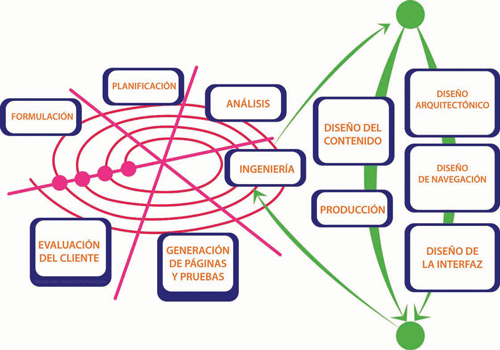
## 

## Marco Metodológico

Modelo de proceso o ciclo de vida del software

La selección del Modelo que se utilizara para la Aplicación a desarrollar será IWEB, ya que se enfoca a la creación, implantación y Capacidad de Mantenimiento y sistemas web, este incluye metodologías, herramientas, técnicas, guías y patrones para cubrir los requisitos únicos de la aplicaciones web y así lograr una aplicación basada en web de alta calidad



*Figura 1. Modelo del Proceso IWEB*

Razones para la elección del Modelo:

* Es un modelo que se especializa en aplicaciones web.
* Este modelo tiene unas etapas que no figuran en otros modelos, como el diseño de interfaz y el diseño de navegación.
* Ya que al escoger este modelo de aplicación web nos asegura que la aplicación sea de calidad, porque todas las etapas de este modelo son esenciales para lograr el desarrollo.

Métodos

El panorama de la tecnología IWeb abarca un conjunto de labores técnicas que permiten al ingeniero Web comprender, caracterizar y luego construir una WebApp de alta calidad. Los métodos IWeb se pueden categorizar de la siguiente manera:

Los **Métodos de comunicación** definen el enfoque con que se facilita la comunicación entre ingeniero Web y los demás participantes de la WebApp (por ejemplo, usuarios finales, clientes de negocio, expertos en problemas de dominio, diseñadores de contenido, líderes de equipo, gestores de proyecto). Las técnicas de comunicación son particularmente importantes durante la recolección de requisitos y siempre que es evaluado un incremento en la WebApp.

Los **Métodos de análisis de requisitos** proporcionan una base para comprender el contenido que entregará una WebApp, la función que proporcionará al usuario final y los modos de interacción que dada clase de usuario requerirá mientras ocurra la navegación por medio de la WebApp.

Los **Métodos de diseño** abarcan una serie de técnicas de diseño que abordan el contenido, la aplicación y la arquitectura de información, así como el diseño de interface y la estructura de navegación de la WebApp.

Los **Métodos de prueba** incorporan revisiones técnicas formales tanto del contenido y el modelo de diseño como de una amplia variedad de técnicas de prueba que abordan conflictos al nivel de componente y arquitectónicos, pruebas de la navegación, pruebas de facilidad de uso, pruebas de seguridad y pruebas de configuración.

El proceso de la ingeniería Web

Comunicación con el cliente. Dentro del proceso de la Ingeniería Web la comunicación con el cliente se caracteriza por medio de dos grandes tareas: análisis de negocio y la formulación.

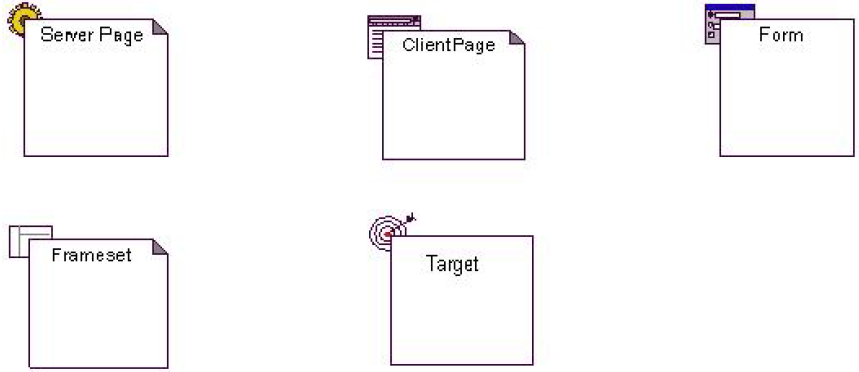
El **análisis del negoci**o se define como el contexto empresarial-organizativo para los sistemas basados en web. Además, se identifican los participantes, se predicen los potenciales cambios en el ambiente o los requisitos del negocio, y se define la integración entre del sistema basado en web y otras aplicaciones de negocios, base de datos y funciones**. La Formulación** es una actividad de recopilación de requisitos que involucra a todos los participantes. El intento es describir el problema que aplicación basada en web habrá de resolver con el aprovechamiento de la mejor información disponible. Además se intenta identificar áreas de incertidumbre y donde ocurrirán cambios potenciales.

* **Planeación**. Se crea el plan del proyecto para el incremento del sistema basado en web. El plan consiste de una definición de tareas y un calendario de plazos respecto al periodo proyectando para el desarrollo del incremento de la aplicación basada en web.
* **Modelado**. Las labores convencionales de análisis y diseño de ingeniería del software se adaptan al desarrollo de la aplicación basado en web, se mezclan y luego se funden en la actividad de modelado Web. El intento es desarrollar análisis “rápidos” y modelos de diseño que definan requisitos y al mismo tiempo representen un sistema y una aplicación basada en web que los satisfará.
* **Construcción**. Las herramientas y la tecnología Web se aplican para construir la aplicación basada en web que se ha modelado. Una vez que se construye el incremento de la aplicación basada en web, se dirige una serie de pruebas rápidas para asegurar que se descubran los errores en el diseño (contenido, arquitectura, interface, navegación). Pruebas adicionales abordan otras características de la aplicación basada en web.
* **Despliegue**. La aplicación basada en web se configura para su ambiente operativo, se entrega a los usuarios finales y luego comienza un periodo de evaluación. La retroalimentación acerca de la evaluación se presenta al equipo Web y el incremento se modifica conforme se requiera.

Proceso de ingeniería web compuesto por las siguientes fases:

* **Planteamiento y formulación:** Identificamos los objetivos de nuestra aplicación, y delimitamos el alcance de la primera iteración.
* **Planificación:** Una vez planteado el problema, podremos estimar costos, riesgos y esfuerzo durante el desarrollo. Recordemos que en la planeación iterativa solamente se detalla la iteración actual, y las iteraciones subsecuentes sólo se plantean de forma general.
* **Análisis:** Durante esta etapa establecemos los requerimientos técnicos, gráficos, y de contenido, que incorporaremos en la iteración.
* **Ingeniería:** La actividad de ingeniería incorpora dos grupos de tareas que se realizan en paralelo: el diseño del contenido y la producción, se enfocan en el diseño, producción y adquisición del contenido de texto, gráfico y video que se vayan a integrar en la aplicación. Estas tareas son realizadas por personal no técnico. Por otro lado, está el diseño arquitectónico, de navegación e interfaz, el cual lidia con los aspectos técnicos.
* **Generación de páginas y pruebas:** Se prueba que el contenido dinámico se genere correctamente, utilizando las plantillas, interfaces y contenidos diseñados en la fase de ingeniería. Posteriormente se realizan las pruebas pertinentes, que dependerán del tipo de aplicación y requerimientos no funcionales (por ejemplo, pruebas de desempeño).
* **Evaluación del cliente:** Al final de cada iteración se debe realizar una evaluación con el cliente, para validar el avance y determinar los cambios o mejoras, en caso de ser necesarios, que se aplicarán en las siguientes iteraciones.

Metodología a utilizar

**La metodología a utilizar para la WebApp será WAE, que es la extensión de UML proporciona la habilidad de asignar iconos diferentes a las clases estereotipadas El problema de una Aplicación Web es que tiene diferentes scripts y variables que se ejecutan en el servidor o del lado del cliente. Para estos conformaremos dos métodos el del servidor y el del cliente y de esta forma separa las funciones que corren en el servidor y en el cliente.

*Figura 2. “Estereotipos de la Metodología WAE”*

1. ***Estereotipos***

|  |  |
| --- | --- |
| Nombre | Página del Servidor |
| Meta-Modelo | Clase |
| Descripcion | Una página del servidor representa una pagina web que tiene scripts que son ejecutado por el servidor. Estos scripts actúan recíprocamente con recursos en el servidor(banco de datos, lógica de negocio, sistemas externos). Los funcionamientos del objeto representan las funciones en el script, y sus alcances de la página(accesible por todas las funciones en la página) |
| Icono |  |
| Restricciones | Las páginas del servidor pueden tener sólo relaciones con objetos en el servidor. |
| Valores Etiquetados | Artefacto de scripting – O el lenguaje o artefacto que deben ser uso para ejecutar o interpretar esta página (JavaScript, VBScripts, Perl, etc) |

|  |  |
| --- | --- |
| Nombre | Página del Cliente |
| Meta-Modelo  de Clase | Clase |
| Descripción | Un caso de una página del cliente es una estructura HTML de la página Web. Como cualquier página HTML es una mezcla de datos, presentación y lógica igual. Las páginas del cliente son dadas por el browser del cliente, y puede contener scripts que son interpretadas por el browser. Las páginas del clientes pueden estar asociadas con otras páginas del cliente o del servidor |
| Icono | C:\Users\mj\Desktop\2.png |
| Restricciones | Ninguna |
| Valores  Etiquetados | TitleTag – Titulo de la página como desplegado por el browser.  BaseTag – El URL de la base para deferencing URLs Relativo  BodyTag – El Juego de atributos para la etiqueta del <body> que pone el texto de fondo por defecto |

|  |  |
| --- | --- |
| Nombre | Formulario (Form) |
| Meta-Modelo de Clase | Clase |
| Descripción | Una Clase estereotipada como un <<Form>> es una colección de campos de entradas que son parte de una página del cliente. Una clase de formulario se mapea directamente a HTML. Sus atributos representan los campos de la entrada de los formularios de HTML (input boxes, text áreas radio buttons, check boxes y los campos ocultos).Un <<form>> se opera desde que no pueden encapsularse su funcionamiento en un formulario. Cualquier funcionamiento en un formulario que actúa recíprocamente con el formulario seria la propiedad de la página que contiene al formulario |
| Icono | C:\Users\mj\Desktop\3.png |
| Restricciones | Ninguna |
| Valores  Etiquetados | Método: Suministra datos a la acción URL, cualquiera GET o POST |

|  |  |
| --- | --- |
| Nombre | Frame Set |
| Meta-Modelo  de Clase | Clase |
| Descripción | Un Juego de frames es un contenedor de múltiples aginas web.  La vista de los rectángulos son áreas divididas en rectángulos más pequeños de frames.  Los contenidos de un Frame pueden estar en una página web o en otro juego de frames.  Un Frame set es una página <<página del cliente >>, que puede tener funcionamiento y atributos también, pero estos solos son activados por el browser que no devuelven frames. |
| Icono | C:\Users\mj\Desktop\4.png |
| Restricciones | Ninguna |
| Valores  Etiquetados | Filas (Rows) – El valor del atributo de las filas es la etiqueta <frameset> del HTML. Esto es una secuencia con los hieghts delimitados de la fila.  Columnas (Cols) – El valor del atributo del cols es la etiqueta <frameset> del HTML Esto es una secuencia de anchuras de columnas delimitadas |

|  |  |
| --- | --- |
| Nombre | Target |
| Meta – Modelo de clase | Clases |
| Descripción | Típicamente un target es un marco en una ventana definida por un frameset, sin embargo un target podría ser un completamente de un nuevo caso del browser o ventana. «Targeted link» las asociaciones especifican targets como el lugar donde una nueva página Web será devuelta. |
| Icono | C:\Users\mj\Desktop\5.png |
| Restricciones | El nombre de un target debe ser único para cada cliente del sistema. Esto significa eso que sólo un caso de un target puede existir en el mismo cliente. |
| Valores etiquetados | Ninguno |

|  |  |
| --- | --- |
| Nombre | JavaScript |
| Meta – Modelo de clase | Clase |
| Descripción | En un Javascript permitido por el browser es posible simular a usuario los objetos definidos con funciones del  Javascript. Los casos del «JavaScript» existen solamente en el contexto de las páginas del cliente. |
| Icono | C:\Users\mj\Desktop\6.png |
| Restricciones | Ninguno |
| Valores etiquetados | Ninguno |

1. **Indicadores de asociación**

* **Link:** en un diagrama de la clase un es una asociación entre una «página del cliente» y cualquiera otro «página del cliente» o a una «página del servidor».
* Target link: es un link donde la página asociada se da en otro target.
* **Submit:** la asociación siempre es entre un «form» y una «página del servidor».
* **Builds:** identifica qué página del servidor es responsable para la creación de una página del cliente.
* **Redirect:** es una asociación unidireccional con otra página Web.
* **Object** se establece entre una página cliente y un objeto que se encuentra en ella

1. **Atributos**

* **Input Element:** Es un atributo de la clase Form que se corresponde con campos de entrada en el Form de la página HTML
* **Select Element**: Es un atributo de la clase Form que representa un campo cuyo valor, o valores, pueden ser seleccionados a partir de los elementos de una lista. Básicamente se traduce a los controles ComboBox y ListBox.c) Text Area Component: es un atributo de la clase Form que representa a un área de texto

1. **Componentes**

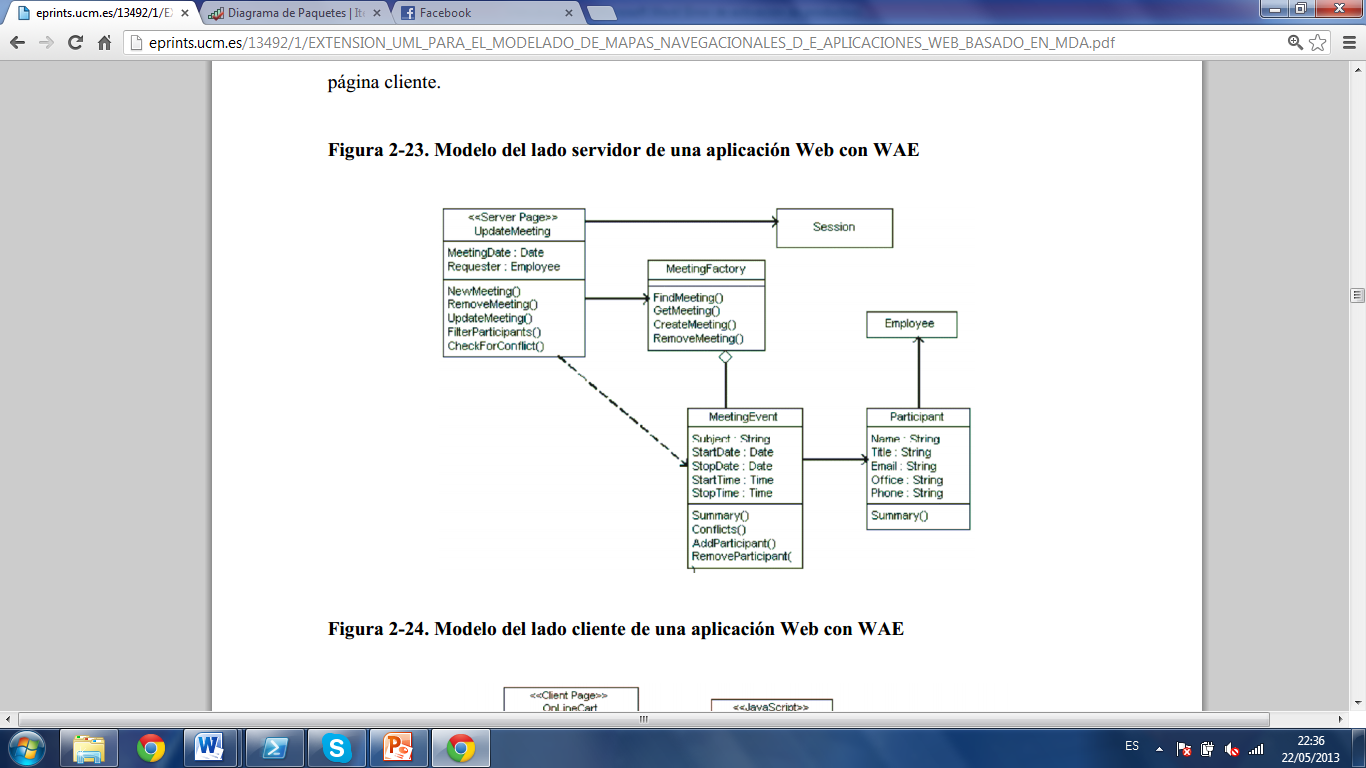
* **Page:** Un componente Page representa a cualquier tipo de página web

**Modelado**

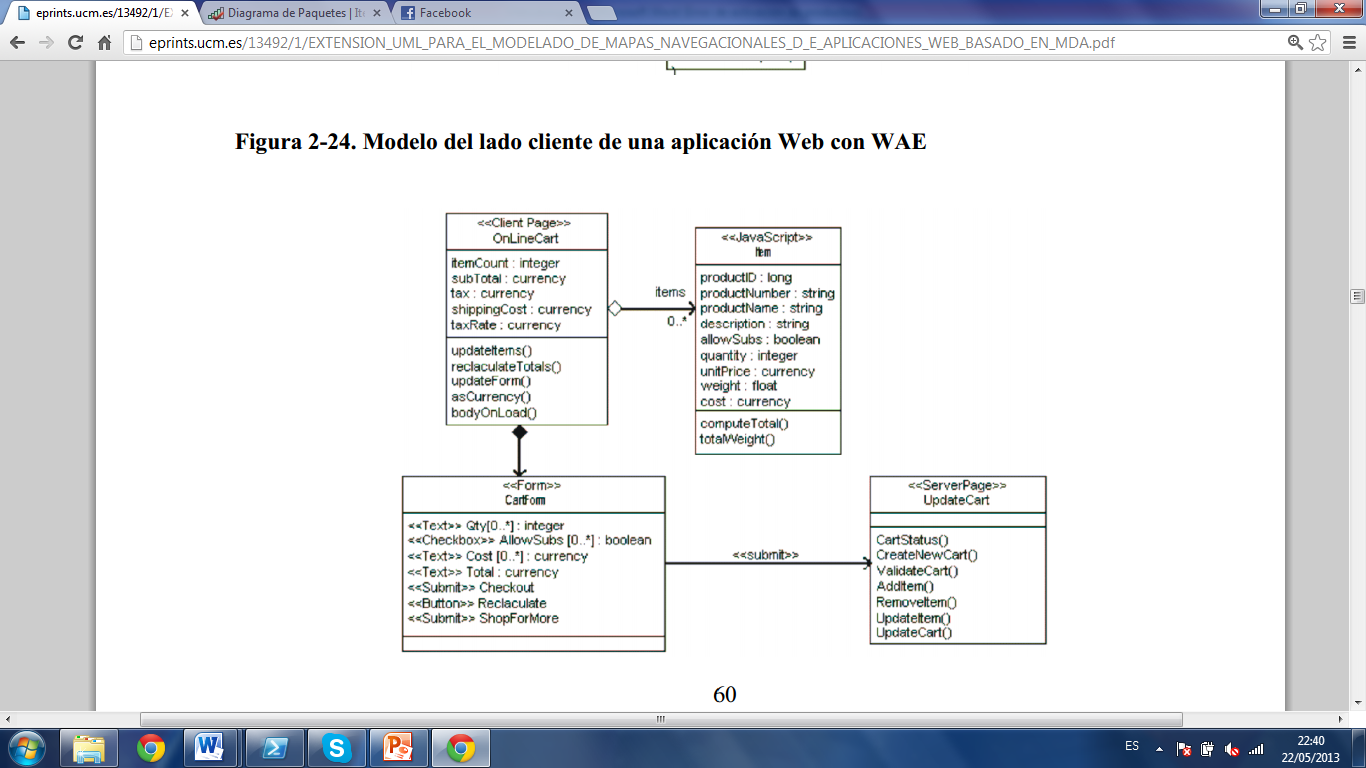
Como se dijo anteriormente, las clases «server page» y «client page» son dos

Abstracciones de una misma página Web, en WAE se deben relacionar a través de una asociación dirigida y estereotipada como «build» que significa que la página de servidor construye la página cliente.

**Modelo del lado servidor de una aplicación Web con WAE**



*Figura 3. ” Modelado al lado del servidor”*

**Modelo del lado cliente de una aplicación Web con WAE**

*Figura 4. “Modelado del lado del Cliente”*

**Descripción de las tareas y condiciones para realizarlas**

**Formulación**

* Se realiza la entrevista con el cliente
* Se identifican las necesidades y requerimientos
* Se determinan los objetivo de la aplicación web
* Se recopilan los requisitos
* Se realizan casos de usos
* Se establecen alcances y limitaciones

**Planificación**

* Se realiza análisis de recurso y factibilidad
* Se realiza un análisis de riesgos

**Análisis**

* **Modelado de contenido:** Contiene los elementos estructurales que proporcionan una visión de los requisitos del contenido para la aplicación web.
* **Modelado de interacción:** Se describe como interactúa el usuario con la aplicación.
  + Se realizan : Casos de uso, diagramas de secuencia, diagrama de estado y prototipo de la interfaz del usuario
* **Modelado funcional:** Se define las operaciones que se aplicaran al contenido de la aplicación y describe otras funciones de procesamiento, independientes del contenido pero necesarias para el usuario final
* **Modelado de configuración:** Describe el ambiente y la infraestructura en la que reside la aplicación

Se realiza un **Análisis relación navegación:** que determina la estructura de relación de una aplicación, esto ayuda a descubrir las relaciones potencialmente útiles en los dominios de la aplicación, y que se puede implementar como vínculos más adelantes. Se realiza los siguientes pasos:

* **Análisis de los participantes:** Determina la estructura de relación de usuarios y establece una apropiada jerarquía de participantes.
* **Análisis de elementos:** Se identifica los objetos de contenido y los elementos funcionales de interés para los usuarios finales.
* **Análisis de Relación:** Se describe las relaciones entre los elementos de la aplicación web.
* **Análisis de navegación:** Se examina como los usuarios pueden acceder a elementos individuales o grupos de elementos.
* Revisión y corrección de errores.

**Ingeniería**

**Diseño**

El diseño conduce a un modelo que contiene la mezcla adecuada de estética, contenido y tecnología. La mezcla varía dependiendo de la naturaleza de la aplicación web, como consecuencia, las actividades de diseño también variaran.

**Diseño de Interfaz**

* Revisar la información contenida en el modelo de análisis.
* Desarrollar un bosquejo aproximado de la plantilla de la interfaz.
* Ordenar los objetivo, el usuario con acciones específicas de la interfaz.
* Definir un conjunto de tareas de usuario que estén asociadas con cada acción.
* Elaborar bosquejos con imágenes de la pantalla para cada acción de la interfaz.
* Refinar la plantilla de la interfaz y los bosquejos con el uso de entradas desde el diseño estético.
* Identificar los objetivos de la interfaz de usuario que se requiera para implementarlo.
* Desarrollar una representación de procedimiento de la interacción de usuario con la interfaz.
* Desarrollar una representación del comportamiento de la interfaz.
* Describir la plantilla de la interfaz para cada estado.
* Refinar y revisar el modelo de diseño de la interfaz.

**Diseño de Contenido**

Define la pantalla, estructura y el bosquejo de todo el contenido que se presenta como parte de la aplicación web, se establece las relaciones entre los objetos de contenido.

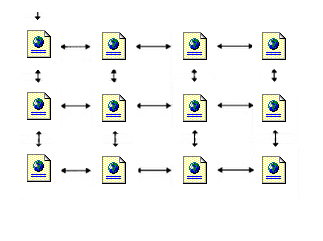
**Diseño Arquitectónico**

Identificar la estructura que utilizara la aplicación web, donde se debe definir la arquitectura de contenido y arquitectura de la aplicación. Los tipos son:

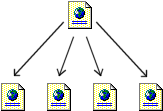
* **Estructuras lineales:** Cuandoes predecible la sucesión de interacciones. Por ejemplo en la entrada y validación de datos.

lineal.gif (163×32)

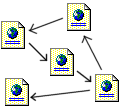
* **Estructura de retícula:** Cuando si el contenido de la web puede ser organizado en dos o más dimensiones. Para ello el contenido debe ser muy regular. Por ejemplo marcas de herramientas y tipos de herramientas.

****

* **Estructuras jerárquicas:** Se fomenta el flujo de control solo a lo largo de las ramas verticales. En una aplicación web se pueden enlazar por hipertexto ramas verticales de la misma estructura.



* **Estructura de red o de web pura:** Es una arquitectura en evolución de los sistemas orientados a objetos. Se enlaza todo con todo. Da mucha flexibilidad de navegación, aunque a veces es confusa para el usuario.



**Diseño de Navegación**

* Identificar la semántica de navegación
* Definir la sintaxis que logra la navegación

**Diseño de Componentes:**

El diseñador debe transformar la información del análisis y los modelos arquitectónicos en una representación de diseño que proporcione suficiente detalle para guiar la etapa de construcción.

**Construcción y Adaptación**

**Construcción de la aplicación:** Se realiza una serie de tereas de codificación que conduce al desarrollo del Aplicación Web, se basan:

**Preparación:**

• Entender a la perfección el problema que se intenta resolver con el desarrollo de la **Aplicación Web**

• Entender los principios básicos del diseño de la Aplicación Web

• Definir el lenguaje de programación

• Definir las herramientas a utilizar en la programación

• Definir estándares de programación

**Codificación:**

• Seleccionar la estructura de datos

• Entender la arquitectura de la aplicación y crear interfaces que sean consistentes al diseño arquitectónico

• Definir las variables según los estándares descritos

• Definir la conexión de base de datos

• Escribir las líneas de códigos para la aplicación web

• Crear las interfaces, verificando su contenido, crear los distintos componentes y desarrollar su forma de navegación

• Conocer los componentes de la base de datos

**Validación:**

• Realizar pruebas de unidad y corregir los errores detectados

• Rehacer el código en caso que sea necesario

• Se realiza una revisión y corrección de errores

**Generación de páginas y Pruebas**

Se realiza las siguientes pruebas:

**Pruebas de Interfaz:**

• Descubrir errores relacionados con la ejecución de los vínculos de menú o los datos que se ingresan a un formulario

• Descubrir errores en la navegación, funcionalidad o despliegue del contenido

**Pruebas de Contenido:**

• Descubrir errores sintácticos

• Descubrir errores semánticos

• Encontrar errores en la organización o estructura del contenido

**Pruebas de Navegación:**

• Probar que el usuario pueda navegar en la aplicación web, siendo funcional y validar que cada unidad de navegación pueda ser alcanzada por la categoría del usuario adecuada

**Pruebas de Componentes:**

• Pruebas de partición de equivalencia: ingreso de valores.

• Análisis de valores al límite: forzar ingreso de datos.

• Pruebas de ruta: pruebas de algoritmo.

**Pruebas de Configuración:**

• Servidor:

• Verificar que los archivos de sistemas, directorios y datos de sistemas relacionados se creen correctamente cuando opere la Aplicación Web

• Cliente:

• Verificar el hardware

• Reconocer el sistema de navegación

• Verificar componentes de interfaz (Java)

• Verificar la conectividad

**Evaluación del cliente**

• Se aprueba la validación del avance

**Instalación de la aplicación web**

• Verificar la configuración de hardware.

• Cargar o montar la aplicación web en el servidor

• Efectuar el proceso de migración de información del cliente.

• Realizar pruebas de puesta en marcha

**Documentación**

• Se realiza Manual de usuario

• Se realiza Manual técnico

**Capacitación**

Se realiza curso de capacitación

Gestión de proyectos de software

Planificación temporal del proyecto

Métricas y estimación

**Factibilidad**

**Factibilidad técnica**

Debido a tratarse de un sistema mediano, pero sumamente importante, sus costos son bajos y la necesidad de equipos potentes es innecesaria.

En la empresa cuenta con estos requisitos, por lo que no habría necesidad de adquirirlos. En cuanto nuestros conocimientos son suficientes para el desarrollo y cumplimiento del software. Se debe de concluir indicando si es factible o no desarrollar técnicamente el proyecto.

Para una mejor explicación de lo que se ha descrito anteriormente, se muestra la siguiente tabla, detallando las características tanto de Hardware como de Software que se necesitan para el desarrollo y funcionamiento del sistema

|  |  |
| --- | --- |
| Nombre | Características |
| Servidor | Procesador AMD Athlon (1.30 GHz),Memoria RAM 4 GB, Disco Duro 160 GB, Sistema operativo DEBIAN |
| Notebook | Procesador AMD Dual Core (1.0 GHz), Memoria RAM 2GB, Disco Duro 320 GB, Pantalla 14.0, Video AMD Radeon, Teclado español, LAN 10/100 Ethernet, WiFi 802 11b/g/n |
| Navegadores | Internet Explorer versión 9.0, Google Chrome versión 13.0, Firefox versión 5.0, Opera versión 11.5 |
| Sistema Operativo | Windows Seven / Ubuntu |
| Internet | Proveedor VTR |
| Desarrollo de la Aplicación Web | Notepadd ++ versión 5.9, KompoZer versión 0.8, Apache versión 2.2, PHP versión 5.3, MySql versión 5.1 |

**Factibilidad Legal**

La herramienta a utilizar para el desarrollo de la aplicación web son Notepadd++ y KompoZer están bajo las licencias:

Licencia Pública de Mozilla (MPL), es una licencia de código abierto y de software libre.

Licencia Pública General de GNU (GNU GPL), protege la libre distribución, modificación y uso de software.

Licencia Pública General Reducida de GNU (GNU LGPL), está diseñado para compartir y modificar dicho software.

En conclusión, el desarrollo de la aplicación web se realiza con programas de licencia gratuita.

**Factibilidad Operacional**

ISO 9001 Software de calidad es una aplicación web, los usuarios cuentan con el conocimiento necesario para utilizar internet, por lo tanto, se debe realizar una capacitación en el ambiente en que se utilizara el sistema, para que los usuarios del sistema sepan cómo utilizarlo y los beneficios que otorga el sistema.

Los usuarios que manejen el sistema deben contar con el conocimiento necesario sobre la ISO 9001 para que el sistema cuente con el uso correcto

**Factibilidad económica**

* + **Inversión Hardware y Software:** La Empresa, cuenta con todo el hardware necesario para implementar la aplicación web.
  + **Servicios básicos** que se gastara durante los 5 meses de desarrollo del proyecto:

Adjuntar documento EXCEL FLUJO Y DEMAS

Flujo de caja:

Estudio de Mercado

Gestión de riegos

Identificación de descripción del riesgo

Análisis de Riesgos

Control de calidad

Organización

Tareas de SQA

Plan de Pruebas

Herramientas CASE

Conclusión