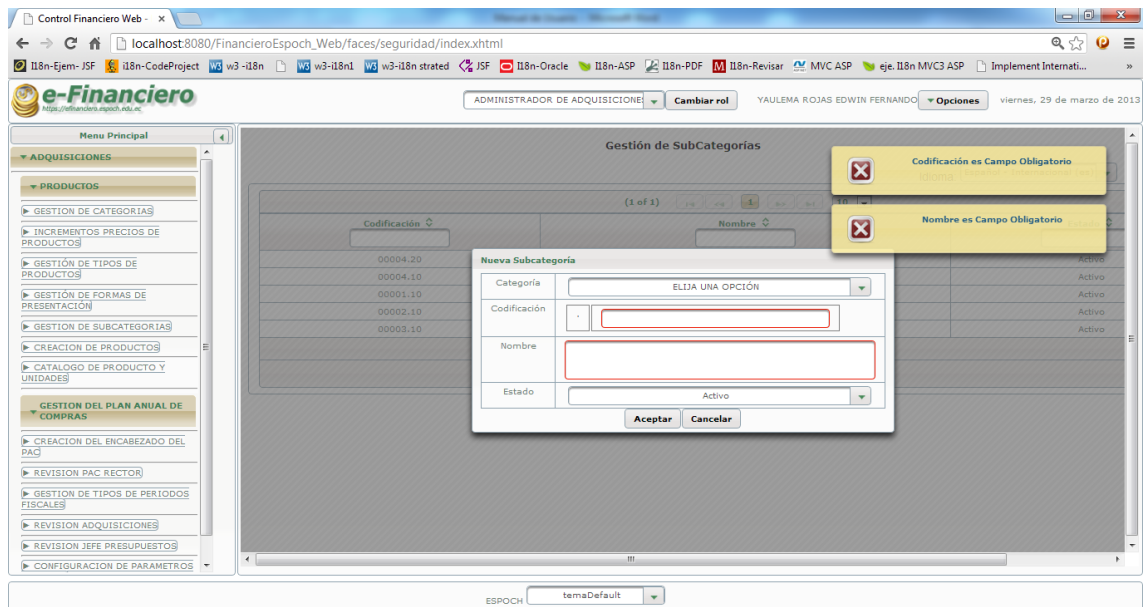


Figura N° 17 Notificaciones de Error de Subcategoría

### 3.2.3 GESTIÓN DE FORMA DE PRESENTACIÓN (UNIDADES)

La clasificación de productos en el catálogo es de suma importancia ya que en base a ello se obtendrá toda la información requerida para la elaboración del PAC Institucional. Debemos elegir la opción **GESTIÓN DE FORMAS DE PRESENTACIÓN** (Figura N° 18).

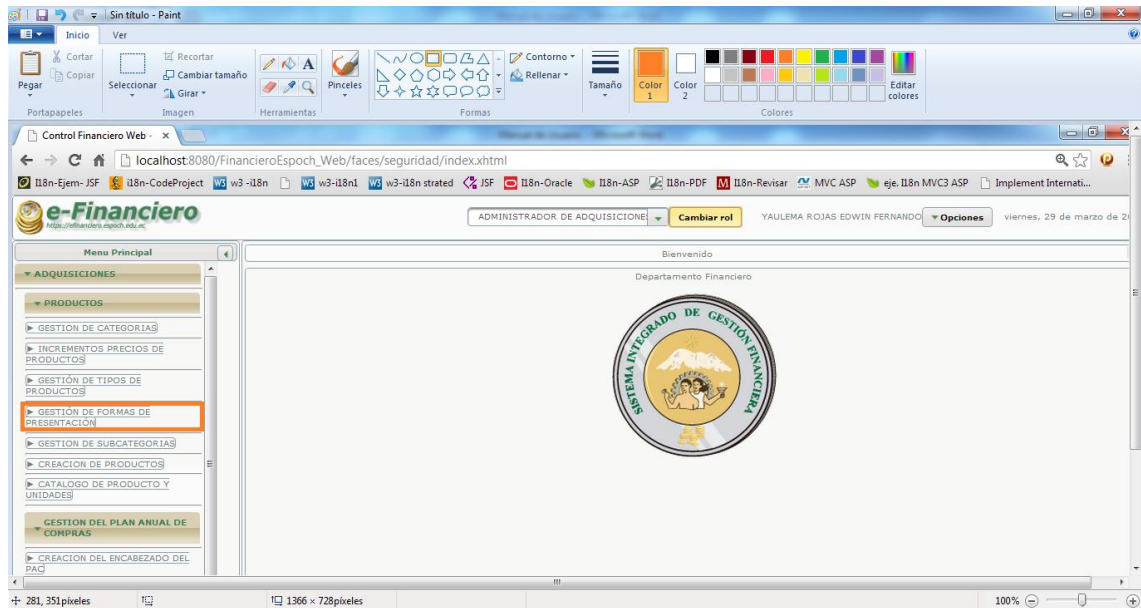


Figura Nº 18 Selección Opción Gestión de Formas de Presentación (Unidades).

Ahora se mostrará la pantalla principal para la Gestión de Formas de Presentación en la que se podrá realizar búsquedas de las categorías almacenadas (Figura Nº 19):

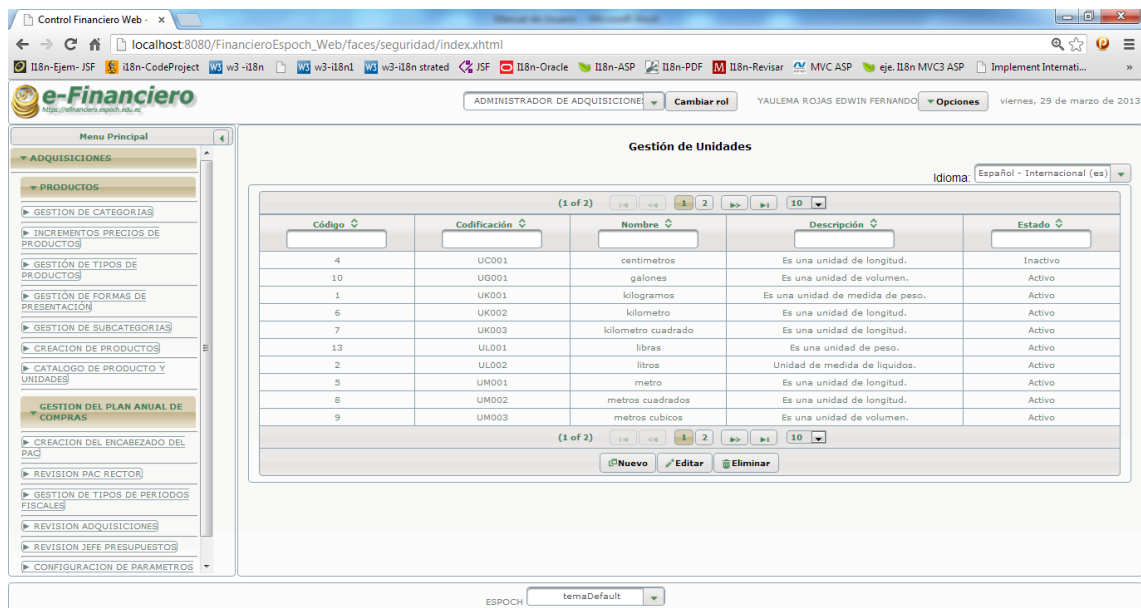


Figura Nº 19 Pantalla Principal de Gestión de Formas de Presentación (Unidades)

### 3.2.3.1 Nueva Forma de Presentación (Unidad):

Esta opción nos permite registrar una nueva Unidad, la cual estará disponible en el Catálogo (Figura Nº 20).



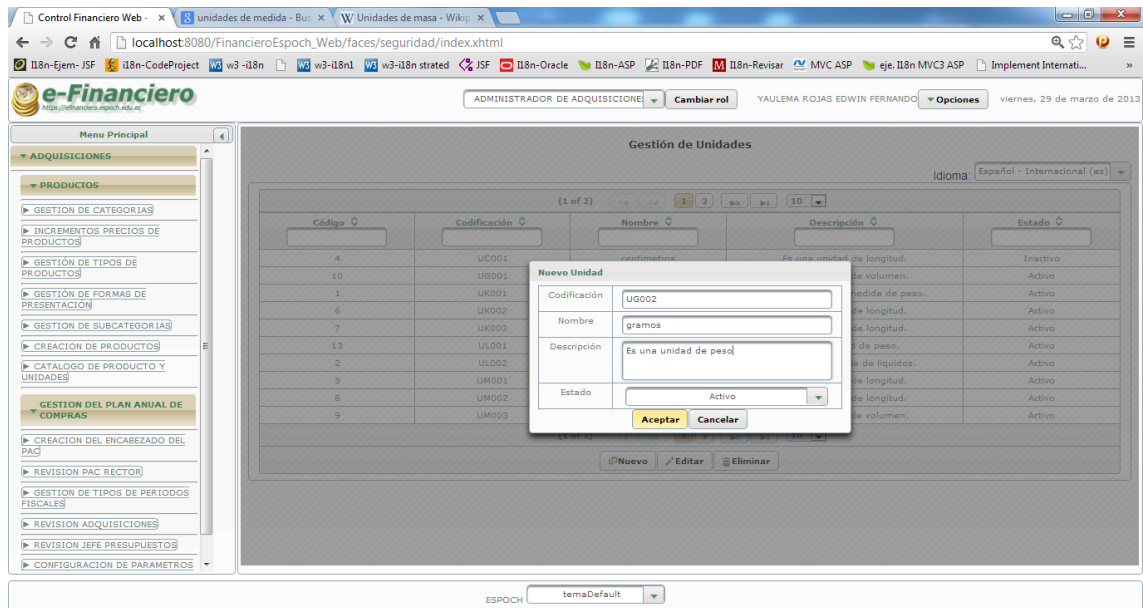


Figura N° 20 Nueva Forma de Presentación (Unidad)

### 3.2.3.2 Editar/Dar de Baja Formas de Presentación (Unidad):

Esta opción nos permite registrar una editar los datos de una Unidad, la cual estará disponible en el Catálogo (Figura N° 21).

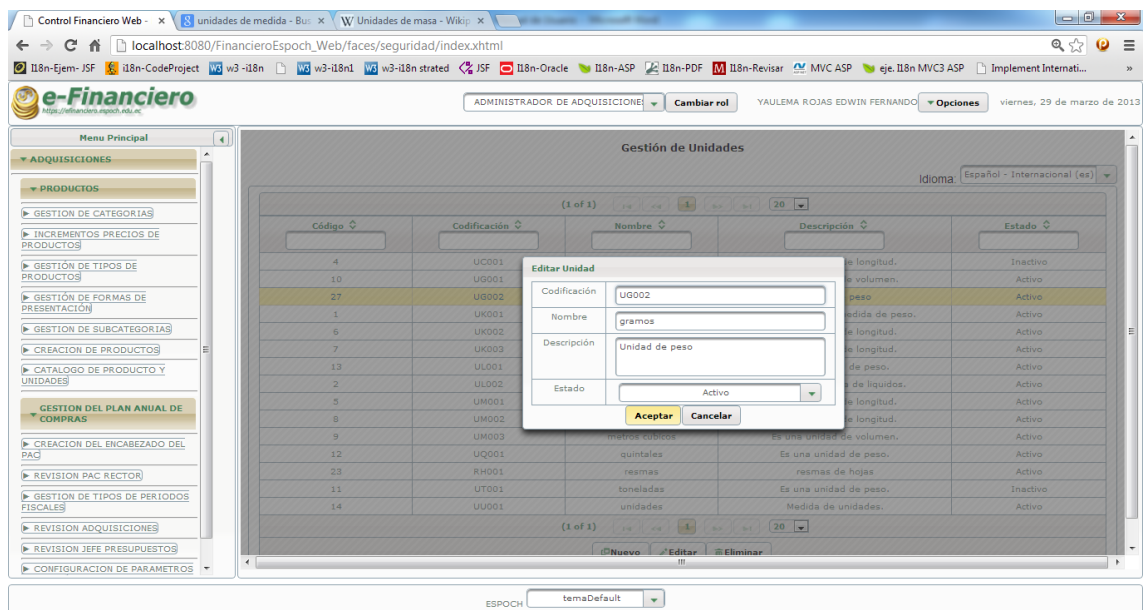


Figura N° 21 Editar Datos Forma de Presentación (Unidades)

### 3.2.3.3 Eliminar Forma de Presentación (Unidad):

Esta opción nos permite eliminar los datos de una Unidad, ésta acción se realizará siempre que la categoría no forme parte en otro registro (Figura N° 22).

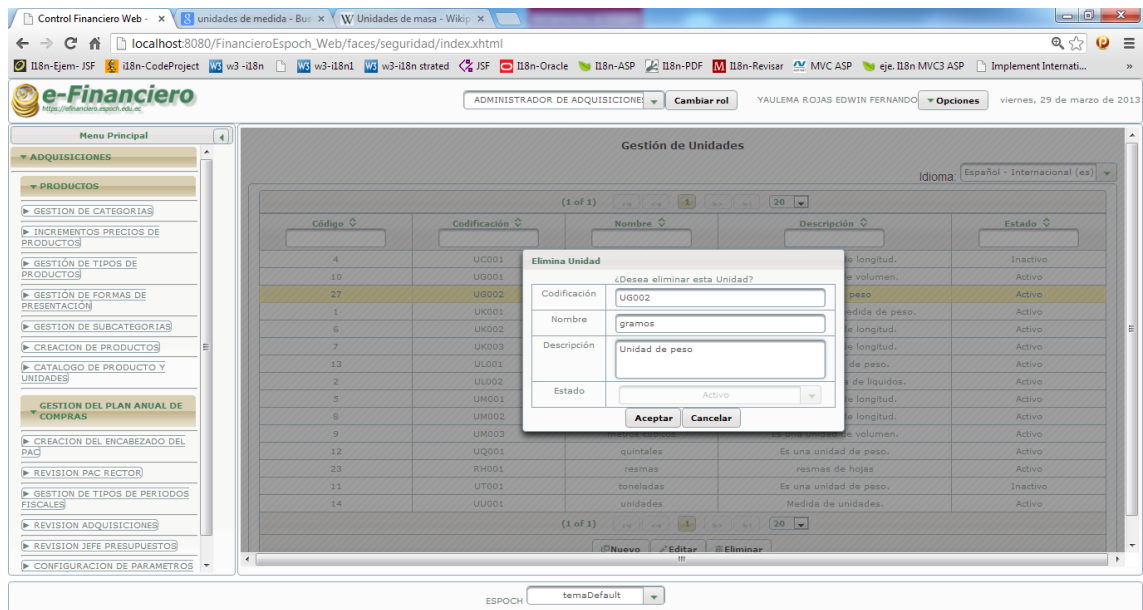


Figura N° 22 Eliminar Datos de Formas de Presentación (Unidades)

### 3.2.3.4 Notificaciones de Error:

Al ingresar una nueva Unidad o al editarla si los campos ingresados no cumplen con el formato o son obligatorios el Módulo de Adquisiciones emitirá notificaciones (Figura N° 23):

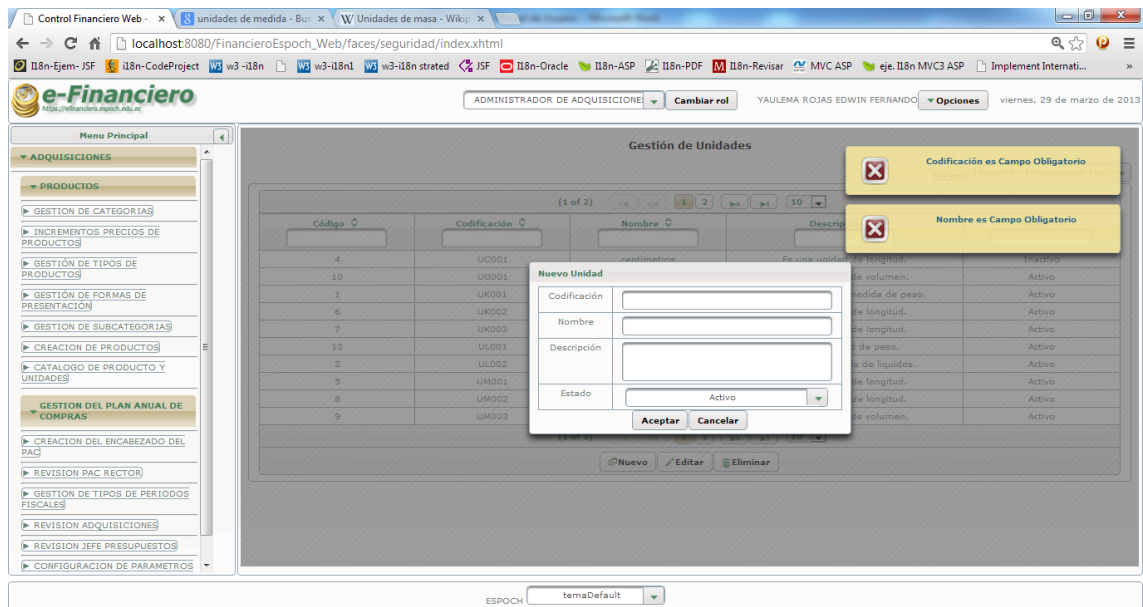


Figura N° 23 Notificaciones de Error de Formas de Presentación (Unidades)

### 3.2.4 GESTIÓN DE TIPOS DE PRODUCTOS

La clasificación de productos en el catálogo es de suma importancia ya que en base a ello se obtendrá toda la información requerida para la elaboración del PAC Institucional. Debemos elegir la opción **GESTIÓN DE TIPOS DE PRODUCTOS** (Bienes, Servicios, etc.) - (Figura N° 24).

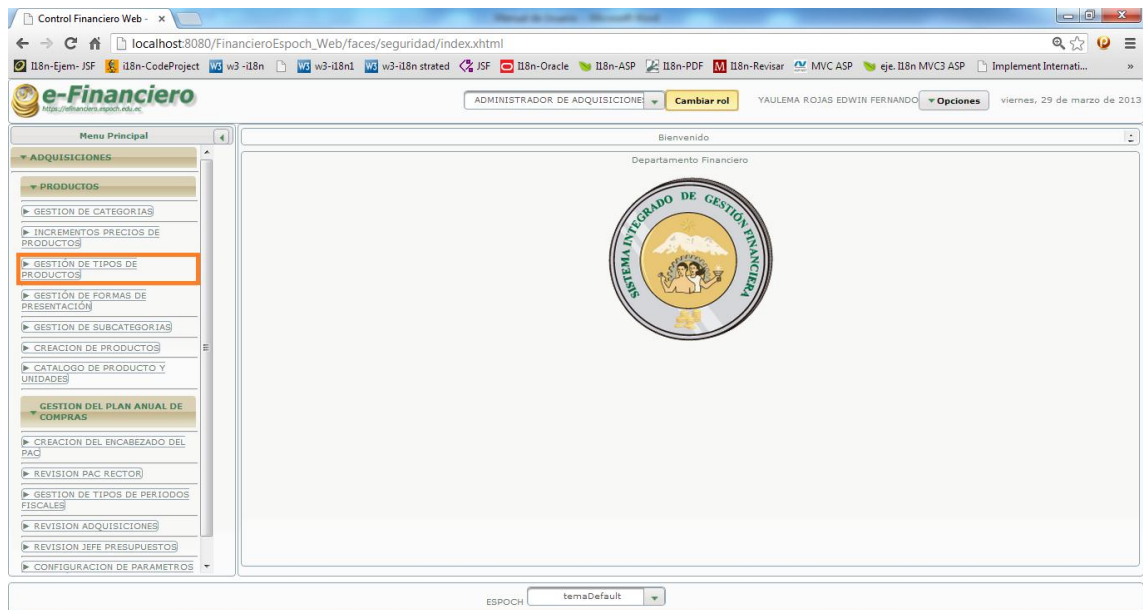


Figura N° 24 Gestión de Tipos de Productos.

Ahora se mostrará la pantalla principal para de la Gestión de Tipos de Productos en la que se podrá realizar búsquedas de las categorías almacenadas (Figura N° 25):

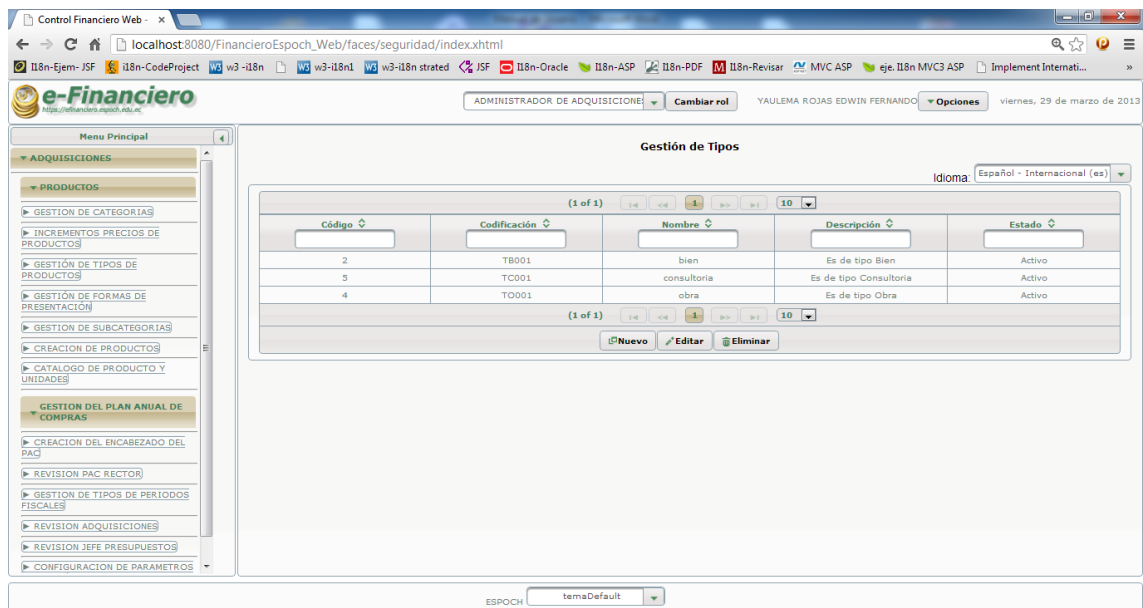


Figura N° 25 Pantalla Principal de Gestión de Tipos de Productos

### 3.2.4.1 Nuevo Tipo de Producto:

Esta opción nos permite registrar un nuevo Tipo de Producto, la cual estará disponible en el Catálogo (Figura N° 26).

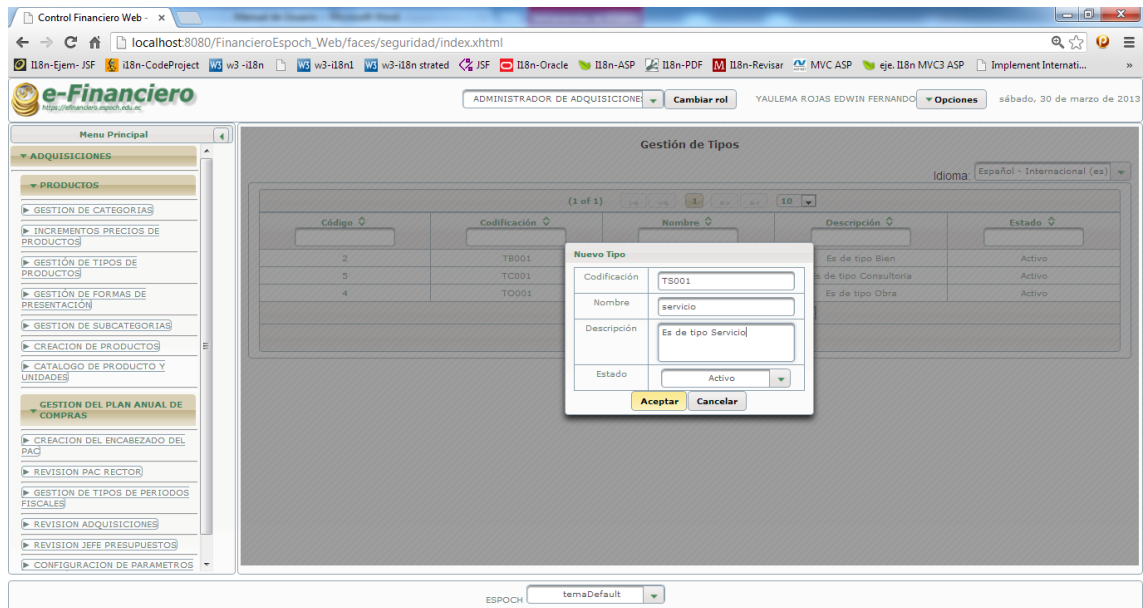


Figura N° 26 Nuevo Tipo de Producto

### 3.2.4.2 Editar/Dar de Baja Tipo de Producto:

Esta opción nos permite registrar una editar los datos de una Unidad, la cual estará disponible en el Catálogo (Figura N° 27).

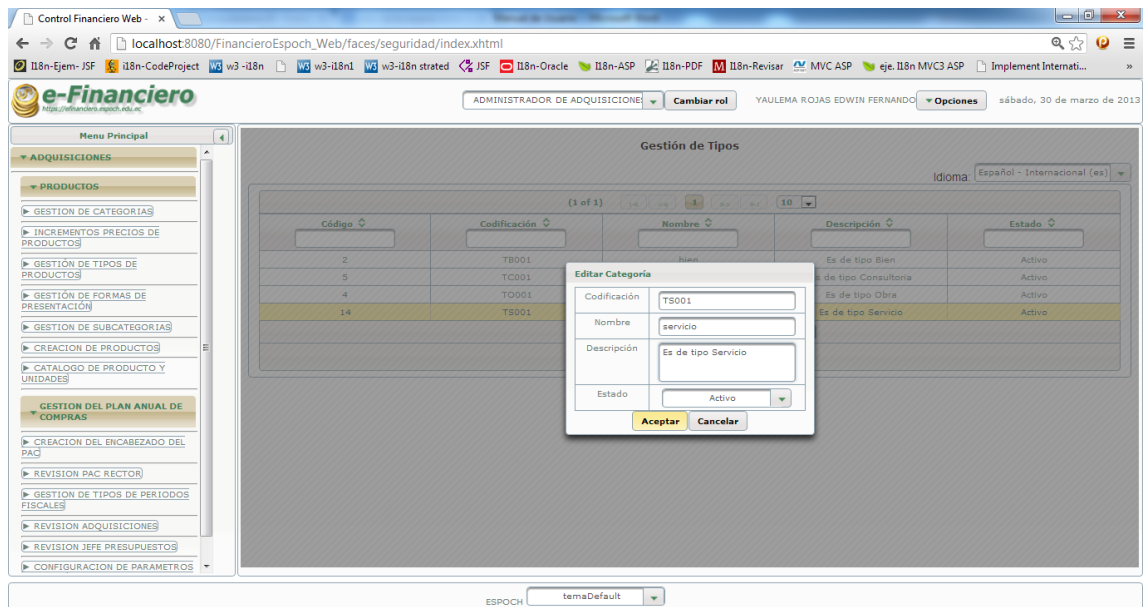


Figura N° 27 Editar Datos Tipo de Producto

### 3.2.4.3 Eliminar Tipo de Producto:

Esta opción nos permite eliminar los datos de una Unidad, ésta acción se realizará siempre que la categoría no forme parte en otro registro (Figura N° 28).

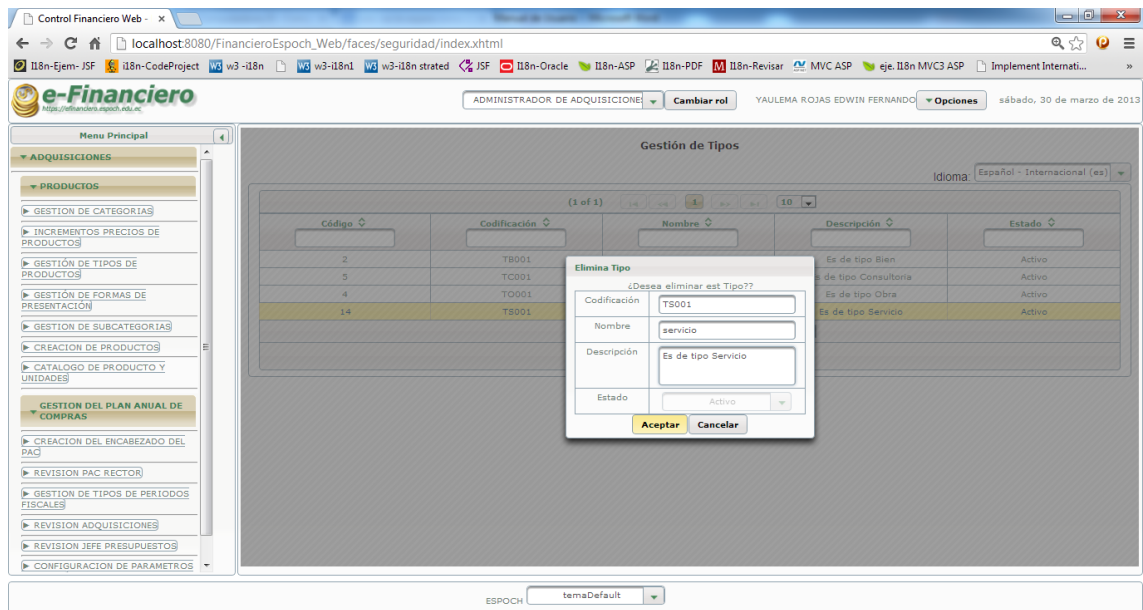


Figura N° 28 Eliminar Datos de Tipos de Productos

#### 3.2.4.4 Notificaciones de Error:

Al ingresar una nueva Unidad o al editarla si los campos ingresados no cumplen con el formato o son obligatorios el Módulo de Adquisiciones emitirá notificaciones (Figura N° 29):

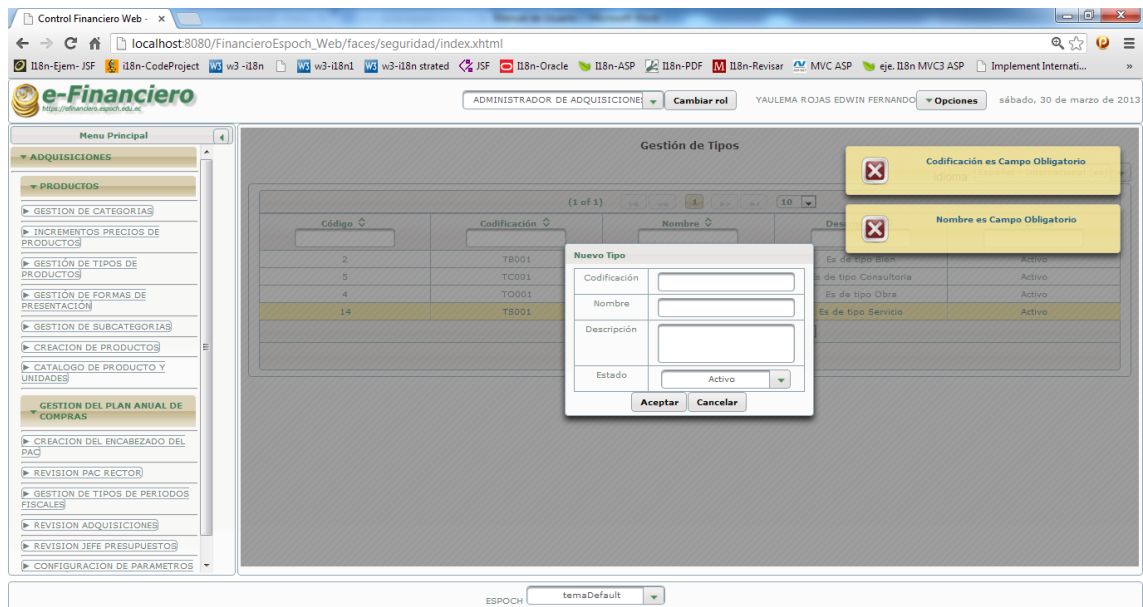


Figura N° 29 Notificaciones de Error de Tipos de Productos

### 3.2.5 GESTIÓN DE PRODUCTOS

La clasificación de productos en el catálogo es de suma importancia ya que en base a ello se obtendrá toda la información requerida para la elaboración del PAC Institucional. Debemos elegir la opción **GESTIÓN DE PRODUCTOS** (Bienes, Servicios, etc.) - (Figura N° 30).

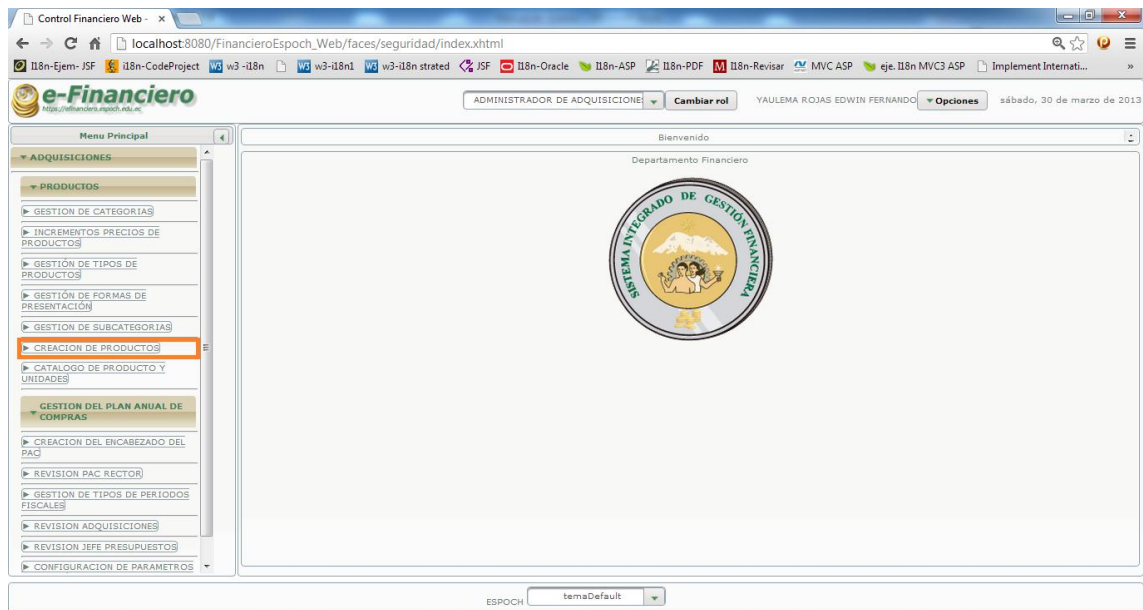


Figura N° 30 Gestión de Productos.

Ahora se mostrará la pantalla principal para de la Gestión de Productos en la que se podrá realizar búsquedas de las categorías almacenadas (Figura N° 31):

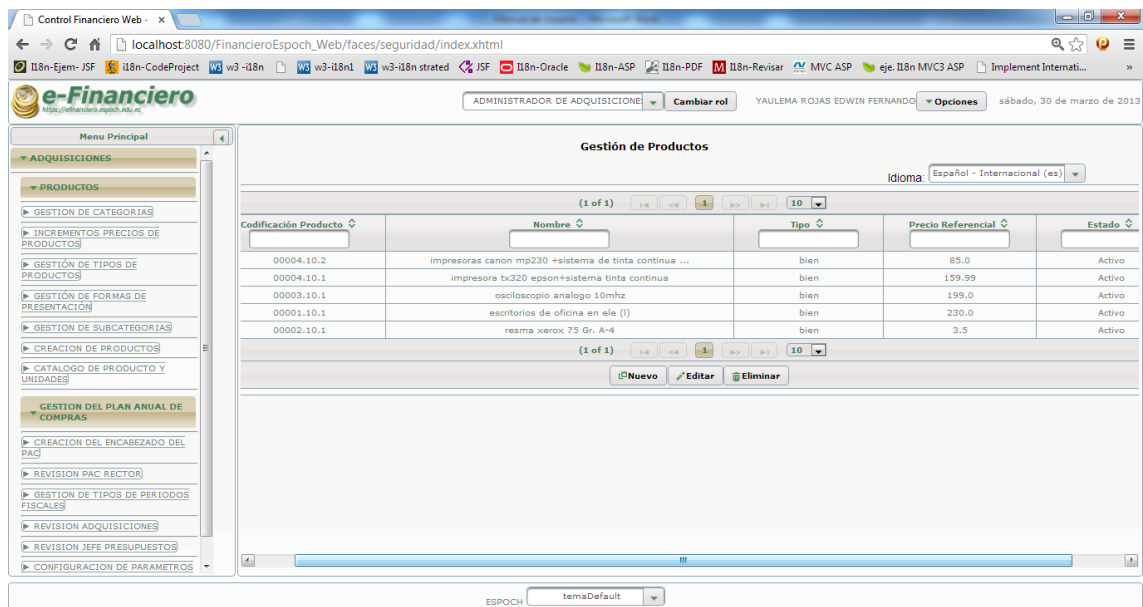


Figura N° 31 Pantalla Principal de Gestión de Productos

### 3.2.5.1 Nuevo Producto:

Esta opción nos permite registrar una nuevo Producto, el cual estará disponible en el Catálogo (Figura N° 32).



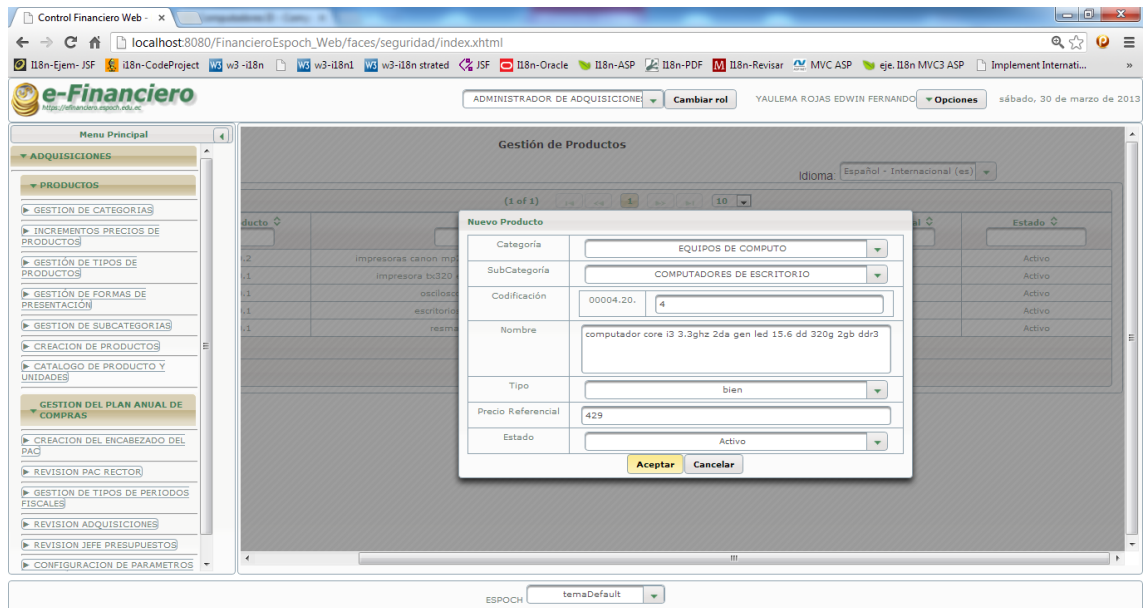


Figura N° 32 Nuevo Producto

### 3.2.5.2 Editar/Dar de Baja de Producto:

Esta opción nos permite registrar una editar los datos de un Producto, el cual estará disponible en el Catálogo (Figura N° 33).

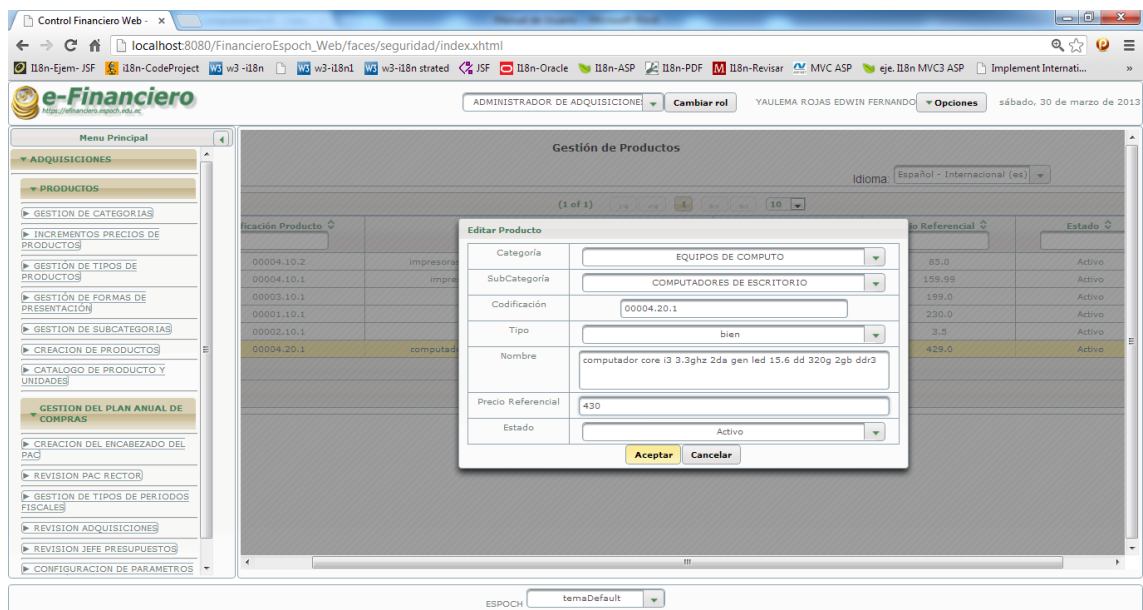


Figura N° 33 Editar Datos de Producto

### 3.2.5.3 Eliminar de Producto:

Esta opción nos permite eliminar los datos de un Producto, ésta acción se realizará siempre que el producto no forme parte en otro registro (Figura N° 34).

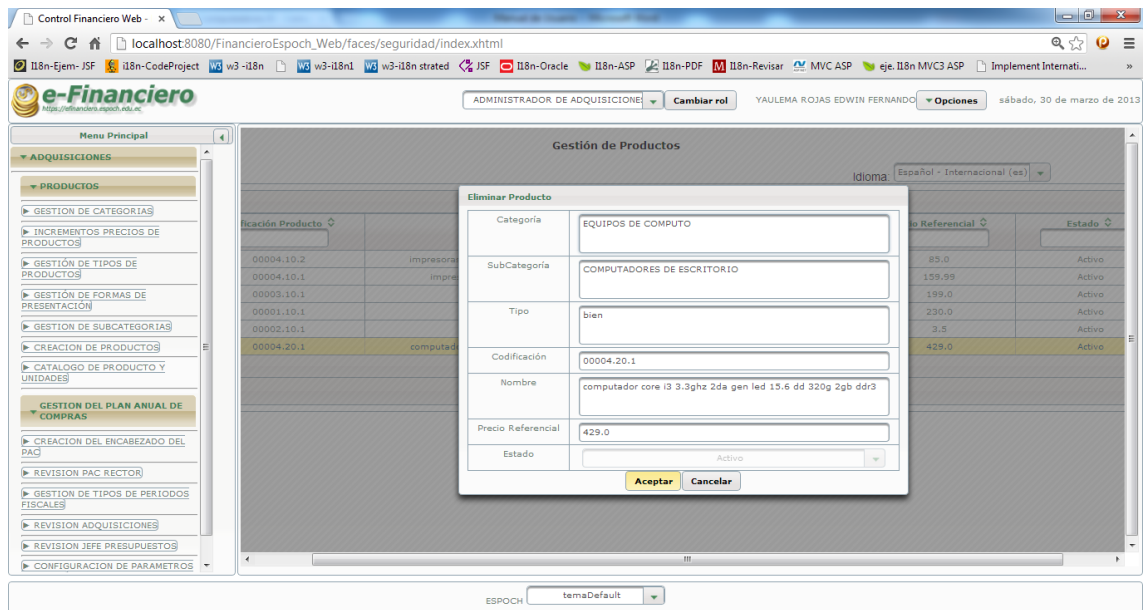


Figura N° 34 Eliminar Datos de Productos

### 3.2.5.4 Notificaciones de Error:

Al ingresar un nuevo Producto o al editarla si los campos ingresados no cumplen con el formato o son obligatorios el Módulo de Adquisiciones emitirá notificaciones (Figura N° 35):

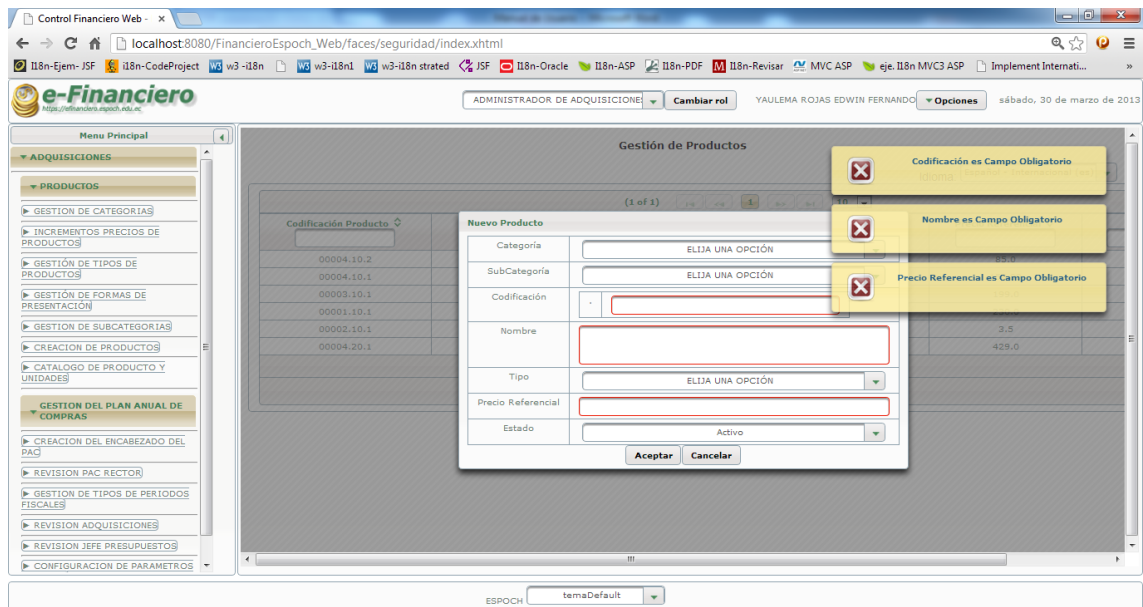


Figura N° 35 Notificaciones de Error de Productos

### 3.2.6 GESTIÓN DEL CATÁLOGO DE PRODUCTOS Y UNIDADES

La clasificación de productos en el catálogo es de suma importancia ya que en base a ello se obtendrá toda la información requerida para la elaboración del PAC Institucional. Debemos elegir la opción **GESTIÓN DE CATÁLOGO DE PRODUCTOS Y UNIDADES** (Figura N° 36).



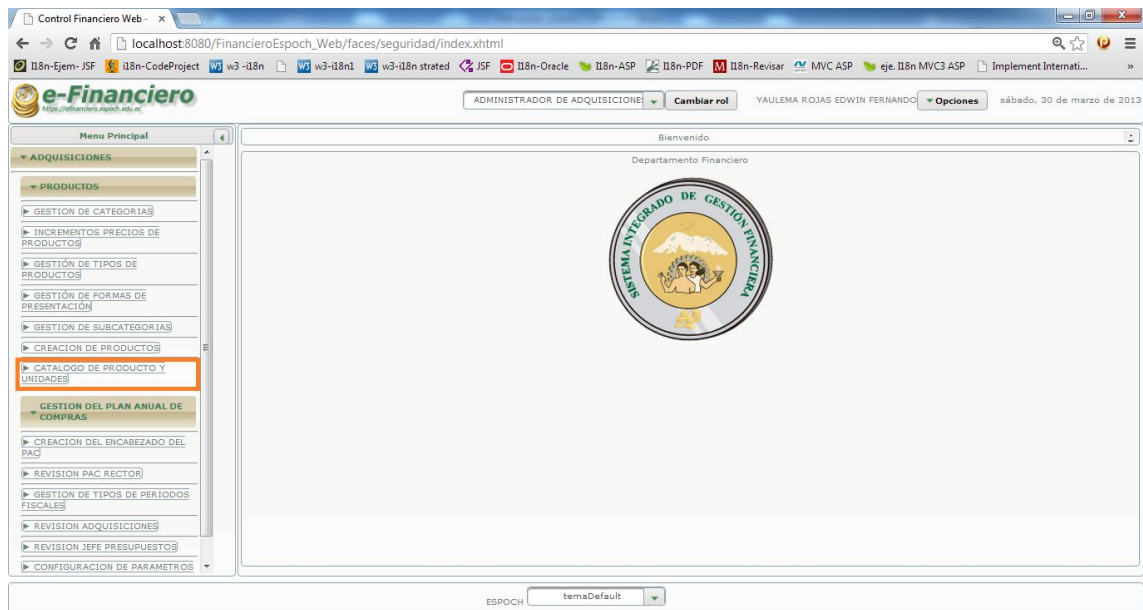


Figura N° 36 Gestión de Catálogo de Productos y Unidades

Ahora se mostrará la pantalla principal para de la Gestión de Productos y Unidades en la que se podrá realizar búsquedas de las categorías almacenadas (Figura N° 37):

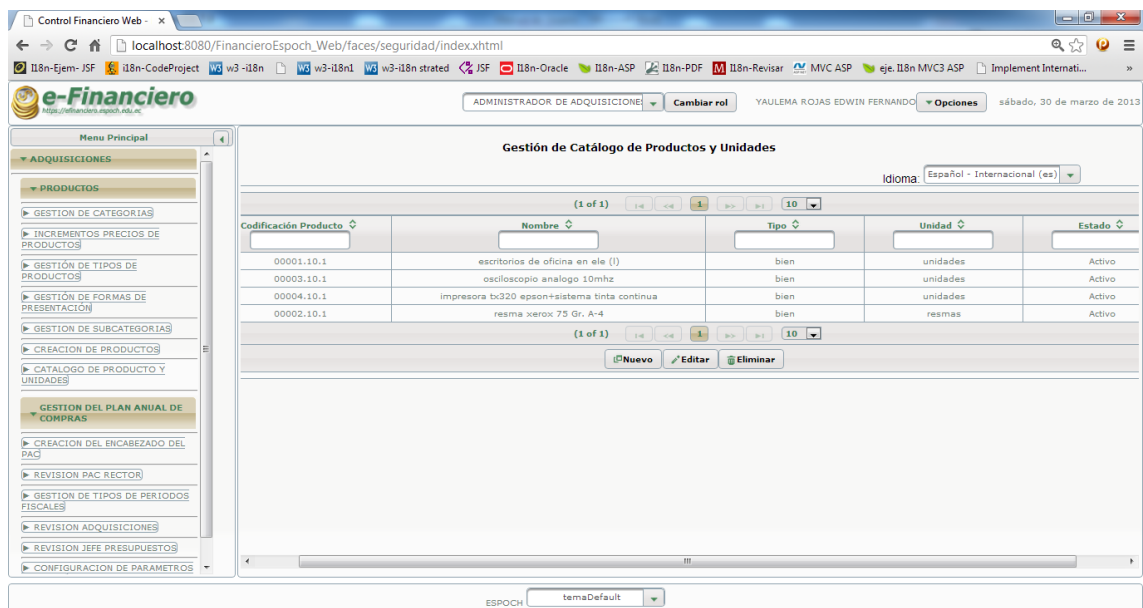


Figura N° 37 Pantalla Principal de Gestión de Catálogo de Productos y Unidades

### 3.2.6.1 Nuevo de Producto a Catálogo:

Esta opción nos permite registrar una nuevo Producto al Catálogo, el cual estará disponible en el Catálogo (Figura N° 38).

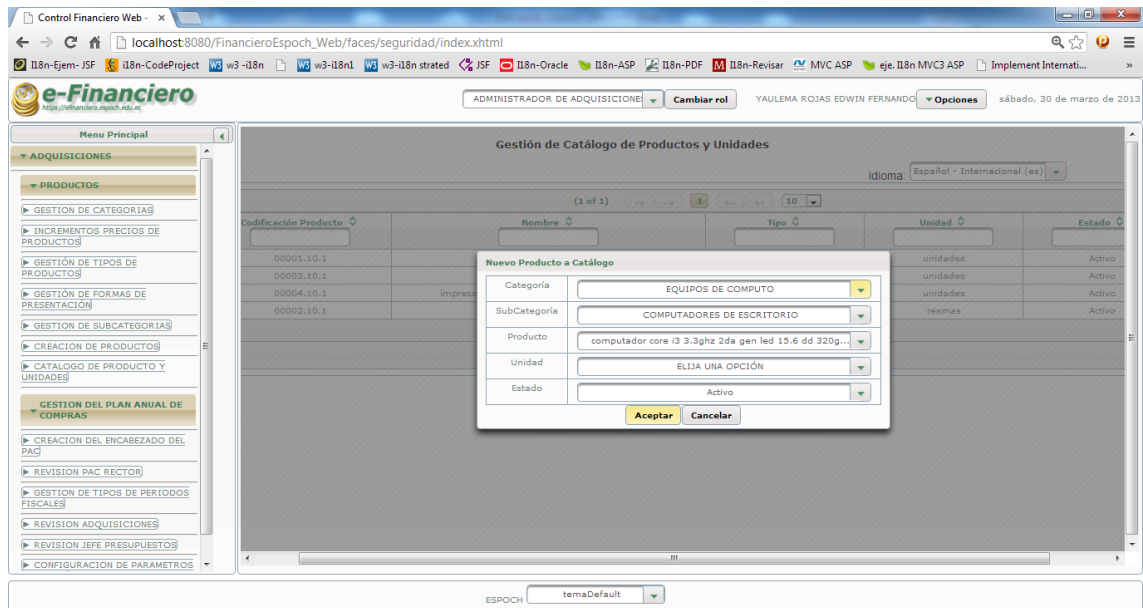


Figura N° 38 Nuevo Producto al Catálogo

### 3.2.6.2 Editar/Dar de Baja de Producto de Catálogo:

Esta opción nos permite registrar una editar los datos de un Producto del Catálogo, el cual estará disponible en el Catálogo (Figura N° 39).

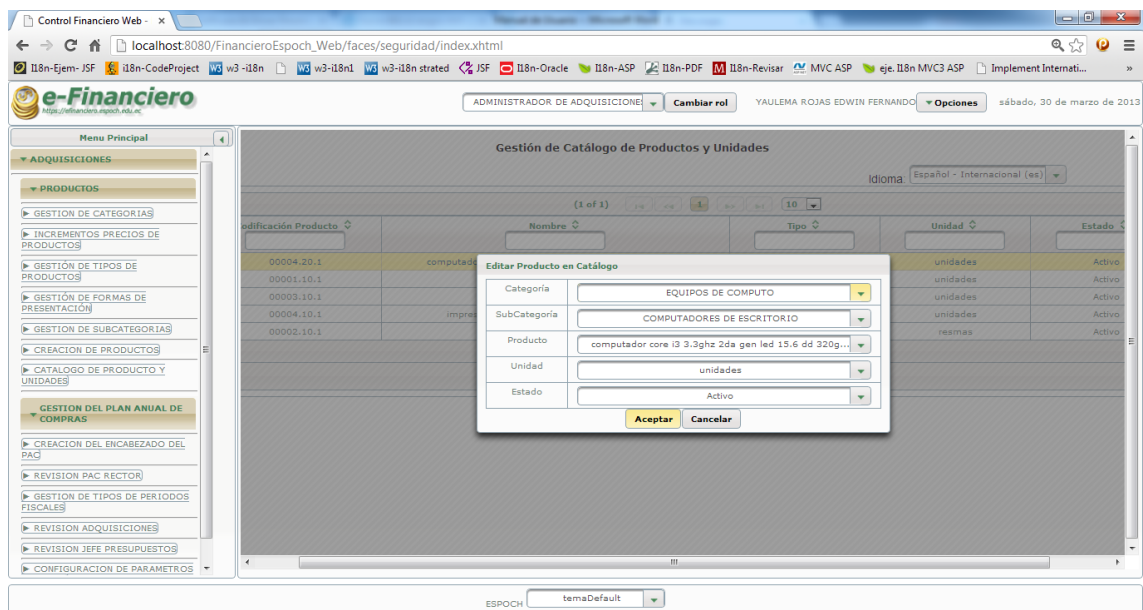


Figura N° 39 Editar Datos de Producto del Catálogo

### 3.2.6.3 Eliminar de Producto del Catálogo:

Esta opción nos permite eliminar los datos de un Producto del Catálogo, ésta acción se realizará siempre que el producto no forme parte en otro registro (Figura N° 40).

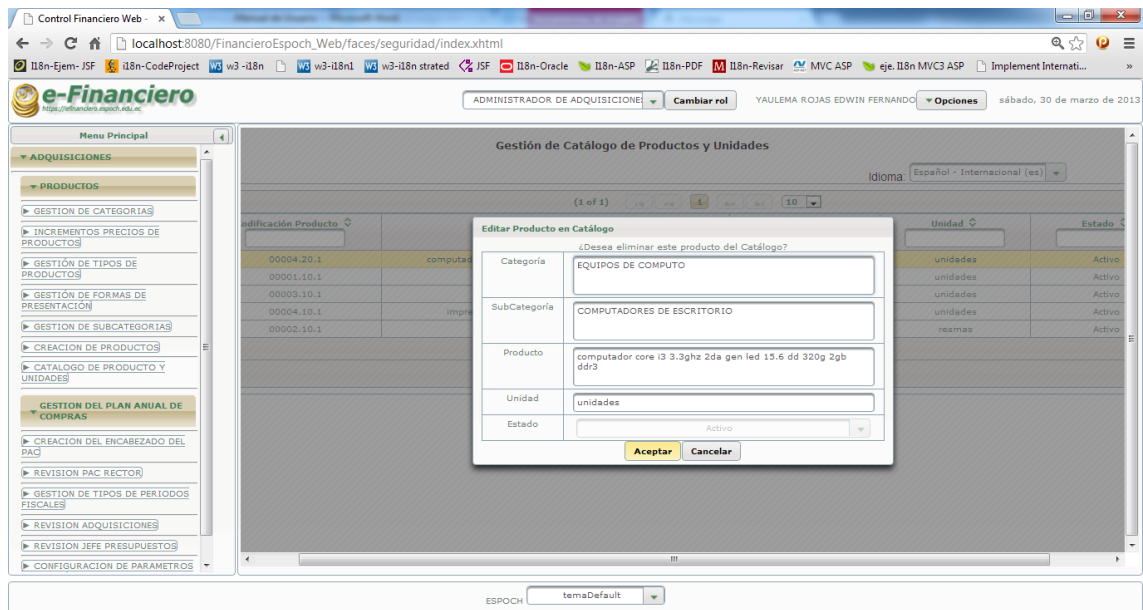


Figura N° 40 Eliminar Datos de Productos del Catálogo

### 3.2.6.4 Notificaciones de Error:

Al ingresar un nuevo Producto al Catálogo o al editarla si los campos ingresados no cumplen con el formato o son obligatorios el Módulo de Adquisiciones emitirá notificaciones (Figura N° 41):

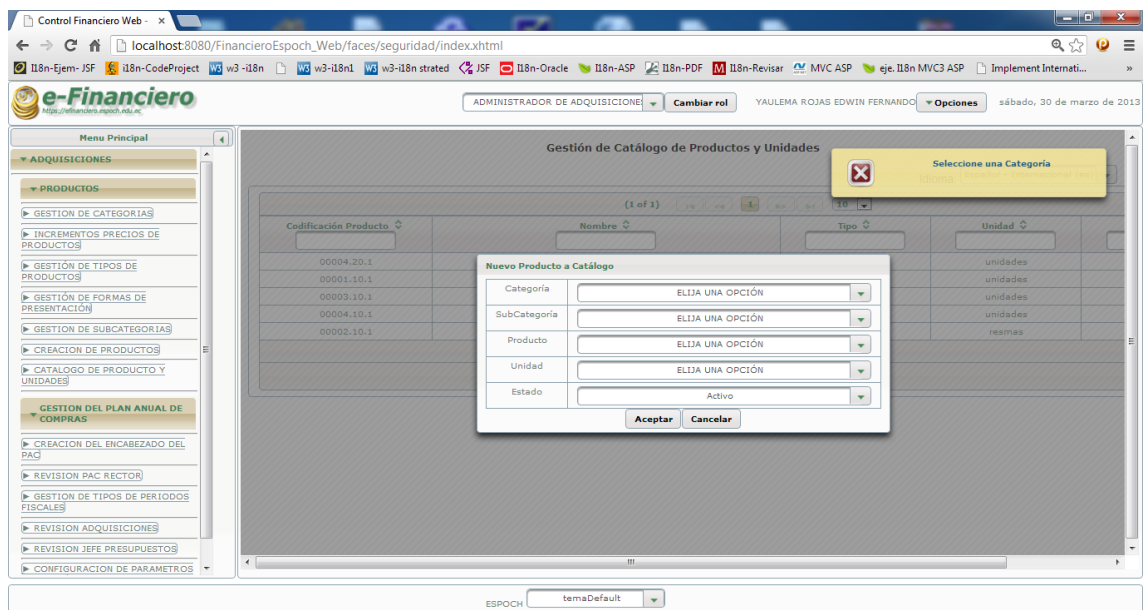


Figura N° 41 Notificaciones de Error de Productos del Catálogo

### 3.2.7 INCREMENTOS PROCENTUALES DE PRECIOS

En el submódulo de incrementos porcentuales se podrá realizar incrementos porcentuales del precio de los productos a través de las categorías, subcategorías y productos almacenados. Esto facilitará incrementar los precios de forma masiva. Debemos elegir la opción **INCREMENTO DE PRECIOS DE PRODUCTOS** (Figura N° 42).

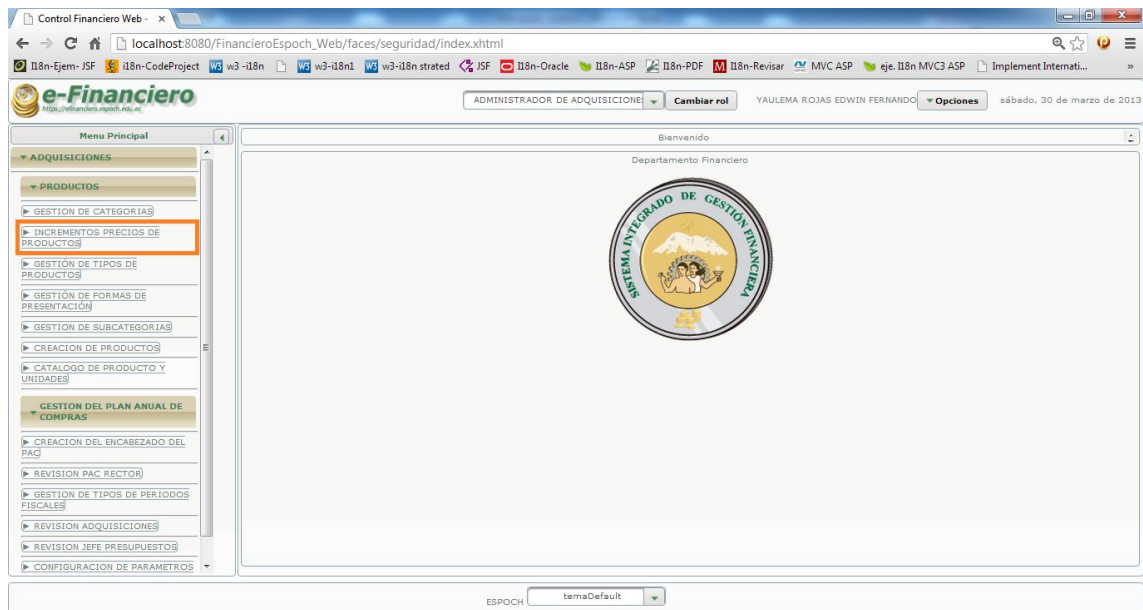


Figura N° 42 Opción Incremento de Precios de Productos

Ahora se mostrará la pantalla donde de Incremento de Precios de Productos en la que se podrá ingresar el valor a incrementar (Figura N° 43):

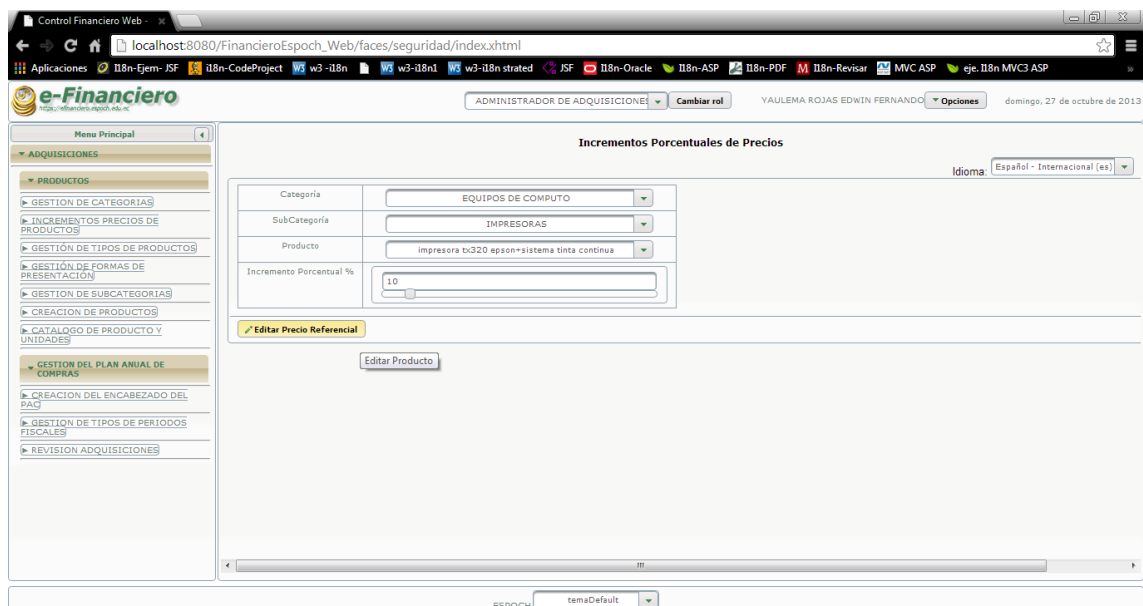


Figura N° 43 Incremento de Precios de Productos

### 3.2.8 INGRESO DE ITEMS AL PAC

El ingreso de ítems al PAC tiene que ver con el proceso de ingreso, edición y eliminación de los diferentes productos, el ingreso de productos es de suma importancia ya que en base a ello se consolidará la información del PAC Institucional. Debemos elegir la opción **INGRESO DE ITEMS AL PAC** (Figura N° 44).

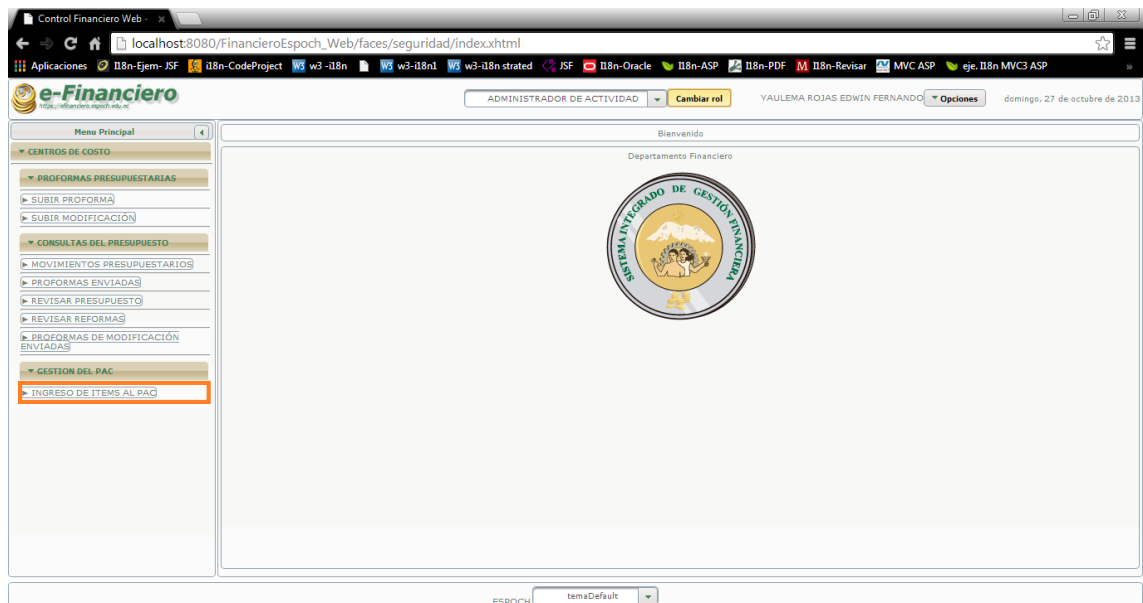


Figura N° 44 Pantalla Principal de Ingreso de Ítems al PAC

Ahora se mostrará la pantalla principal de Ingreso de Ítems al PAC en la que se podrá visualizar los ítems o productos ingresados (Figura N° 45):

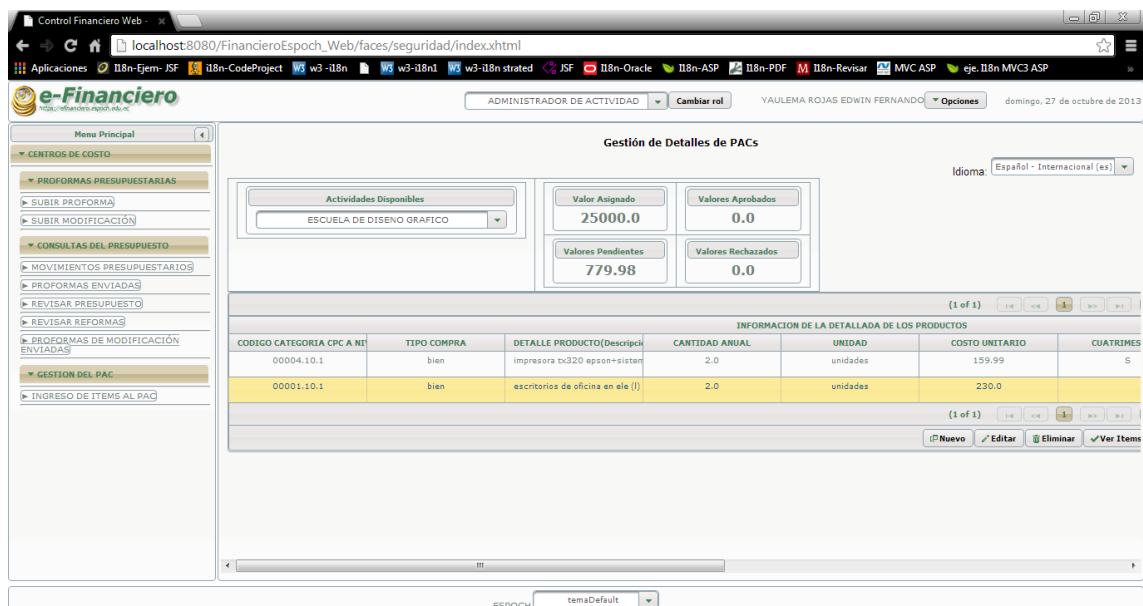


Figura N° 45 Pantalla Principal de Gestión de Catálogo de Productos y Unidades

### 3.2.8.1 Nuevo de Item:

Esta opción nos permite registrar un nuevo Producto al PAC, el cual estará disponible para el proceso de aprobación (Figura N° 46).

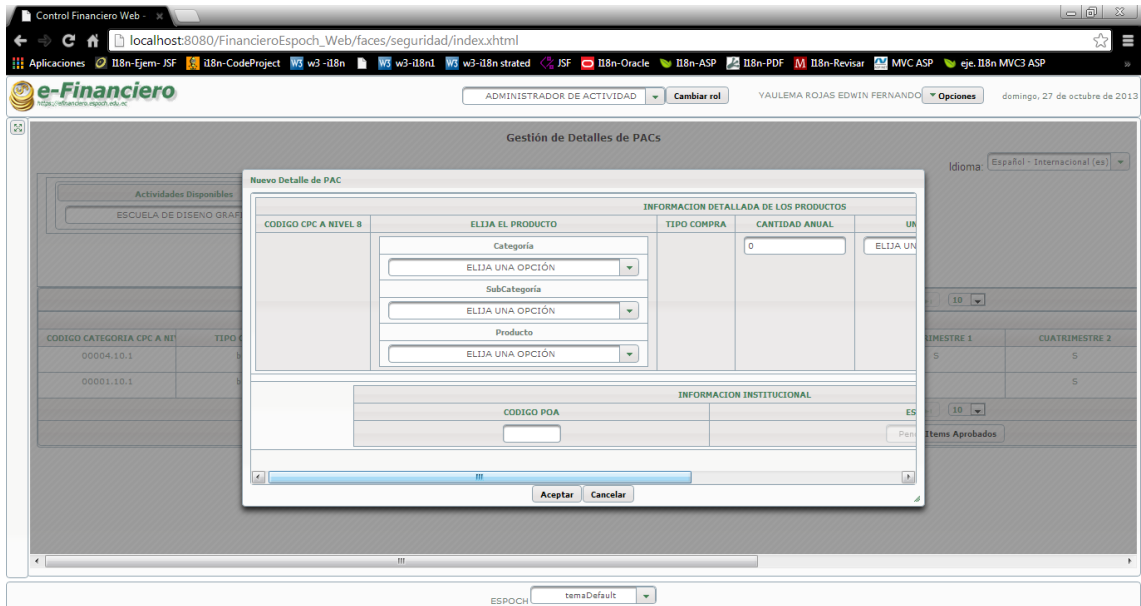


Figura N° 46 Nuevo Producto para el PAC

### 3.2.8.2 Editar Producto del PAC:

Esta opción nos permite registrar una editar los datos de un Producto del PAC (Figura N° 47).

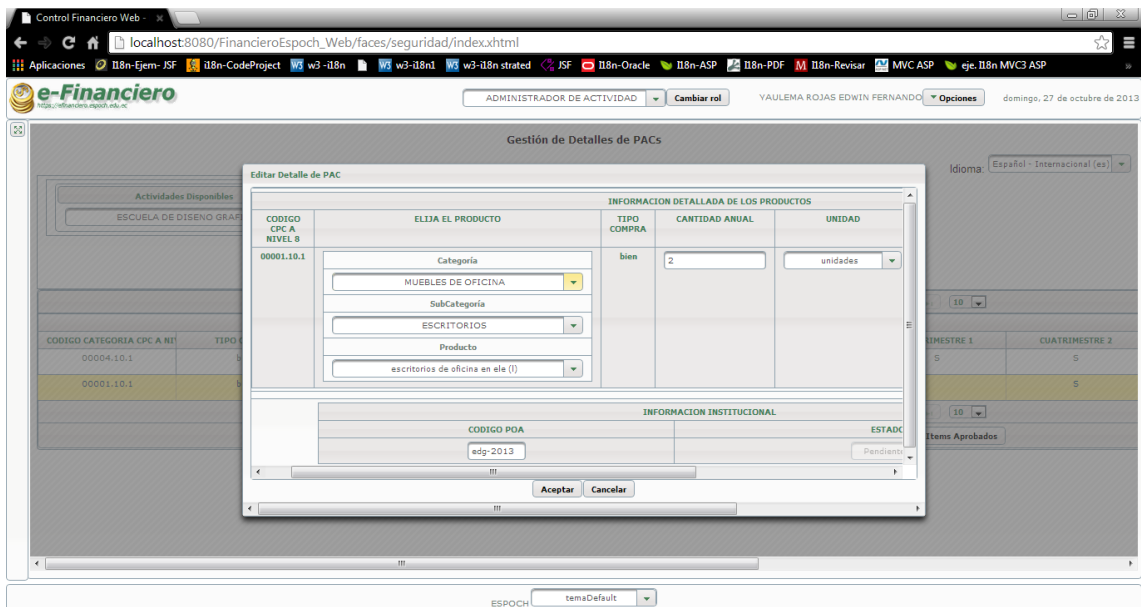


Figura N° 47 Editar Datos de Producto del PAC

### 3.2.8.3 Eliminar Producto del PAC:

Esta opción nos permite eliminar los datos de un Producto del PAC (Figura N° 40).

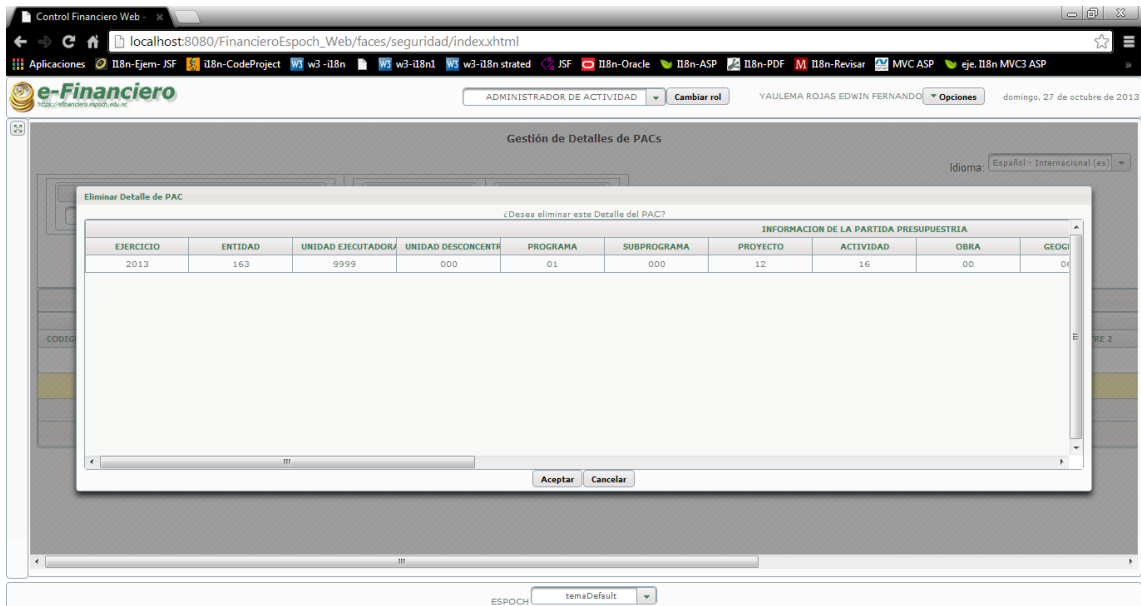


Figura N° 48 Eliminar Productos del PAC

#### 3.2.8.4 Notificaciones de Error:

Al ingresar un nuevo Producto al PAC o al editarla si los campos ingresados no cumplen con el formato o son obligatorios el Módulo de Adquisiciones emitirá notificaciones (Figura N° 49):

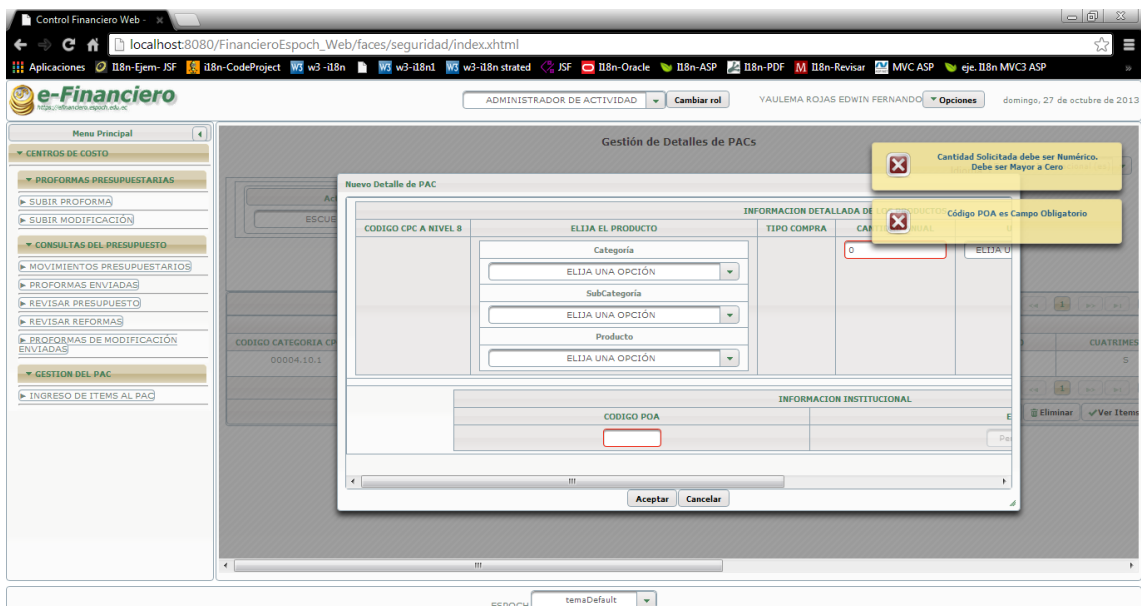


Figura N° 49 Notificaciones de Error de Productos del PAC

#### 3.2.9 REVISIÓN DE ADQUISICIONES

La Revisión de Adquisiciones tiene que ver con la validación de la información de los ítems o productos de ingresados al PAC por cada Actividad (Escuela, Departamento, etc.). Debemos elegir la opción **REVISIÓN DE ADQUISICIONES** (Figura N° 50).



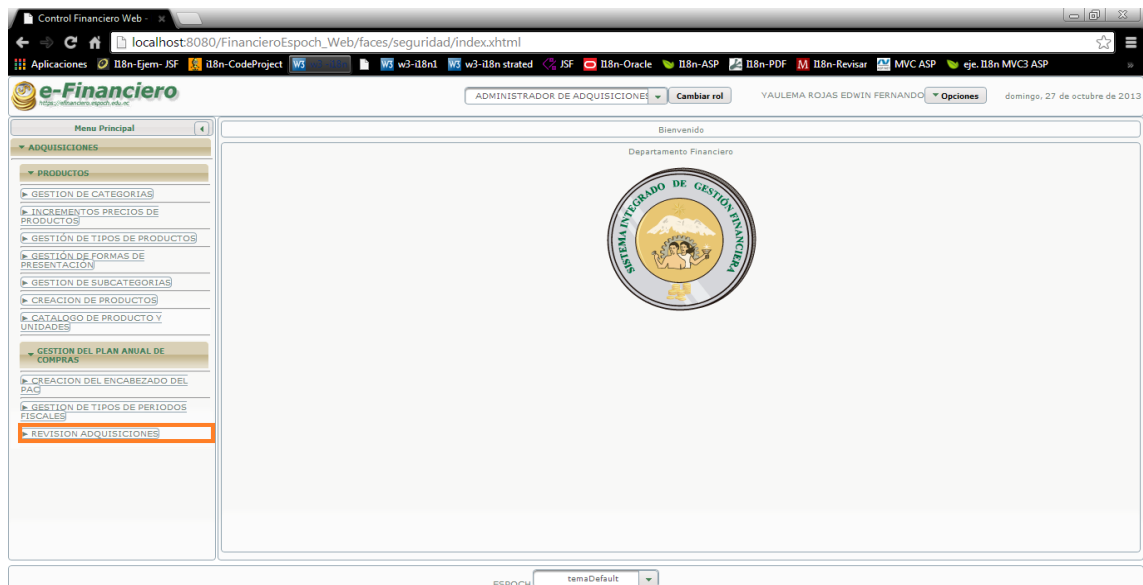


Figura N° 50 Revisión del Rol Adquisiciones

Como se visualiza en la Figura N° 51 se puede editar la información concerniente a los productos ya sea el tipo de compra, el producto, la forma de presentación del producto, también tiene la opción de aprobar o rechazar un determinado ítem y el período en el se ejecutará la compra y finalmente se presiona el botón guardar.

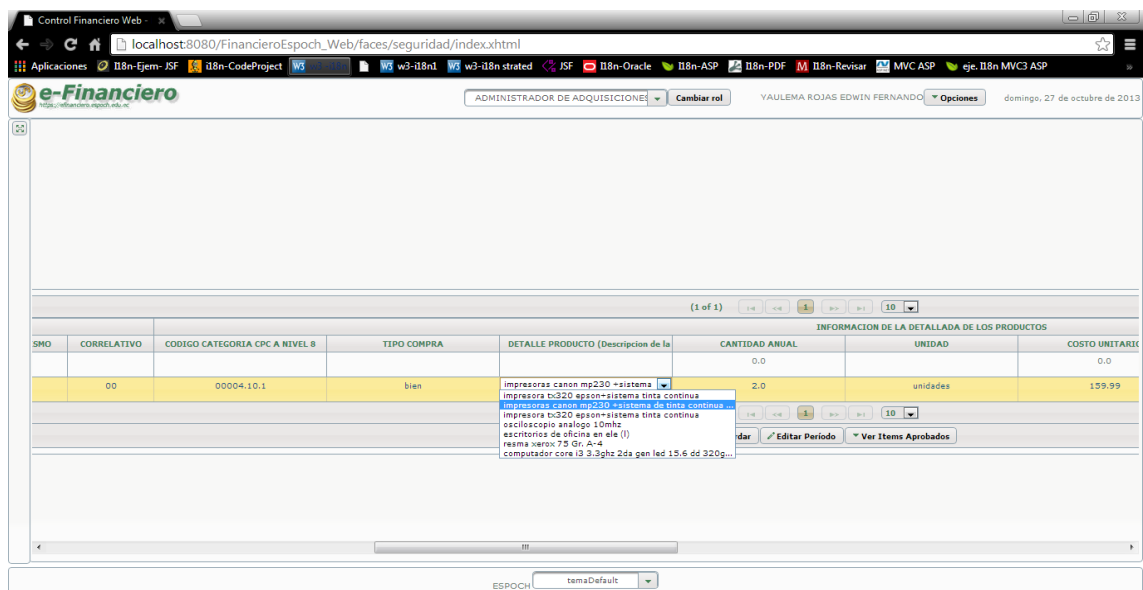


Figura N° 51 Revisión Adquisiciones - Información de Productos

### 3.2.10 REVISIÓN JEFE DE PRESUPUESTOS

La Revisión del Jefe de Adquisiciones tiene que ver con la validación de la información de la sección presupuestaria del PAC. Debemos elegir la opción **REVISIÓN JEFE PRESUPUESTOS** (Figura N° 52).



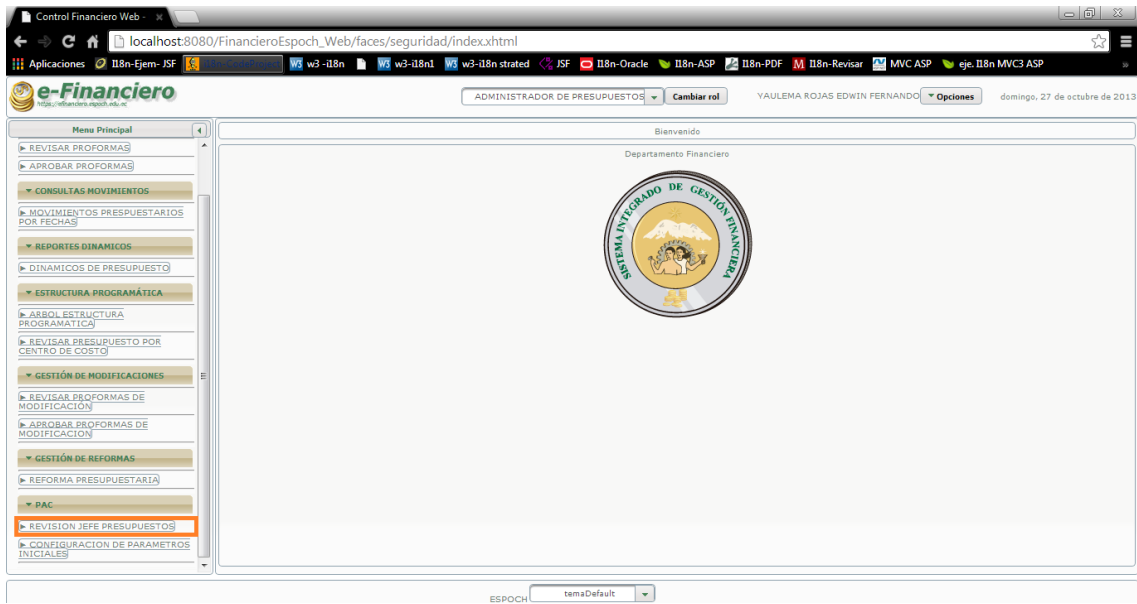


Figura N° 52 Revisión Jefe de Presupuestos

Como se visualiza en la Figura N° 53 se puede editar la información de la sección presupuestaria de los productos y el período en la que se ejecutará la compra, también tiene la opción de aprobar o rechazar un determinado item y finalmente se presiona el botón guardar

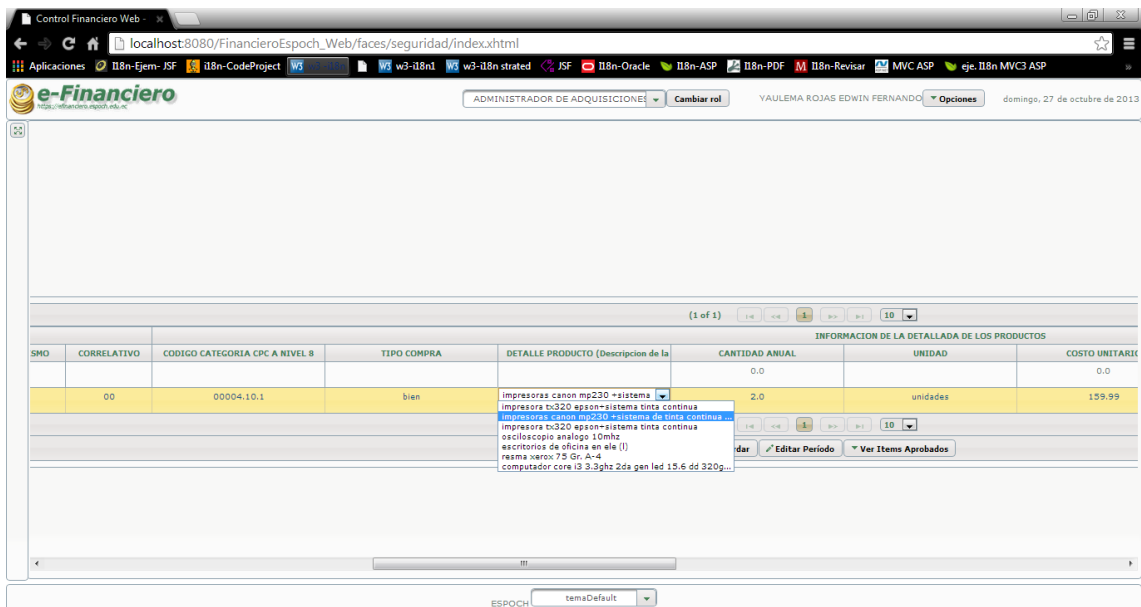


Figura N° 53 Revisión Sección Presupuestaria

### 3.2.10 REVISIÓN RECTOR

La Revisión del Sr. Rector tiene que ver con la validación de toda la información del PAC. Debemos elegir la opción **RECTOR** (Figura N° 54).

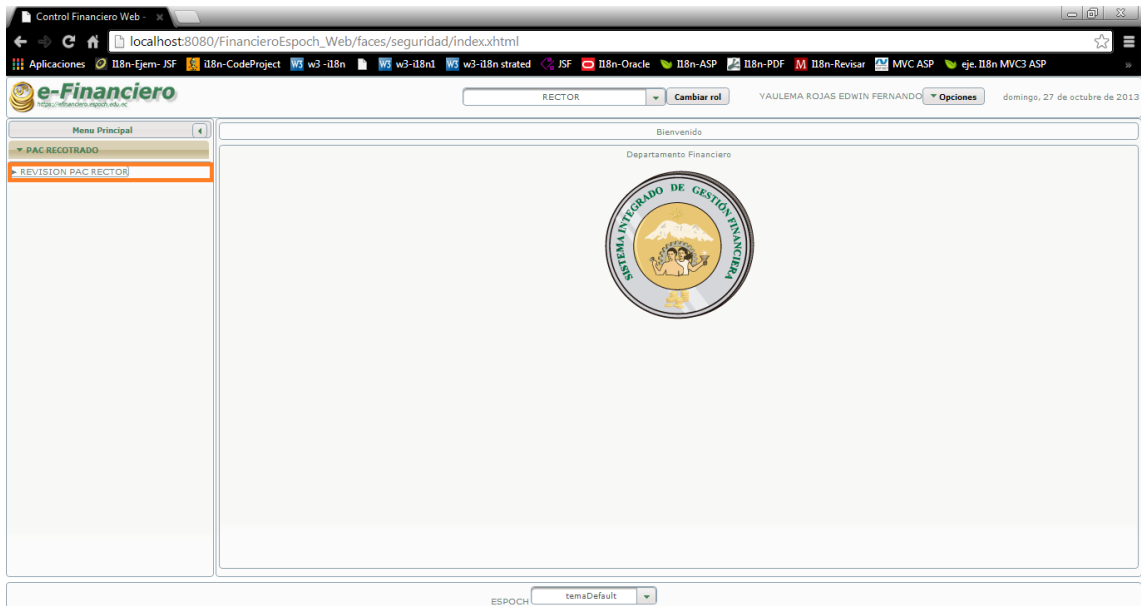


Figura N° 54 Revisión Rector

Como se visualiza en la Figura N° 55 se puede editar toda la información de todas las secciones del PAC también tiene la opción de aprobar o rechazar ítems. Finalmente se presiona el botón guardar para almacenar y posterior a ello publicar el PAC aprobado.

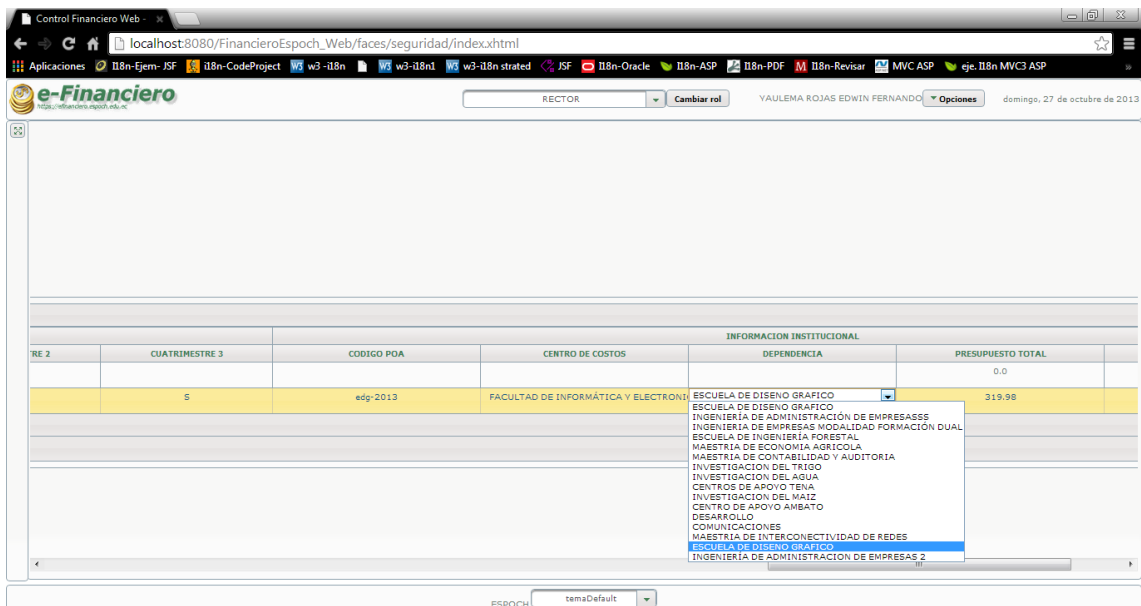


Figura N° 55 Revisión y Aprobación del PAC

### 3.2.10 REPORTE DEL PAC

Para la emisión del Reporte del PAC debemos presionar el botón **Ver Items Aprobados**, y se desplegará dos opciones las cuales con **Externo e Institucional**. La opción Externo emitirá el reporte con la información requerida por el portal de compras públicas y el Institucional el formato que maneja actualmente la ESPOCH. Esta opción se encuentra disponible para la Revisión de Adquisiciones, Presupuesto y Rector (Figura N° 56).

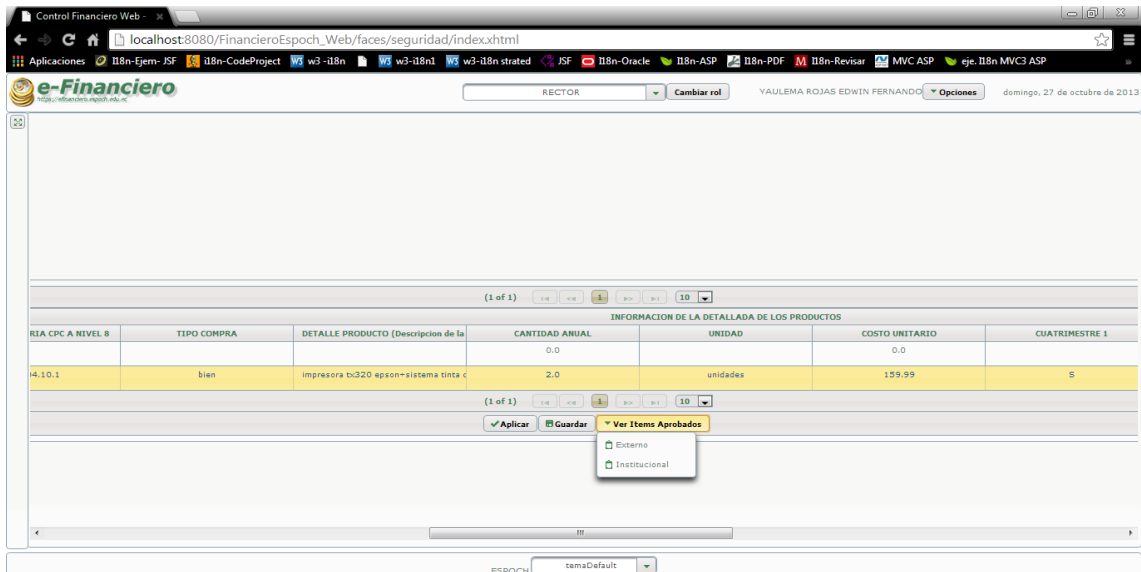


Figura N° 56 Ver Items Aprobados

Como se visualiza en la Figura N° 57 se puede editar toda la información de todas las secciones del PAC también tiene la opción de aprobar o rechazar ítems. Finalmente se presiona el botón guardar para almacenar y posterior a ello publicar el PAC aprobado.

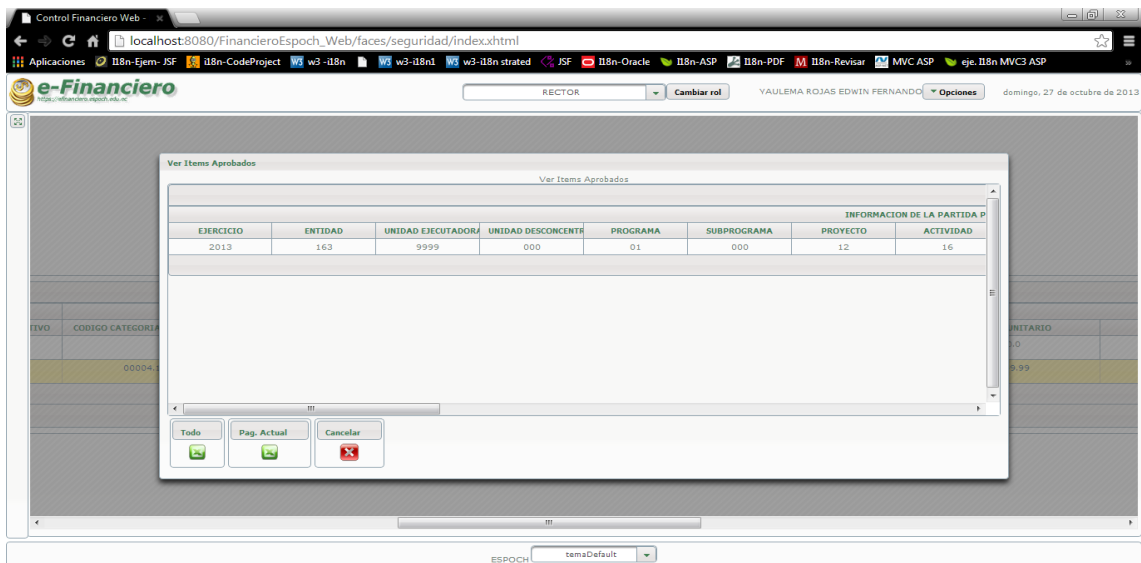


Figura N° 57 Visualización de Items Aprobados del PAC

Finalmente al presionar la opción presionamos sobre el ícono del Microsoft Excel y el reporte se exportará con ese formato el cual es el requerido por la ESPOCH (Figura N° 58).

Pac\_Institucional\_Todo\_2013 (4) [Modo de compatibilidad] - Microsoft Excel (Error de activación de productos)

Inicio Insertar Diseño de página Fórmulas Datos Revisar Vista Complementos Equipo

Portapapeles Fuente Alineación Número

Formato condicional Dar formato como tabla Estilos de celda Insertar Eliminar Formato Celdas

Autosuma Rellenar Ordenar y filtrar Buscar y seleccionar Modificar

A1 EJERCICIO

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T		
	EJERCICIO	ENTIDAD	UNIDAD E	UNIDAD D	PROGRAMA	SUBPROYECTO	PROYECTO	ACTIVIDAD	OBRA	GEOGRAFIA	RENGLON	RENGLON	FUENTE	ORGANISMO	CORRELA	CODIGO	TIPO DE	DETALLE	CANTIDAD	UNIDAD	COS	
1																						
2	2013	163	9999	000	01	000	12	16	00	0601		00		00	00	00004.10	1	bien	impresora	2.0	unidades	159.
3																						
4																						
5																						
6																						
7																						
8																						
9																						
10																						
11																						
12																						
13																						
14																						
15																						
16																						
17																						
18																						
19																						
20																						
21																						
22																						
23																						
24																						
25																						
26																						
27																						
28																						
29																						

Sheet0

100%

Figura Nº 58 Reporte del PAC Aprobado y Consolidado

# **DICCIONARIO DE DATOS**

## DICCIONARIO DE DATOS

**Tabla categoría:** Esta tabla almacenará toda la información que de las categorías de los productos disponibles en el catálogo del PAC

Column Name	Data Type	Primary Key	Not Null	AutoInc
Código	int8	YES	YES	YES
Codificación	Varchar	NO	YES	NO
Nombre	Varchar	NO	YES	NO
Descripción	Varchar	NO	NO	NO
Estado	int4	NO	YES	NO
<b>IndexName</b>	<b>Columns</b>			
categoria_codificacion_key	Codificacion			
categoria_pkey	Código			

**Tabla encabezado\_pac:** Esta tabla almacenará la información de cada PAC elaborado, también servirá como historial de los PACs realizados.

Column Name	Data Type	Primary Key	Not Null	AutoInc
Código	int8	YES	YES	YES
codigo_dependencia	int8	NO	YES	NO
codigo_usuario	int8	NO	YES	NO
Codificación	varchar	NO	NO	NO
Fecha	int8	NO	YES	NO
Total	float8	NO	NO	NO
Ejercicio	varchar	NO	YES	NO
nombre_dependencia_entidad	varchar	NO	NO	NO
Observación	varchar	NO	NO	NO
Estado	int4	NO	YES	NO
Entidad	varchar	NO	NO	NO
codigo_tipo_periodo	int8	NO	NO	NO
<b>IndexName</b>	<b>Columns</b>			
encabezado_pac_ejercicio_key	Ejercicio			
encabezado_pac_codificacion_key	Codificacion			
encabezado_pac_pkey	Código			
<b>Constraint Name</b>	<b>Constraint Kind</b>	<b>Check Expression</b>	<b>Columns</b>	
encabezado_pac_pkey	PRIMARY KEY		codigo	

encabezado_pac_codificacion_key	UNIQUE		codificacion	
encabezado_pac_ejercicio_key	UNIQUE		ejercicio	

**Tabla parámetros:** Esta tabla almacenará la información inicial de los campos de la sección de del PAC.

Column Name	Data Type	Primary Key	Not Null	AutoInc
Código	int8	YES	YES	YES
unidad_ejecutora	varchar	NO	YES	NO
unidad_desconcentrada	varchar	NO	YES	NO
Subprograma	varchar	NO	YES	NO
Proyecto	varchar	NO	YES	NO
Obra	varchar	NO	YES	NO
Geográfico	varchar	NO	YES	NO
renglon_auxiliar	varchar	NO	YES	NO
Organismo	varchar	NO	YES	NO
Correlativo	varchar	NO	YES	NO
Descripción	varchar	NO	NO	NO
IndexName	Columns			
parametros_pkey	codigo			
Constraint Name	Constraint Kind	Check Expression	Columns	
parametros_pkey	PRIMARY KEY		codigo	

**Tabla Producto:** La tabla producto contendrá la información de los productos disponibles.

Column Name	Data Type	Primary Key	Not Null	AutoInc
Código	int8	YES	YES	YES
codigo_subcategoria	int8	NO	NO	NO
codigo_tipo	int8	NO	NO	NO
Codificación	varchar	NO	YES	NO
Nombre	varchar	NO	NO	NO
precio_referencial	float8	NO	NO	NO
Descripción	varchar	NO	NO	NO
Estado	int4	NO	NO	NO
IndexName	Columns			
producto_codificacion_key	codificacion			
producto_pkey	codigo			

Constraint Name	Constraint Kind	Check Expression	Columns	
producto_pkey	PRIMARY KEY		codigo	
producto_codificacion_key	UNIQUE		codificacion	

**Tabla producto\_unidad:** La tabla producto unidades contiene un catálogo de productos en diferentes formas de presentación.

Column Name	Data Type	Primary Key	Not Null	AutoInc
Código	int8	YES	YES	YES
codigo_producto	int8	NO	NO	NO
codigo_unidad	int8	NO	NO	NO
Estado	int4	NO	NO	NO
IndexName	Columns			
producto_unidad_pkey	codigo			
Constraint Name	Constraint Kind	Check Expression	Columns	
producto_unidad_pkey	PRIMARY KEY		codigo	

**Tabla subcategoría:** La tabla subcategoría almacenará todas las subcategorías disponibles.

Column Name	Data Type	Primary Key	Not Null	AutoInc
Código	int8	YES	YES	YES
codigo_categoria	int8	NO	NO	NO
Codificación	varchar	NO	YES	NO
Nombre	varchar	NO	NO	NO
Descripción	varchar	NO	NO	NO
Estado	int4	NO	NO	NO
IndexName	Columns			
subcategoria_codificacion_key	codificacion			
subcategoria_pkey	codigo			
Constraint Name	Constraint Kind	Check Expression	Columns	
subcategoria_pkey	PRIMARY KEY		codigo	
subcategoria_codificacion_key	UNIQUE		codificacion	

**Tabla tipo:** La tabla tipo contendrá los diferentes tipos de productos en que se pueden presentar los productos.



Column Name	Data Type	Primary Key	Not Null	AutoInc
Código	int8	YES	YES	YES
Codificación	varchar	NO	NO	NO
Nombre	varchar	NO	YES	NO
Descripción	varchar	NO	NO	NO
Estado	int4	NO	YES	NO
IndexName	Columns			
tipo_codificacion_key	codificacion			
tipo_pkey	codigo			
Constraint Name	Constraint Kind	Check Expression	Columns	
tipo_pkey	PRIMARY KEY		codigo	
tipo_codificacion_key	UNIQUE		codificacion	

**Tabla tipo\_periodo:** Esta tabla almacenará el tipo de periodo fiscal ya sea por mensual, trimestral etc.

Column Name	Data Type	Primary Key	Not Null	AutoInc
Código	int8	YES	YES	YES
anio_fiscal	int8	NO	YES	NO
Codificación	varchar	NO	NO	NO
Nombre	varchar	NO	YES	NO
Valor	int4	NO	YES	NO
Descripción	varchar	NO	NO	NO
Estado	int4	NO	YES	NO
IndexName	Columns			
tipo_periodo_codificacion_key	codificacion			
tipo_periodo_pkey	codigo			
Constraint Name	Constraint Kind	Check Expression	Columns	
tipo_periodo_pkey	PRIMARY KEY		codigo	
tipo_periodo_codificacion_key	UNIQUE		codificacion	

**Tabla tipo\_periodo\_detalle\_pac:** Esta tabla almacenará los periodos que tuvo un determinado periodo fiscal.

Column Name	Data Type	Primary Key	Not Null	AutoInc
Código	int8	YES	YES	YES
codigo_tipo_periodo	int8	NO	YES	NO
codigo_detalle_pac	int8	NO	YES	NO
Valor	bool	NO	NO	NO
Estado	int4	NO	YES	NO
IndexName	Columns			
tipo_periodo_detalle_pac_pkey	codigo			
Constraint Name	Constraint Kind	Check Expression	Columns	
tipo_periodo_detalle_pac_pkey	PRIMARY KEY		codigo	

**Tabla unidad:** La tabla unidad contendrá las formas de presentación que puede tener un producto.

Column Name	Data Type	Primary Key	Not Null	AutoInc
codigo	int8	YES	YES	YES
codificacion	varchar	NO	NO	NO
nombre	varchar	NO	YES	NO
descripcion	varchar	NO	NO	NO
estado	int4	NO	YES	NO
IndexName	Columns			
unidad_codificacion_key	codificacion			
unidad_pkey	codigo			
Constraint Name	Constraint Kind	Check Expression	Columns	
unidad_pkey	PRIMARY KEY		codigo	
unidad_codificacion_key	UNIQUE		codificacion	

**Tabla detalle\_pac:** Almacenará toda la información de cada ítem que se ingrese al PAC.

Column Name	Data Type	Primary Key	Not Null	AutoInc
Código	int8	YES	YES	YES
codigo_encabezado_pac	int8	NO	YES	NO
codigo_proveedor_producto	int8	NO	YES	NO
unidad_ejecutora	Varchar	NO	NO	NO
unidad_desconcentrada	Varchar	NO	NO	NO
Programa	Varchar	NO	NO	NO
Subprograma	Varchar	NO	NO	NO

Proyecto	Varchar	NO	NO	NO
Actividad	Varchar	NO	NO	NO
Obra	Varchar	NO	NO	NO
Geográfico	Varchar	NO	NO	NO
Renglón	Varchar	NO	NO	NO
renglon_auxiliar	Varchar	NO	NO	NO
Fuente	varchar	NO	NO	NO
Organismo	varchar	NO	NO	NO
Correlativo	varchar	NO	NO	NO
codigo_cpc	varchar	NO	NO	NO
tipo_compra	varchar	NO	NO	NO
detalle_producto	varchar	NO	NO	NO
cantidad_anual	float8	NO	NO	NO
Unidad	varchar	NO	NO	NO
costo_unitario	float8	NO	NO	NO
codigo_poa	varchar	NO	NO	NO
centro_costos	varchar	NO	NO	NO
Dependencia	varchar	NO	NO	NO
descripcion_caracteristica	varchar	NO	NO	NO
presupuesto_total	float8	NO	NO	NO
Observación	varchar	NO	NO	NO
Estado	int4	NO	NO	NO
<b>IndexName</b>	<b>Columns</b>			
detalle_pac_modificada_pkey	codigo			
<b>Constraint Name</b>	<b>Constraint Kind</b>	<b>Check Expression</b>	<b>Columns</b>	
detalle_pac_modificada_pkey	PRIMARY KEY		codigo	

# **HISTORIA DE USUARIO DEL PROTOTIPO EN ASP.**

<b>Tarea – Desarrollo del Prototipo en ASP</b>	
<b>Número de tarea:</b> -	<b>Número de historia:</b> -
<b>Nombre de tarea:</b> Implementación del Prototipo en ASP.	
<b>Tipo de tarea:</b> Desarrollo	<b>Esfuerzo:</b> Alto
<b>Fecha inicio:</b> 01-08-2012	<b>Fecha fin:</b> 05-10-2012
<b>Programador responsable:</b> Edwin Yaulema Rojas	
<b>Descripción:</b> Consistirá en analizar, diseñar e implementar las clases de necesarias para implementación del prototipo en ASP.	