

3. Fase Inicial

3.1 Modelado de negocios

En este apartado se definirá de forma clara el problema que se quiere resolver con la finalidad de:

- Describir en su totalidad los requerimientos de construcción.
- No omitir en la construcción ningún punto clave de resolución.

3.1.1 Definición del problema

3.1.1.1 Introducción

Hoy en día el desarrollo de sistemas web de alta calidad se basa en el uso de herramientas que:

- Faciliten el diseño y la construcción del mismo.
- Disminuyan notablemente el tiempo de desarrollo de dichos proyectos.
- Optimicen los recursos de los lenguajes sobre los cuales están programados.
- Provean soluciones sobre las tareas más comunes en el desarrollo de los sistemas web.
- Ayuden a la organización de código en base al patrón Modelo Vista Controlador.
- Permitan y faciliten la introducción de contenido interactivo.

Una de esas herramientas son los llamados **Frameworks** o estructuras de soporte. Los frameworks se encargan de automatizar y/o estandarizar los procesos más utilizados en la elaboración de un sistema con el fin de que los desarrolladores del mismo centren su atención en la lógica del negocio.

Los frameworks se han desarrollado a partir de las necesidades de un grupo específico de desarrolladores y como es de esperarse, estas necesidades son tan diversas que dan lugar a frameworks que resuelven problemas específicos tales como la interfaz de usuario o el control de vistas.

3.1.1.2 Tareas básicas de los sistemas web

Si se le pidiera a un conjunto de desarrolladores que construyeran una lista de las tareas más comunes o básicas de un sistema web, dichas listas serían totalmente diferentes entre ellas ya que cada desarrollador ha enfrentado necesidades de negocio muy diversas, por este hecho, las tareas que se tomarán en cuenta serán únicamente aquellas que fueron identificadas por los desarrolladores web del Laboratorio de Multimedia e Internet.

Durante los análisis de requerimientos de los sistemas web SIAEFI y SSIMAD se identificaron tareas comunes, este tipo de tareas se definirán entonces, como básicas para un sistema web y son:

A) Envío y recepción de mensajes de correo

Muchas aplicaciones de internet ofrecen una bandeja de mensajes y un buzón personalizado, en la mayoría de los casos el propósito de esta utilidad es mantener a los usuarios de un sistema comunicados sin la necesidad de hacer uso de otras plataformas web o de escritorio que solamente alargarían el tiempo necesario para consultar un mensaje.

Se debe considerar entonces contar con servidor de correo electrónico. El objetivo es enviar correos en texto plano o en formato HTML, además se

debe soportar el envío de uno o varios archivos adjuntos y que el funcionamiento se pueda habilitar y deshabilitar fácilmente desde un archivo de configuración en las aplicaciones.

B) Manejo de sesiones

Un requerimiento muy común en las aplicaciones web es proveer de una sesión, esta sesión permite que el sistema se comporte de una manera específica teniendo de referencia al usuario que lo utiliza (identificación y autenticación).

Regularmente el inicio de una sesión personalizada se hace a través de un identificador conocido como nombre usuario y de una palabra secreta llamada contraseña, es necesario proveer una pantalla genérica en la que se pida el nombre de usuario y la clave secreta, esta pantalla debe ser configurable y fácil de habilitar, por otra parte se deberá proveer de los elementos para hacer posible la personalización del sistema.

Otro punto importante de controlar es cuando el usuario intenta tener acceso a páginas no permitidas para él (autorización).

C) Manejo de seguridad

C.1 Permisos y protección de información

La manera más eficiente de proteger información es controlar quien puede tener acceso a ella, dicho control es posible a través del manejo de sesiones, cuando se inicia una sesión el sistema identifica quién es el usuario que está

solicitando datos y verifica en sus políticas de seguridad para saber si es posible proveer al usuario lo que solicita.

De acuerdo a lo anterior podemos concluir que la seguridad y manejo de sesiones están altamente ligados ya que una parte primordial de la personalización recae en los permisos que un usuario puede tener en un sistema.

Las políticas de seguridad por otra parte son una serie de reglas que los sistemas siguen para saber qué partes del sistema están disponibles para cada usuario.

Se necesita brindar una manera sencilla de crear políticas de seguridad sobre las cuales se basen las sesiones y las posibles vistas de cualquier sistema, por otra parte es necesario contemplar que se debe proveer a los desarrolladores de una forma de ampliar la plataforma de seguridad.

C.2 Protección de Contraseñas

Como ya se mencionó anteriormente se denomina *contraseña* a la palabra clave que se pide a un usuario para iniciar una sesión dentro de un sistema web, la contraseña está almacenada en el servidor y cada vez que es solicitada se verifica que la cadena de caracteres que ingreso el usuario sea exactamente igual a la que tiene registrada el servidor, algunas veces el hecho de que las contraseñas estén almacenadas en el servidor puede ser riesgoso ya que en casos extraordinarios puede haber intrusos que accedan a esta información confidencial afectando directamente a la organización a la que pertenece el sitio web.

Si se desea construir un sistema genérico debe tener un buen soporte para proteger las contraseñas de los usuarios, se debe utilizar un método efectivo pero fácil del usar.

C.3 Limpieza y prevención de SQL injection

Conocido en español como inyección de código SQL, este tipo de ataque puede afectar a los sistemas web que hacen uso de las bases de datos informáticas, una inyección de SQL consiste en mandar peticiones al servidor de bases de datos de forma “oculta”, para que muestre o inserte información no deseada en el sitio web.

Para crear una arquitectura segura para el desarrollo de sistemas web, se debe tener presente la posibilidad de ataques de inyección de código SQL, por lo que se debe analizar el problema y ofrecer una solución lo más confiable posible.

C.4 Historiales (Logs)

Una vez que un sistema está listo y en funcionamiento es necesario monitorearlo ya que algunos usuarios intentarán realizar acciones indebidas sobre él. Los historiales permiten tener una relación de ¿Quién hace qué?, si se hace un buen sistema de historiales es posible detectar la **dirección ip** desde la que se realizan acciones sobre el sistema, este tipo de control es muy conveniente para detectar a las personas que desean dañar o conseguir información no autorizada.

Ofrecer una manera sencilla de crear historiales ahorra mucho tiempo de programación que puede ser utilizado en otras actividades de importancia empresarial, sería muy cómodo para los programadores habilitar o deshabilitar los historiales dependiendo de las especificaciones del sitio web que se desea construir.

C.5 Caducidad de páginas

Los botones del navegador “adelante” y “atrás” pueden ser un problema de seguridad alto cuando no se tiene el suficiente cuidado en el manejo de cierre de sesión y más aun cuando los usuarios olvidan cerrar su sesión y dejan el navegador abierto. Para corregir este tipo de errores de seguridad se recurre a lo que se conoce como caducidad de página, esto es que una vez que la página deja de mostrarse en el navegador dicha página se destruye o caduca y no se puede mostrar de nuevo sin una sesión válida. La caducidad de páginas debe ser contemplado en el desarrollo de la arquitectura base y de ser posible ofrecer una forma de habilitarlo o deshabilitarlo.

D) Conexión a bases de datos

Las aplicaciones web necesitan una forma eficiente de búsqueda y despliegue de información. Actualmente la forma más efectiva es a través de las bases de datos. En las aplicaciones web de java, por ejemplo, las bases de datos tienen un lugar clave en su funcionamiento ya que la mayoría de las páginas modernas crean su contenido en forma dinámica: básicamente la lógica de negocio recurre a las bases de datos para extraer, modificar e insertar información.

Otra de las ventajas clave de las bases de datos es que ofrecen la capacidad de modificar la información sin necesidad de modificar las vistas. La mayoría de los lenguajes de alto nivel ofrecen métodos para la comunicación a las bases de datos.

Aunque existen ya muchos frameworks que resuelven el acceso a las bases de datos, a veces es muy complicado adaptarlos a sistemas web que no puedan sortear todas las restricciones que dichos frameworks manejan, si se logra integrar una sola funcionalidad que opere sobre el lenguaje que se utilizará para desarrollar el sistema web, se facilitará el proceso de construcción de los sistemas.

E) Manejo de archivos y directorios

Existen hoy en día un conjunto de formatos de archivo que son muy utilizados en internet, entre ellos podemos mencionar:

- **PDF.-** (*Portable document format*) creado por Adobe Systems para la compresión de textos con letras e imágenes.
- **SWF.-** Archivo de formato de video o animaciones especialmente diseñados para internet desarrollado por Adobe System para su entorno de desarrollo Flash.
- **XLS y XLSX.-** Archivo de Microsoft Excel para descargar información asentada en tablas de datos.
- **ZIP.-** Formato para la compresión de archivos, se agrupan archivos dentro de un contenedor, ocupando menor espacio en disco que los archivos en formato nativo.

Una arquitectura base debe por lo menos ofrecer soporte para crear archivos

PDF o XLS a partir de la información mostrada en el navegador, debe ser capaz de crear o de descomprimir contenedores de archivos en formato ZIP y además no debe restringir ni afectar las etiquetas HTML diseñadas para poder desplegar cualquier archivo SWF.

Otro de los puntos medulares para un buen funcionamiento de un sistema web es que la información del mismo esté bien organizada. Esto se logra a través del uso de directorios o carpetas que permiten hacer una búsqueda más eficiente de la información y almacenarla de manera correcta. Muchas veces la información de las aplicaciones se genera de forma dinámica por lo que es necesario que se puedan crear carpetas este en mismo modo. Otras operaciones necesarias para la modificación y creación de contenido dinámico son la escritura, la modificación y la eliminación de archivos.

F) Servicio de Chat

Existen muchas plataformas que ofrecen servicios de Chat, pero la mayoría están desarrollados para **PHP**, es necesario crear un chat capaz de ser configurable y que este se adapte a cualquier aplicación desarrollada sobre la plataforma Struts y YUI, otra ventaja de crear un módulo de chat es que estaría acoplado con todo el desarrollo posterior del proyecto incluyendo la base de datos.

G) Interfaz entre YUI y Struts

YUI ofrece una amplia gama de Componentes que se utilizan para dar interactividad o bien para facilitar creación de las vistas web, sin embargo el paso de los elementos de java a YUI requiere trabajo extra que muchas veces no se hace

estandarizado y se queda al libre albedrío el desarrollador que lo realiza. Este hecho ocasiona que todo el sistema contenga código diferente para solucionar el mismo problema: Pasar los elementos Java a Java Script, en una arquitectura robusta se debe contemplar dicha interacción y ofrecer un método que haga lo más transparente posible la transferencia de elementos inclusive si ésta se realiza por llamadas Ajax.

I) Interfaz entre Flash y Struts

Aun hoy en día existen pocas herramientas que ofrecen una interacción efectiva con Flash y los JSP's, estas herramientas son muchas veces escasas, difíciles de usar o bien el costo de sus licencias es alto. Es necesario proveer un método para la interacción de la arquitectura JSP inclusive utilizando Flash, ya que dicho Software sigue siendo ampliamente utilizado para generar vistas interactivas.

Existen muchas formas de realizar la interacción, se debe realizar una amplia investigación acerca de lo que es más factible utilizar para dicha interacción: web services, onLoadVars podrían ser alguna alternativa para el manejo de la interactividad Web para mejorar la comunicación entre Java y ActionScript 3.0.

3.2 Identificar y valorar los riesgos

A continuación se muestra una tabla con la descripción de los riesgos, su valoración y las estrategias de resolución disponibles con el fin de crear un buen documento de visión en el cual se expresan los requerimientos del proyecto.

Probabilidad.- La probabilidad de que un riesgo ocurra (expresado en porcentaje).

Exposición.- (magnitud) Representa que tan susceptible está el proyecto ante el riesgo, se calcula multiplicando el impacto por la probabilidad de aparición.

Impacto.- Establece categorías de acuerdo a la magnitud del impacto del riesgo, lo que implica desviaciones en cuanto a planificación, esfuerzos o costes.

Estrategia de Elución (EE).- Reorganización del proyecto, reducir el ámbito del sistema, eliminar requisitos no fundamentales.

Estrategia de Mitigación (EM).- Son planes para reducir la probabilidad de riesgo ó para reducir su impacto en el proyecto.

Estrategia de Contingencia (EC): Es necesario cuando las estrategias de elusión y mitigación han fallado, entonces el riesgo se debe afrontar directamente. El indicador de riesgos señala cuándo el riesgo se ha convertido en una realidad ó se debe reconocer el evento de pérdidas

Tabla 3.2.1 Descripción de parámetros de impacto.

| Risk ID | Fecha de identificación | Título | Descripción | Impacto | Probabilidad | Magnitud | Management Plan | Estrategia |
|---------|-------------------------|--|--|---------|--------------|----------|---|------------|
| 1 | 17/04/2010 | Posible retraso por no tener poca experiencia en YUI | Si no se define bien el alcance de BawaMx, se tendrá peligro de no entregar algo funcional para los desarrolladores | 1 | 20 % | 0.2 | Se debe de definir el alcance de BawaMx con YUI en el núcleo del sistema para evitar retrasos y desorganización en el código. | EE |
| 2 | 17/04/2009 | Posible retraso por proyectos externos de los colaboradores del proyecto | Durante un periodo de 4 meses los colaboradores no podrán dedicar el tiempo necesario al proyecto debido a compromisos laborales | 1 | 20 % | 0.2 | Establecer los días en que no trabajarán buscando una solución intermedia que no signifique la pérdida de todos los cuatro meses. | EM |
| 3 | 23/04/2010 | análisis incorrecto de las tecnologías multimedia | Las tecnologías multimedia a integrar deben ser bien analizadas antes de integrarlas, como hay poco | 2 | 30 % | 0.6 | Analizar si es viable la de integración todas las tecnologías multimedia incluyendo Flash para primera versión de BawaMx | EE |

| | | | conocimiento de ellas, puede causar un grave retraso en el desarrollo del sistema | | | | | |
|-------|------------|---|---|---|------|-----|--|----|
| 4 | 23/04/2010 | Crecimiento lento de la curva de aprendizaje de los desarrolladores que utilizarán BawaMx | | 3 | 60 % | 1.8 | Será responsabilidad del equipo de BawaMx proveer toda la documentación y capacitación a los desarrolladores web que utilicen la arquitectura. | EC |

Tabla 3.2.2 Descripciones de riesgo

3.3 Requerimientos y documento de Visión

Los requerimientos se han asentado en el documento de visión del proyecto, este documento además de resumir y poner en claro todos los elementos para dar inicio al desarrollo del sistema especifica la prioridad y alcances de desarrollo.

Los puntos más importantes a destacar el documento de visión son:

- Declaración del Problema
- Resumen de interesados.
- Resumen de usuarios
- Necesidades
- Características del producto
- Perspectiva del producto
- Licenciamiento e instalación.
- Rangos de calidad
- Documentación
- Guías de instalación.

Para ver el documento detallado consultar **Anexo A. Documento de Visión.**