

Universidad de las Ciencias Informáticas
“Facultad 3”



**Título: Implementación de la Capa Interfaz del Módulo
Seguimiento del Proyecto Convenio Cuba – Venezuela**

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autor: Yuli Ernesto Martínez Barzaga

Tutores: Lic. Evelyn Anania Infante
Ing. Yurisleidys Leiva Zuñiga

Mayo del 2009

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Firma del Autor

Yuli Ernesto Martinez Barzaga

Firma del Tutor

Ing. Yurisleidys Leiva Zuñiga

Firma del Tutor

Lic. Evelyn Anania Infante

AGRADECIMIENTOS

Un trabajo siempre es el resultado del sacrificio y apoyo de muchas personas por eso quiero agradecer:

A mis tutoras, por estar presentes para brindarme su ayuda y conocimientos.

A todos los profesores que me dedicaron parte de su valioso tiempo con profesionalidad, especialmente a mi amigo y profesor ante todo Zarate, Yoandris y otros más que los quiero y aprecio muchísimo.

A todos mis amigos que me brindaron sus conocimientos y apoyo, sin excepción de ninguno, fundamentalmente a todos aquellos que convivieron conmigo estos cinco años.

Y a todas esas personas que me brindaron todo, sin esperar nada a cambio.

A todos los quiero mucho, muchas gracias por estar ahí en el momento más difícil de toda la carrera.

DEDICATORIA

En el transcurso de nuestras vidas existen muchos seres queridos que siempre te están dando ánimo y cariño a los cuales les dedico esta investigación y el resultado de esta:

A mi madre, por su empeño en creer en mí y siempre estar ahí en todo momento,

Gracias mamá.

A mi querida novia, que siempre me ha servido de guía y apoyo en cada momento de estos duros cinco años, y nunca desistir de mí, aún en momentos difíciles para ella, muchas gracias por existir.

A mis hermanos, que siempre han sido mi impulso y apoyo.

A mis tías, por siempre estar ahí en todo momento, aunque estén muy lejos, por apoyarme y siempre creer en mí y brindarme todo su apoyo y confianza, aunque nunca lo dude.

A todos mis amigos que están en las buenas y en las malas.

A mis compañeros con los que he compartido en estos 5 años de carrera.

A todas aquellas personas que me ayudan a seguir viviendo la vida como un poema.

RESUMEN

A partir del desarrollo alcanzado en las relaciones entre los países de Cuba y Venezuela, se tomó la decisión de informatizar el Convenio Integral de Cooperación Cuba Venezuela, debido a que el control desde inicio a fin de los proyectos, se realizaba de forma manual, por lo que se tornaba un proceso engorroso por el tamaño de los datos que en estos se podría manejar. El sistema a realizar debía ser sencillo, agradable y funcional, pero siempre cumpliendo con los requisitos del cliente y brindando la seguridad necesaria, por la importancia de este. También la correcta solución del software solicitada por el cliente debería de ser implementado en el tiempo acordado por ambas partes.

En el presente trabajo se explica el proceso de desarrollo de la Interfaz de usuario del Módulo Seguimiento del proyecto CCV, para la cual se han definido pautas de diseño gráfico, para su correcta implementación, siempre rigiéndose por los estándares web definidos a nivel mundial con el fin de lograr un producto de alta calidad.

PALABRAS CLAVES

Convenio Integral de Cooperación Cuba Venezuela, Interfaz Gráfica de Usuario, Diseño Web, Estándares Web, Pautas de Diseño.

TABLA DE CONTENIDOS

DECLARACIÓN DE AUTORÍA	I
AGRADECIMIENTOS	I
DEDICATORIA	I
RESUMEN	I
TABLA DE CONTENIDOS	I
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	6
Introducción.....	6
1.1 Metodologías de desarrollo del software	6
1.1.1 <i>Proceso de desarrollo del software</i>	6
1.1.2 <i>Metodologías</i>	7
1.1.2.1 <i>Extreme Programing (XP)</i>	7
1.1.2.2 <i>Microsoft Solution Framework (MSF)</i>	10
1.1.2.3 <i>Rational Unified Process (RUP)</i>	11
1.2 ¿Qué es una Interfaz?	14
1.2.1 <i>¿Qué es una Interfaz de Usuario?</i>	14
1.2.2 <i>¿Qué es una Interfaz Web?</i>	14
1.2.3 <i>Partes que componen una Aplicación Web</i>	15
1.2.2 <i>Evolución de Interfaces</i>	16
1.2.2.1 <i>Interfaz de Línea de mandato (Consola)</i>	16
1.2.2.2 <i>Interfaces de menús</i>	17
1.2.2.3 <i>Interfaces Web</i>	18
1.3 Estándares para el diseño Web	21
1.3.1 <i>CSS</i>	22
1.3.2 <i>HTML</i>	23
1.3.3 <i>XML</i>	24
1.3.4 <i>DOM</i>	25
1.4 Tecnologías para el desarrollo de Interfaces Web.....	25
1.4.1 <i>Tecnologías del lado del cliente</i>	26
1.4.1.1 <i>AJAX</i>	27
1.4.1.2 <i>Java Script</i>	29
1.4.2 <i>Tecnologías del lado del servidor</i>	30
1.4.2.1 <i>PHP</i>	30
1.4.2.2 <i>JSP</i>	31
1.4.2.3 <i>ASP.NET</i>	32

1.4.2.4 EXT 2.0.....	33
Conclusiones.....	34
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	35
Introducción.....	35
2.1 Estudio y Modelado del Sistema.....	35
2.1.1 <i>Requisitos no Funcionales</i>	35
2.2 Arquitectura de la Información.....	36
2.3 Características del Sistema	37
2.3.1 <i>Usuarios</i>	37
2.3.2 <i>Tipo de Aplicación</i>	38
2.4 Convenciones de Archivos.....	38
2.4.1 <i>Estándares web</i>	40
2.5 Diseño del Sistema Web	42
2.5.1 <i>Elementos del Sistema de Navegación</i>	43
2.5.2 <i>Distribución de la página</i>	44
2.5.3 <i>Diseño de la Estructura de las Páginas</i>	47
2.5.4 <i>Validaciones</i>	53
2.6 Subsistema de Implementación	54
Conclusiones.....	57
CAPÍTULO 3: ANÁLISIS DE LA SOLUCIÓN PROPUESTA.....	58
Introducción.....	58
3.1 Pruebas de Software.....	58
3.1.1 <i>Pruebas de Caja Negra</i>	59
3.2 Casos de Pruebas	61
3.3 Resultados Obtenidos.....	72
Conclusiones.....	73
CONCLUSIONES GENERALES	74
RECOMENDACIONES.....	75
BIBLIOGRAFÍA.....	76
ANEXOS.....	77
GLOSARIO DE TÉRMINOS	95

INTRODUCCIÓN

Hoy en día los sistemas informáticos están presentes en todas las esferas de la sociedad, en la política, en lo social, en la cultura y en la economía. El uso de aplicaciones informáticas ha sido vital para realizar trabajos que se tornan complejos, tediosos, que requieren de mucho tiempo. Estos trabajos son realizados de forma manual por lo que están expuestos a errores y muchas veces se sufre pérdida de la información.

La humanidad se ha dado cuenta de la necesidad de informatizar y automatizar los procesos que enmarcan un trabajo o alguna situación determinada. Estos garantizan un aprovechamiento del tiempo de trabajo, reducción de errores, permitiendo muchas facilidades y una interacción más amena y cómoda entre los usuarios y sus puestos de trabajo. Por lo anteriormente mencionado se puede afirmar que si se crearan aplicaciones informáticas se resolverían muchos problemas que hoy en día se presentan.

En el marco de la Alternativa Bolivariana para las Américas (ALBA) para fortalecer los tradicionales lazos de amistad entre Cuba y Venezuela, se establecieron de común acuerdo, compromisos para elaborar programas y proyectos de cooperación que están inmersos en el Convenio Integral de Cooperación Cuba Venezuela. Los mismos fueron creados con el objetivo de fomentar el progreso de sus economías, y las ventajas recíprocas que resultan de una cooperación que tenga resultados efectivos en su avance económico y social.

El Convenio Integral de Cooperación Cuba Venezuela ha crecido y ganado en diversidad, lo que influye de manera positiva en importantes sectores del desarrollo económico y social de ambos países. Lo que a su vez esto a traído consigo que el seguimiento que se le realizan a estos proyectos que era de forma semiautomática, debido a que se realizaba por vía telefónica u por correo electrónico, se haya convertido un poco engorroso y muy difícil de controlar, lo que a traído consigo que muchos proyectos no se hayan terminado o ni siquiera empezado. Es por esto que surge la necesidad de iniciar un proyecto para automatizar todo el proceso que se realiza antes, durante y después de realizada la Comisión Mixta de cada año. Esta no es más, que el marco en el cual ambas partes

involucradas en los proyectos se reúnen para pactar nuevos acuerdos u proyectos de cooperación mutua.

El producto informático ideado para informatizar y automatizar las Comisiones Mixtas entre ambas naciones, se nombra “Sistema de Gestión para Seguimiento de los Proyectos del Convenio Integral de Cooperación Cuba Venezuela”, que dentro de la Universidad de las Ciencias Informáticas se nombró como proyecto CCV. Este está dirigido a informatizar el proceso de gestión de los proyectos que se realizan en el marco del Convenio Cuba y Venezuela, proceso que es chequeado por las Secretarías Técnicas de ambos países: Ministerio del Poder Popular para la Energía y Petróleo (MPPEP) en Venezuela y Ministerio para la Inversión Extranjera y la Colaboración Económica (MINVEC) en Cuba.

La solución de software debe ser utilizada por los usuarios que tienen la responsabilidad de llevar a cabo la ejecución de los proyectos, estos no son más que las partes involucradas en cada proyecto, que son los entes ejecutores, ministerios, y secretarías técnicas. Estos, de acuerdo a sus responsabilidades, serán los encargados de darle un uso correcto a la aplicación en algunas de las funcionalidades siguientes: revisar y aprobar las propuestas de los proyectos; coordinar los períodos de presentación de proyectos; consensar las propuestas de contrato de los proyectos; chequear la ejecución física y financiera de los proyectos; y tomar decisiones sobre procedimientos complejos que requiere el buen funcionamiento de un proyecto.

Para dar solución a lo anteriormente planteado, el proyecto CCV contará dentro de sus módulos con uno dedicado específicamente a informatizar los procesos relacionados con el seguimiento de los proyectos en cuanto a la ejecución física y financiera se refiere. Este se enmarca en la planificación, seguimiento y control de las actividades de cada uno de los proyectos; de permitir la re-planificación de actividades, la cual está orientada a las modificaciones en los recursos o en tiempo de ejecución de los proyectos. Además, debe brindar la posibilidad de modificar contratos a través de documentos anexados a los proyectos.

La República Bolivariana de Venezuela ha solicitado a la Universidad la necesidad de crear una aplicación capaz de realizar automatización de los proyectos del Convenio

Integral de Cooperación Cuba Venezuela, que cumpla con todos los requisitos necesarios y que permita el control del seguimiento de los proyectos de ambos países. Esto impone una **situación problemática** a resolver, que es la necesidad de realizar una aplicación capaz de cumplir las necesidades inferidas por las partes involucradas en este Convenio, la cual a su vez debe de ser amigable, flexible, de fácil entendimiento a los usuarios. Estos tres últimos factores pueden tener bastante complejidad, por la magnitud que algunos de estos proyectos puede alcanzar, por tanto el buen desarrollo de interfaces web es vital para obtener la aceptación del sistema, pues será con la que el usuario involucrado interactúe.

El **problema** a resolver es: ¿Cómo implementar la Capa de Interfaz del Módulo de Seguimiento del proyecto Sistema de Gestión del Convenio Integral de Cooperación Cuba – Venezuela?

Donde el **Objeto de Estudio** es: El proceso de desarrollo de la solución de software en el Módulo de Seguimiento del Proyecto CCV.

Del anterior objeto de estudio se deriva el siguiente **Campo de Acción**: Desarrollo de la Capa Interfaz de Usuario en el Módulo de Seguimiento del proyecto CCV.

El **Objetivo General** es: Implementar la Capa de Interfaz de Usuario del Módulo Seguimiento del proyecto CCV, para lograr una aplicación flexible, segura y de fácil intercambio con el usuario.

Para poder brindar solución al objetivo general se derivan los siguientes **Objetivos Específicos**:

- ✓ Elaborar el marco teórico de la investigación, analizando las metodologías, tecnologías y otros recursos que pueden ser utilizados posteriormente.
- ✓ Implementar los componentes identificados de las Interfaces de Usuario en el Módulo de Seguimiento en el proyecto CCV.
- ✓ Evaluar la implementación de los componentes realizada.

El presente trabajo está basado en la siguiente **Hipótesis**: con una correcta implementación de la Capa de Interfaz del Módulo de Seguimiento del proyecto CCV, se

podrá contribuir al desarrollo de una aplicación flexible, segura y de fácil intercambio con el usuario.

Los **Métodos Científicos** utilizados en el desarrollo de la investigación se utilizaron para estudiar la realidad natural, el pensamiento, los distintos fenómenos sociales con el objetivo de descubrir su esencia y sus relaciones, estos fueron:

- ✓ **Analítico-Sintético:** Hizo posible que a través del estudio de las necesidades de los clientes y los requerimientos solicitados por este, se arribe a la creación de una aplicación que le brinde solución a estas necesidades.
- ✓ **Histórico-Lógico:** Ayudó a realizar un estudio histórico sobre las aplicaciones que realizaban trabajos similares, para así poder tener un mayor conocimiento y saber aplicarlo en la solución del problema presentado.
- ✓ **Hipotético-Deductivo:** Brindó la posibilidad de determinar las principales herramientas que facilitarían el desarrollo de la aplicación.

Como resultado de este trabajo se obtendrán las funcionalidades para el manejo o consulta de la información de los proyectos que se encuentran en la fase de Seguimiento. Además, el Módulo de Seguimiento del proyecto CCV, representa una porción importante de todo el sistema. Por último, el trabajo, demostrara que el uso de la tecnología EXT 2.0 u otra superior en el desarrollo de interfaces webs en aplicaciones en la universidad, es viable.

Estructuración del contenido.

El documento está compuesto por 3 capítulos.

El **Capítulo 1** “Fundamentación Teórica”, está dedicado al estudio del estado del arte del tema relacionado con la investigación. Se realiza un estudio de las tecnologías actuales sobre las cuales se apoya la propuesta para facilitar el desarrollo de las interfaces necesarias en el Módulo Seguimiento Perteneiente al proyecto CCV.

El **Capítulo 2** “Características del Sistema”, muestra cómo se dio solución a la creación de las interfaces de usuario del módulo.

El **Capítulo 3** “Análisis de la Solución Propuesta”, expone los resultados obtenidos luego de la aplicación de pruebas de calidad a las Interfaces de usuarios realizadas.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Introducción

En el presente capítulo se abordan conceptos relacionados con el desarrollo de las Interfaces de usuario así como las tecnologías que pueden ser utilizadas en la implementación del sistema. Además, se tratan cuestiones como: una breve historia acerca de la evolución de las interfaces de usuarios, ¿qué es una Interfaz?, ¿qué es AJAX?, y las ventajas del uso de esta última tecnología en la creación de Interfaces de Usuario respecto a las que no la utilizan. Se realiza un estudio de varias tecnologías para el desarrollo del software como la EXT 2.0, ASP.net y otras. De igual forma se ven tecnologías del lado del cliente como el Java Script.

1.1 Metodologías de desarrollo del software

Cada vez que un grupo de personas enfrenta la tarea de desarrollar un software llega una pregunta que es de vital importancia para todo el proceso que se avecina, ¿qué metodología debo usar para el desarrollo de un software? El desarrollo de todo producto informático es difícil de controlar, pero si no se trabaja por la guía de una metodología de desarrollo, la insatisfacción final del usuario será un premio seguro y con ello también la de los desarrolladores. Tampoco resulta fácil tener en cuenta que metodología utilizar para el desarrollo de un producto ya que estas varían teniendo en cuenta la envergadura de lo que se va a desarrollar.

1.1.1 Proceso de desarrollo del software

Un proceso define “quién” está haciendo “qué”, “cuándo” y “cómo”, para alcanzar un determinado objetivo. Un Proceso de Desarrollo de Software es la definición del conjunto de actividades que guían los esfuerzos de las personas implicadas en el proyecto, a modo de plantilla que explica los pasos necesarios para terminar el mismo.

El proceso de desarrollo de software tiene la misión de transformar los requerimientos del usuario en un producto de software; de manera que los integrantes del equipo y todo aquel que pueda estar interesado en el producto final, tenga la misma visión. Por lo tanto, las piedras angulares del proceso de desarrollo del software son: el proyecto, las personas y el producto; siendo las características del cliente, el entorno de desarrollo y las condiciones del negocio, elementos que influyen en el proceso. Existe una estrecha relación entre personas, proyecto, producto y proceso. Estos términos son conocidos como las cuatro "P" en el desarrollo de software (Pressman, 2001).

El resultado final de un proyecto software es un producto, donde intervienen personas a través de un proceso de desarrollo de software que guía los esfuerzos de las personas implicadas en el proyecto.

1.1.2 Metodologías

Es importante tener presente que es la organización quien debe adaptarse a la metodología, la más correcta será aquella capaz de guiar a la organización en el desarrollo de sus objetivos. A continuación se establecen algunos aspectos relevantes de algunas tecnologías más populares actualmente.

1.1.2.1 Extreme Programming (XP)

La metodología de desarrollo de software XP ha resultado ser una de las más exitosas en los últimos años, utilizadas para proyectos de corto plazo, equipos pequeños y cuyo plazo de entrega es muy corto. La metodología consiste en una programación rápida o extrema, se destaca entre sus rasgos particulares el tener como parte del equipo, al cliente, pues es uno de los requisitos para llegar al éxito del proyecto.

En esta metodología se distinguen tres aspectos principales:

- ✓ Pruebas Unitarias: se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándose en algo hacia el futuro, se pueda hacer pruebas de las fallas que pudieran ocurrir.
- ✓ Re fabricación: se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- ✓ Programación en pares: una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Mientras uno programa el otro está guiando y revisando ese trabajo, por lo que este par de desarrolladores debe tener una buena comunicación y técnicas de desarrollo similares.

La metodología XP para guiar el desarrollo de proyectos va desde lo pequeño y agregando funcionalidades a medida que se avance en el proceso, considerando siempre la constante retroalimentación y el manejo de cambios, teniendo en cuenta que el costo de los cambios no depende en ninguna medida de la fase o etapa en la que se encuentre el producto. A lo largo del proceso de desarrollo se van agregando funcionalidades a medida que se vayan haciendo necesarias, pero nunca se implementará nada que no haga falta en ese momento. Otro aspecto que distingue esta metodología es que el cliente está muy involucrado con los desarrolladores en todo momento, es prácticamente un miembro más del equipo.



Figura 1 Ciclo de vida de un proyecto XP

La metodología de desarrollo de software XP cuenta de cuatro actividades fundamentales:

- ✓ Codificar
- ✓ Testear
- ✓ Atender
- ✓ Diseñar

Define los siguientes roles:

- ✓ Programador: trabajando en pares, son los responsables de diseñar, construir el sistema y realizar las pruebas.
- ✓ Jefe de proyecto: asegura condiciones adecuadas para el proyecto y organiza y guía las reuniones.
- ✓ Cliente: este también forma parte del equipo, determina que construir y cuando, organiza las pruebas funcionales.
- ✓ Probador: ayuda a que las pruebas funcionales sean superadas.
- ✓ Rastreador: lleva los datos históricos del proyecto.
- ✓ Entrenador: responsable del proceso.

En XP es de vital importancia la comunicación que se establece entre usuario y el equipo de desarrollo, la simplicidad a la hora de implementar los diferentes módulos y la constante y concreta retroalimentación entre usuarios finales, clientes y los desarrolladores (Fernández Escribano, 2002).

1.1.2.2 Microsoft Solution Framework (MSF)

Esta es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.



Figura 2 Ciclo de vida MSF

MSF tiene las siguientes características:

- ✓ **Adaptable:** es parecido a un compás, usado en cualquier parte como un mapa, del cual su uso es limitado a un específico lugar.
- ✓ **Escalable:** puede organizar equipos tan pequeños entre 3 o 4 personas, así como también, proyectos que requieren 50 personas a más.

- ✓ Flexible: es utilizada en el ambiente de desarrollo de cualquier cliente.
- ✓ Tecnología Agnóstica: porque puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología.

Es una metodología compuesta por varios modelos encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto: Modelo de Arquitectura del Proyecto, Modelo de Equipo, Modelo de Proceso, Modelo de Gestión del Riesgo, Modelo de Diseño de Proceso y finalmente el modelo de Aplicación (Rational, 1998).

1.1.2.3 Rational Unified Process (RUP)

El Proceso Unificado de Desarrollo (RUP) es un proceso de software genérico que puede ser utilizado para una gran cantidad de tipos de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de competencia y diferentes tamaños de proyectos. Provee un enfoque disciplinado en la asignación de tareas y responsabilidades dentro de una organización de desarrollo. Su meta es asegurar la producción de software de muy alta calidad que satisfaga las necesidades de los usuarios finales, dentro de un calendario y presupuesto predecible.

Esta metodología esta divida en cuatro fases y cada una de esas faces tienes sus propios flujos de trabajo, existen en total nueve flujos de trabajo, seis de ingeniería y tres de soporte.

Fases y objetivos que define RUP:

- ✓ Inicio: determinar la visión del proyecto y poner en marcha al equipo de trabajo.
- ✓ Elaboración: definir una óptima arquitectura para el software.
- ✓ Construcción: lograr la capacidad funcional inicial del software.
- ✓ Transición: obtener un producto de software acabado y probado.

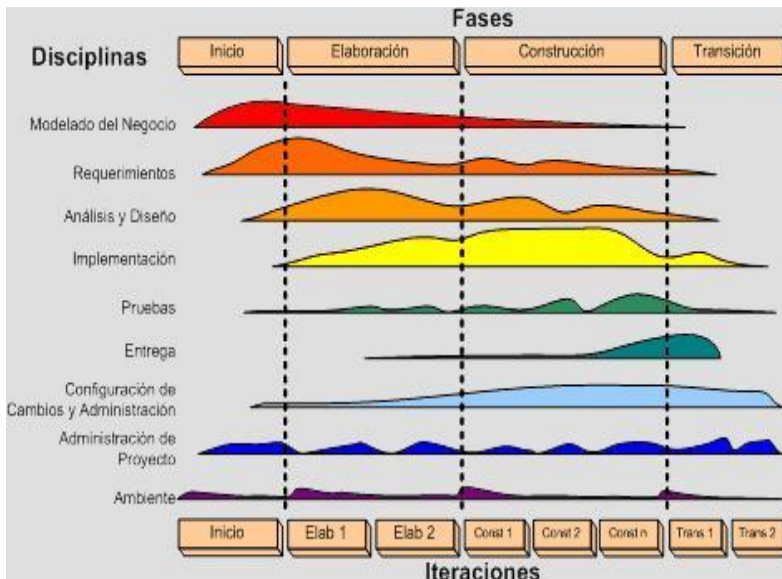


Figura 3 Ciclo de vida RUP

Flujos de ingeniería:

- ✓ Modelamiento del negocio: Se describe el negocio, define las actividades a automatizar y el personal que participa en ellas.
- ✓ Levantamiento de requisitos: Se define que debe hacer el sistema, identificando las funcionalidades que desea el usuario y las restricciones que esto impone.
- ✓ Análisis y diseño: Se describe como el sistema debe cumplir las funcionalidades previstas y las restricciones, queda indicado en detalle lo que se va a programar.
- ✓ Implementación: Se comienza a desarrollar el sistema en términos de clases y objetos.
- ✓ Pruebas: Se comienza a identificar y eliminar los errores y defectos surgidos durante el proceso de desarrollo.
- ✓ Despliegue: Se pone un release del producto en manos de los usuarios finales.

Flujos de soporte:

- ✓ Gestión de configuración y cambios: Se describe como llevar a cabo el control de los artefactos generados en el proyecto, de manera que se mantenga al tanto a todo equipo con las últimas versiones.

- ✓ Gestión de proyectos: Engloba las actividades encaminadas a lograr un producto que realmente satisfaga las expectativas del cliente.
- ✓ Gestión de entornos: Contiene las actividades que describen procesos y herramientas que empleara el equipo de trabajo y el proceso para llevar a cabo dichas actividades en la organización.

RUP posee tres características que lo identifican y lo hacen diferente a las demás metodologías:

- ✓ Centrado en la arquitectura: Muestra la visión común del sistema con la que deben estar de acuerdo el equipo de desarrollo y los usuarios, describe los elementos del modelo que son más importantes para su implementación. Describe las bases del sistema comprenderlo y desarrollarlo de manare económica.
- ✓ Guiado por casos de uso: reflejan las necesidades de los usuarios en el modelo de negocio y se representan en los requerimientos. Son ellos quienes guían el proceso de desarrollo mediante las diferentes realizaciones de casos de uso.
- ✓ Iterativo e incremental: Se desarrolla por iteraciones, de lo pequeño a lo más complejo, cada una de estas iteraciones tiene impacto en todos los flujos de trabajo.

La metodología RUP es la más acertada para el desarrollo de proyectos de gran envergadura ya que posee una manera disciplinada de asignar tareas y responsabilidades dentro de un grupo de trabajo, realizando estimaciones de tiempo y costo por actividades lo que permite anticiparse a soluciones que resulten poco factibles desarrollar.

En esta metodología se realiza un levantamiento detallado de los requisitos, depositando gran interés y tiempo en estos y el diseño, la puesta en práctica de las funcionalidades recibidas de los clientes se ven retrasadas mientras no se tenga una arquitectura bien definida que sea firme y robusta para el desarrollo del software. Está basada en las mejores prácticas que se han realizado y probado en los últimos tiempos (Jacobson, et al., 2000).

1.2 ¿Qué es una Interfaz?

El término interfaz (del inglés interface, superficie o sistemas independientes de contacto) se define como una conexión física y funcional entre dos aparatos. Generalizando esta definición, dados dos sistemas cualesquiera (en este caso, usuario - ordenador) que se deben comunicar entre ellos, la interfaz será el mecanismo, entorno o herramienta que hace posible dicha comunicación (Lores, 2001).

La interfaz de usuario de un sistema, consiste de aquellos aspectos del sistema con los que el usuario entra en contacto, físicamente, perceptivamente o conceptualmente.

1.2.1 ¿Qué es una Interfaz de Usuario?

La interfaz gráfica de usuario (GUI¹) es un tipo de interfaz de usuario que utiliza un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Habitualmente las acciones se realizan mediante manipulación directa para facilitar la interacción del usuario con la computadora. (Royo, 2004)

1.2.2 ¿Qué es una Interfaz Web?

Una Interfaz Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar y enviar formularios.

Es una aplicación de software que genera dinámicamente una serie de páginas en un formato estándar, como HTML² o XHTML³, que son soportados por los navegadores web comunes. Se utilizan lenguajes interpretados en el lado del cliente, tales como Java Script, para añadir elementos dinámicos a la interfaz de usuario. Generalmente cada página web

¹ Graphical user interface

² Lenguaje de Marcas de Hipertexto, del inglés HyperText Markup Language.

³ Lenguaje Extensible de Marcado de Hipertexto, del inglés eXtensible HyperText Markup Language.

en particular se envía al cliente como un documento estático, pero la secuencia de páginas ofrece al usuario una experiencia interactiva. Durante la sesión, el navegador web interpreta y muestra en pantalla las páginas, actuando como cliente para cualquier aplicación web. Esta no es simplemente una aplicación del diseño convencional sobre Internet ya que requiere tener en cuenta cuestiones tales como: navegabilidad, interactividad, usabilidad, arquitectura de la información y la interacción de medios como el audio, texto, imagen y video. (Hassan, et al., 2000)



Figura 4 Pirámide de Diseño Web.

Es útil imaginar las aplicaciones Web como pirámides, donde cada característica particular varía según el tipo de aplicación a desarrollar.

1.2.3 Partes que componen una Aplicación Web

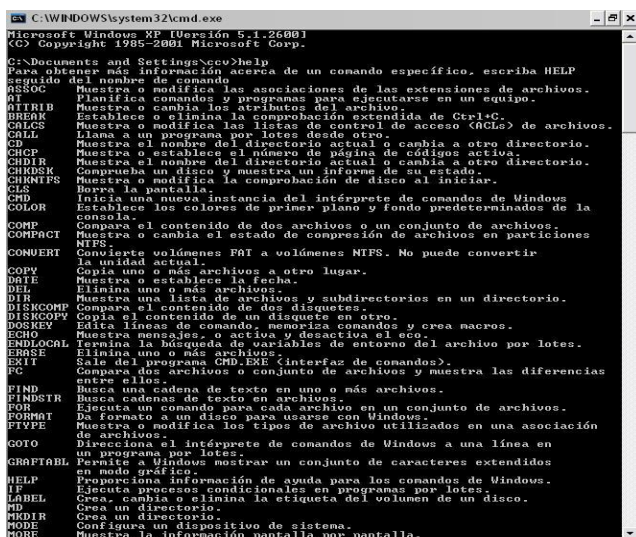
Aunque existen muchas variaciones posibles según la arquitectura que se haya utilizado, una aplicación web está normalmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador web ofrece la primera capa y un motor capaz de usar alguna tecnología web dinámica, (ejemplo: PHP, Java Servlets o Ruby on Rails (Hansson, 2005)) constituye la capa del medio. Por último, una base de datos constituye la tercera y última capa. El navegador web manda peticiones a la capa de en medio que ofrece servicios valiéndose de consultas y actualizaciones a la base de datos y a su vez proporciona una interfaz de usuario.

1.2.2 Evolución de Interfaces

Paralelo a la evolución de los sistemas operativos las interfaces también han venido desarrollándose, ya que el desarrollo alcanzado por la humanidad hasta el momento ha permitido la creación de nuevas herramientas para su creación, que están más allá de la vieja consola de comandos que se utilizaba para poder interactuar con los sistemas operativos. Las nuevas herramientas de diseño que han surgido con el de cursar de los tiempos, ha permitido que se desarrollen interfaces de usuarios que brinden más información y utilidad que las anteriores. Han evolucionado tanto al punto de ir de la primera interfaz que surgió, que fue la interfaz de línea de mandato (Consola), pasando por las nuevas interfaces de menús, hasta la interfaz gráfica.

1.2.2.1 Interfaz de Línea de mandato (Consola)

Antes de la llegada de las GUI se trabajaba fundamentalmente en las computadoras en lo que se conocía como modo texto o consola. En este tipo de interfaces, que aún hoy se siguen utilizando en los terminales X⁴, en las consolas de sistemas Unix o en una consola de MSDOS, la interfaz que se le ofrece al usuario espera de este que sepa introducir comandos a ser procesados y recibir unos resultados.



```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\cco>help
Para obtener más información acerca de un comando específico, escriba HELP
seguido del nombre de comando.
ASSOC Muestra o modifica las asociaciones de las extensiones de archivos.
AT Planifica comandos y programas para ejecutarse en un equipo.
ATTRIB Muestra o cambia los atributos del archivo.
BREAK Establece o elimina la comprobación extendida de Ctrl+C.
CACLS Muestra o modifica las listas de control de acceso (ACLs) de archivos.
CALL Llama a un programa por lotes desde otro.
CD Muestra el nombre del directorio actual o cambia a otro directorio.
CHCP Muestra o establece el número de página de códigos activa.
CHDIR Muestra el nombre del directorio actual o cambia a otro directorio.
CHKDSK Comprueba un disco y muestra un informe de su estado.
CHKNTFS Muestra o modifica la comprobación de disco al iniciar.
CLS Borra la pantalla.
CMD Inicia una nueva instancia del intérprete de comandos de Windows
COLOR Establece los colores de primer plano y fondo predeterminados de la
consola.
COMP Compara el contenido de dos archivos o un conjunto de archivos.
COMPACT Muestra o cambia el estado de compresión de archivos en particiones
NTFS.
CONVERT Convertir volúmenes FAT a volúmenes NTFS. No puede convertir
la unidad actual.
COPY Copia uno o más archivos a otro lugar.
DATE Muestra o establece la fecha.
DEL Elimina uno o más archivos.
DIR Muestra una lista de archivos y subdirectorios en un directorio.
DISKCOMP Compara el contenido de dos disquetes.
DISKCOPY Copia el contenido de un disquete a otro.
DOSKEY Edita líneas de comando, memoriza comandos y crea macros.
ECHO Muestra mensajes, o activa y desactiva el eco.
ENDLOCAL Termina la búsqueda de variables de entorno del archivo por lotes.
ERASE Elimina uno o más archivos.
EXIT Sale del programa CMD.EXE (interfaz de comandos).
FC Compara dos archivos o conjunto de archivos y muestra las diferencias
entre ellos.
FIND Busca una cadena de texto en uno o más archivos.
FINDSTR Busca cadenas de texto en archivos.
FOR Ejecuta un comando para cada archivo en un conjunto de archivos.
FORMAT Da formato a un disco para usarse con Windows.
TYPE Muestra o modifica los tipos de archivo utilizados en una asociación
de archivos.
GOTO Dirige al intérprete de comandos de Windows a una línea en
un programa por lotes.
GRAFTABL Permite a Windows mostrar un conjunto de caracteres extendidos
en modo gráfico.
HELP Proporciona información de ayuda para los comandos de Windows.
IF Ejecuta procesos condicionales en programas por lotes.
LABEL Crea, cambia o elimina la etiqueta del volumen de un disco.
MD Crea un directorio.
MMDIR Crea un directorio.
MORE Configura un dispositivo de sistema.
MORE Muestra la información pantalla por pantalla.

```

Figura 5 Ejemplo de Interfaz de Consola

⁴ Es un ordenador que ejecuta programas situados en otro ordenador (servidores)

A muchos usuarios aún les siguen gustando estas interfaces, que no necesitan para su uso de dispositivos como el ratón, y que permiten centrarse muy bien en las labores que se están llevando a cabo. Además, son interfaces que consumen muy pocos recursos del sistema y a las que es sencillo por ejemplo acceder de forma remota utilizando muy poco ancho de banda.

Uno de los primeros problemas que se encuentran a la hora de utilizar interfaces modo texto es su limitada capacidad expresiva. En las interfaces iniciales sólo se disponía de un tipo de texto sin colores y con unos caracteres limitados. Poco a poco se fue enriqueciendo las posibilidades de estos terminales modo texto. Estas trajeron consigo varios inconvenientes como fueron:

- ✓ El usuario no recibe mucha información por parte del sistema.
- ✓ El modelo de Interfaz es el del programador no del usuario.
- ✓ Carga de memoria del usuario.
- ✓ Nombre no siempre adecuados en los mandatos.
- ✓ Inflexible en la sintaxis.

1.2.2.2 Interfaces de menús

La sencillez de uso y robustez que se puede lograr dentro de una interfaz gráfica son mucho mayores que en las interfaces texto. Por ejemplo, mediante el uso de menús se logra hacer que el usuario pueda ejecutar comandos de forma guiada, sin necesidad de conocer la sintaxis concreta del mismo. Con un diseño adecuado de los menús de una aplicación, se pueden poner a disposición del usuario decenas de comandos fácilmente accesibles y que pueden guiar al usuario a la hora de ejecutar los comandos. Incluso los comandos más utilizados se pueden incluir en una barra de herramientas de fácil acceso para un uso más rápido de la interfaz.

Algunas de las grandes ventajas de estas es que permiten navegar dentro del propio sistema o poder seleccionar elementos de una lista que representan acciones o propiedades que el usuario desea realizar. Estas aparecen cuando el ordenador se

convierte en una herramienta del usuario y no del programador. Estas, bien estructuradas pueden ser buenas para usuarios noveles o esporádicos.

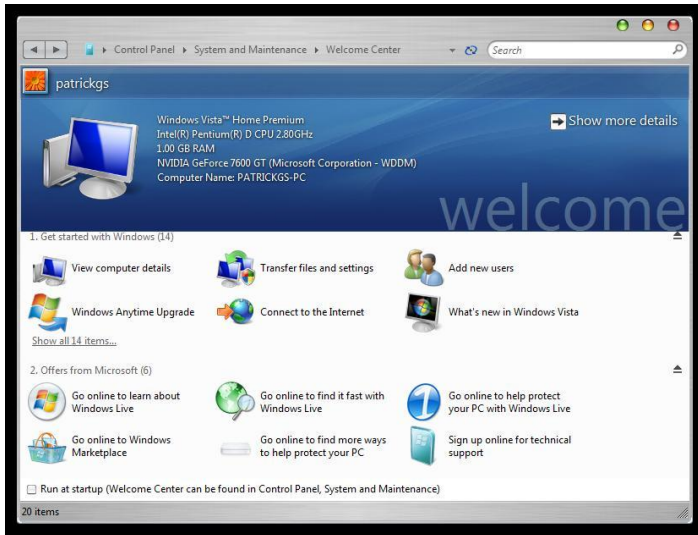


Figura 6 Ejemplo de Interfaz de Menús

En resumen, las interfaces gráficas son una evolución natural de las interfaces modo texto. Permiten interfaces mucho más ricas visualmente, mucho más expresivas y más sencillas de utilizar, pudiendo ser tan accesibles y rápidas de utilizar como interfaces modo texto.

1.2.2.3 Interfaces Web

El desarrollo de la humanidad y las ansias por cada vez descubrir nuevas formas de comunicación, servicios, etc. trajo consigo el surgimiento de la Internet. Esta produce un aceleramiento del continuo cambio que se viene dando respecto a la necesidad de interacción cotidiana de los usuarios con sistemas informáticos tanto a nivel local como mundial. Cada día se hace más necesario interactuar con interfaces diferentes: cajeros automáticos, consultas de información sobre viajes, actividades, eventos; búsqueda de información en bibliotecas, múltiples y variados sistemas de gestión, como ser sistemas empresariales, de comercio, de enseñanza, etc. y especialmente el uso de Internet como un proveedor mundial de servicios, comunicación y fuente de información. A su vez, los

sistemas informáticos son cada vez más completos y tienen más interacción con otros sistemas.

Para muchos, las respuestas a estas preguntas provienen de la investigación y desarrollo de interfaces más inteligentes, que se adapten al usuario de manera natural y progresiva, tratando de detectar sus características para que el sistema se adecue a su nivel y preferencias. Estas interfaces parten de la premisa que “los sistemas deben adaptarse a las personas”, y no lo contrario.

Las páginas Web supusieron la aparición de las interfaces Web, interfaces gráficas de usuario con unos elementos comunes de presentación y navegación que pronto se convirtieron en estándares. Este tipo de interfaces sirven de intermediarias entre usuarios genéricos, no acostumbrados generalmente al uso de aplicaciones informáticas, y unos sistemas de información y procesos transaccionales que son transparentes para el usuario, debiendo posibilitar la localización de la información deseada, el entendimiento claro de las funcionalidades ofrecidas, la realización práctica de tareas específicas por parte de los usuarios y la navegación intuitiva por las diferentes páginas que forman la aplicación Web.



Figura 7 Ejemplo de Interfaz Web

Desarrollados inicialmente por la compañía Xerox y popularizadas por Apple, le brindaron a la computación una nueva forma de comunicación entre maquina-hombre. Esta

se ha convertido en el medio de acceso a información y a programas más utilizado del mundo. De esta forma, el inmenso repositorio de la Web está al alcance de cualquiera, con sólo disponer de un navegador.

Ventajas:

La interfaz web constituye hoy en día el medio más cómodo, flexible, de fácil aprendizaje y ampliamente extendido para que los usuarios accedan a datos y programas. Es la Interfaz que le permite al usuario el acceso a información y servicios de la web a través de otros canales. Por ejemplo, recortes de prensa de páginas para incorporarlas a portales, conversión dinámica de páginas para mostrarlas por diferentes dispositivos o pasarelas multicanal a sitios web. Permite la gestión y automatización de procesos de negocio. Sustitución de actividades manuales realizadas sobre una página web por actividades automáticas, más fáciles de gestionar e integrar. Como también brindan servicios de consolidación y comparación, aplicaciones que permiten extraer datos y realizar transacciones sobre varios sitios web: compra comparativa, marketplaces, e-procurement.

Desventajas:

Las interfaces Web tienen ciertas limitantes en la funcionalidad del cliente. Los desarrolladores Web comúnmente utilizan lenguajes interpretados del lado del cliente para añadir más funcionalidad, especialmente para crear una experiencia interactiva que no requiera recargar la página cada vez que se realiza una petición, algo que suele molestar a los usuarios. Desde el punto de vista de estilos, la información no es presentada de la misma forma en todos los navegadores, por ejemplo Mozilla Firefox e Internet Explorer. (Moreno, 2003)

1.3 Estándares para el diseño Web

Los estándares web son un conjunto de recomendaciones dadas por el W3C⁵ y otras organizaciones internacionales a cerca de cómo crear e interpretar documentos basados en el Web. Son un conjunto de tecnologías orientadas a brindar beneficios a la mayor cantidad de usuarios, asegurando la vigencia de todo documento publicado en el Web.

El objetivo es crear una Web que trabaje mejor para todos, con sitios accesibles a más personas y que funcionen en cualquier dispositivo de acceso a Internet.

La importancia del diseño de la aplicación se basa en que éste será el que modele la interacción entre usuario y aplicación, y por tanto posibilitará o no la consecución de los objetivos perseguidos por el usuario (encontrar información, comprar, comunicarse, aprender...).

Es fácil inferir que un buen diseño deberá ser comprensible, fácil de usar, amigable, claro, intuitivo y de fácil aprendizaje para el usuario. Para poder asegurar que un diseño cumple con estos requisitos no basta simplemente con una actitud responsable del diseñador durante el desarrollo de la aplicación; es imprescindible la adopción por parte de éste de técnicas, procedimientos y métodos que aseguren empíricamente la adecuación del diseño a las necesidades, habilidades y objetivos del usuario. Para dar cumplimiento a ese correcto diseño existen los estándares Web definidos por la W3C, estos se basan en tres aspectos fundamentales: en la interacción con el usuario, buscando una comunicación armoniosa entre el mismo y la aplicación; en la optimización del modo en que se le presentan los datos al usuario, pensando siempre en el cliente, y ayudando a que el usuario sepa en qué parte del ambiente de trabajo está situado en un determinado momento; y por último la estructura del sistema, optimizando la ubicación de los componentes y las distintas secciones que se definen.

Estos tienen como principio lo siguiente:

- ✓ La Web se basa en una gran diversidad de tecnologías.

⁵ World Wide Web Consortium.

- ✓ Ninguna tecnología puede pretender cubrir todas las necesidades de la Web.
- ✓ La construcción de los cimientos de la Web requiere seguir un proceso claro y efectivo.
- ✓ Es imprescindible asegurar la interoperabilidad de las tecnologías.
- ✓ La Web debe ser universal.
- ✓ La Web evoluciona hacia la Semántica.

Existen hoy en día estándares y tecnologías capaces de ayudar al desarrollador de interfaces a crear un interfaz web con todos los requisitos para que cualquier usuario final quede satisfecho. Con el uso de estos se pueden lograr resultados como:

- ✓ Una mayor consistencia visual.
- ✓ Los documentos que separan apariencia de contenido usan menos código.
- ✓ Utilizan menos ancho de banda y se muestran más rápido a los usuarios.
- ✓ Compatibilidad con todos los navegadores actuales, y lo será con versiones futuras.
- ✓ Es más fácil de mantener y actualizar, el código es más simple, de esta forma se elimina la dependencia de un solo desarrollador.
- ✓ Es más accesible, permitiendo a personas con discapacidades utilizar su contenido. (Hassan, et al., 2004) (Alonso, 2005)

1.3.1 CSS

El gran impulso de los lenguajes de hojas de estilos se produjo con el boom de Internet y el crecimiento exponencial del lenguaje HTML para la creación de documentos electrónicos. La guerra de navegadores y la falta de un estándar para la definición de los estilos dificultaban la creación de documentos con la misma apariencia en diferentes navegadores.

El organismo W3C, encargado de crear todos los estándares relacionados con la web, propuso la creación de un lenguaje de hojas de estilos específico para el lenguaje HTML.

CSS⁶ no es más que un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas.

Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, ya que obliga a crear documentos HTML/XHTML bien definidos y con significado completo (también llamados "documentos semánticos"). Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes.

Al crear una página web, se utiliza en primer lugar el lenguaje HTML/XHTML para marcar los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla, lista de elementos.

Una vez creados los contenidos, se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página (Eguíluz Pérez, 2007).

1.3.2 HTML

Internet utiliza para la transferencia de información entre clientes y servidores el protocolo HTTP⁷, protocolo de transferencia de hipertexto. Este protocolo establece los requisitos que se deben cumplir para la transferencia de información y exige que la información a transferir sea definida mediante un sistema estándar, comprensible para cualquier equipo. Mucho antes del surgimiento de la Internet, con el inicio de las primeras redes de ordenadores por parte de organismos científicos y militares surge esta necesidad de crear

⁶ Hojas de Estilo en Cascada, del inglés Cascade StyleSheets.

⁷ Protocolo de Transferencia de Hipertexto, del inglés HyperText Transfer Protocol.

un sistema que permita la edición estructurada de documentos, evitando la disparidad de formatos y a la vez que facilite el intercambio con los mismos. Con la intención de satisfacer esta necesidad se desarrollo el HTML (Marshall, 1997).

HTML está definido como un lenguaje de definición de hipertextos compuesto por una serie de comandos, marcas o etiquetas también denominadas (Tags) que permiten definir la estructura lógica de un documento web y puede describir hasta un cierto punto, la apariencia de un documento del mismo (color del texto, contenidos multimedia, hipervínculos).

En resumen, es un lenguaje que permite crear páginas web y para ello utiliza unos comandos o etiquetas que indican o marcan que se debe mostrar y de qué forma (Raymond, 2003).

1.3.3 XML

El XML⁸ se ha convertido en el formato estándar para la representación de datos en Internet. Los datos XML se pueden procesar e interpretar en cualquier plataforma. Es un metalenguaje usado para definir documentos que contienen datos estructurados, desarrollado por la W3C, permite definir la gramática de lenguajes específicos, Por lo que XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades. Las características y beneficios del XML se pueden agrupar en estas áreas principales:

- ✓ Extensibilidad: como metalenguaje, XML puede usarse para crear sus propios lenguajes de marcas. Hoy en día existen muchos lenguajes de marcas basados en XML, incluyendo WML⁹.
- ✓ Estructura precisa: HTML sufre de una pobre estructura que hace difícil procesar eficientemente documentos HTML. Por otro lado, los documentos XML están bien

⁸ Lenguaje de Marcas Extensible, del inglés eXtensible Markup Language.

⁹ Lenguaje de Marcas para Inalámbricos, del inglés Wireless Markup Language.

estructurados, cada documento tiene un elemento raíz y todos los demás elementos deben estar anidados dentro de otros elementos.

Los documentos XML contienen datos portables. Esto significa que un documento XML puede ser transformado en HTML, WML o cualquier otro lenguaje de marcas. XML no ha nacido sólo para su aplicación en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo, etc.

XML posee un papel muy importante en la actualidad ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil (González, 2005).

1.3.4 DOM

El DOM¹⁰, Modelo en Objetos para la representación de Documentos, es esencialmente un modelo computacional a través del cual los programas y scripts pueden acceder y modificar dinámicamente el contenido, estructura y estilo de los documentos HTML y XML. Su objetivo es ofrecer un modelo orientado a objetos para el tratamiento y manipulación en tiempo real (o de forma dinámica) a la vez que de manera estática de páginas de Internet. El responsable del DOM es el consorcio W3C. En efecto, el DOM es una API¹¹ para acceder, añadir y cambiar dinámicamente contenido estructurado en documentos con lenguajes como Java Script (Autores, 1998).

1.4 Tecnologías para el desarrollo de Interfaces Web

Para el desarrollo de una interfaz web es necesaria la utilización de lenguajes interpretados ya que estos facilitan la portabilidad, ya que lo único necesario es disponer en cada plataforma del intérprete adecuado. Es por eso que para la realización de sitios web

¹⁰ Modelo en Objetos para la representación de Documentos, del inglés Document Object Model.

¹¹ Interfaz de Programación de Aplicaciones, del inglés Application Programming Interface.

son necesarios estos tipos de lenguajes ya que en Internet la portabilidad es imprescindible.

También hay que tener en cuenta de que existen dos tipos de tecnologías: las tecnologías del lado del cliente y las del lado del servidor. Del lado del servidor existen hoy en día muchas tecnologías capaces de desarrollar una interfaz web, ejemplos de estas son: PHP, JSP, ASP.NET. Del lado del cliente existen: Java Script, Visual Basic Script, Ext. 2.0. Todas son tecnologías muy potentes para lograr un resultado satisfactorio. Se podría desarrollar una interfaz web con cualquiera de estas herramientas, pero antes hay que tener en cuenta ventajas y desventajas.

1.4.1 Tecnologías del lado del cliente

Es una tecnología en la cual los programas residen junto a la página web en el servidor, pero son transferidos al cliente para que este los ejecute. Para esto el cliente necesita de un software para tener acceso al correspondiente servicio en un sitio web. Ejemplo de este software son los navegadores o exploradores Web, pues son las tecnologías necesarias para mostrarle al cliente en una interfaz web el servicio solicitado; estos son capaces de convertir el lenguaje Java Script, HTML o Ajax, en una interfaz de usuario. También se podría decir que las tecnologías del lado del cliente son totalmente independientes del servidor, lo cual permite que la página pueda ser albergada en cualquier lugar.



Figura 8 Ejemplo del funcionamiento de la programación del lado del cliente.

1.4.1.1 AJAX

AJAX¹², es una técnica de desarrollo web para crear aplicaciones interactivas o RIA¹³. Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

AJAX es una tecnología asíncrona, en el sentido de que los datos adicionales se requieren al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. Java Script es el lenguaje interpretado (scripting language) en el que normalmente se efectúan las funciones de llamada de AJAX mientras que el acceso a los datos se realiza mediante *XMLHttpRequest*, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

AJAX es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dados que está basado en estándares abiertos como DOM. Este no es una tecnología en sí mismo. En realidad, se trata de varias tecnologías independientes que se unen de formas nuevas y sorprendentes.

Las tecnologías que forman AJAX son:

- ✓ XHTML y CSS, para crear una presentación basada en estándares.
- ✓ DOM, para la interacción y manipulación dinámica de la presentación.
- ✓ XML, XSLT y JSON, para el intercambio y manipulación de información.
- ✓ XMLHttpRequest, para el intercambio asíncrono de información.
- ✓ Java Script, para unir todas las demás tecnologías.

¹² Asynchronous JavaScript And XML que al español significa aproximadamente JavaScript asíncrono y XML.

¹³ Rich Internet Applications que al español significa aproximadamente Aplicaciones de Internet Enriquecidas.

Ventajas:

Las aplicaciones construidas con AJAX eliminan la recarga constante de páginas mediante la creación de un elemento intermedio entre el usuario y el servidor. La nueva capa intermedia de AJAX mejora la respuesta de la aplicación, ya que el usuario nunca se encuentra con una ventana del navegador vacía esperando la respuesta del servidor. Su utilización hace más vistosa y atractiva la navegabilidad por las páginas. No requiere ningún tipo de plugin. Prácticamente todos los navegadores actuales lo soportan. Presenta reducción en el tamaño de información a intercambiar. Por lo que las peticiones HTTP al servidor se sustituyen por peticiones Java Script que se realizan al elemento encargado de AJAX. Las peticiones más simples no requieren intervención del servidor, por lo que la respuesta es inmediata. Si la interacción requiere una respuesta del servidor, la petición se realiza de forma asíncrona mediante AJAX. En este caso, la interacción del usuario tampoco se ve interrumpida por recargas de página o largas esperas por la respuesta del servidor. Esto contribuye al ahorro de tiempo por parte de la aplicación en brindar respuesta a las peticiones del usuario y menor consumo de recursos por parte de la aplicación a los servidores.

Desventajas:

En las aplicaciones web tradicionales, las acciones del usuario en la página como, pinchar en un botón, seleccionar un valor de una lista, etc., desencadenan llamadas al servidor. Una vez procesada la petición del usuario, el servidor devuelve una nueva página HTML al navegador del usuario. Esta técnica tradicional para crear aplicaciones web funciona correctamente, pero no crea una buena sensación al usuario. Al realizar peticiones continuas al servidor, el usuario debe esperar a que se recargue la página con los cambios solicitados. Si la aplicación debe realizar peticiones continuas, su uso se convierte en algo molesto. (Eguíluz Pérez, 2005)

1.4.1.2 *Java Script*

Con unas aplicaciones web cada vez más complejas y una velocidad de navegación tan lenta, surgió la necesidad de un lenguaje de programación que se ejecutara en el navegador del usuario. De esta forma, si el usuario no rellenaba correctamente un formulario, no se le hacía esperar mucho tiempo hasta que el servidor volviera a mostrar el formulario indicando los errores existentes.

Este es un lenguaje interpretado basado en guiones que son integrados directamente en el código HTML. El código es transferido al cliente para que este lo interprete al cargar la página. La primera versión de este lenguaje apareció con el navegador Netscape 2.0 en 1995, con el nombre original de LiveScript y soportando gran cantidad de las instrucciones que tiene hoy en día.

Las principales características de este lenguaje son:

- ✓ Es un lenguaje interpretado.
- ✓ No necesita compilación.
- ✓ Multiplataforma.
- ✓ Lenguaje de alto nivel.
- ✓ Admite programación estructurada.
- ✓ Basado en objetos.
- ✓ Maneja la mayoría de los eventos que se pueden producir sobre la página web.
- ✓ No se necesita ningún kit o entorno de desarrollo.

Java Script es un lenguaje de programación que se utiliza principalmente en la creación páginas web dinámicas. Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

Técnicamente, Java Script es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. Con una sintaxis semejante a la del

lenguaje Java y el lenguaje C. Al igual que Java, Java Script es un lenguaje orientado a objetos propiamente dicho, ya que dispone de herencia, si bien esta se realiza siguiendo el paradigma de programación basada en prototipos, ya que las nuevas clases se generan clonando las clases base (prototipos) y extendiendo su funcionalidad. En otras palabras, los programas escritos con Java Script se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. (Eguíluz Pérez, 2008)

1.4.2 Tecnologías del lado del servidor

Es una tecnología que consiste en el procesado de una petición de un usuario mediante la interpretación de un script en el servidor web para generar páginas HTML dinámicamente como respuesta. La ventaja que existe al usar tecnologías del lado del servidor es la seguridad, ya que al ejecutarse el código fuente en el servidor, la programación es transparente al cliente, permitiendo ocultar así los detalles de implementación.

1.4.2.1 PHP

PHP¹⁴ fue creado originalmente por Rasmus Lerdorf en 1994, aunque no hubo una versión utilizable por otros usuarios hasta principios de 1995. Esta primera versión se llamó, Personal Home Page Tools. La implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar de facto para PHP, al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre. Es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno.

¹⁴ Preprocessed Hypertext Pages que al español significa aproximadamente Páginas de Hipertexto Pre procesadas.

Ventajas:

- ✓ Lenguaje multiplataforma.
- ✓ Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- ✓ Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- ✓ Permite las técnicas de Programación Orientada a Objetos.
- ✓ No requiere definición de tipos de variables.

Este posee desventajas como: si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar (muchos otros lenguajes tampoco lo hacen), aun estando dirigido a alguna en particular, el programador puede aplicar en su trabajo cualquier técnica de programación y/o desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del patrón de diseño Modelo Vista Controlador, que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes (Autores, 2005) .

1.4.2.2 JSP

La tecnología JSP¹⁵ es una especificación abierta (y gratis) disponible y desarrollada por Sun Microsystems como una alternativa a ASP¹⁶ de Microsoft, y son un componente dominante de la especificación de (J2EE¹⁷). Muchos de los servidores de aplicaciones comercialmente ya utilizan tecnología JSP. Las páginas JSP están escritas en lenguaje Java. Por lo tanto, las páginas JSP son independientes de la plataforma. Esta permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

¹⁵ Java Server Pages traducido al español es aproximadamente Páginas de Servidor Java.

¹⁶ Active Server Pages, traducido al español es aproximadamente Páginas Activas del lado del Servidor.

¹⁷ Java 2 Enterprise Edition es una plataforma de programación para desarrollar y ejecutar software de aplicaciones en Lenguaje de programación Java.

Las JSP permiten la utilización de código Java mediante scripts. Además es posible utilizar algunas acciones JSP predefinidas mediante etiquetas.

El funcionamiento general de la tecnología JSP es que el Servidor de Aplicaciones interpreta el código contenido en la página JSP para construir el código Java del servlet a generar. Este servlet será el que genere el documento (típicamente HTML) que se presentará en la pantalla del Navegador del usuario. Con JSP podemos crear aplicaciones web que se ejecuten en variados servidores web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Esta tecnología posee muchas ventajas como:

Las páginas JSP son compiladas en Servlets Java y cargadas en memoria la primera vez que se las llama, y son ejecutadas para todas las llamadas siguientes. Esto le da a las páginas JSP gran velocidad y escalabilidad.

Las páginas JSP tienen una característica avanzada conocida como etiquetas extensibles. Este mecanismo permite a los desarrolladores crear etiquetas personalizadas. En otras palabras, las etiquetas extensibles nos permiten extender la sintaxis de las etiquetas de las páginas JSP (Welling, et al., 2005).

1.4.2.3 ASP.NET

Es un framework¹⁸ para aplicaciones web, desarrollado y comercializado por Microsoft. Es usado por programadores para construir sitios web dinámicos, aplicaciones web y servicios web XML. Surge en enero de 2002 con la versión 1.0 del .NET Framework, y es la tecnología sucesora de la tecnología ASP. ASP.NET está construido sobre el Common Language Runtime, permitiendo a los programadores escribir código ASP.NET usando cualquier lenguaje admitido por el .NET Framework.

Las páginas de ASP.NET, son conocidas oficialmente como "web forms" (formularios web), son el principal medio de construcción para el desarrollo de aplicaciones web. Los formularios web están contenidos en archivos con una extensión .aspx, en idioma de programación, estos archivos típicamente contienen etiquetas HTML o XHTML estático, y

¹⁸ Plataforma de desarrollo de software.

también etiquetas definiendo controles web que se procesan del lado del servidor y controles de usuario donde los desarrolladores colocan todo el código estático y dinámico requerido por la página web. Adicionalmente, el código dinámico que se ejecuta en el servidor puede ser colocado en una página dentro de un bloque `<!--código dinámico-- >` que es muy similar a otras tecnologías de desarrollo como PHP, JSP y ASP, pero esta práctica es, generalmente, desaconsejada excepto para propósitos de enlace de datos pues requiere más llamadas cuando se genera la página (Gates, 2000).

1.4.2.4 EXT 2.0

Amparado bajo GPLv3¹⁹, Ext. 2.0 es un framework completo y extremadamente avanzado. Nace como solución a tareas comunes pero complejas. Esta ha creado una solución atractiva y revolucionaria en el tema de la creación de páginas web dinámicas, la tecnología Ext 2.0, que es un lenguaje de desarrollo neutro con respecto a las plataformas. Este framework basa toda su funcionalidad en JS a través de librerías ya muy conocidas: YUI, jQuery para la creación de componentes personalizados y así poder crear web innovadoras con fabulosas interfaces de usuario. Por lo que permite que los programas puedan ser escritos una sola vez y desplegados en cualquier ordenador, explorador o sistema operativo.

Ventajas:

- ✓ Existe un balance entre Cliente – Servidor. La carga de procesamiento se distribuye, permitiendo que el servidor, al tener menor carga, pueda manejar más clientes al mismo tiempo.
- ✓ Comunicación asíncrona. En este tipo de aplicación el motor de render puede comunicarse con el servidor sin necesidad de estar sujeta a un clic o una acción del usuario, dándole la libertad de cargar información sin que el cliente se de cuenta.

¹⁹ Licencia para la distribución del Software Libre versión 3.

- ✓ Eficiencia de la red. El tráfico de red puede disminuir al permitir que la aplicación elija que información desea transmitir al servidor y viceversa, sin embargo la aplicación que haga uso de la pre-carga de datos puede que revierta este beneficio por el incremento del tráfico.

Conclusiones

En este capítulo se han definido los conceptos fundamentales para lograr el desarrollo de un sistema web. Después de haber analizado los conceptos y definiciones abordadas fue posible percatarse de las ventajas que proporciona el uso de herramientas potentes para el desarrollo de una aplicación con las características y complejidad que necesita el cliente, como el uso de la Ext2.0 para el desarrollo de esta para agilizar la implementación y garantizar un mejor mantenimiento de la aplicación. Y el uso del RUP, como metodología de desarrollo del software, por las grandes iteraciones que esta posee así como la gran cantidad de plantillas que esta genera, para así lograr un mayor control sobre el producto.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

Introducción

En este capítulo se abordarán temas como las principales características del sistema y algunos conceptos e importancia de estos. Primeramente habría que decir que el diseño es la comunicación textual (expresada a través de contenidos) como en un sistema web. La unión de un buen diseño con una jerarquía bien elaborada de contenidos aumenta la eficiencia de la Web como canal de comunicación e intercambio de datos, y brinda posibilidades como usabilidad, rapidez y una navegación familiar y simple. (McGOVERN, 2005)

2.1 Estudio y Modelado del Sistema

En el Estudio y Modelado del Sistema es donde se obtiene una primera presentación de cómo será la interfaz de usuario (UI). En esta fase como indica su nombre, es donde se modela el sistema, se hace el levantamiento de requisitos, o sea aquí se estudia el entorno donde está enmarcado el negocio desde el punto de vista de la UI. Se agrupan los requisitos no funcionales de usabilidad y UI, con el objetivo de obtener información subliminar acerca de la preferencia de los clientes.

2.1.1 Requisitos no Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. En muchos casos los requerimientos no funcionales son fundamentales en el éxito del producto ya estos forman una parte significativa de la especificación.

Existen múltiples categorías para clasificar a los requerimientos no funcionales y dentro de estos esta los de IU. Este tipo de requerimiento describe la apariencia del producto. Es importante destacar que no se trata del diseño de la interfaz en detalle sino que especifican

cómo se pretende que sea la interfaz externa del producto. Los requerimientos de apariencia se vuelven más importantes a medida que los productos de software se mueven hacia áreas más orientadas al consumidor. Requerimientos de apariencia también son necesidades de cumplir con normas estándares. (JACOBSON, et al., 2000) (BOOCH, et al., 2000)

2.2 Arquitectura de la Información

En el proyecto CCV la Arquitectura de la Información se define como un proceso interactivo que actúa durante todo el diseño del sistema y en cada una de sus fases para asegurar el cumplimiento real de los objetivos de su producción y del desarrollo de la interfaz. De esta manera es posible que diseñadores y desarrolladores de interfaces cuenten con un estilo de trabajo concreto, a través del cual resuelvan todas las dudas de los elementos que componen esta página.

El uso de la Arquitectura de Información en un proyecto como el desarrollado en CCV, brinda muchas ventajas, ya que con su uso se facilita el proceso de corrección, debido a que siempre existe una interacción constante con el cliente, lo que trae consigo que ningún elemento en las páginas queda puesto al azar sino que responde a necesidades puntuales que se han detectado.

Todo esto es posible, debido a que la arquitectura de información cuenta con varias fases capaces de lograr un sistema capaz de satisfacer todas las expectativas del cliente, como son el diseño de la estructura de páginas, este permite la generación de dibujos sólo lineales que describen los componentes de cada una de las pantallas del sitio, con el objetivo de verificar la ubicación de cada uno de ellos. Otra de las fases con las que cuenta es con la creación de bocetos de diseño el cual es capaz de generar dibujos digitales acabados de la forma que tendrán las páginas principales del sitio que se desarrolla. Dando seguimiento a los borradores de página, existe la creación de borradores de páginas, que no es más que tomando el resultado de la etapa anterior se genera un prototipo de interfaz programada, para mostrar los resultados de el uso de tecnologías de creación de interfaces web.

Por último para brindar un resultado satisfactorio se encuentra la etapa de la maqueta web, en la cual se genera todo el sitio utilizando imágenes y contenidos reales.

2.3 Características del Sistema

Se trata de un sistema gubernamental, al servicio del Ministerio de Energía y Petróleo por parte de la República Bolivariana de Venezuela y al Ministerio para la Inversión Extranjera y la Colaboración Económica por la parte de la República de Cuba, lo que requiere un compromiso extremo. El proyecto CCV debe implementar un sistema eficiente y a la vez fácil de manejar, para ello se deben tener en cuenta dos características fundamentales: el tipo de usuario y tipo de aplicación.

2.3.1 Usuarios

Debido a que se pretende realizar un sistema web para el uso y control de proyectos entre dos países, los usuarios que interactúan en el son usuarios de un alto nivel cultural pero aun así hay que tratarlos como usuarios que son, por lo que se trató de eliminar un error común en los sistemas web, que es que los sitios se construyen más para los diseñadores y sus necesidades que para los verdaderos usuarios. Esto es muy común debido a que muchos diseñadores no siempre comprenden las solicitudes de los clientes o ignoran las necesidades de estos, por lo que el resultado final de esto es la creación de un diseño en el cual el cliente no tiene ni el más mínimo conocimiento, ya que lo que el diseñador entiende no es lo que un usuario entenderá. Como diseñador, tiene profundo conocimiento del sitio Web, conoce dónde se encuentran la información, sabe cómo instalar programas complementarios, dispone de una resolución óptima de pantalla, etc. Por lo que el diseñador debe de ser capaz de comprender que él no es el usuario, y aceptar el hecho de que la mayoría de los usuarios no tienen un profundo conocimiento del sitio que se está construyendo. Por lo que en el proyecto CCV se pensó en el usuario, preguntándoles para diseñar el sitio de la manera que ellos desean, logrando así un diseño más sencillo y comprensible.

2.3.2 Tipo de Aplicación

En este punto se abordará sobre el tipo de aplicación y la envergadura que alcanza el sistema, debido a que esta aplicación controlará grandes flujos de información de gran importancia y confidencialidad para los países involucrados en ella. Además de que en este sistema existen usuarios de alto nivel como pueden ser desde Vice-presidentes y Ministros hasta un director de empresa. También es un sistema que demanda mucho tiempo de intercambio entre el usuario y el ordenador, lo que exige la creación de una interfaz formal, pero siempre con un diseño refrescante a la vista, aplicando para esto un correcto uso de colores.

2.4 Convenciones de Archivos

El proyecto Web que se creará en el Eclipse para el desarrollo del sistema será “CCV”, al igual que la URL para acceder al mismo. Todas las clases y las páginas JS y otros ficheros del sistema estarán agrupadas en una estructura de carpetas y paquetes.

Las clases y ficheros de recursos se encontrarán dentro de la carpeta WebContent, haciendo uso de las convenciones de nombrado de paquetes y adaptándolas al sistema quedaría como paquete principal:

styles

A partir de este nivel se agregaría la separación por subsistemas o módulos:

styles.administracion (Subsistema Configuración y Administración)

styles.common (Subsistema que contendría las clases y componentes comunes para el resto de los subsistemas)

styles.contratacion (Subsistema Contratación de Proyectos)

styles.financiamiento (Subsistema Financiamiento de Proyectos)

styles.misiones (Subsistema Misiones)

styles.presentacion (Subsistema Presentación de Proyectos)

styles.seguimiento (Subsistema Seguimiento de Proyectos)

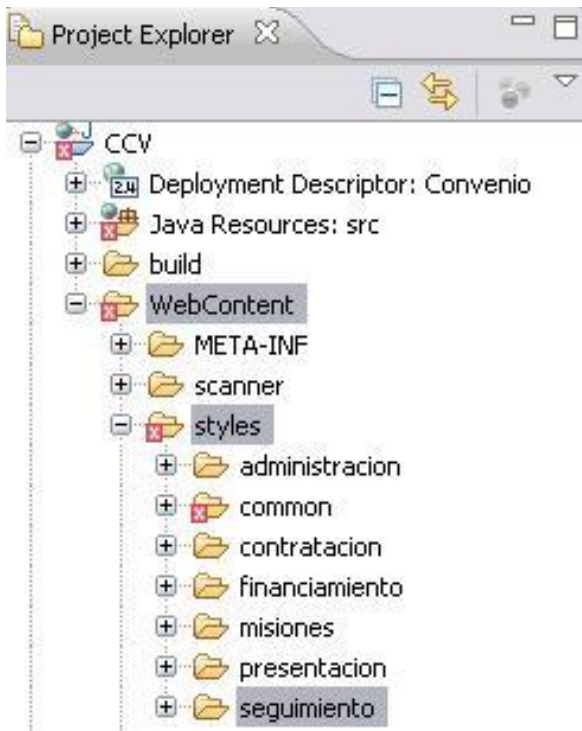


Figura 9 Organización de los subsistemas o paquetes.

Dentro de cada subsistema se agregarían varias subcarpetas en las cuales se encuentran la de la CSS, la subcarpeta de la JS y por último la subcarpeta XML, a utilizar por el subsistema. En este caso se muestra como se vería en el subsistema Seguimiento de Proyectos.

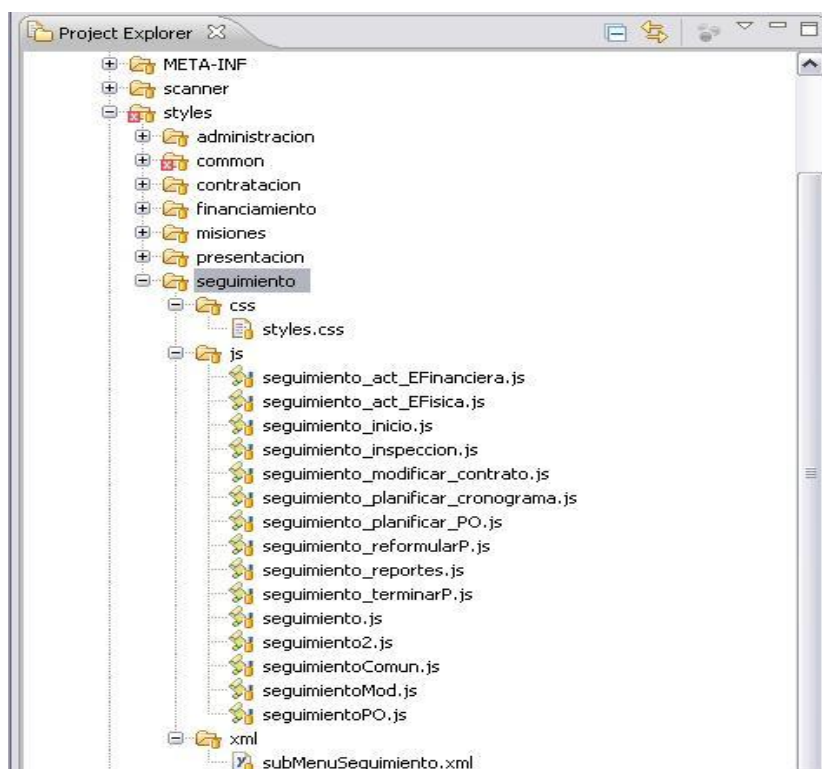


Figura 10 Organización del subsistema o paquete Seguimiento

styles.seguimiento.css (Dentro de esta subcarpeta se encuentra la .css, que no es más que el estilo que se le aplicara al subsistema de seguimiento)

styles.seguimiento.js (Aquí se encuentra las .js, que es donde se encuentra la implementación de los componentes que se utilizaran en el subsistema seguimiento)

styles.seguimiento.xml (En esta subcarpeta se encuentra el .xml, que este contiene el menú del subsistema de seguimiento)

2.4.1 Estándares web

Se ha comprobado que el uso de estándares Web es de vital importancia en el desarrollo de aplicaciones web de este tipo, la funcionalidad del sistema puede rendir al máximo pero sin una buena arquitectura de la información el funcionamiento no tiene validez. Sin una correcta ubicación de los contenidos, los componentes ordenados siguiendo una secuencia

lógica, sin un adecuado agrupamiento de los elementos comunes no tiene sentido alguno que detrás de la interfaz se haya desarrollado una compleja programación para hacer funcionar la vista, ya que el usuario que interactúa con esa pantalla pierde la noción de lo que debe hacer o puede hacerse, por lo que tiende a confundirse fácilmente.

El diseño del sistema de CCV es lo que el cliente observa, es la portada del sistema, la primera impresión que el usuario capta, por lo que el sistema tiene que ser funcional al 100% pero a su vez debe de agradar al cliente, brindar usabilidad y una buena estructuración de los contenidos. Todos los componentes de una aplicación Web son fundamentales para lograr los objetivos y obtener un buen resultado, pero con lo que el usuario interactúa es con la interfaz de usuario, por lo que el buen uso de estándares Web y la aplicación de una correcta Arquitectura de la Información son imprescindibles. La implementación de interfaz de usuario, consiste en desarrollar las pantallas para cada uno de los Casos de Uso del sistema, empleando estándares Web, por lo que se emplean tecnologías que cumplan con los mismos, y aplicando las pautas de diseño establecidas.

El proyecto CCV está construido en una plantilla escrita en lenguaje Java Script, que es la página inicial y a través de la cual se estructura el sistema, cada una está igualmente dividida en las distintas secciones de la aplicación, secciones que servirán para generar la presentación desde el archivo CSS. El estilo CSS es el encargado de aplicar los estándares de diseño antes establecidos y controlar todo lo que a presentación, ubicación y estética se refiere. Se emplean propiedades generales para todo el documento y, mediante identificadores, se distinguen cada una de las secciones o áreas de la aplicación con objeto de designar estilos específicos para cada una de ellas.

El estilo está desarrollado en un solo fichero CSS que se carga una vez al inicio cuando se ejecuta el subsistema de seguimiento, lo que hace óptimo el mantenimiento de la interfaz. En él se define el espaciado, separación, tipo y color de letra, tamaño de los componentes, cambio del color de las letras que componen una actividad al pasar de un estado a otro, entre otros. Todos estos estilos generales le dan la presentación definida a cada una de las vistas del sistema siempre que se utilice la estructura correcta en la creación de las páginas JS.

La funcionalidad e interactividad de la interfaz se logra con la utilización de Java Script, lenguaje de programación del cliente que está siendo fundamental para el desarrollo de aplicaciones Web dada su inclusión en la tecnología de punta AJAX. Esta tecnología tiene como una de sus más importantes funcionalidades el no tener que enviar en una petición todo el documento al servidor, sino solamente los datos necesarios, así cuando se realiza una petición desde el menú o una pantalla funcional solamente el área de contenido se actualiza. El uso de este resuelve asuntos de usabilidad comunes como la navegación y detección del navegador, soportar cambios de direcciones y la habilidad de degradar cuando Java Script no es completamente soportado en el cliente. Proporciona una amplia gama de opciones en una sola biblioteca Java Script.

2.5 Diseño del Sistema Web

Con el desarrollo del aspecto visual del modulo seguimiento del sistema web, se pretende lograr los siguientes objetivos:

- ✓ Reflejar la imagen e identidad de la oficina del Convenio Cuba Venezuela ante las empresas y organizaciones de ambas naciones que laboran en el marco del ALBA por la integridad y las mejoras económicas y sociales.
- ✓ Optimizar el flujo de información y los niveles de acceso que actualmente tienen los involucrados que laboran en el marco del Convenio.
- ✓ Crear un nuevo espacio de comunicación entre los involucrados que facilite la gestión de los trámites necesarios para que los proyectos de colaboración sean concebidos y ejecutados de manera más eficiente.
- ✓ Facilitar la consulta de informaciones de la ejecución de los proyectos de colaboración entre ambas naciones que apoye la toma de decisiones y la gestión de gobierno.

2.5.1 Elementos del Sistema de Navegación

En este epígrafe se abordará como se desarrollará la navegación en el sistema, ya que esta deberá de ser fácil al usuario que interactúa con el sistema, pero siempre logrando las funcionalidades requeridas por estos.

Menú General: siempre presente en todo el sitio, permite el acceso a cada una de las áreas del sitio.

Pié de Página: usualmente ubicado en la parte inferior de cada página, indica el nombre de la institución, teléfonos, dirección física y de correo electrónico.

Barra Corporativa: ofrece diversas opciones de información respecto del sitio y tal como el anterior, se muestra en todas las páginas.

Ruta de Acceso: listado que aparece en la parte superior de cada página y que muestra el trazado de páginas que hay entre la Portada del sitio hasta la página actual que se esté revisando; cada una de ellas debe tener un enlace, para acceder al área de la cual depende la página. Cada uno de los elementos que conforman este «camino» debe tener un enlace que permita el acceso a esas áreas. En la literatura internacional en inglés sobre este tema, se llama a este elemento como «breadcrumbs».

Fecha de publicación: para saber la vigencia de publicación del contenido desplegado.

Botón Inicio: para ir a la portada

Botón Mapa del sitio: para ver el mapa del Sitio Web

Botón Contacto: para enviar un mensaje al encargado del sitio

Botón Ayuda: para recibir ayuda sobre qué hacer en cada pantalla del sitio.

Botón Imprimir: para imprimir el contenido de la página; se espera que el formato de impresión del documento que se muestra en pantalla sea más simple que la página normal del Sitio Web, para dar la impresión al usuario de que hay una preocupación por ayudarlo en la tarea de llevar impreso el contenido.

Los siguientes Botones demás botones que estarán presentes en el sistema, serán botones comunes y tendrán los siguientes textos en dependencia de función:

Aceptar: para aceptar las acciones propuestas en un determinado contesto .

Rechaza: para rechazar las acciones propuestas en un determinado contexto.

Siguiente: para avanzar a siguientes escenarios.

Atrás: para regresar a escenarios anteriores.

Finalizar: para finalizar la creación o revisión de un documento determinado.

Cancelar: para cancelar la acción en curso.

2.5.2 Distribución de la página

La diagramación general de la aplicación quedó estructurada como sigue:

- ✓ Menú lateral: Casos de uso.
- ✓ Área de Trabajo.
- ✓ Identidad Visual Institucional.
- ✓ Cabezal o área de Identificación y fecha.
- ✓ Trazabilidad.

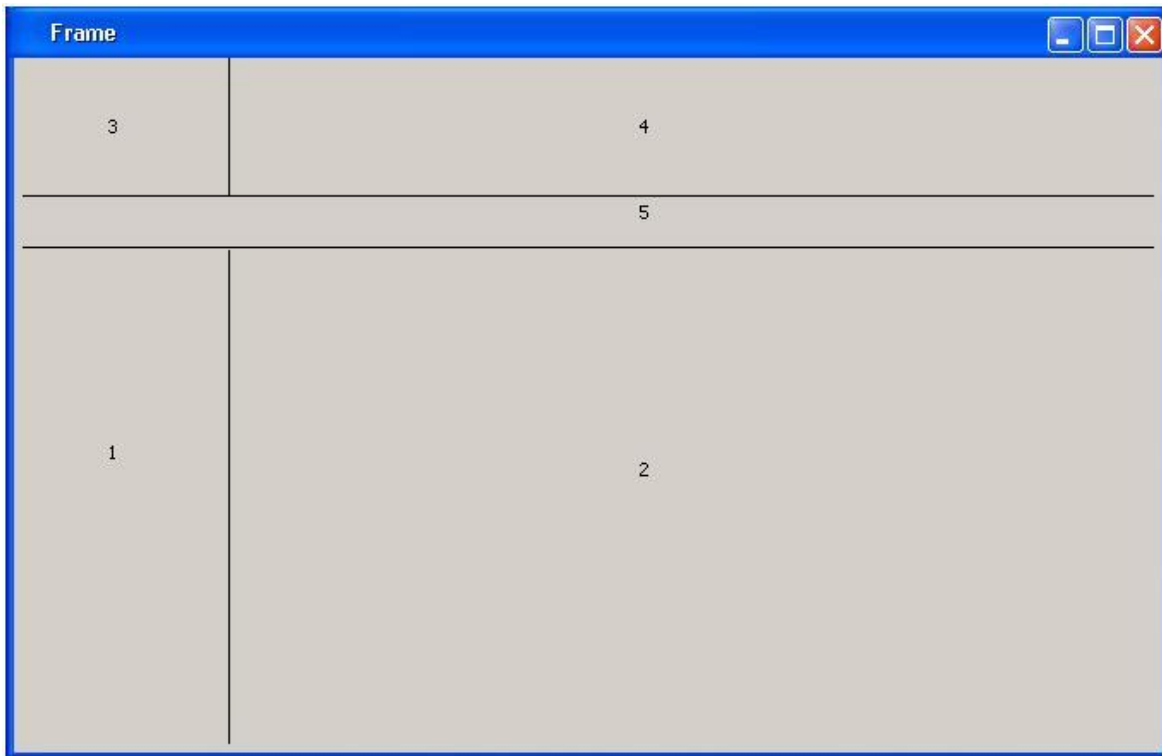


Figura 11 Prototipo del diseño general del sistema.

Partiendo de las pautas diseñadas en general se definieron estándares por secciones con el objetivo de diferenciar cada una de las áreas de acuerdo al objetivo de cada uno de ellos. Cada sección tiene un significado y una importancia que se hace resaltar en la definición de pautas. Una vez especificadas se muestran al cliente con el objetivo de obtener su aprobación, y a partir de ese paso se comienza a desarrollar.

Menú lateral:

El menú lateral está ubicado a la izquierda como buena práctica de diseño ya que el usuario acostumbra a verlo en esta parte de su navegador. En este se muestran los casos de uso correspondientes. Cuando se pase el ratón por encima se manifiesta un cambio de color y se restablece el estado anterior cuando se retira el ratón del ítem. Al hacer clic en un ítem éste se despliega mostrando todos los elementos que este contiene.

Área de Trabajo:

Es en esta área donde se mostrarán todas y cada una de las vistas del sistema, siendo algunas de estas simples, pero muchas otras de complejidad alta, se debe lograr una correcta distribución de los elementos que permita aprovechar el espacio. Se tratará en lo posible de ubicar toda la información dentro del área visible del navegador, Los objetivos son mostrar al usuario una imagen limpia y poco cargada, fácil de comprender. Para todo ello se utiliza una tonalidad clara, buscando el contraste con la tipografía y una separación adecuada de los controles. La tecnología empleada para la implementación de la interfaz de usuario brinda la posibilidad de sólo actualizar el Área de Contenido, esto es, cuando se hace clic en determinado Caso de Uso en el menú lateral, se muestra una vista en el Área de Contenido, y ésta es la única sección de la aplicación que es actualizada, lo que proporciona una mayor rapidez de interacción e implica un mejor funcionamiento del sistema.

Identidad Visual Institucional:

En esta se presenta el logo de la institución. Es una sección muy importante porque es donde se representa la validez legal de la aplicación además que aporta un elemento muy particular con la que se identifica el sistema.

Cabecal o área de Identificación y fecha:

Es en se muestra el nombre del sistema junto a las banderas de los dos países representados en el sistema, a la derecha se muestra la fecha y la hora. También en la parte superior derecha se muestran un conjunto de funcionalidades como es la de salir del sistema y otras que brindan soporte al usuario, como las de calculadora, escáner, mensajes y una de las más importantes como es la ayuda.

Trazabilidad:

En este se visualiza los ítems por los cuales el usuario se ha ido desplazando.

2.5.3 Diseño de la Estructura de las Páginas.

Interfaz de la página de inicio del sistema

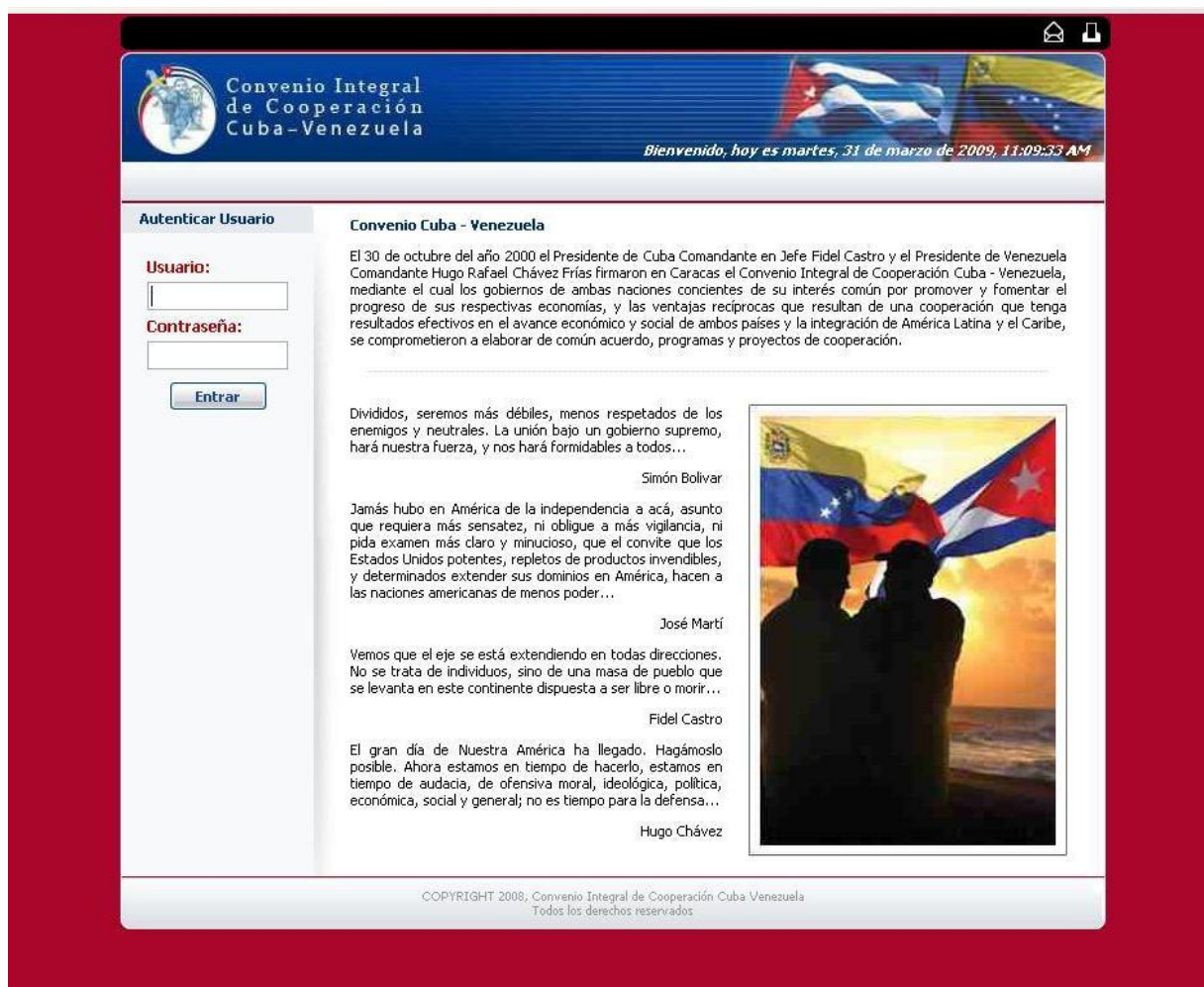


Figura 12 Interfaz de inicio del sistema.

La página de inicio de sesión es a través de la cual se accede al sistema y subsistemas que tiene acceso el usuario de acuerdo a los privilegios establecidos en la jerarquía de usuarios. En esta pantalla de inicio de sesión es donde el empleado se registra con su nombre de usuario, contraseña para acceder a la siguiente vista donde se listan los subsistemas o módulos en los cuales puede trabajar.

Interfaz de Subsistemas o módulos

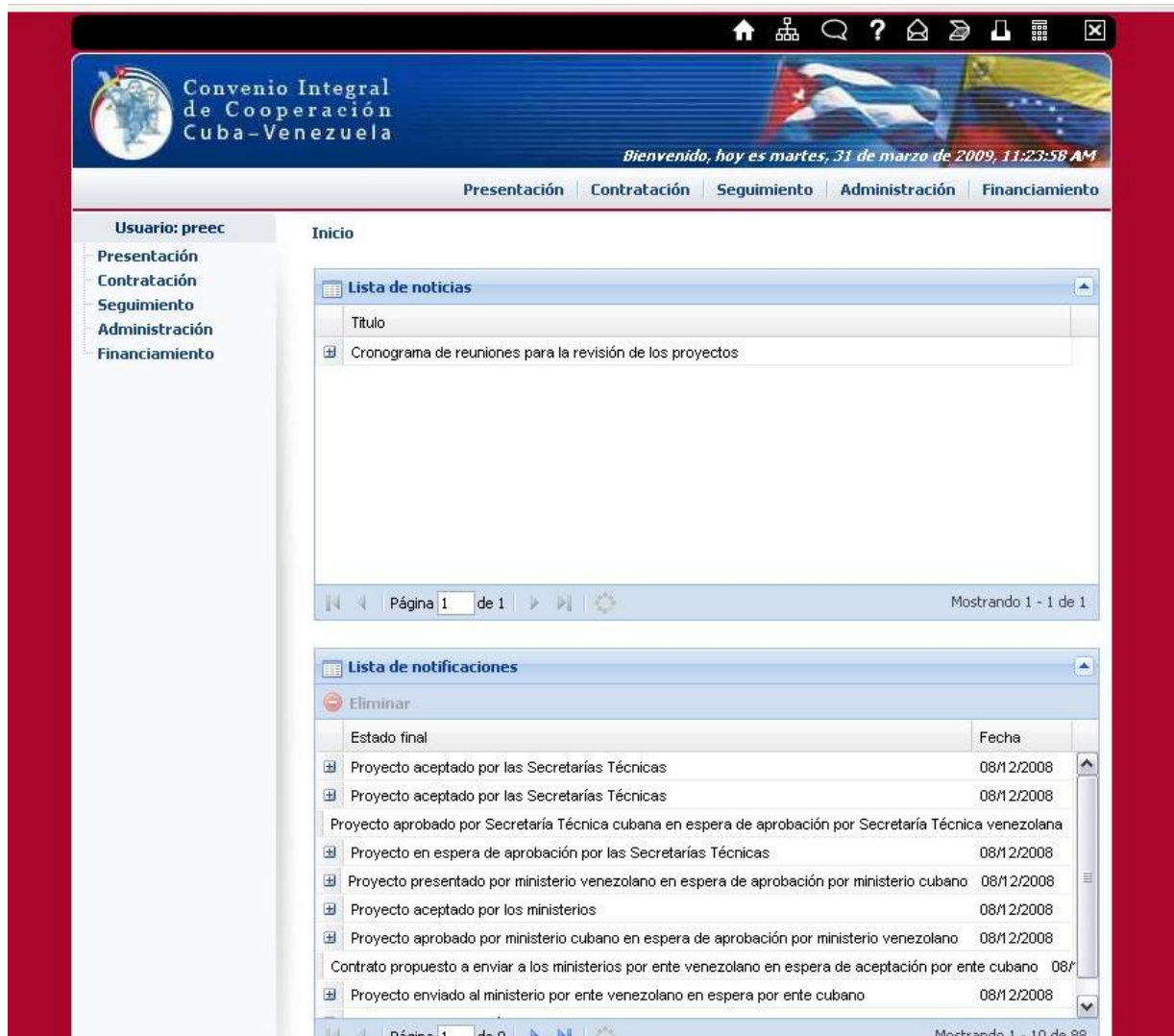


Figura 13 Interfaz de subsistemas o módulos del sistema.

A la página de subsistemas o módulos se accede posterior al registro de usuario en la página de inicio de sesión, en la parte superior de la pantalla, y debajo del identificador del sistema, se muestra el nombre del empleado registrado actualmente en el sistema y los subsistemas correspondientes. Además, dada la posible necesidad que el usuario una vez registrado desee volver a la página de inicio de sesión para entrar al sistema como otro usuario, dispone en la parte superior derecha de la pantalla un enlace “Salir” que lo transporta hacia dicha vista.

Interfaz que contienen una tabla.

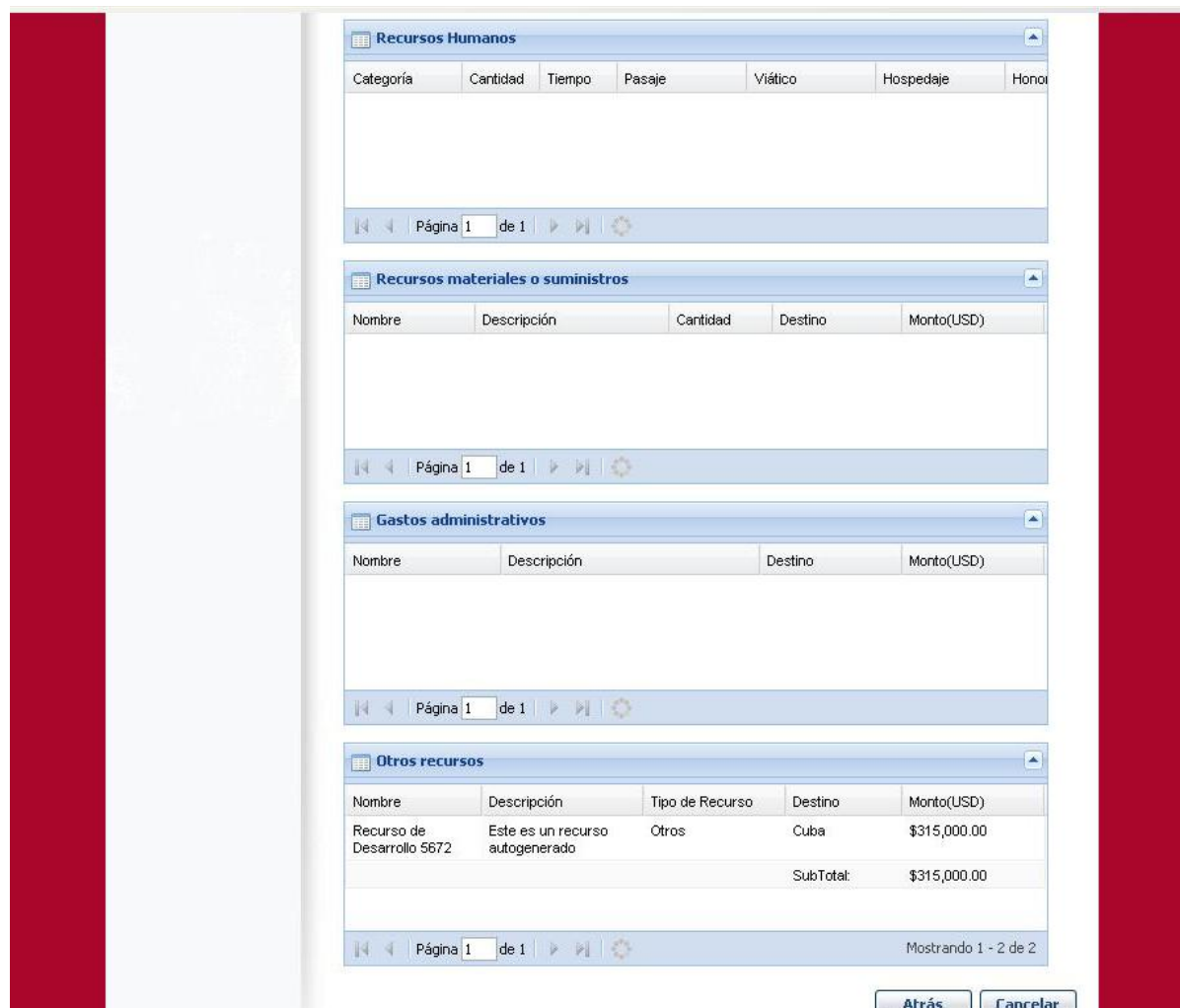
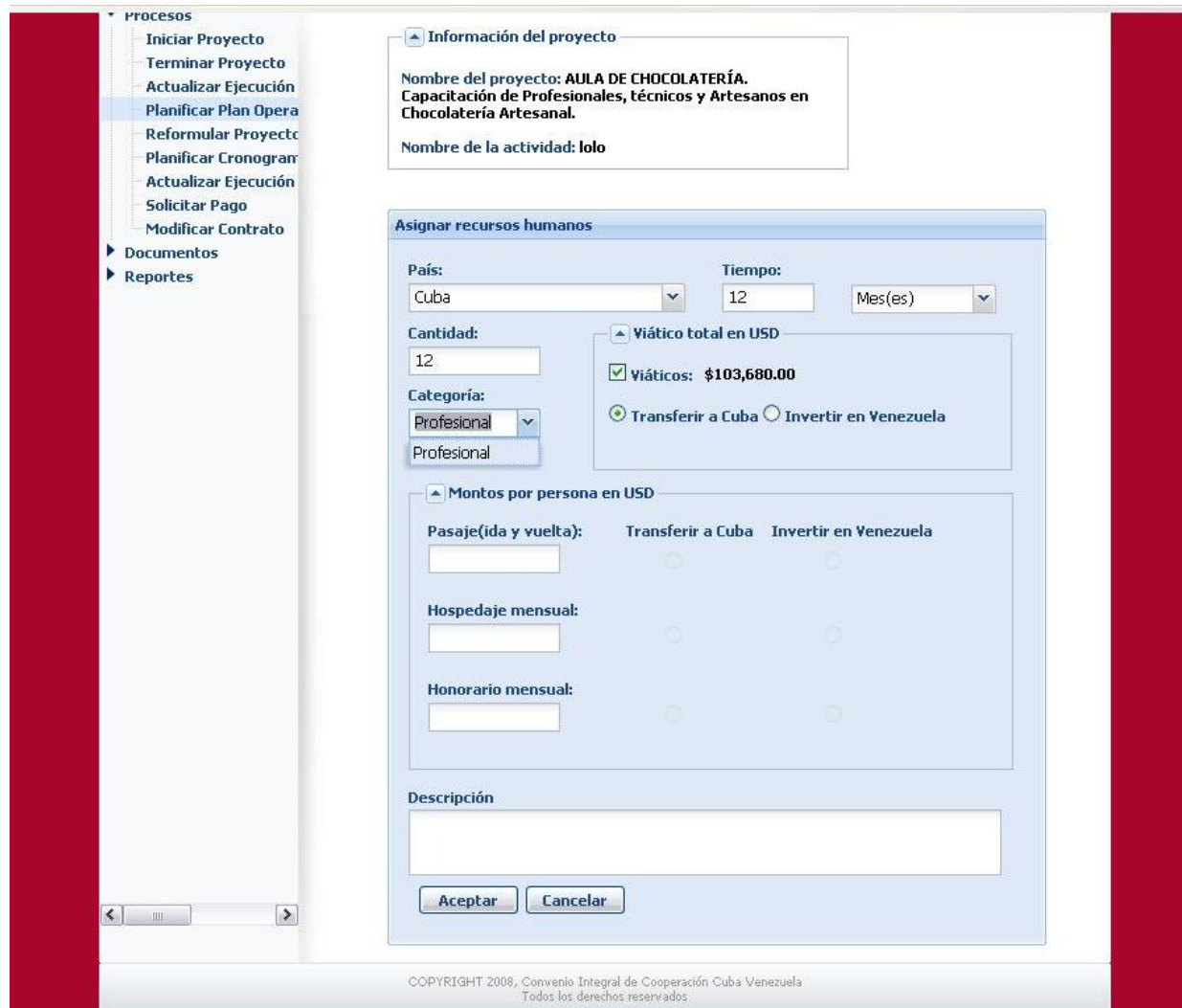


Figura 14 Interfaz de tablas.

Las tablas creadas en la EXT 2.0 ofrecen facilidades para el manejo de datos, acciones como ordenar descendente o ascendentemente. Para las filas seleccionadas se crea un estilo visual, consistente en cambiar el color de fondo por uno más fuerte, que resalte al usuario con objeto de distinguir fácilmente entre aquellos elementos seleccionados y los demás. Se le añadió un sistema de paginado para no hacer demasiada extensa la tabla cuando se manejen grandes cantidades de datos, cumpliendo siempre con las pautas de diseño establecidas. De esta forma se puede mostrar un número de filas adecuado a cada caso. El paginado permite navegar de página en página a través de los controles anterior y siguiente.

Componentes



Procesos

- Iniciar Proyecto
- Terminar Proyecto
- Actualizar Ejecución
- Planificar Plan Opera
- Reformular Proyecto
- Planificar Cronogram
- Actualizar Ejecución
- Solicitar Pago
- Modificar Contrato

Documentos

Reportes

Información del proyecto

Nombre del proyecto: AULA DE CHOCOLATERÍA.
Capacitación de Profesionales, técnicos y Artesanos en Chocolatería Artesanal.

Nombre de la actividad: lolo

Asignar recursos humanos

País: Cuba Tiempo: 12 Mes(es)

Cantidad: 12

Categoría: Profesional

Viático total en USD

Viáticos: \$103,680.00

Transferir a Cuba Invertir en Venezuela

Montos por persona en USD

	Transferir a Cuba	Invertir en Venezuela
Pasaje(ida y vuelta):	<input type="text"/>	<input type="text"/>
Hospedaje mensual:	<input type="text"/>	<input type="text"/>
Honorario mensual:	<input type="text"/>	<input type="text"/>

Descripción

Aceptar Cancelar

COPYRIGHT 2008, Convenio Integral de Cooperación Cuba Venezuela. Todos los derechos reservados.

Figura 15 Interfaz de componentes.

Se han implementados componentes de forma que ayuden al usuario a trabajar con el sistema, partiendo de la filosofía de los expedientes manuales con los cuales estos trabajaban anteriormente. En los cuales se recogen los datos de las actividades de los proyectos, tales como fecha de culminación, recursos financieros y profesionales, cronogramas de ejecución financiera de estas, etc. A la vez que se está trabajando en crear o modificar actividades, se le mostrara un mensaje de error si el usuario ha cometido alguno, mostrando el componente subrayado de una línea roja y mostrando al lado de este un signo

de interrogación con el motivo de error. Cada forma consta de botones Aceptar y Cancelar que se encargan de controlar el proceso de almacenamiento de datos.

Flujo de trabajo

The screenshot displays a web-based interface for project management. On the left, a vertical sidebar contains a list of actions: Terminar Proyecto, Actualizar Ejecución, Planificar Plan Opera, Reformular Proyecto, Planificar Cronogram, Actualizar Ejecución, Solicitar Pago, and Modificar Contrato. Below these are sections for 'Documentos' (Ficha de Proyecto, Contratos, Plan Operativo, Cronograma de Ejec, Solicitud de Pago, Observaciones) and 'Reportes'. The main area is a form titled 'Datos del proyecto'. It contains the following fields: 'Nombre del proyecto:' with the value 'Asistencia técnica para el fortalecimiento institucional c'; 'Duración en meses:' with a dropdown menu set to '12'; a 'Modalidades' section with a list of options: 'Asistencia Técnica y Suministro', 'Soluciones Integrales', 'Servicios', and 'Proyecto Investigación', each with a checkbox and a small icon; 'Ministerio responsable venezolano:' with a dropdown menu set to 'Ministerio del Poder Popular para la Agricultura y Tierras'; 'Ente ejecutor venezolano:' with a dropdown menu set to 'Corporación Venezolana Agraria'; 'Ministerio responsable cubano:' with a dropdown menu set to 'Ministerio de la Industria Alimenticia'; and 'Ente ejecutor cubano:' with a dropdown menu set to 'Instituto para la Investigación para la Industria Alimenticia'. At the bottom of the form are three buttons: 'Siguiete', 'Aceptar', and 'Cancelar'.

Figura 16 Interfaz de flujo de trabajo.

La necesidad de crear componentes que guíen al usuario paso a paso durante el flujo de trabajo es evitar que este cometa errores en un flujo delicado, como en el caso de reformular proyecto donde se debe controlar bien los pasos de los usuarios debido a que de este depende el futuro que puede tomar un proyecto. También surge esta necesidad debido a que en un proyecto un flujo puede ser muy grande, por lo que se decide dividir los mismos en varias pantallas para ubicar en el área visible del navegador.

Mensajes

En el sistema se definieron un conjunto de mensajes en forma de popUps, se muestran en ventanas auxiliares porque requieren la total atención por parte del usuario. Estos mensajes pueden ser de tipo informativos ya que se muestran cuando es necesario brindar una información al usuario, o bien preguntar por una operación o confirmar una acción. Las opciones que brinda este tipo de mensaje son: Sí, No y Aceptar. Como se muestra en la siguiente imagen se muestra un mensaje informativo para confirmar una acción por el usuario.

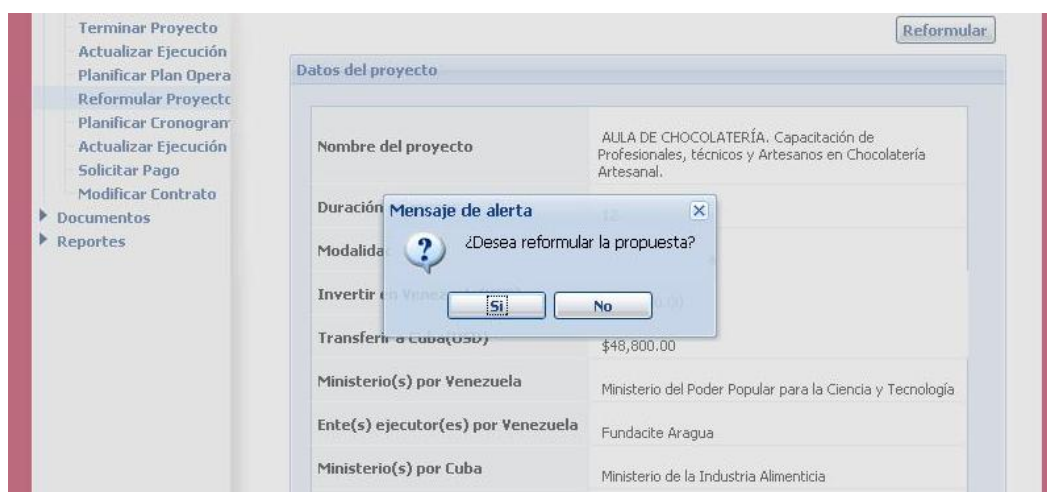


Figura 17 Mensaje de información.



Figura 18 Mensaje de alerta.

También existe el mensaje de inactividad, este se muestra cuando el usuario que está conectado al sistema se pasa un tiempo prudente sin ejecutar ninguna acción dentro del sistema. Este surge por la necesidad de brindarle seguridad al sistema, debido a que el usuario por una razón cualquiera se ha alejado de la PC y dejó el sistema abierto, así que al salir este habría que volver a introducir el usuario y contraseña nuevamente.



Figura 19 Mensaje de inactividad.

2.5.4 Validaciones

La validación en los componentes creados en la interfaz de usuario es fundamental ya que asegura que los datos lleguen correctamente al servidor de base de datos, y reduce al mínimo el margen de error. Con objeto de evitar al máximo los errores de entrada de datos por parte del usuario, siempre que sea posible se utilizan componentes donde el usuario seleccione valores y no efectúe entradas de datos, como el caso de checkbox, radiobutton y selects, entre otros.

Hay casos particulares de datos requeridos para los que se valida toda forma posible de entrada de datos, mediante expresiones regulares se definen los posibles caracteres de entrada y cuando se detecta un error el campo se muestra subrayado de una línea roja y mostrando al lado de este un signo de interrogación con el motivo de error.

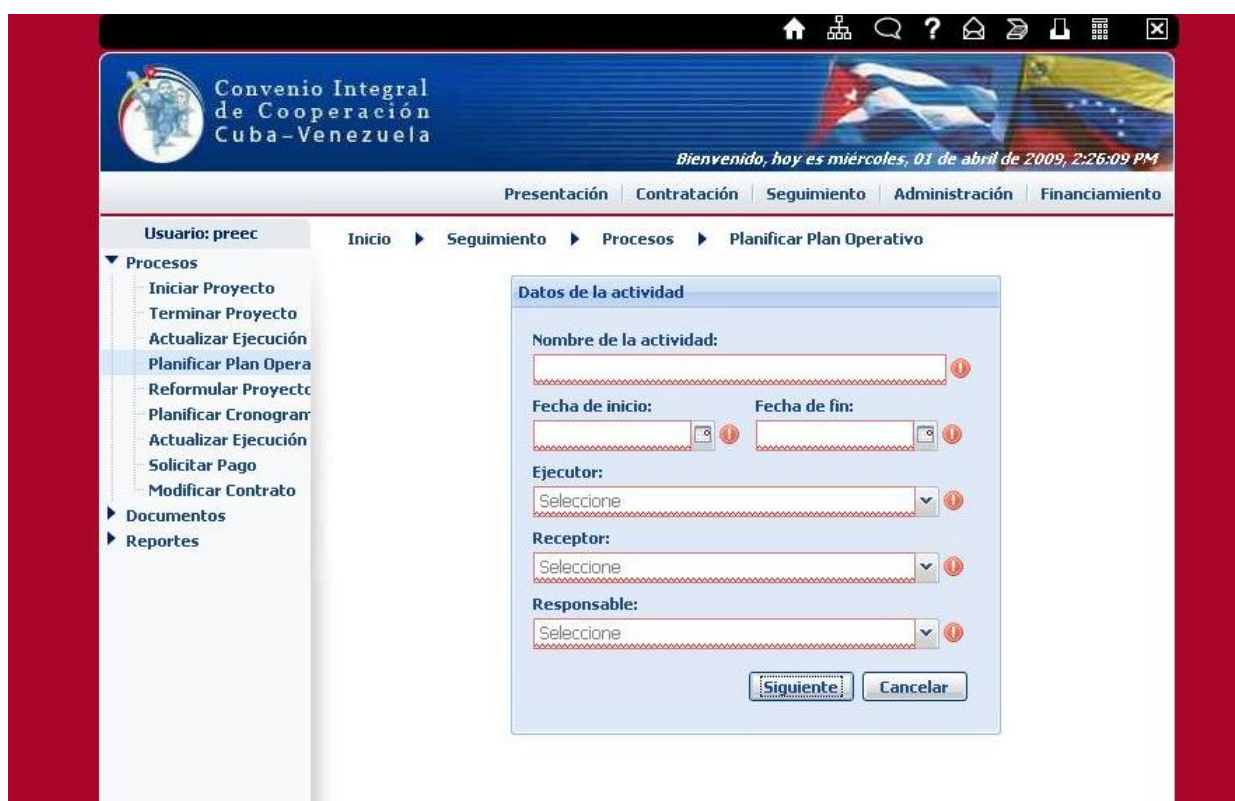


Figura 20 Interfaz de validación de errores.

2.6 Subsistema de Implementación

Una colección de componentes y otros subsistemas de implementación usados para estructurar el modelo de implementación y dividirlos en pequeñas partes que pueden ser integradas y probadas de forma separada. Los subsistemas de implementación incluyen dependencias y otras informaciones.

También podrían incluir modelos claves del subsistema (diagramas de componentes, modelo de despliegue). Además un subsistema puede implementar las interfaces que representan la funcionalidad que exportan en forma de operaciones.

CAPÍTULO 2: Características del Sistema

El diagrama que se presenta a continuación muestra la relación entre los Subsistema de Implementación en los que queda estructurado la capa UI. Este representa una división física del sistema.

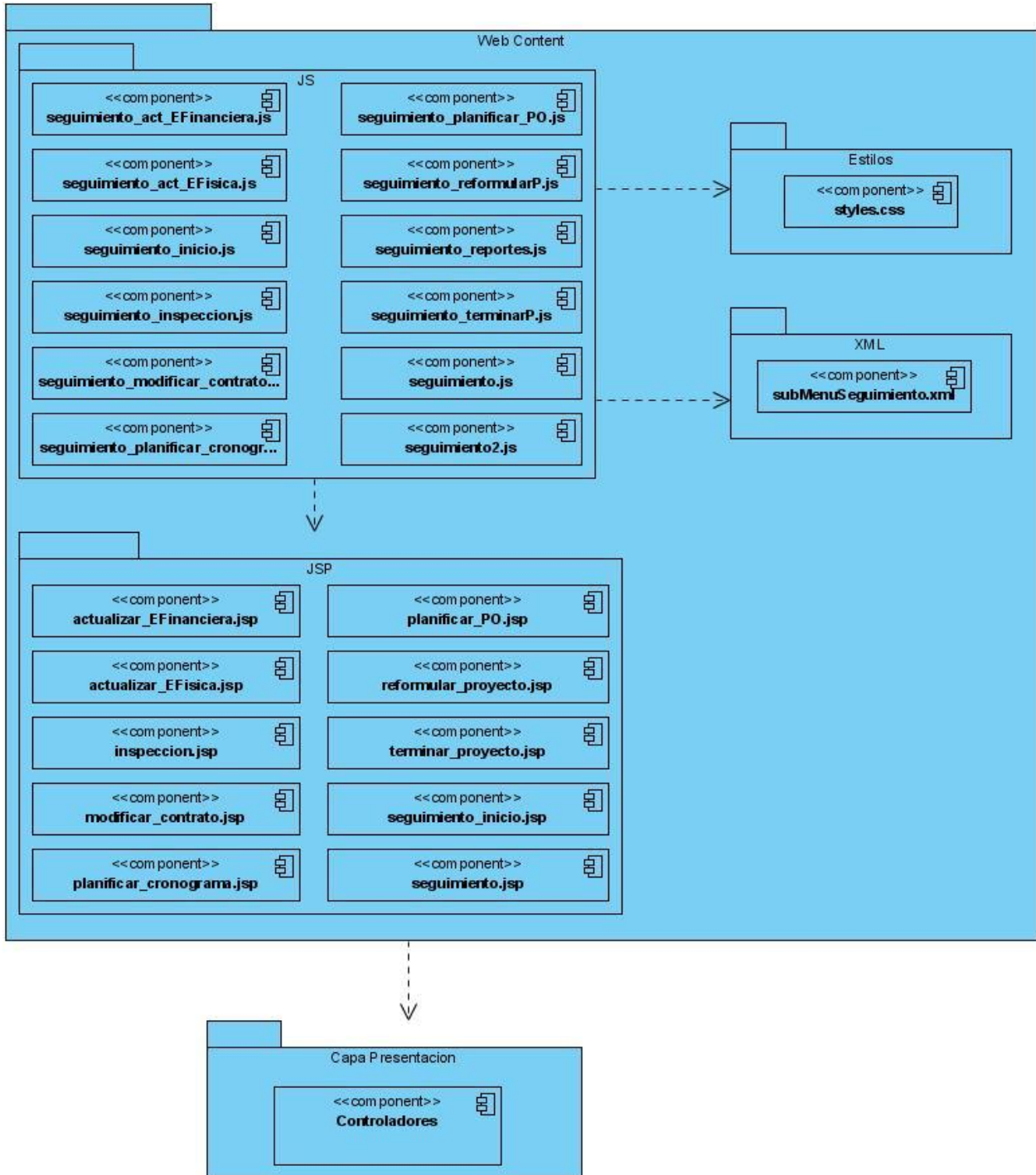


Figura 21 Subsistema de implementación.

Como se muestra en la figura anterior, el desarrollo de la capa de Interfaz de usuario está organizado de la siguiente manera. Está compuesta por un paquete Java Script (JS), en este se encuentran todas las páginas Java Script, la cual implementan los componentes utilizados como sus funcionalidades. También cuenta con un paquete Java server page (JSP), en esta se encuentran las páginas .JSP las cuales son las encargadas de publicar las paginas .JS. Y por último existe una relación muy estrecha entre el paquete .JS con el paquete CSS, en este se encuentra el estilo definido de cada componente, así como el estilo que tomará el diseño de la interfaz final. Todo esto se relaciona con la Capa Presentación, que es capa intermedia entre las capa de Negocio de un sistema y la de Interfaz.

2.6.1 Modelo de Componentes

Un componente es una parte física que se encuentra en la computadora como por ejemplo librerías, ejecutables, archivos de datos entre otros. Con la realización del diagrama de componentes se tiene una versión de las relaciones que tendrán los componentes del sistema. A continuación se muestran los diagramas de componentes por cada uno de los subsistemas definidos para implementar el la capa interfaz del módulo Seguimiento:

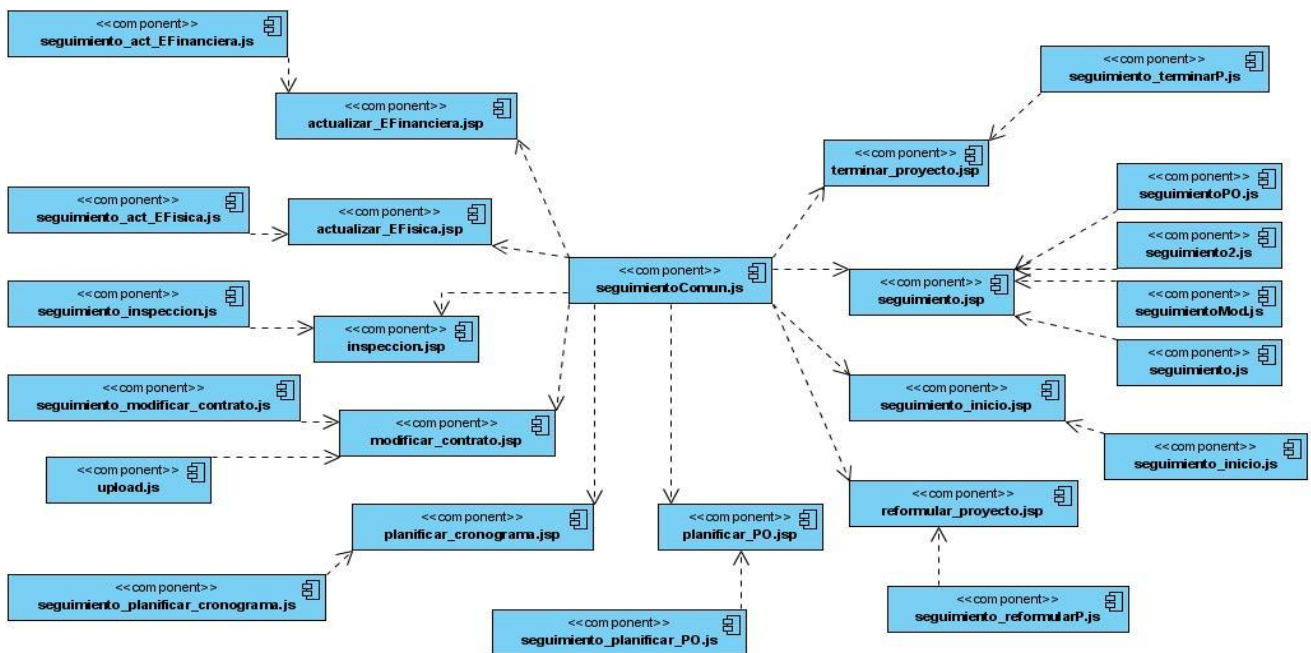


Figura 22 Diagrama de Componentes de la Capa Interfaz perteneciente al módulo Seguimiento.

Como se muestra en la imagen anterior, se especifica la relación existente entre las JS y las JSP, ya que en las páginas JS es donde se desarrolla el código java script que es el encargado de crear los componentes necesarios para desarrollar las funcionalidades requeridas por el cliente y que componen la interfaz de usuario, siempre utilizando los estilos definidos, y este se vincula estrechamente con las paginas JSP, ya que estas son las encargadas de mostrarlo.

Conclusiones

De manera general en este capítulo se presentó una propuesta para darle solución a la interfaz de usuario del módulo Seguimiento, donde se estructuró el la capa en paquetes, para su posterior implementación, se modeló el diagrama de componentes, se abordaron las estrategias utilizadas en cada componente desarrollado describiendo sus funcionalidades. Logrando así un sistema confiable y funcional.

CAPÍTULO 3: ANÁLISIS DE LA SOLUCIÓN PROPUESTA

Introducción

Las pruebas son de gran importancia en la garantía del software. Por lo general están orientadas a comprobar la funcionalidad (el sistema debe comportarse a la altura de los requisitos especificados por el cliente) y lógica de negocio. En ella se realizan diferentes procesos con el objetivo de asegurar la calidad, entre otros factores de gran importancia.

Es importante destacar que los objetivos principales de la realización de una prueba son:

- ✓ Detectar un error.
- ✓ Tener un buen diseño de caso de prueba.
- ✓ Descubrir un error no descubierto antes.

3.1 Pruebas de Software

Una de las últimas fases del ciclo de vida antes de entregar un programa para su explotación, es la fase de pruebas. Esta no es más que los procesos que permiten verificar y revelar la calidad de un producto software. Las pruebas de software se integran dentro de las diferentes fases del ciclo del software dentro de la Ingeniería de software. Así se ejecuta un programa y mediante técnicas experimentales se trata de descubrir que errores tiene. Para determinar el nivel de calidad se deben efectuar unas medidas o pruebas que permitan comprobar el grado de cumplimiento respecto de las especificaciones iniciales del sistema.

En general, los informáticos distinguen entre errores de programación y defectos de forma. En un defecto de forma, el programa no realiza lo que el usuario espera. Por el contrario, un error de programación puede describirse como un fallo en la semántica de un programa de ordenador. Éste podría presentarse, o no, como un defecto de forma si se llegan a dar ciertas condiciones de cálculo.

Una práctica común es que el proceso de pruebas de un programa sea realizado por un grupo independiente de "tester" (persona que realiza las pruebas) al finalizar su desarrollo y antes de sacarlo al mercado.

Existen varios tipos de pruebas para poder asegurar la calidad del software como:

- ✓ Pruebas unitarias
- ✓ Pruebas funcionales
- ✓ Pruebas de Integración
- ✓ Pruebas de validación
- ✓ Pruebas de sistema
- ✓ Caja blanca (sistemas)
- ✓ Caja negra (sistemas)
- ✓ Las pruebas existentes para asegurar la calidad de la interfaz de usuario son las pruebas de caja negra. (Pressman, 2001)

3.1.1 Pruebas de Caja Negra

En teoría de sistemas y física, se denomina caja negra a aquel elemento que es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno. En otras palabras, de una caja negra nos interesará su forma de interactuar con el medio, entendiendo qué es lo que hace, pero sin dar importancia a cómo lo hace. Por tanto, de una caja negra deben estar muy bien definidas sus entradas y salidas, es decir, su interfaz; en cambio, no se precisa definir ni conocer los detalles internos de su funcionamiento.

Esta es conocida también por sus sinónimos:

- ✓ pruebas de caja opaca.
- ✓ pruebas funcionales.

CAPÍTULO 3: Análisis de la Solución Propuesta

- ✓ pruebas de entrada/salida.
- ✓ pruebas inducidas por los datos

El Método de la Caja Negra se centra en los requisitos fundamentales del software y permite obtener entradas que prueben todos los requisitos funcionales del programa. Con este tipo de pruebas se intenta encontrar:

- ✓ Funciones incorrectas o ausentes.
- ✓ Errores de interfaz.
- ✓ Errores en estructuras de datos o en accesos a las bases de datos externas.
- ✓ Errores de rendimiento.
- ✓ Errores de inicialización y terminación.

Pressman presenta la partición equivalente como un método de prueba de caja negra que divide el campo de entrada de un programa en clases de datos de los que se pueden derivar casos de prueba. Un caso de prueba ideal descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico. La partición equivalente se dirige a la definición de casos de prueba que descubran clases de errores, reduciendo así el número total de casos de prueba que hay que desarrollar.

Una clase de equivalencia representa un conjunto de estados válidos o no válidos para condiciones de entrada. Típicamente, una condición de entrada es un valor numérico específico, un rango de valores, un conjunto de valores relacionados o una condición lógica.

En el diseño de casos de prueba para partición equivalente se procede en dos pasos :

- ✓ *Se identifican las clases de equivalencia.* Las clases de equivalencia son identificadas tomando cada condición de entrada (generalmente una oración o una frase en la especificación) y repartiéndola en dos o más grupos.

- ✓ *Se define los casos de prueba.* El segundo paso es el uso de las clases de equivalencia para identificar los casos de prueba. El proceso es como sigue: se asigna un número único a cada clase de equivalencia. Hasta que todas las clases de equivalencia válidas han sido cubiertas por los casos de prueba, se escribe un nuevo caso de prueba que cubra la clase de equivalencia válida. Y por último hasta que los casos de prueba hayan cubierto todas las clases de equivalencia inválidas, se escribe un caso de la prueba que cubra una, y solamente una, de las clases de equivalencia inválidas descubiertas.

3.2 Casos de Pruebas

En este epígrafe se comentara todo lo referente a los casos de pruebas que se le realizaron al Módulo Seguimiento del Proyecto CCV por Calisoft²⁰. En este se abordaran los casos de usos más críticos del sistema.

3.2.1 Caso de uso Actualizar Ejecución Física.

Descripción General

El caso de uso inicia cuando un ente ejecutor necesita cambiar el estado de ejecución de una de las actividades del plan operativo, las mismas pueden estar en los siguientes estados: Sin ejecutar, En ejecución, Cerrada, Terminada.

Condiciones de Ejecución:

El sistema debe estar instalado y ejecutado correctamente. El usuario debe estar autenticado con los permisos necesarios.

²⁰ Entidad en la UCI encargada de asegurar la calidad de los proyectos antes de estos ser desplegados.

CAPÍTULO 3: Análisis de la Solución Propuesta

Secciones a probar en el Caso de Uso:

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC 1: Cambiar estado a En ejecución	EC 1.1: El actor solicita la búsqueda de los proyectos	El sistema muestra los posibles estados en los que puede estar la actividad Sin ejecutar En ejecución El sistema modifica temporalmente el estado de la actividad de Sin ejecutar a En ejecución. El sistema cambia el estado del plan operativo a Actualización de plan operativo en espera de aceptación por ente. El caso de uso termina.	Procesos/ Actualizar Ejecución Física/ En ejecución
	EC 1.2: El actor cancela la operación.	El sistema no ejecuta ninguna acción, la actividad se queda en estado de Sin ejecutar. El caso de uso termina.	El actor debe cancelar.
SC 2: Cambiar estado a Cerrada	EC 2.1: Cambiar estado a Cerrada	El sistema muestra los posibles estados en los que puede estar la actividad En ejecución Cerrada Terminada El sistema modifica temporalmente el estado de la actividad de En Ejecución a Cerrada. El sistema muestra un mensaje de redistribución del monto de la actividad que no se ha consumido. El sistema cambia el estado del plan operativo a Actualización de plan operativo en espera de aceptación por ente. El caso de uso termina.	Procesos/ Actualizar Ejecución Física/ En ejecución/ Cerrada
	EC 2.2: El actor cancela la operación.	El sistema no ejecuta ninguna acción, la actividad se queda en estado de En ejecución. El caso de uso termina.	El actor debe cancelar.

CAPÍTULO 3: Análisis de la Solución Propuesta

SC 3: Cambiar estado a Terminada	EC 3.1: Cambiar estado a Terminada	El sistema muestra los posibles estados en los que puede estar la actividad En ejecución Cerrada Terminada El sistema modifica temporalmente el estado de la actividad de En Ejecución a Terminada. El sistema cambia el estado del plan operativo a Actualización de plan operativo en espera de aceptación por ente. El caso de uso termina.	Procesos/ Actualizar Ejecución Física/ En ejecución/ Terminada
	EC 3.2: El actor cancela la operación.	El sistema no ejecuta ninguna acción, la actividad se queda en estado de En ejecución. El caso de uso termina.	El actor debe cancelar.

1.1 SC 1: Cambiar estado a En Ejecución

Id del escenario	Escenario	Sin ejecutar	En ejecución	Respuesta del Sistema	Resultado de la Prueba
EC 1	El actor solicita la búsqueda de los proyectos	NA	NA	El sistema cambia el estado del plan operativo a Actualización de plan operativo en espera de aceptación por ente.	Se obtiene un resultado satisfactorio.
EC 2	El actor cancela la operación.	NA	NA	El sistema no ejecuta ninguna acción, la actividad se queda en estado de En ejecución. El caso de uso termina.	Se obtiene un resultado satisfactorio.

1.2 SC 2: Cambiar estado a Cerrada

Id del escenario	Escenario	En ejecución	Cerrada	Terminada	Respuesta del Sistema	Resultado de la Prueba
------------------	-----------	--------------	---------	-----------	-----------------------	------------------------

CAPÍTULO 3: Análisis de la Solución Propuesta

EC 1	Cambiar estado a Cerrada	I	V	I	El sistema muestra un mensaje de redistribución del monto de la actividad que no se ha consumido. El sistema cambia el estado del plan operativo a Actualización de plan operativo en espera de aceptación por ente.	Se obtiene un resultado satisfactorio.
EC 2	El actor cancela la operación.	NA	NA	NA	El sistema no ejecuta ninguna acción, la actividad se queda en estado de En ejecución. El caso de uso termina.	Se obtiene un resultado satisfactorio.

1.3 SC 3: Cambiar estado a Terminada

Id del escenario	Escenario	En ejecución	Cerrada	Terminada	Respuesta del Sistema	Resultado de la Prueba
EC 1	Cambiar estado a Terminada	I	I	V	El sistema modifica temporalmente el estado de la actividad de En Ejecución a Terminada. El sistema cambia el estado del plan operativo a Actualización de plan operativo en espera de aceptación por ente.	Se obtiene un resultado satisfactorio.
EC 2	El actor cancela la operación.	NA	NA	NA	El sistema no ejecuta ninguna acción, la actividad se queda en estado de En ejecución. El caso de uso termina.	Se obtiene un resultado satisfactorio.

CAPÍTULO 3: Análisis de la Solución Propuesta

3.2.2 Caso de uso Gestionar otros tipos de recursos.

Descripción General

El caso de uso inicia cuando un ente decide realizar cambios respecto a los otros tipos de recursos asociados a una actividad.

Condiciones de Ejecución:

El sistema debe estar instalado y ejecutado correctamente. El usuario debe estar autenticado con los permisos necesarios.

Secciones a probar en el Caso de Uso:

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC1: Agregar otros tipos de recursos	EC 1.1: El actor solicita agregar un nuevo tipo de recurso a una actividad.	El sistema solicita los siguientes datos para registrar el nuevo tipo de recurso a la actividad: Nombre del recurso Tipo de recurso Monto Descripción Destino Verifica los datos. Y registra la información temporalmente, cambia el estado del plan operativo a Plan operativo en espera de aceptación por ente. El caso de uso termina.	El actor introduce los datos para agregar el nuevo tipo de recurso a la actividad.
	El actor introduce los datos errados	El sistema muestra un mensaje de error en la entrada de datos. El caso de uso termina.	El actor introduce los datos errados

CAPÍTULO 3: Análisis de la Solución Propuesta

SC 2: Modificar otros tipos de recursos	EC 2.1: El actor solicita modificar los datos de un tipo de recurso seleccionado.	El sistema muestra los datos del tipo de recurso seleccionado. El sistema verifica los datos. El sistema registra la información temporalmente. El sistema cambia el estado del plan operativo Plan operativo en espera de aceptación por ente. El caso de uso termina.	El actor modifica los datos siguientes del tipo de recurso seleccionado: Monto Breve descripción Destino
	EC 2.2: El actor introduce mal los datos.	El sistema muestra un mensaje de error en la entrada de datos. El caso de uso termina.	El actor introduce mal los datos.
SC3: Eliminar otros tipos de recursos	EC 3.1: El actor solicita eliminar un tipo de recurso determinado.	El sistema elimina temporalmente el recurso de la actividad. El sistema cambia el estado del plan operativo a Plan operativo en espera de aceptación por ente. El caso de uso termina..	El actor solicita eliminar un tipo de recurso determinado. Y confirma la eliminación.
	EC 3.2: El actor selecciona la opción de cancelar	El sistema no ejecuta ninguna acción. El caso de uso termina.	El actor selecciona la opción de cancelar

2.1 SC 2: Modificar otros tipos de recursos

Id del escenario	Escenario	Monto	Breve descripción	Destino	Respuesta del Sistema	Resultado de la Prueba
EC 1	El actor solicita modificar los datos de un tipo de recurso seleccionado	V	V	V	El sistema registra la información temporalmente. El sistema cambia el estado del plan operativo Plan operativo en espera de aceptación por ente.	Se obtiene un resultado satisfactorio
		V	V	V		
		V	V	V		
EC 2	El actor introduce mal los datos.	V	V	I	El sistema muestra un mensaje de error en la entrada de datos.	Se obtiene un resultado satisfactorio
		I	V	I		
		I	I	I		

CAPÍTULO 3: Análisis de la Solución Propuesta

2.2 SC 3: Modificar otros tipos de recursos

Id del escenario	Escenario	Monto	Breve descripción	Destino	Respuesta del Sistema	Resultado de la Prueba
EC 1	El actor solicita eliminar un tipo de recurso determinado.	NA	NA	NA	El sistema elimina temporalmente el recurso de la actividad. El sistema cambia el estado del plan operativo a Plan operativo en espera de aceptación por ente.	Se obtiene un resultado satisfactorio
EC 2	El actor selecciona la opción de cancelar	NA	NA	NA	El sistema no ejecuta ninguna acción	Se obtiene un resultado satisfactorio

3.2.3 Caso de uso Modificar Contrato

Descripción General

El caso de uso inicia cuando uno de los entes ejecutores necesita modificar un contrato y se descargan documentos en caso de que el usuario lo desee.

Condiciones de Ejecución:

El sistema debe estar instalado y ejecutado correctamente. El usuario debe estar autenticado con los permisos necesarios. Existencia de al menos un contrato.

CAPÍTULO 3: Análisis de la Solución Propuesta

Secciones a probar en el Caso de Uso:

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC1: Flujo normal Modificar contrato	EC 1.1: Modificar contrato	Consiste en modificar un contrato.	El actor solicita la búsqueda, se muestra un listado de los contratos selecciona el que desee, el sistema muestra los datos del contrato y modifica el mismo.
	EC 1.2 Flujo alternativo Descargar documento	Consiste en descargar un documento en caso de que no desee hacerlo todo quedaría igual.	Dar clic en Descargar contrato.

3.1 SC 1: Sección #1 a revisa

Id del escenario	Escenario	Buscar	Contrato	Modificar contrato	Respuesta del Sistema	Resultado de la Prueba
EC 1.1	Modificar contrato	V	V	V	El sistema muestra un listado con los contratos a modificar.	Se modifican los contratos satisfactoriam ente.
		V	I	V		
		V	V	I		
EC 1.2	Flujo alterno Descargar documento	NA	NA	NA	El sistema continúa mostrando la pantalla de modificación del contrato.	El sistema continúa con la modificación satisfactoriam ente.
		V	I	V		
		V	V	I		

CAPÍTULO 3: Análisis de la Solución Propuesta

3.2.3.1 Caso de uso Firmar modificación de contrato

Descripción General

El caso de uso inicia cuando uno de los ministerios necesita firmar un contrato modificado.

Condiciones de Ejecución:

El sistema debe estar instalado y ejecutado correctamente. El usuario debe estar autenticado con los permisos necesarios. Existencia de al menos un contrato modificado.

Secciones a probar en el Caso de Uso:

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC1: Flujo normal Firmar modificación del contrato.	EC 1.1: Firmar del modificación contrato	Consiste en firmar modificación del contrato.	El actor solicita la búsqueda, se muestra un listado de los contratos selecciona el que desee firmar, el actor acepta firmar luego de ser visualizado los documentos que posee el contrato.

4.1 SC 1: Sección #1 a revisar

Id del escenario	Escenario	Contratos Modificados	Firmar	Respuesta del Sistema	Resultado de la Prueba
EC 1.1	Firmar modificación del contrato	V	V	El sistema muestra un listado con los contratos a modificar.	Se firman los contratos satisfactoriamente.
		V	/		

CAPÍTULO 3: Análisis de la Solución Propuesta

3.2.3.2 Caso de prueba Revisar modificación de contrato (EE)

Descripción General

El caso de uso inicia cuando uno de los entes ejecutores necesita revisar la modificación de un contrato. El mismo acepta o rechaza la modificación.

Condiciones de Ejecución:

El sistema debe estar instalado y ejecutado correctamente. El usuario debe estar autenticado con los permisos necesarios. Existencia de contratos en estado de: Contrato con adendum en espera de aceptación por ente.

Secciones a probar en el Caso de Uso:

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC1: Aceptar modificación.	EC 1.1: Aceptar opción de modificación	Consiste en que el actor acepta la modificación y se cambia el estado del contrato a Contrato con adendum aceptado por entes.	El actor solicita la búsqueda, se muestra un listado de los contratos selecciona el que desee revisar, revisa los documentos y adendum y si están aceptados y sin problemas acepta la modificación.
SC 2: Rechazar modificación.	EC 2.1: Rechazar la opción de modificación del contrato	Consiste en rechazar la modificación. Se cambia el estado del contrato a Contrato con adendum rechazado por ente.	El usuario escoge la opción rechazar mostrando mediante un mensaje que no se aprobó por uno de los entes.

CAPÍTULO 3: Análisis de la Solución Propuesta

5.1 SC 1: Sección #1

Id del escenario	Escenario	Contrato modificado	Aceptar contrato	Rechazar contrato	Respuesta del Sistema	Resultado de la Prueba
EC 2.1	Rechazar la opción de modificación del contrato	V	V	V	El sistema muestra un listado con los contratos aceptados y queda con modificaciones de contratos rechazadas.	Se rechazan los contratos satisfactoriamente.
		NA	NA	NA		

3.2.3.3 Caso de uso Revisar modificación de contrato (M)

Descripción General

El caso de uso inicia cuando uno de los entes ejecutores necesita revisar la modificación de un contrato. El mismo acepta o rechaza la modificación.

Condiciones de Ejecución:

El sistema debe estar instalado y ejecutado correctamente. El usuario debe estar autenticado con los permisos necesarios. Existencia de contratos en estado de: Contrato con adendum en espera de aceptación por ente.

CAPÍTULO 3: Análisis de la Solución Propuesta

Secciones a probar en el Caso de Uso:

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC1: Aceptar modificación.	EC 1.1: Aceptar opción de modificación	Consiste en que el actor acepta la modificación y se cambia el estado del contrato a Contrato con adendum aceptado por entes.	El actor solicita la búsqueda, se muestra un listado de los contratos selecciona el que desee revisar, revisa los documentos y adendum y si están aceptados y sin problemas acepta la modificación.

Ver anexos para las demás pruebas de Caja Negra.

3.3 Resultados Obtenidos

Como resultados obtenidos después de realizadas las pruebas de caja Blanca al sistema, se puede hablar de que:

- ✓ La aplicación mostro ser completamente funcional, dando solución a todos los requisitos planteados por los clientes
- ✓ Todos los problemas de tratamiento de errores que fueron descubiertos durante el desarrollo fueron solucionados, por lo que la aplicación es capaz de detectar cualquier tipo de error.
- ✓ Mostró una gran eficiencia en el tiempo de respuesta al cliente cuando este hace un pedido, siempre teniendo en cuenta la integración de la capa interfaz y presentación. También se debe de tener en cuenta que en esto influye mucho la condición de conexión.
- ✓ Mostró ser una aplicación de fácil navegación, pese a que esta fue pensada para usuarios medios.

- ✓ Debido a que el software posee una arquitectura por capas, la mantenibilidad del software es poco costosa, así como su mantenimiento debido a que se cumplieron con todos los estándares de codificación.
- ✓ El software puede ser utilizado con varios navegadores web, como Internet Explorer y Mozilla Firefox, y además en cualquier sistema operativo.

Todas las pruebas dieron resultados satisfactorios, de esta manera se demuestra que el sistema cumple con cada uno de los casos de uso que lo conforman.

Conclusiones

El empleo de estas pruebas permitió verificar la solidez de la implementación de la capa interfaz de usuario en el sistema, utilizadas posteriormente por la capa inmediata inferior o capa de Presentación para completar la total integración entre las estas capas que conforman la aplicación.

Al concluir este capítulo se cuenta con el código ejecutable, o sea los componentes diseñados e implementados. Además se realizaron las pruebas lo cual constituyó la certificación de que los componentes soportan las funcionalidades requeridas.

CONCLUSIONES GENERALES

Se puede afirmar que con los resultados obtenidos en este trabajo de diploma se le da cumplimiento a los objetivos específicos propuestos, pues se logró implementar una capa de interfaz de usuario flexible y segura para la realización del Módulo Seguimiento del Proyecto CCV y a su vez logrando satisfacer todas las necesidades planteadas por el cliente de una forma correcta.

Con un estudio realizado sobre el Módulo Seguimiento del Proyecto CCV, se logró comprender lo que el cliente necesitaba, para así poder lograr un correcto entendimiento de los flujos que se deben realizar. Así se pudo diseñar e implementar las interfaces de usuario con un conjunto de pautas y estándares que brindaron respuestas a las solicitudes del cliente, logrando una interfaz sencilla, agradable y centrada en el usuario final. Se definió gracias a una arquitectura de información con una correcta estructuración de los contenidos. Se lograron explotar nuevas herramientas en el desarrollo de interfaces web, como fue la utilizada en el desarrollo de esta.

Finalmente se le realizaron unas pruebas exhaustivas a toda la interfaz implementada, siempre mostrando este una respuesta positiva, lo que demuestra que la aplicación cumple satisfactoriamente con los requisitos del módulo Seguimiento. De esta forma se logró dar cumplimiento a los objetivos planteados.

RECOMENDACIONES

Se recomienda el uso de esta aplicación en gestiones gubernamentales de este tipo con otros países debido a su gran seguridad y eficiencia, siempre y cuando se le realicen algunas especificaciones propias de los mismos. Así como la extender la experiencia adquirida en el desarrollo de interfaces web en la tecnología EXT2.0 a otros proyectos productivos en la universidad.

BIBLIOGRAFÍA

- Alonso, José Manuel. 2005.** *Estándares Web del W3C.*
- Autores, Colectivo de. 1998.** *Document Object Model (DOM) Level 1 Specification.* 1998.
- 2005.** *PHP y MySQL Tecnologías para el desarrollo de aplicaciones web.*
- Bonsiepe, Gui. 1995.** *Del objeto a la interfase.*
- BOOCH, Grady,, RUMBAUGH, James y JACOBSON, Ivar. 2000.** “*El lenguaje unificado de modelado*”.
- Eguíluz Pérez, Javier. 2005.** *Introducción a AJAX.* [aut. libro] Jesse James Garrett. *Ajax: A New Approach to Web Applications.* 2005.
- 2007.** *Introducción a CSS.*
- 2008.** *Introducción a JavaScript.*
- Fernández Escribano, Gerardo. 2002.** *Introducción a Extreme Programming.* [aut. libro] Beck Kent. *Extreme Programming.*
- Gates, Bill. 2000.** *Professional Developers Conference.*
- González, Óscar. 2005.** *Xml (edición Revisada Y Ampliada 2005).*
- Hansson, David Heinemeier. 2005.** *Agile Web Development with Rails.*
- Hassan, Yusef, Martín Fernández, Francisco J. y Iazza, Ghzala. 2004.** *Diseño Web Centrado en el Usuario: Usabilidad y Arquitectura de la Información.*
- Hassan, Yusef, Martín Fernández, Francisco y Iazza, Ghzala. 2000.** *Diseño Web Centrado en el Usuario: Usabilidad y Arquitectura de la Información.*
- JACOBSON, Ivar, BOOCH, Grady y RUMBAUGH, James. 2000.** “*El Proceso Unificado de Desarrollo de Software*”.
- Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000.** *El Proceso Unificado de desarrollo del Software.*
- Lores, J. 2001.** *La interacción persona-ordenador.*
- Marshall, James. 1997.** *HTTP Made Really Easy.*
- McGOVERN, GERRY. 2005.** *Graphic design plays a minor role on the Web.*
- Moreno, Luciano. 2003.** *desarrolloweb.com. desarrolloweb.com.* [En línea]
<http://www.desarrolloweb.com/articulos/2171.php>.
- Pressman, Roger S. INGENIERIA DEL SOFTWARE UN ENFOQUE PRACTICO .** España : Mc. Graw Hill .
- 2001.** *Ingeniería del software: un enfoque práctico.*
- Rational, Microsoft y. 1998.** *A White Paper on the Benefits of Integrating Microsoft Solutions Framework and The Rational Process.*
- Raymond, Eric Steven. 2003.** *The Art of Unix Programming.*
- Royo, Javier. 2004.** *Diseño Digital.* s.l. : Ediciones Paidós Ibérica,
- Welling, Thomson, Luke & y Laura. 2005.**

ANEXOS

Anexo 1 casos de pruebas de Caja Negra.

1.1 Caso de uso Gestionar envío de cronograma

Descripción General

El caso de uso inicia cuando uno de los entes ejecutores necesita gestionar el envío de la propuesta de la re-planificación del cronograma de ejecución financiera a ambos ministerios, la gestión está enmarcada en el envío, aceptación o rechazo de la propuesta.

Condiciones de Ejecución:

El sistema debe estar instalado y ejecutado correctamente. El usuario debe estar autenticado con los permisos necesarios. Tienen que existir cronogramas de ejecución financiera en los siguientes estados: Cronograma de ejecución financiera aceptado por entes y/o Cronograma de ejecución financiera en espera de envío por ente.

Secciones a probar en el Caso de Uso:

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC 1: Búsqueda de los proyectos.	EC 1.1: Búsqueda de los proyectos	Consiste en la búsqueda de los proyectos dado: Nombre del proyecto, Ministerio contraparte, Ente contraparte, Marco de aprobación.	Actualizar ejecución/ buscar.
SC 2: Enviar re-planificación del cronograma de ejecución financiera	EC 2.1: Enviar re-planificación del cronograma de ejecución financiera	Consiste en enviar re-planificación del cronograma.	Actualizar ejecución/Ver detalles/enviar.
SC 3: Aceptar envío de re-planificación del cronograma de ejecución financiera	EC 3.1: Aceptar envío de re-planificación del cronograma de ejecución financiera	Consiste en aceptar la re-planificación del cronograma de ejecución financiera.	Actualizar ejecución/Ver detalles/aceptar .

SC 4: Rechazar envío de re-planificación del cronograma de	EC 4.1: Rechazar envío de re-planificación del cronograma de	Consiste en rechazar el envío.	Actualizar ejecución/Ver detalles/rechazar.
--	--	--------------------------------	---

SC 1: Búsqueda de los proyectos

Id del escenario	Escenario	Nombre del proyecto	Ministerio o contraparte	Ente contraparte	Marco de aprobación	Respuesta del Sistema	Resultado de la Prueba
EC 1.1	Búsqueda de los proyectos	I	V	V	V	El sistema muestra un listado de los proyectos que cumplen con los criterios de búsqueda. El sistema ejecuta alguna de las siguientes acciones: Envío de re-planificación a ministerios. Ejecución financiera. Aceptar envío de re-planificación a ministerios. Rechazar envío de re-planificación a	Se realiza la búsqueda satisfactoriamente.
		V	I	V	I		
		V	V	I	V		

SC 2: Enviar re-planificación del cronograma de ejecución financiera

Id del escenario	Escenario	Re planificación cronograma	Re planificación cronograma aceptado EE	Cronograma de ejecución financiera en espera de envío	Respuesta del Sistema	Resultado de la Prueba
EC 2.1	Enviar re-planificación del cronograma de ejecución financiera	NA	NA	NA	El sistema muestra una interfaz para enviar temporalmente a ministerios la re-planificación del cronograma de ejecución financiera aceptada por los entes ejecutores. Hasta que el EE contraparte no apruebe este envío no llegara a los ministerios. El sistema cambia el estado del cronograma a	Se hace el envío satisfactoriamente.

SC 3: Aceptar envío de re-planificación del cronograma de ejecución financiera

Id del escenario	Escenario	Re planificación cronograma	Re planificación cronograma aceptado EE	Cronograma de ejecución financiera Aceptado	Respuesta del Sistema	Resultado de la Prueba
EC 3.1	Aceptar envío de re-planificación del cronograma de ejecución	NA	NA	NA	El sistema muestra una interfaz para aceptar el envío a los ministerios de la propuesta de re-planificación del cronograma de	Se acepta el envío satisfactoriamente.

SC 4: Rechazar envío de re-planificación del cronograma de ejecución financiera

Id del escenario	Escenario	Re planificación cronograma	Re planificación cronograma aceptado EE	Cronograma de ejecución financiera rechazado	Respuesta del Sistema	Resultado de la Prueba
EC 4.1	Rechazar envío de re-planificación del cronograma de ejecución financiera	NA	NA	NA	El sistema muestra una interfaz para rechazar el envío a los ministerios de la propuesta de re-planificación del cronograma de ejecución financiera. El sistema envía la nota de rechazo al FF	Se rechaza el envío satisfactoriamente.

1.2 Caso de uso Actualizar Ejecución Financiera

Descripción General

El caso de uso inicia cuando un ente ejecutor necesita cambiar el estado de ejecución de una de las actividades del plan operativo, las mismas pueden estar en los siguientes estados: Sin ejecutar, En ejecución, Cerrada, Terminada.

Condiciones de Ejecución:

El sistema debe estar instalado y ejecutado correctamente.

El usuario debe estar autenticado con los permisos necesarios.

Secciones a probar en el Caso de Uso:

Nombre de la sección	Escenarios de la sección	Descripción de la	Flujo Central
----------------------	--------------------------	-------------------	---------------

		funcionalidad	
SC 1: Cambiar estado a En ejecución	EC 1.1: El actor solicita la búsqueda de los proyectos	<p>El sistema muestra los posibles estados en los que puede estar la actividad</p> <p>Sin ejecutar</p> <p>En ejecución</p> <p>El sistema modifica temporalmente el estado de la actividad de Sin ejecutar a En ejecución.</p> <p>El sistema cambia el estado del plan operativo a Actualización de plan operativo en espera de</p>	<p>Procesos/Actualizar</p> <p>Ejecución</p> <p>Física/ En ejecución</p>
	EC 1.2: El actor cancela la operación.	<p>El sistema no ejecuta ninguna acción, la actividad se queda en estado de Sin ejecutar. El caso de uso termina.</p>	<p>El actor debe cancelar.</p>

<p>SC 2: Cambiar estado a Cerrada</p>	<p>EC 2.1: Cambiar estado a Cerrada</p>	<p>El sistema muestra los posibles estados en los que puede estar la actividad En ejecución Cerrada Terminada El sistema modifica temporalmente el estado de la actividad de En Ejecución a Cerrada. El sistema muestra un mensaje de redistribución del monto de la actividad que no se ha consumido. El sistema cambia el estado del plan operativo a Actualización de plan operativo en espera de aceptación por ente. El caso de uso termina.</p>	<p>Procesos/Actualizar Ejecución Física/ En ejecución/ Cerrada</p>
	<p>EC 2.2: El actor cancela la operación.</p>	<p>El sistema no ejecuta ninguna acción, la actividad se queda en estado de En ejecución. El caso de uso termina.</p>	<p>El actor debe cancelar.</p>
<p>SC 3: Cambiar estado a Terminada</p>	<p>EC 3.1: Cambiar estado a Terminada</p>	<p>El sistema muestra los posibles estados en los que puede estar la actividad En ejecución Cerrada</p>	<p>Procesos/Actualizar Ejecución Física/ En ejecución/</p>

	EC 3.2: El actor cancela la operación.	El sistema no ejecuta ninguna acción, la actividad se queda en estado de En ejecución. El caso de uso termina.	El actor debe cancelar.
--	--	--	-------------------------

SC 1: Cambiar estado a En Ejecución

Id del escenario	Escenario	Sin ejecutar	En ejecución	Respuesta del Sistema	Resultado de la Prueba
EC 1	El actor solicita la búsqueda de los proyectos	NA	NA	El sistema cambia el estado del plan operativo a Actualización de plan operativo en espera de aceptación por ente.	Se obtiene un resultado satisfactorio.
EC 2	El actor cancela la operación.	NA	NA	El sistema no ejecuta ninguna acción, la actividad se queda en estado de En ejecución. El caso de uso termina.	Se obtiene un resultado satisfactorio.

SC 2: Cambiar estado a Cerrada

Id del escenario	Escenario	En ejecución	Cerrada	Terminada	Respuesta del Sistema	Resultado de la Prueba
------------------	-----------	--------------	---------	-----------	-----------------------	------------------------

EC 1	Cambiar estado a Cerrada	I	V	I	El sistema muestra un mensaje de redistribución del monto de la actividad que no se ha consumido. El sistema cambia el estado del plan operativo a Actualización de plan operativo en espera de aceptación por ente.	Se obtiene un resultado satisfactorio.
EC 2	El actor cancela la operación.	NA	NA	NA	El sistema no ejecuta ninguna acción, la actividad se queda en estado de En ejecución. El caso de uso termina.	Se obtiene un resultado satisfactorio.

SC 3: < Cambiar estado a Terminada >

Id del escenario	Escenario	En ejecución	Cerrada	Terminada	Respuesta del Sistema	Resultado de la Prueba
------------------	-----------	--------------	---------	-----------	-----------------------	------------------------

EC 1	Cambiar estado a Terminada	I	I	V	El sistema modifica temporalmente el estado de la actividad de En Ejecución a Terminada. El sistema cambia el estado del plan operativo a Actualización de plan operativo en	Se obtiene un resultado satisfactorio.
EC 2	El actor cancela la operación.	NA	NA	NA	El sistema no ejecuta ninguna acción, la actividad se queda en estado de En ejecución. El caso de uso termina.	Se obtiene un resultado satisfactorio.

1.3 Caso de uso caso de uso Gestionar Inspección

Descripción General

La secretaría técnica tiene la posibilidad de crear, mostrar, modificar o eliminar una inspección de la ejecución física de un proyecto.

Condiciones de Ejecución:

El sistema debe estar instalado y ejecutado correctamente. El usuario debe estar autenticado con los permisos necesarios. Existencia de inspecciones de la EF de un proyecto.

Secciones a probar en el Caso de Uso:

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC 1: " Crear inspección"	EC 1.1: Nueva inspección.	El actor solicita crear una nueva inspección.	Seguimiento/ Documentos/ Inspecciones/ Nueva Inspección.
	EC 1.2: Registrar datos.	Registrar datos de Ministerio, Entes, Marco de aprobación, Nombre del proyecto, Fecha, Nombre del supervisor, Observaciones de la inspección.	Seguimiento/ Documentos/ Inspecciones/ Nueva Inspección.
	EC 1.3: Editar los datos.	El actor edita los datos de la inspección y acepta la operación.	Seguimiento/ Documentos/ Inspecciones/ Nueva Inspección.
	EC 1.4 Verificar y registrar.	Verificar los datos y registrar la información.	Seguimiento/ Documentos/ Inspecciones/ Nueva Inspección/ Crear.

SC 2: "Mostrar inspección"	EC 2.1: Mostrar una inspección.	El actor solicita mostrar una inspección.	Seguimiento/ Documentos/ Inspecciones/ Seleccionar una inspección/ Ver detalles.
	EC 2.2: Datos de una inspección.	Se muestran los datos de una inspección: Ente ejecutor, Ministerio, Marco de aprobación del proyecto y Nombre del proyecto. Fecha de emisión de la inspección Nombre del supervisor Observaciones de la inspección Número de inspección	Seguimiento/ Documentos/ Inspecciones/ Seleccionar una inspección/ Ver detalles.
SC 3: "Modificar inspección"	EC 3.1: Seleccionar la inspección.	El actor selecciona la inspección que desea modificar.	Seguimiento/ Documentos/ Inspecciones/ Seleccionar una inspección/ Modificar.
	EC 3.2: Detalles de la inspección.	Se muestran los detalles de la inspección y permite hacer modificaciones en todos los datos de la inspección.	Seguimiento/ Documentos/ Inspecciones/ Seleccionar una inspección/ Ver detalles.

EC 3.3: Modificar y aceptar.	El actor modifica los datos de la inspección y acepta la operación.	Seguimiento/ Documentos/ Inspecciones/ Seleccionar una inspección/ Modificar/Aceptar.
EC 3.4: Verificar y modificar.	Se verifican los datos de la inspección y se modifica la información en la base de datos.	Seguimiento/ Documentos/ Inspecciones/ Seleccionar una inspección/ Modificar/Aceptar.
EC 4.1: Eliminar una inspección.	El actor solicita eliminar una inspección de un determinado proyecto.	Seguimiento/ Documentos/ Inspecciones/ Seleccionar una inspección/ Eliminar.
EC 4.2: Se muestra un mensaje.	Se muestra un mensaje de confirmación de eliminación de la inspección.	Seguimiento/ Documentos/ Inspecciones/ Seleccionar una inspección/ Eliminar.

	EC 4.3: Aceptar la operación	El actor acepta la operación de eliminación.	Seguimiento/ Documentos/ Inspecciones/ Seleccionar una inspección/ Eliminar/Si.
	EC 4.4: Eliminar	El sistema elimina la inspección.	Seguimiento/ Documentos/ Inspecciones/ Seleccionar una inspección/ Eliminar/Si.

1.1 SC 1: “Crear inspección”

Id del escenario	Escenario	Ministerio	Entes	Marco de aprobación	Nombre del proyecto	Fecha	Nombre del supervisor	Observaciones de la inspección	Respuesta del Sistema	Resultado de la Prueba
EC 1.1	Nueva inspección.	NA	NA	NA	NA	NA	NA	NA	El sistema da la opción de crear una nueva inspección	La prueba resultó satisfactoria.

EC 1.2	Registrar datos.	NA	NA	NA	NA	NA	NA	NA	El sistema muestra una interfaz para registrar.	La prueba resultó satisfactoria
EC 1.3	Editar los datos.	I	V	V	V	V	V	V	El sistema muestra un mensaje informativo.	La prueba resultó satisfactoria.
EC 1.4	Verificar y registrar.	NA	NA	NA	NA	NA	NA	NA	Se escribe el resultado que se espera al realizar la prueba.	La prueba resultó satisfactoria.

1.2 SC 2: “Mostrar inspección”

Id del escenario	Escenario	Ente ejecutor	Ministerio	Marco de aprobación del proyecto	Nombre del proyecto	Fecha de emisión de la inspección	Nombre del supervisor	Observaciones de la inspección	Número de inspección	Respuesta del Sistema	Resultado de la Prueba
------------------	-----------	---------------	------------	----------------------------------	---------------------	-----------------------------------	-----------------------	--------------------------------	----------------------	-----------------------	------------------------

EC 2.1	Mostrar una inspección.	NA	NA	NA	NA	NA		NA	NA	El sistema muestra las inspecciones	La prueba resultó satisfactoria.
EC 2.2	Datos de una inspección.	NA	NA	NA	NA	NA		NA	NA	El sistema muestra una interfaz con los datos de la inspección.	La prueba resultó satisfactoria.

1.3 SC 3: “Modificar inspección”

Id del escenario	Escenario	Ministerio	Entes	Marco de aprobación	Nombre del proyecto	Fecha	Nombre del supervisor	Observaciones de la inspección	Respuesta del Sistema	Resultado de la Prueba
EC 3.1	Seleccionar la inspección.	NA	NA	NA	NA	NA	NA	NA	El sistema muestra la pantalla de listado de proyectos.	La prueba resultó satisfactoria.

EC 3.2	Detalles de la inspección.	NA	NA	NA	NA	NA	NA	NA	El sistema permite hacer modificaciones en todos los datos de la inspección.	La prueba resultó satisfactoria.
EC 3.3	Modificar y aceptar.	I	V	V	V	V	V	V	El sistema te permita aceptar los cambios.	La prueba resultó satisfactoria.
EC 3.4	Verificar y modificar.	NA	NA	NA	NA	NA	NA	NA	El sistema modifica la base de datos.	La prueba resultó satisfactoria.

1.4 SC 4: “Eliminar inspección”

Id del escenario	Escenario	Ministerio	Entes	Marco de aprobación	Nombre del proyecto	Fecha	Nombre del supervisor	Observaciones de la inspección	Respuesta del Sistema	Resultado de la Prueba
------------------	-----------	------------	-------	---------------------	---------------------	-------	-----------------------	--------------------------------	-----------------------	------------------------

EC 4.1	Eliminar una inspección.	NA	NA	NA	NA	NA	NA	NA	El sistema da la opción de eliminar una inspección.	La prueba resultó satisfactoria.
EC 4.2	Se muestra un mensaje.	NA	NA	NA	NA	NA	NA	NA	El sistema muestra mensaje para verificar que este seguro de querer eliminar.	La prueba resultó satisfactoria.
EC 4.3	Aceptar la operación	NA	NA	NA	NA	NA	NA	NA	El sistema acepta que sea eliminada la inspección.	La prueba resultó satisfactoria.

EC 4.4	Eliminar	NA	NA	NA	NA	NA	NA	NA	El sistema elimina la inspección.	La prueba resultó satisfactoria.
--------	----------	----	----	----	----	----	----	----	-----------------------------------	----------------------------------

GLOSARIO DE TÉRMINOS

1-) **GUI:** Es el artefacto tecnológico de un sistema interactivo que posibilita, a través del uso y la representación del lenguaje visual, una interacción amigable con un sistema informático.

2-) **Marketplaces:** Estos espacios globales para los negocios son también conocidos como e-markets, o B2B Trade Communities. Son sitios en la web que brindan la posibilidad de relacionar un gran número de empresas, clientes y proveedores, en un sólo punto de encuentro, planteándose así como ecosistemas globales y completos, que enlazan automáticamente toda la cadena de operaciones, desde los proveedores hasta el consumidor final.

3-) **E-procurement:** modalidad de comercio electrónico, las empresas se unen para comprar a través de internet algunos servicios y productos no estratégicos que necesitan para su actividad.

4-) **XHTML:** Es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas web. En su versión 1.0, XHTML es solamente la versión XML de HTML, por lo que tiene, básicamente, las mismas funcionalidades, pero cumple las especificaciones, más estrictas, de XML. Su objetivo es avanzar en el proyecto del World Wide Web Consortium de lograr una web semántica, donde la información, y la forma de presentarla estén claramente separadas.

5-) **W3C:** El Consorcio World Wide Web es un consorcio internacional donde las organizaciones miembro, personal a tiempo completo y el público en general, trabajan conjuntamente para desarrollar estándares Web. La misión del W3C es: Guiar la Web hacia su máximo potencial a través del desarrollo de protocolos y pautas que aseguren el crecimiento futuro de la Web.

6-) **WML:** es un lenguaje cuyo origen es el XML. Este lenguaje se utiliza para construir las páginas que aparecen en las pantallas de los teléfonos móviles y los asistentes personales digitales (PDA).