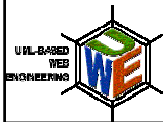


Metodologías para el Desarrollo de Aplicaciones Web: UWE



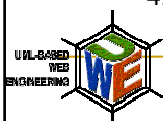
Trabajo realizado por:

Daniel Mínguez Sanz.

Emilio José García Morales.

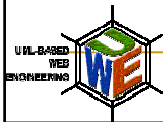
ÍNDICE

1. ¿Qué es una Aplicación Web?
 - 1.1 Tipos de Aplicación Web.
 - 1.2 Características de una Aplicación Web
 - 1.3 Requisitos del desarrollo de una Aplicación Web.
 - 1.4 Requisitos para la aplicación de una Aplicación Web.
2. ¿El por qué de una Metodología de Desarrollo Web?
 - 2.1 Evolución de las metodologías de Desarrollo Web.
3. ¿Qué es UWE?
 - 3.1 Principales aspectos.
 - 3.2 Fases del Desarrollo Web.
4. Caso Práctico: *A Conference Review System*.
 - 4.1 Definición de Actores.
 - 4.2 Definición de Procesos.
 - 4.3 Análisis de Casos de Uso.
 - 4.3.1 Caso Práctico.



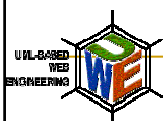
ÍNDICE

- 4.4 Representación UML del Modelo Conceptual.
 - 4.4.1 Elementos de modelado.
 - 4.4.2 Caso Práctico.
- 4.5 Diseño Navegacional.
 - 4.5.1 Modelado del Espacio Navegacional.
 - 4.5.1.1 Elementos de modelado.
 - 4.5.1.2 Caso Práctico.
 - 4.5.2 Estructura del Modelo Navegacional.
 - 4.5.2.1 Elementos de Modelado.
 - 4.5.2.2 Caso Práctico.



ÍNDICE

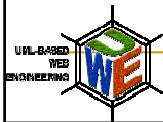
- 4.5.3 Añadir Menús: Pasos.
 - 4.5.3.1 Elementos de modelado.
 - 4.5.3.2 Caso Práctico.
- 4.6 Flujo de Presentación y Storyboarding.
 - 4.6.1 Storyboards
 - 4.6.1.1 Elementos de modelado.
 - 4.6.1.2 Caso Práctico.
 - 4.6.2 Flujo de Presentación
 - 4.6.2.1 Elementos de modelado.
 - 4.6.2.2 Caso Práctico.
- 5. Bibliografía.



1. ¿Qué es una Aplicación Web?

Es un SI donde una gran cantidad de datos volátiles, altamente estructurados, van a ser consultados, procesados y analizados mediante navegadores.

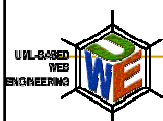
Una de las principales características va a ser su alto grado de interacción con el usuario, y el diseño de su interfaz debe ser claro, simple y debe estar estructurado de tal manera que sea orientativo para cada tipo de usuarios.



1.1 Tipos de Aplicación Web.

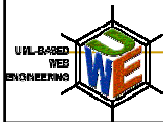
A lo hora de establecer una clasificación la podemos realizar atendiendo a criterios como pueden ser la complejidad de los datos, de la propia aplicación, la volatilidad, la estructuración de los datos o la intencionalidad de la aplicación. De entre todas las posibles clasificaciones, la que aparece a continuación esta hecha en base a la intencionalidad de la aplicación:

- ✓ Informacionales: Orientadas a la difusión de información personalizada o no, y con acceso a la BD o sin el.
- ✓ Orientados a la descarga de datos: Servidores de material didáctico, servidores de canciones, ...
- ✓ Interactivas: Orientadas ala interacción con el usuario.



1.1 Tipos de Aplicación Web.

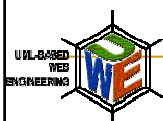
- ✓ Orientas al Servicio: Sistemas de ayuda financiera, simuladores, ...
- ✓ Transaccionales: Compra electrónica, banca electrónica,...
- ✓ De Flujo de Datos: Sistemas de planificación en línea, manejo de inventario, ...
- ✓ Entornos de Trabajo Colaborativo: Herramientas de diseño colaborativo, sistemas de autoría distribuidos,...
- ✓ Comunidades on-line (Sistemas C2C): Foros de debate, servicios de subastas,...
- ✓ Portales Web: Centros comerciales de compra electrónica, intermediarios en línea,...
- ✓ Orientados al análisis de datos: Dataworkhousing, aplicaciones OLAP,...



1.2 Características de una Aplicación Web

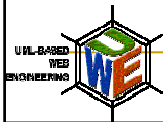
Las Aplicaciones Web tienen una serie de rasgos comunes que diferencia a unos tipos de aplicaciones software de otros, y que son:

- Desde el punto de vista del usuario, se ha universalizado su accesibilidad: Actualmente un usuario experto y un usuario con habilidad limitada en el uso de aplicaciones informáticas acceden al mismo tipo de aplicación. Aún más, el número y tipo de usuario de las Aplicaciones Web no siempre es predecible, lo que obliga a tener el concepto de facilidad de uso aún más presente que en otros tipos de aplicaciones.



1.2 Características de una Aplicación Web

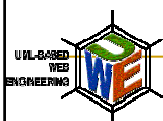
- Desde el punto de vista de la plataforma se realiza un uso intensivo de la red y la conexión se establece desde distintos tipos de dispositivo de acceso.
- Desde el punto de vista de la información, asistimos en la actualidad a una disponibilidad global de fuentes heterogéneas de información, estructurada y no estructurada, pertenecientes a distintos dominios y que colaboran en el cumplimiento de los objetivos de la aplicación.



1.3 Requisitos del desarrollo de una Aplicación Web.

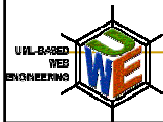
Cada una de estas perspectivas introduce una serie de requisitos que deben ser tenidos en cuenta durante el proceso de desarrollo de cualquier tipo de Aplicación Web con el fin de incrementar su probabilidad de éxito de implantación y que pueden ser estructuradas como sigue

- Portabilidad. Debido a la dinamicidad del entorno tecnológico, a menudo es necesario implantar una misma aplicación en distintas plataformas, con distintas arquitecturas, con distintas tecnologías y/o atendiendo a distintos dispositivos de acceso, lo que obliga a desarrollar técnicas, modelos y herramientas que faciliten la reutilización e independiza hasta donde sea posible en el desarrollo de la aplicación.



1.3 Requisitos del desarrollo de una Aplicación Web.

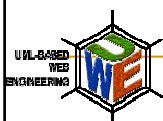
- Inmediatez (Rapidez de Implantación). El desarrollo de aplicaciones web requiere un período de implantación mucho *más* reducido, que influye en todo su ciclo de desarrollo.
- Inmediatez (Rapidez de Implantación). El desarrollo de aplicaciones web requiere un período de implantación mucho *más* reducido, que influye en todo su ciclo de desarrollo.
- Creación de contenidos como parte integrante de la fase de ingeniería de la aplicación. Aunque en este trabajo nos centramos en la especificación de aplicaciones orientadas a ofrecer funcionalidad compleja, más allá de la mera



1.3 Requisitos del desarrollo de una Aplicación Web.

diseminación de información, el diseño y producción de textos, gráficos, vídeos etc. que conforman la estructura informacional de la aplicación es una tarea que debería ser realizada en paralelo al diseño de la propia aplicación.

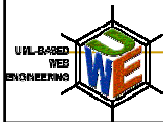
- Integración (disponibilidad global) de fuentes hetero-géneas de información. La posible necesidad de manejo integrado de contenido estructurado y no estructurado, almacenado en distintos formatos (bases de datos, sistemas de ficheros, dispositivos multimedia) y accesibles de forma distribuida mediante múltiples aplicaciones es otro de los factores que condiciona el proceso de diseño de este tipo de aplicaciones.



1.4 Requisitos para la aplicación de una Aplicación Web.

Los requisitos que vamos a destacar en una Aplicación Web, son los siguientes:

- Evolución orgánica. Es un aspecto fundamental en el ámbito de la web, donde tanto el contenido como los requisitos de las aplicaciones evolucionan a una velocidad vertiginosa. Esto es en parte debido a que los clientes de este tipo de aplicaciones suelen tener un conocimiento muy pobre de sus necesidades y de las posibilidades del sistema.
- Seguridad en la comunicación. Debido a que las aplicaciones web se encuentran disponibles a través de una red, es difícil limitar el grupo de usuarios finales que pueden acceder a ella. Es por ello que se hacen necesarios mecanismos para proteger información sensible y proporcionar modos seguros de transmisión de datos.



1.4 Requisitos para la aplicación de una Aplicación Web.

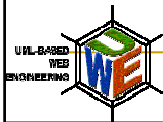
- Calidad (margen de error cero). La permisividad mostrada por los usuarios ante los errores en aplicaciones Web (robustez, facilidad de uso o rendimiento) es muy limitada: enlaces erróneos o información desactualizada provocan la pérdida de usuarios de la aplicación. Es por ello que en el desarrollo de este tipo de aplicaciones es primordial disponer de mecanismos exhaustivos de control de calidad que minimicen las posibilidades de fracaso de la aplicación.
- Velocidad. El uso intensivo de la red provoca que la elección de protocolos de comunicación y el mantenimiento de una velocidad de acceso adecuada sean una parte clave de diseño de dichas aplicaciones.



1.4 Requisitos para la aplicación de una Aplicación Web.

- Importancia de la interfaz. La necesidad de implementar interfaces de usuario más intuitivas, capaces de capturar la atención del usuario y facilitar el acceso a la información a aquéllos que poseen una habilidad limitada en el uso de aplicaciones informáticas.
- Necesidad de personalización. Debido, a la facilidad de migración del usuario a otras aplicaciones y la variedad de este tipo de aplicaciones, la personalización es un elemento significativo del diseño, y da valor añadido a un contenido que debe además ser accesible y estar actualizado.

A estos requisitos debemos añadirles seguridad de la propia aplicación, escalabilidad, disponibilidad, interoperabilidad con sistemas propietarios, etc.

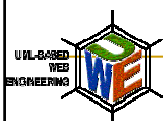


2. ¿El por qué de una Metodología de Desarrollo Web?

Los principales problemas que nos encontramos es la falta de fiabilidad, seguridad, escalabilidad, mantenimiento, integración y la alta dependencia para su desarrollo e implantación junto con la falta de estándares.

Lo que deseamos es controlar el caos que han provocado en el pasado procesos creativos de desarrollo con el fin de proporcionar un proceso sistemático orientado a la mejora de la calidad de la aplicación final. En esta nueva disciplina se parte de la base de que las necesidades de evolución, mantenimiento, la adaptación a nuevos dispositivos de acceso y la migración a nuevas plataformas y entornos de desarrollo deben dirigir el proceso del ciclo de vida.

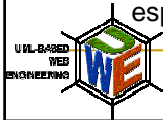
Para todo esto se han desarrollado metodologías que permiten estructurar, comunicar, entender, simplificar y formalizar tanto el dominio como las decisiones de diseño, así como disponer de documentación detallada para posibles cambios del software.



2.1 Evolución de las metodologías de Desarrollo Web.

Las distintas metodologías se pueden dividir en tres generaciones en base a su sofisticación, estas son:

- Primera Generación:(Principios de los 90) Se sientan las bases de la ingeniería Web, en los que se incluyen conceptos como construcción de navegación, separación entre estructuras y el contenido durante el ciclo de desarrollo.
- Segunda Generación: (Segunda mitad de los 90) Se refinan los primeros modelos y se añaden los soportes de funcionalidad básica y se llevan a cabo los primeros esbozos de proceso donde se delimitan los modelos conceptual, lógico y físico.
- Tercera generación: (A partir del 2000): Se lleva a cabo la profundización en el soporte para la funcionalidad, enfatización de la figura del usuario en los métodos, y se avanza hacia la estandarización de notaciones, procesos y lenguajes de especificación.

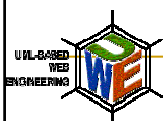


3. ¿Qué es UWE?

La propuesta de Ingeniería Web basada en UML (UWE (Koch, 2000)) es una metodología detallada para el proceso de autoría de aplicaciones con una definición exhaustiva del proceso de diseño que debe ser utilizado. Este proceso, iterativo e incremental, incluye flujos de trabajo y puntos de control, y sus fases coinciden con las propuestas en el Proceso Unificado de Modelado.

UWE está especializada en la especificación de aplicaciones adaptativas, y por tanto hace especial hincapié en características de personalización, como es la definición de un modelo de usuario o una etapa de definición de características adaptativas de la navegación en función de las preferencias, conocimiento o tareas de usuario.

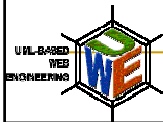
Otras características relevantes del proceso y método de autoría de UWE son el uso del paradigma orientado a objetos, su orientación al usuario, la definición de un meta-modelo (modelo de referencia) que da soporte al método y el grado de formalismo que alcanza debido al soporte que proporciona para la definición de restricciones sobre los modelos.



3.1 Principales aspectos.

Los principales de aspectos en los que se fundamenta UWE son los siguientes:

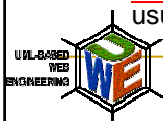
- Uso de una notación estándar, para todos los modelos (UML : Lenguaje de modelado unificado).
- Definición de métodos: Definición de los pasos para la construcción de los diferentes modelos.
- Especificación de Restricciones: Se recomienda el uso de restricciones escritas (OCL: Lenguaje de restricciones de objetos) para aumentar la exactitud de los modelos.



3.2 Fases del Desarrollo Web.

Por lo que respecta al proceso de autoría de la aplicación, UWE hace un uso exclusivo de estándares reconocidos como UML y el lenguaje de especificación de restricciones asociado OCL. Para simplificar la captura de las necesidades de las aplicaciones web, UWE propone una extensión que se utiliza a lo largo del proceso de autoría. Este proceso de autoría está dividido en cuatro pasos o actividades:

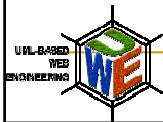
- Análisis de Requisitos: Fija los requisitos funcionales de la aplicación Web para reflejarlos en un modelo de casos de uso.
- Diseño Conceptual: Materializado en un modelo de dominio, considerando los requisitos reflejados en los casos de uso.
- Diseño Navegacional: Lo podemos subdividir en :
 - Modelo del Espacio de Navegacional.
 - Modelo de la Estructura de navegación: Muestra la forma de navegar ante el espacio de navegación.
- Diseño de Presentación: Representa las vistas del interfaz del usuario mediante modelos estándares de interacción UML.



4. Caso Práctico: *A Conference Review System.*

Para hacer mas comprensible el modelo de desarrollo que sigue UWE a continuación vamos a mostrar un ejemplo donde iremos explicando los pasos que sigue el modelo de desarrollo de aplicaciones Web.

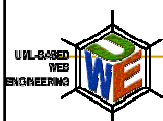
El caso Práctico que vamos a exponer consiste en el desarrollo de una aplicación Web para un Sistema de Revisión de Conferencias.



4.1 Definición de Actores.

El propósito del sistema es apoyar el proceso de evaluación y selección de papeles para una conferencia. Los actores que encontramos son:

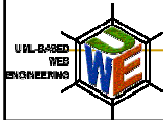
- ❑ **Presidente del comité del programa:** Es el responsable de programar la conferencia, determinar las sesiones que tendrá y los temas, el comité del programa y decidir los artículos aceptados o rechazados, aconsejados por los miembros del comité. El presidente también va a definir los plazos de conferencia: La revisión, envío y notificación.
- ❑ **Miembros del Comité del Programa:** Es el responsable de evaluar los artículos que se le asignan, pudiendo delegar a otra persona la revisión de otro artículo. También pueden aconsejar al presidente sobre los artículos aceptados y rechazados.
- ❑ **Revisor:** Responsable de revisar un artículo.
- ❑ **Autor:** Envía los artículos para la conferencia, que deben registrados con diferentes identificaciones. Tanto los miembros del comité como los revisores pueden ser autores.



4.1 Definición de Actores.

En cuanto a los actores, las reglas que lo controlan las acciones que pueden realizar son las siguientes:

- Todos actores menos el presidente deben registrarse en el sistema en el orden en cual van a participar.
- Los miembros del comité son pre-registrados por el presidente pero deben confirmar su asistencia. Estos pueden modificar tanto su clave y datos sobre forma de contactar. También pueden indicar sus preferencias en cuanto a temas y sesiones cuando estas estén definidas.
- Los revisores pueden ser registrados por el presidente o por los miembros del comité, pero deben confirmar su asistencia. No pueden indicar sus preferencias en cuanto a temas y sesiones.
- Los actores deben registrarse antes de enviar un artículo.
- Un autor puede registrar a un coautor, en que el caso el coautor debe registrarse también para tener el acceso a el sistema.



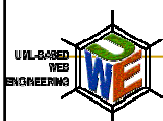
4.2 Definición de Procesos.

- **Envío de Artículos:** Todos los autores registrados pueden enviar un artículo. Para enviarlo debe indicar el título, el resumen, el número de sesión, conjunto de temas elegidos de una lista previamente determinada por el presidente.

El sistema, después de verificar que está registrado le asigna ID al artículo y le permite modificarlo o cambiarlo por uno nuevo, hasta la fecha tope de presentación.

- **Asignación de artículos a revisores:** Un Miembro de PC puede reasignar un papel a un crítico. En este caso, el crítico será responsable de la evaluación del artículo, pero el miembro será el responsable final de la evaluación del artículo.

Cada vez que un revisor termina el proceso de revisión, el sistema avisa al miembro asociado.

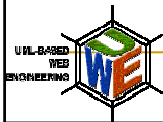


4.2 Definición de Procesos.

- **Asignación de artículos a los miembros:** Una vez acabado el plazo el presidente habilita una lista de artículos aceptados, que serán leídos por los miembros. Estos evalúan su interés, calificándolos del 0 al 2(dónde 0 no indican el interés, 1 indica el interés moderado y 2 indican el interés fuerte).

El presidente basados en la lista de artículos preferidos, asigna los artículos para que lo revisen. Una vez hecho cada miembro recibe un mensaje de correo donde le indican la lista de artículos y la dirección de página donde puede acceder a los artículos.

El presidente pueden asignar un mismo articulo a varias miembros.



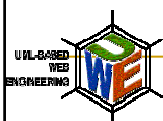
4.2 Definición de Procesos.

- **Revisión:** Un miembro del comité o revisor, debe revisar los artículos que se le han asignado, cada revisión obtiene una calificación del 1 al 5.

El revisor puede indicar si la revisión realizada es final o no lo es. Si la revisión no es final, él puede cambiar cualquier parte de la revisión hasta la fecha tope de la revisión definida por el miembro del comité.

Los miembros pueden revisar otros artículos, si ya tiene revisados los suyos.

El presidente tiene acceso durante todo el proceso de revisión a todos los artículos.

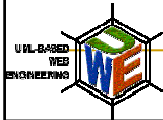


4.2 Definición de Procesos.

- **Artículos Aceptados/Rechazados:** Una vez la fecha tope de la revisión se ha alcanzado, el proceso de la revisión está cerrado. En este momento a todos los Miembros se les permite ver todos los papeles y sus revisiones respectivas. El sistema puede mostrar el la lista de artículos en función del criterio pertinente.

Después de la deliberación, el presidente teniendo en cuenta las recomendaciones de los Miembros y críticos, escoge los papeles que se aceptarán y se rechazarán. También se mostraran las estadísticas sobre los artículos.

Una vez el proceso ha finalizado, el sistema emite una notificación a los autores que incluyen las partes apropiadas de las revisiones sometidos por los Miembros y los críticos. También es posible visualizar la lista de aceptados y rechazados.



4.3 Análisis de Casos de Uso

Siguiendo el Proceso de Desarrollo de Software Unificado aplicado a la Ingeniería Web, se propone los casos de uso necesarios para capturar los requisitos del sistema. Ésta es una técnica que fuerza a definir quienes son los actores (usuarios) de la aplicación y ofrece una manera intuitiva de representar la funcionalidad y aplicación en cada uno de los actores.

Basado en la descripción textual de los requisitos del Sistema de Revisión de Conferencia, podemos ver que los usuarios pueden actuar en los papeles siguientes:

Presidentes.

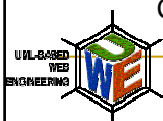
Miembros del comité.

Crítico.

Autor.

Coautor.

La críticos pueden ser modelados como una generalización de Miembros.

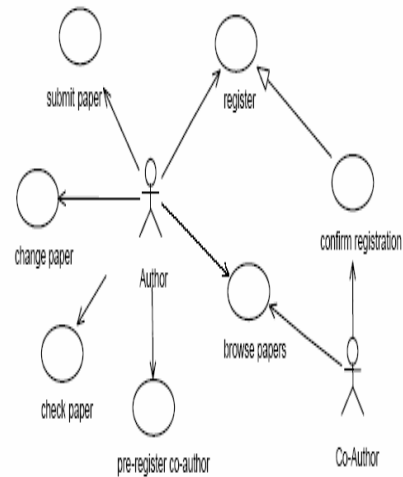


4.3.1 Caso Práctico

En este diagrama podemos observar como el autor puede llevar a cabo cambios, modificaciones y comprobaciones sobre los artículos.

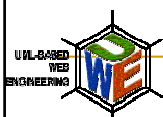
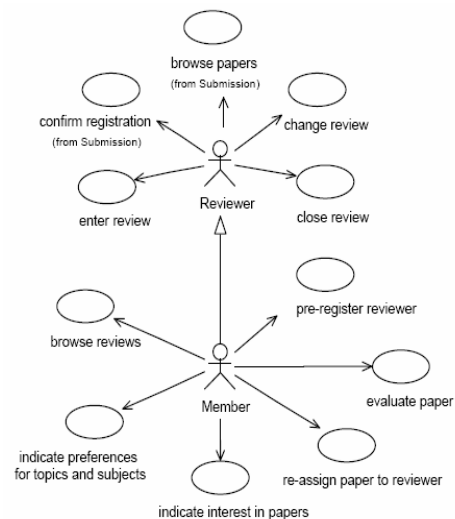
También puede registrarse a si mismo y a un coautor.

El coautor debe confirmar su registro, para ello hacemos una extensión de la clase registro.



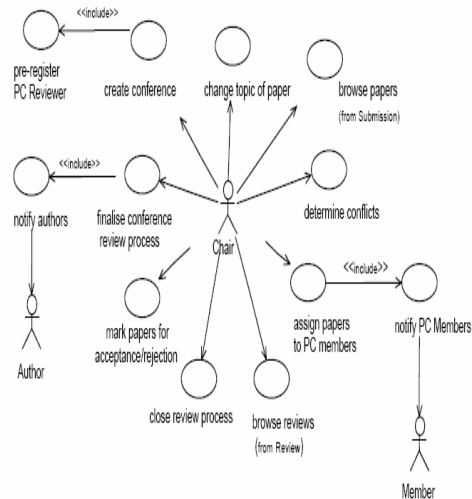
4.3.1 Caso Práctico

En este otro diagrama se muestra los funciones que va a llevar a cabo el revisor y los miembros del comité. También podemos ver la relación que se establece entre ellos dos.



4.3.1 Caso Práctico

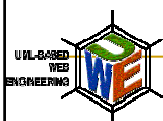
Por ultimo en este otro diagrama se nos muestra como el presidente gestiona todos los casos de uso incluidos en los paquetes.



4.4 Representación UML del Modelado Conceptual.

El diseño conceptual se basa en el análisis de requisitos del paso anterior. Esto incluye los objetos involucrados entre los usuarios y la aplicación.

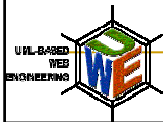
El modelo conceptual propone construir un modelo de clases con estos objetos, ignorándolos los aspectos de navegación: Presentación e Interacción, que serán tratados posteriormente. Los principales elementos de modelado son; las clases, asociaciones y paquetes.



4.4.1 Elementos de modelado.

El diseñador va seguir un proceso de modelado orientado a objetos basado en UML, los pasos que seguiremos son:

- ❑ Distinguir las clases, como la Conferencia, Usuario, Papel, y Revisión.
- ❑ Especificar los atributos más importantes y funcionamiento, por ejemplo el título, el ID, etc.,
- ❑ Determinar las asociaciones entre las clases.
- ❑ Agregar las clases e identificar la composición de estas, como entre la clase Conferencia y Temas y entre la clase Revisión y Evaluación,...
- ❑ Definir las jerarquías de herencia, por ejemplo la jerarquía de roles de usuario.
- ❑ Definir las restricciones de los métodos.

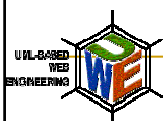


4.4.2 Caso Práctico.

Basándonos en la descripción textual y en los casos de uso basados en el análisis de requisitos del paso anterior nosotros identificamos los objetos, relaciones y operaciones necesarias para construir al modelo conceptual del sistema. Diferenciamos tres vistas del problema:

- ❑ La Vista de artículos: Representa el proceso que se va a llevar a cabo sobre un artículo.
- ❑ La Vista del Usuario: Representa las distintas formas de acceso a la aplicación Web que tienen cada usuario, ya sea presidente, miembro,...
- ❑ La Vista de la Sesión: Representa la forma de interacción entre los distintos roles, es decir el funcionamiento global de la aplicación (Relación entre los usuarios-aplicación, artículos-revisores...).

Ellos se representan mediante diagramas de clase UML.

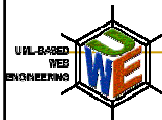


4.4.2 Caso Práctico.

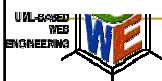
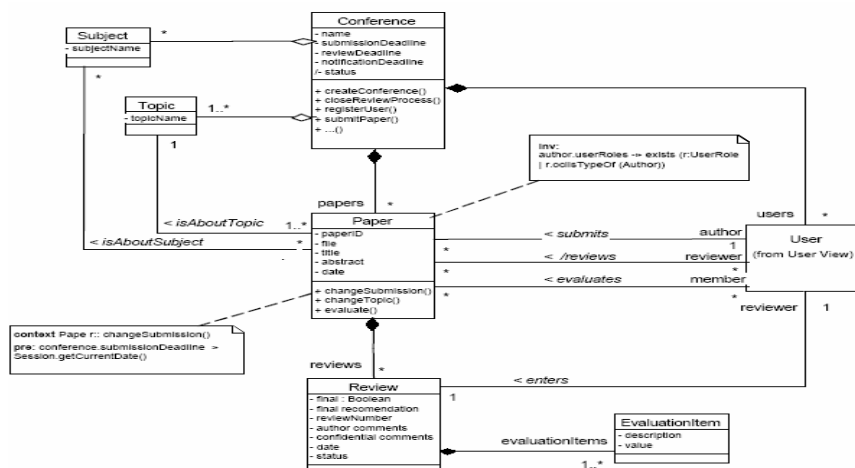
La clase conferencia contiene la información, sobre el nombre de la conferencia y las fechas tope de la conferencia. El estado de la conferencia es un atributo derivado cuyo valor cambia cuando se alcanzan fecha tope de inscripción, fecha tope de la revisión o fecha tope de la notificación.

Un artículo viene descrito por un título, un tema y la fecha en la que se envió. Tiene asociado un archivo donde se almacena el artículo y una identificación generada por el sistema. A cada artículo se le asignan un máximo de 3 revisores. Cada revisor se identifica por un número, si la revisión esta en proceso o ya ha terminado, comentarios del autor, información confidencial, fecha de la revisión.

Los autores tienen permiso para poder llevar la modificación de los artículos.

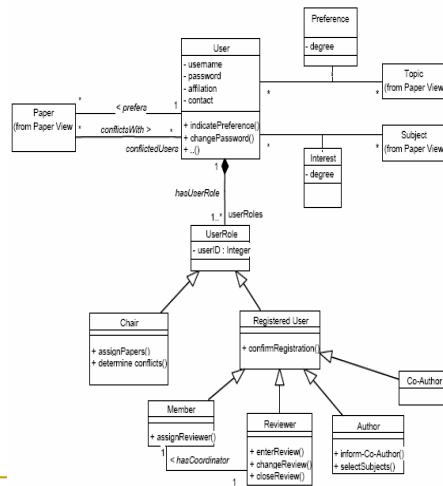


4.4.2 Caso Práctico.



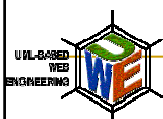
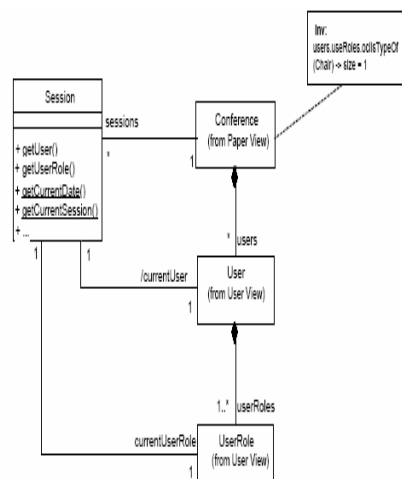
4.4.2 Caso Práctico.

Tenemos una clase principal usuario, que va a gestionar tipos de usuarios (presidente, miembro,...) y vamos a poderle asignar a cada tipo de usuario (los diferenciamos a través de su id.) sus privilegios sobre el trato de los artículos, fechas revision, trato sobre conferencias.



4.4.2 Caso Práctico.

En esta vista de sesión se muestra la relaciones entre la conferencias, las distintas sesiones y los distintos usuarios de las sesiones. Esta vista modela en tiempo de ejecución la información que es relevante para modelo conceptual



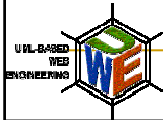
4.5 Diseño Navegacional.

El diseño navegacional no es solo útil para la generación de la documentación de la estructura de la aplicación sino que también permite mejorar la estructura de navegabilidad.

El modelo de la navegación comprende de:

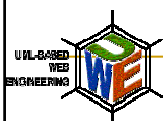
- El modelo de espacio de navegación que especifica **qué** objetos pueden ser visitados a través de la aplicación Web.
- El modelo de estructura de navegación que define **como** se alcanzan estos objetos a través de la Web.

En el proceso de construir el modelo espacial de navegación las decisiones del diseñador están basadas en el modelo conceptual y los requisitos de la aplicación definidos en el modo de caso de uso.



4.5.1 Modelado del Espacio Navegacional.

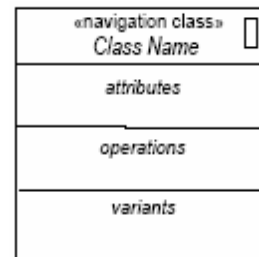
El modelo del espacio navegacional se construye con las clases de navegación y asociaciones entre las mismas y son representadas por un diagrama de clase en UML.



4.5.1.1 Elementos de modelado.

Se usan dos elementos de modelado para la construcción del modelo del espacio navegacional:

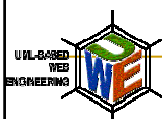
- Las clases de navegación : Es una clase cuyas instancias son visitadas por los usuarios durante la navegación. Se les dan el mismo nombre que a sus correspondientes clases conceptuales. Se representan a través de los estereotipos UML de la figura. Los atributos derivados del modelo conceptual no son incluidos en el modelo navegacional, estos serán denotados mediante / delante del nombre.



4.5.1.1 Elementos de modelado.

- La navegabilidad directa: las asociaciones entre el espacio navegacional representan la navegabilidad directa entre la clase de navegación inicial y la clase de navegación final

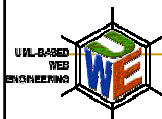
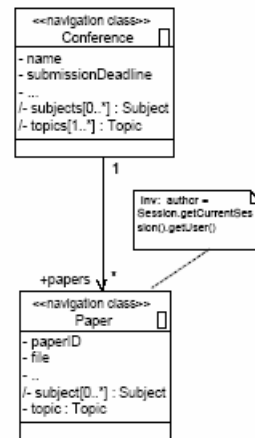
Para determinar las direcciones de la navegación usaremos flechas, en la que se indica el rol y su multiplicidad. Si falta un nombre por convenio se hace lo siguiente si la multiplicidad es menor que uno o igual el nombre de la clase de destino se usa como el nombre del rol; si es mayor que uno se usa el plural del nombre de la clase destino.



4.5.1.2 Caso Práctico.

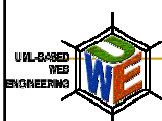
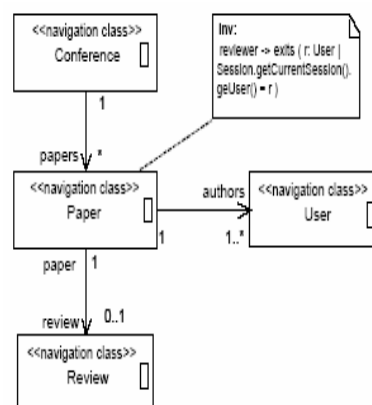
Cada actor, ya sea Miembro, Crítico... tiene una vista diferente del espacio de navegación. Estas vistas se representan como un modelo de clases UML construidos con las clases de navegación y estereotipos de navegabilidad directa.

- Espacio de navegación de un autor: Partiendo de la pagina inicial del sistemas, un autor solo podrá navegar por sus propios artículos y puede cambiar algunos datos informativos sobre el artículo antes de que se termine el plazo de envío



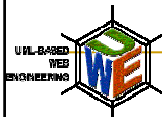
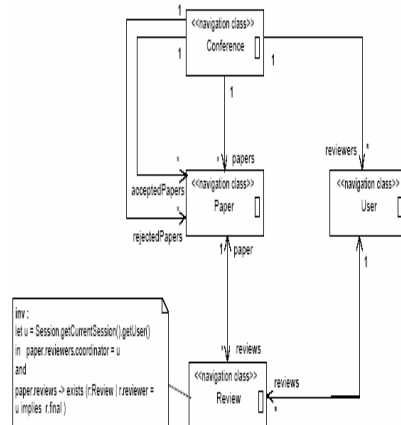
4.5.1.2 Caso Práctico.

- Espacio de navegación del revisor: Puede navegar a través de la pagina de inicio del sistema, revisar artículos, navegar sobre los comentarios de las revisiones y los objetos de navegación que ya conoce.



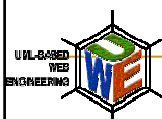
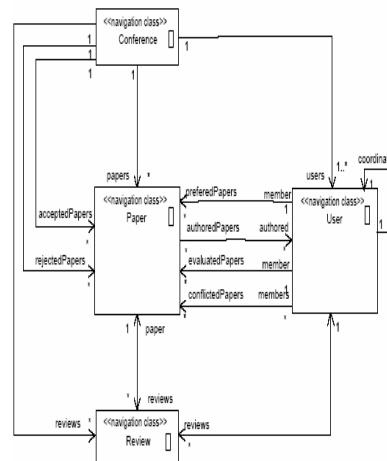
4.5.1.2 Caso Práctico.

- Espacio de navegación de los miembros del comité: Incluye la pagina de inicio del sistema, y todos los artículos asignados a cada miembro si los tiene junto a sus respectivas revisiones. Una vez finalizado el proceso de revisión pueden acceder a la lista de artículos aceptados y rechazados



4.5.1.2 Caso Práctico.

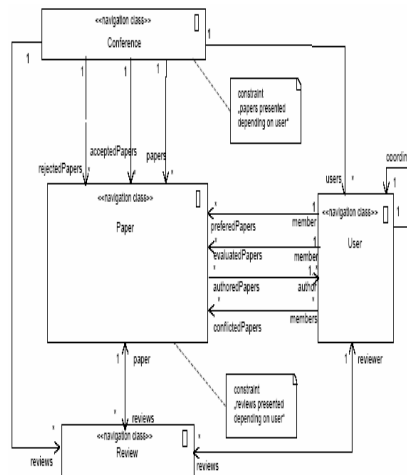
- Espacio de navegación del presidente: administra la información de las conferencias y los artículos, incluye la pagina de inicio, todos los artículos, las revisiones y la lista de usuarios del sistema.



4.5.1.2 Caso Práctico.

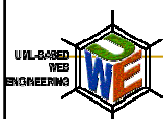
- Espacio de navegación global para los autores, revisores y miembros del comité:

En el vemos como las restricciones (OCL) se representan mediante una nota.



4.5.2 Estructura del Modelo Navegacional.

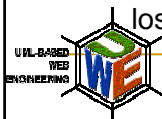
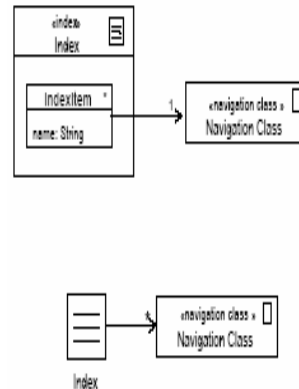
En el modelo de estructura de navegacional los elementos de acceso son los índices, consultas, y giras guiadas. Los caminos de la navegación junto con los elementos de acceso se representan por un modelo de clases que puede construirse en dos pasos: El primer paso consiste en reforzar el espacio de navegación mediante los índices, tutoriales y consultas. El segundo paso consiste en derivar los menús directamente del modelo.



4.5.2.1 Elementos de Modelado.

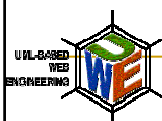
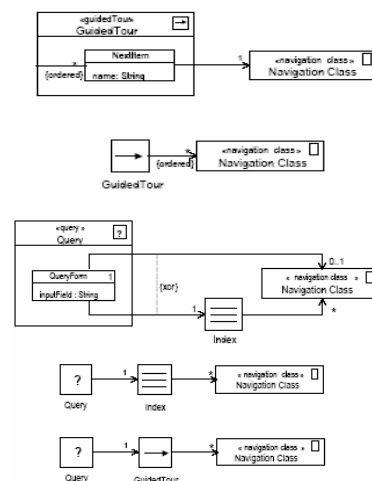
Los elementos modelado que usamos son los índices, consultas y tutoriales. Su los estereotipos y los iconos asociados son:

- Índices: Es un índice que permite el acceso directo a las instancias de las clases navegacionales. Se modelan por un objeto compuesto formado por un objeto, que que tiene un nombre que identifica cada instancia y posee un enlace a una instancia de una clase de navegación. Se modelan en base a los estereotipos UML.



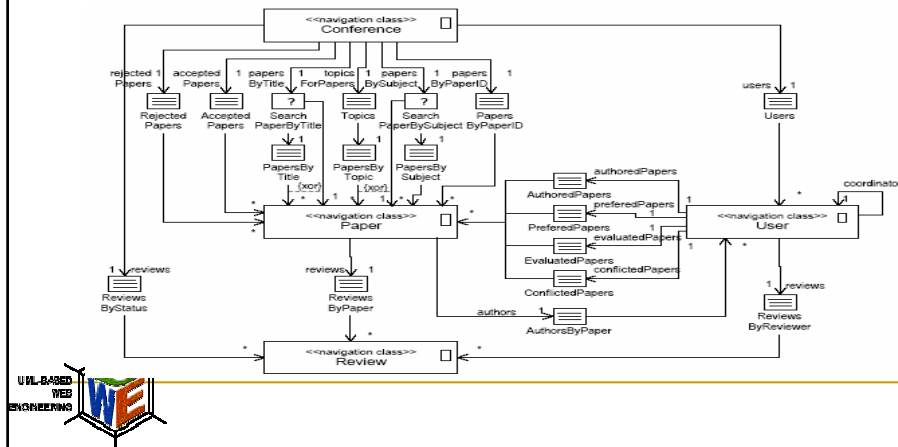
4.5.2.1 Elementos de Modelado.

- Vuelta Guiada: Proporciona el acceso secuencial a las instancias de una clase de navegación. Estas deben ser controladas por el usuario o por el sistema.
- Consultas: Se modelan mediante una clase con cadena de consulta como un atributo. Esta puede ser por ejemplo una operación de selección en OCL.



4.5.2.2 Caso Práctico.

Veamos ahora como el modelo del espacio navegacional puede ser construido mediante índices, vueltas guiadas y consultas.

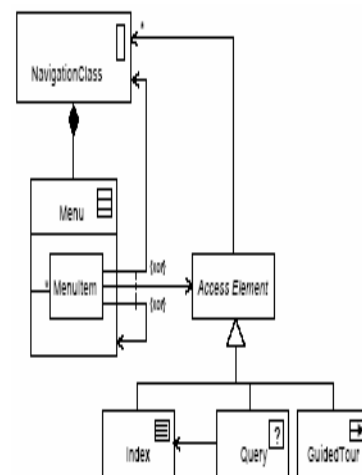


4.5.3 Añadir Menús: Pasos

El modelo del espacio navegacional es enriquecido por los elementos de acceso de menú siguiendo las siguientes reglas:

- Considerar que las asociaciones tienen como fuente una clase de la navegación.
- Asocio a cada clase de navegación del modelo anterior que tiene al menos una asociación saliente una clase menú.
- Reorganizar un menú en submenús.
- Introduzca para cada rol del extremo de una asociación dirigida del modelo anterior, su correspondiente elementos de menú.
- Cualquier asociación del modelo anterior que tiene ahora como su fuente una clase de la navegación.
- Se vuelve una asociación del rol del menú correspondiente introducido en el paso 4.

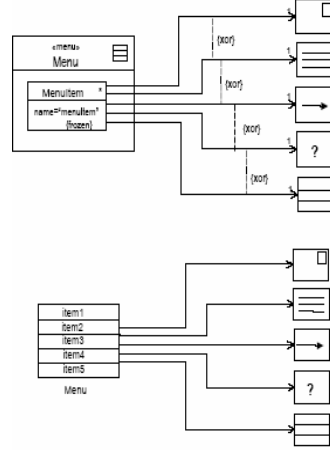
Note que pueden realizarse todos los pasos en el método anterior de una manera totalmente automática. Como resultado obtenemos un modelo de estructura de navegación de la aplicación.



4.5.3.1 Elementos de modelado.

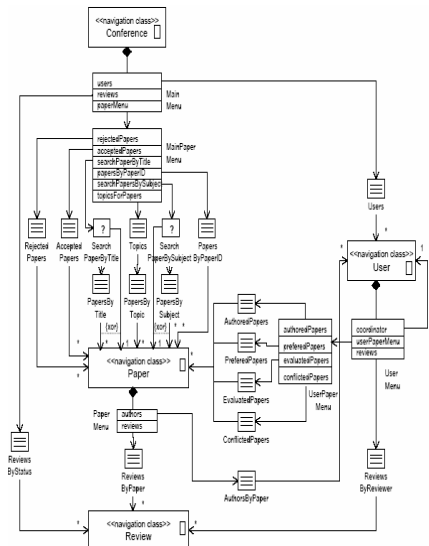
El elemento de modelado menú es acceso primitivo adicional que puede se añadido a la lista presentada anteriormente (consultas, índices y giras guiadas):

- Menú: Un menú es un índice de un conjunto heterogéneo de elementos, como un índices, giras guiada, consultas, u otro el menú. Este es el modelado por un objeto compuesto que contiene un número fijo de artículos del menú. Cada artículo del menú tiene un el nombre constante y posee un enlace al elemento navegacional de acceso.



4.5.3.2 Caso Práctico.

Ahora modelo de estructura de navegación de la etapa anterior del sistema se enriquece por los menús. Cada clase del menú es asociada con su correspondiente clase de la navegación. Nota que el nombre del rol coincide con los nombres de los índices correspondientes.



4.6 Flujo de Presentación y Storyboarding.

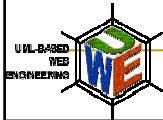
El diseño de presentación soporta la construcción de un modelo de presentación basado en el modelo de estructura de navegación y la información adicional recogida durante el análisis de requisitos. El modelo de la presentación consiste en un conjunto que muestran el contenido y la estructura de nodos

Se propone la construcción de bocetos, storyboards y un modelo de flujo de presentación.

Primero, el diseñador Web propone un boceto de cada vista de interfaz del usuario principal, es decir el plan de interfaces abstractas de usuario

En el segundo paso esta basado en el modelo del storyboard en el que el diseñador puede decidir si él quiere usar la técnica de ventanas múltiples y/o si él quiere usar frames.

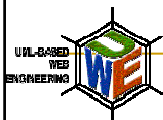
El objetivo del modelo de flujo de presentación es mostrar las vistas de interfaz de usuario del modelo storyboard que se le presentaran al usuario, es decir en que frame o ventana. También muestra como los contenidos son reemplazados cuando el usuario interactúa con el sistema.



4.6.1 Storyboards: Pasos.

Para el diseño de los storyboard (diagramas de cajas) empezamos con un primer modelo de navegación de la aplicación Web. Las reglas que nos van a guiar son:

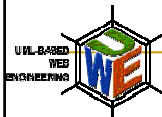
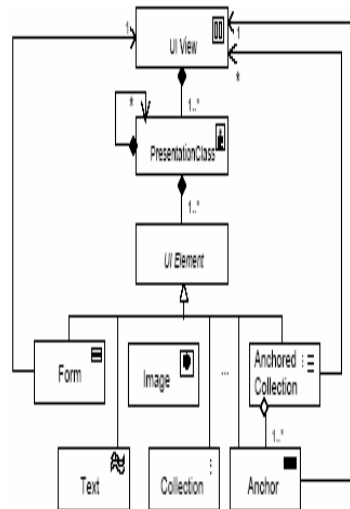
- ❑ Construir una clase presentación para cada clase de navegación que aparece en el modelo de estructura de navegación.
- ❑ Construir una clase de presentación para cada menú e índice que aparece en el modelo de estructura de navegación.
- ❑ Construir una clase de la presentación para cada consulta y vuelta guiada.
- ❑ Construir una clase presentación para apoyo de la navegación.
- ❑ Agregar los enlaces a las clases de la presentación permitir creación, destrucción y ejecución de operaciones sobre el modelo conceptual
- ❑ Determinar qué elementos de la presentación deben presentarse juntos al usuario (en la ventana).
- ❑ Agregar las restricciones OCL, si son necesarias
- ❑ Construir el escenario de storyboarding representados por las sucesiones de vistas de interfaz de usuario



4.6.1.1 Elementos de modelado.

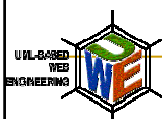
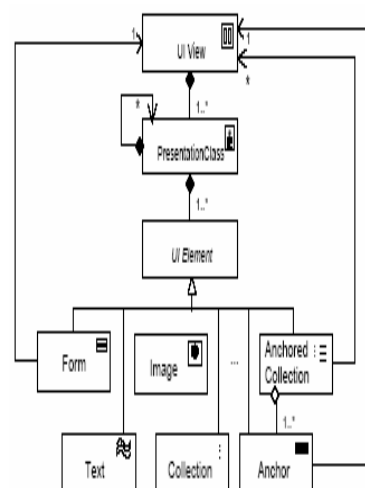
Para la construcción de los bocetos nosotros proponemos un conjunto elementos modelados algunos de los cuales son los siguientes:

- Vista de interfaz de usuario :
Especifica que cada instancia de esta clase es un contenedor de todos los elementos de interfaz de usuario abstractos que se presentan simultáneamente al usuario.



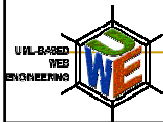
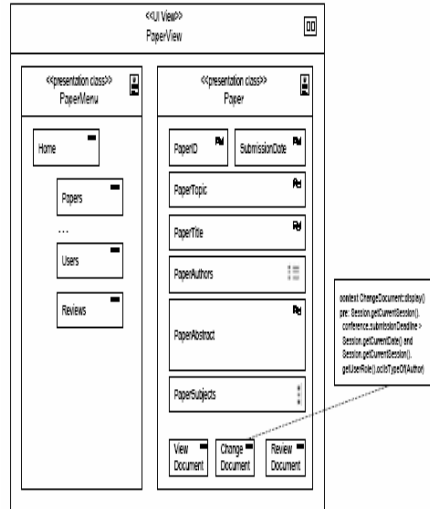
4.6.1.1 Elementos de modelado.

- La clase presentación: Es un estructura única la cual permite la división en vistas de interfaz de usuario, dentro de grupos de elementos de interfaz de usuario.
- Elementos de interfaz de usuario: Es un una clase abstracta la cual tiene elemento de interfaz de usuario describiendo los elementos particulares de la interfaz.



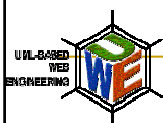
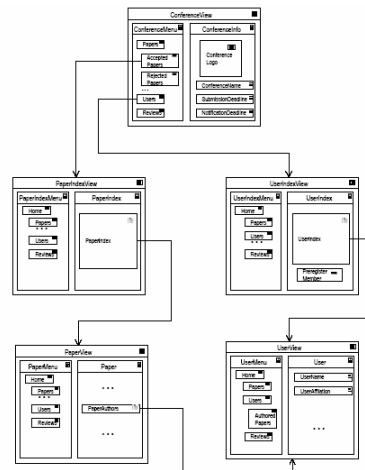
4.6.1.2 Caso Práctico.

Podemos ver una UI del sistema, esta está compuesta por la presentación de un artículo y un árbol de navegación construido en base al menú principal y el menú de artículos.



4.6.1.2 Caso Práctico.

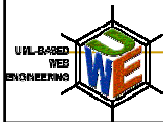
A continuación se muestra un escenario storyboard el cual muestra como el presidente encuentra la información sobre los artículos y sus autores.



4.6.2 Flujo de Presentación

Vamos a modelar el flujo de la presentación, mostrando “dónde” los objetos de navegación y elementos de acceso al usuario se presentarán, es decir, en qué frame o ventana el contenido se desplegará y qué enlaces se activarán.

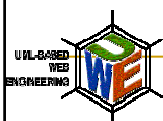
El diseñador tiene que especificar si usará una sola o múltiples ventanas, si se usarán los frames, y en ese caso, en cuántos framesets se dividirán.



4.6.2.1 Pasos a seguir.

El modelo de presentación requiere que el diseñador tome decisiones, sobre el número de ventanas que va a ser usado y numerar los frames. La construcción de la estructura de presentación no puede automatizada totalmente, pero hay ciertas pautas que el diseñador puede seguir:

- ❑ Escoger si usar una única ventana o múltiples ventas.
- ❑ Escoger el estilo del marco.
- ❑ Representar la estructura de presentación con un diagrama de clases UML.
- ❑ Definir que camino del modelo navegacional del diagrama de estructura de navegación que será empleado.
- ❑ Represente el usuario, las ventanas y/o objetos del marco en la dimensión horizontal.
- ❑ Especificar un mensaje display para cada objeto de la presentación que será posteriormente mostrado al usuario.
- ❑ Incluir un mensaje select para cada acción del usuario que selecciona una ancla o un botón.
- ❑ Especificar un mensaje fill y submit para cada acción del usuario en que consiste en suministrar datos en un formato de consulta
- ❑ Incluir un mensaje cada vez que se abra o cierre una ventana.
- ❑ Especificar el período de tiempo que una ventana o el frame es activo.

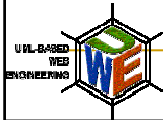


4.6.2.2 Elementos de modelado

El modelo del flujo de presentación de una aplicación Web se construye con las clases estereotipadas siguientes:

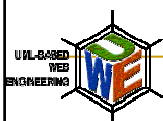
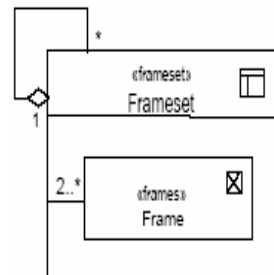
- *Ventana*: Una ventana es el área de la interfaz del usuario dónde se despliegan los objetos. Una ventana puede moverse, el maximizarse/ minimizarse, el redimensionarse, reducirla a un icono y/o cerrarla.

Para realizar estas acciones una ventana contiene los botones especiales. Además, las ventanas incluyen dos scrollbars: un horizontal y un scrollbar vertical de que permite visualizar el volumen entero.



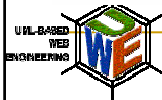
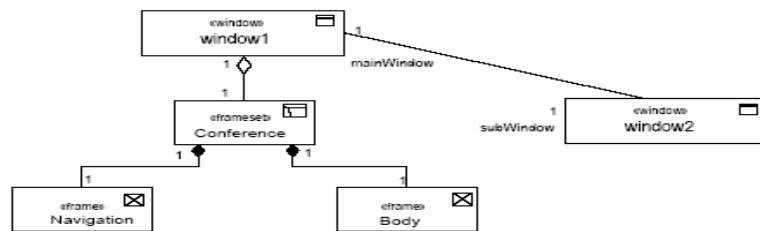
4.6.2.2 Elementos de modelado

- *Frameset (Pestañas)*: Un frameset es que un elemento modelado que es usado para definir múltiples áreas de visualización dentro de una ventana.
- *Frame*: Un frame siempre forma parte de un frameset, esta definido un área del correspondiente conjunto de marcos donde se muestra el contenido.



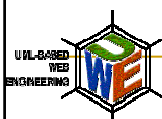
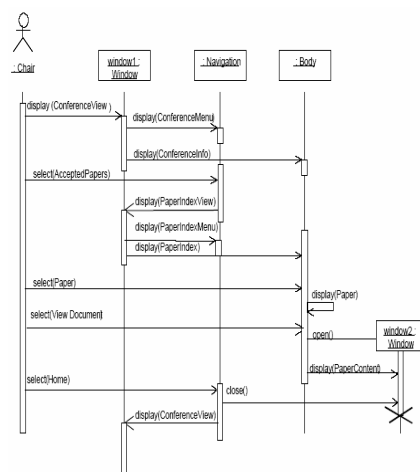
4.6.2.3 Caso Práctico.

Vamos a mostrar la ventana seleccionada y la estructura del framesets. Esta representación es importante cuando la aplicación posee muchas ventanas y diferentes framesets.



4.6.2.3 Caso Práctico.

A continuación mostramos un flujo de presentación que representa un escenario de una secuencia de posibles actividades de navegación que puede realizar el presidente de la conferencia.



5. Bibliografía.

- **The Authoring Process of the UML-based Web Engineering Approach** (Nora Koch 1,2, Andreas Kraus1, Rolf Hennicker1)
- **OO-H: Una extensión de los métodos OO para el modelado y generación automática de interfaces hipermediales.** (Cristina Cachero Castro).

