# 4. METODOLOGÍA WAE PARA EL DESARROLLO DE LAS APLICACIONES WEB PARA UML

### WAE (Extensión de Aplicaciones Web para UML)

En la explicación de lo conceptos Web se utilizará los diagramas e iconos utilizados por Jim Conallen. Ya que Conallen propone una extensión (o estereotipos) a UML para diseñar aplicaciones Web los cuales se describen a continuación:

### Estereotipos [4]

|  |  |
| --- | --- |
| Nombre | Página del servidor |
| Meta-modelo de clase | Clase |
| Descripción | Una página del servidor representa una página Web que tiene scripts que son ejecutadas por el servidor. Estos scripts actúan recíprocamente con recursos en el servidor (bancos de datos, lógica de negocio, sistemas externos, etc). Los funcionamientos del objeto representan las funciones en el script, y sus atributos representan las variables que son visibles en el alcance 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, VBScript, 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 browsers del cliente, y puede |

|  |  |
| --- | --- |
|  | contener scripts que son interpretadas por el browser.  Las páginas del cliente pueden estar asociaciones con otras páginas del cliente o del servidor. |
| Icono |  |
| Restricciones | Ninguna |
| Valores etiquetados | TitleTag – El título de la página como desplegado por el browser.  BaseTag – El URL de la base para dereferencing URLs relativo.  BodyTag – El juego de atributos para la etiqueta del  <body> que pone el texto de fondo y valor por defecto. |

|  |  |
| --- | --- |
| Nombre | Formulario (Form) |
| Meta-modelo de clase | Clase |
| Descripción | Una clase estereotipada como un «form» es una colección de campos de entrada que son parte de una página del cliente. Una clase del formulario se mapea directamente a HTML. Sus atributos representan los campos de la entrada del formulario de HTML (input boxes, text areas, radio buttons, check boxes, y los campos ocultos).  Un «form» se opera, desde que no pueden encapsularse su funcionamiento en un formulario. Cualquier funcionamiento que actúa recíprocamente con el formulario sería la propiedad de la página que contiene al formulario. |
| Icono |  |
| Restricciones | Ninguna |
| Valores etiquetados | Método - el 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 páginas Web.  La vista de los rectángulos son áreas divididas en rectángulos más pequeños de frames. Cada frame puede ser asociado mediante un solo nombre «target» aunque no necesariamente.  Los contenidos de un frame pueden estar en una página Web o en otro juego de frames.  Un esteriotipado de la clase de Frame se mapea directamente a una página Web, y el HTML enmarca etiqueta.  Un frameset es una «página del cliente», que puede tener funcionamientos y atributos también, pero éstos sólo son activados por browsers que no devuelve frames. |
| Icono |  |
| 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 con anchuras de columna delimitadas. |

|  |  |
| --- | --- |
| Nombre | Target |
| Meta-modelo de clase | Clase |
| 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 |  |
| 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 |  |
| Restricción | Ninguna |
| Valores etiquetados | Ninguno |

**Link:** un link es un indicador de una página del cliente a otra «Página». En un diagrama de la clase un link es una asociación entre una «página del cliente» y cualquiera otro

«página del cliente» o a una «página del servidor».

**Target link:** similar a un «link» la asociación, un «targeted link» es un link donde la página asociada se da en otro target. Esta asociación traza directamente a la HTML, con el target especificado por el atributo del target de la etiqueta.

**Submit:** un «submit» la asociación siempre es entre un «form» y una «página del servidor». Los Forms envían sus valores del campo al servidor a través de «página del

servidor» para procesar. El servidor del Web procesa la «página del servidor» que acepta y usa la información en la Form enviada.

Esta relación indica que página (o páginas) puede procesar el Form, y qué Forms una

«página del servidor» tiene un poco de conocimiento.

**Builds:** el «builds» la relación es una relación especial que conecta el vació entre el cliente y páginas del servidor. Las páginas del servidor sólo existen en el servidor. Ellos son acostumbrados a construir páginas del cliente.

El «build» la asociación identifica qué página del servidor es responsable para la creación de una página del cliente. Ésta es una relación direccional, desde que la página del cliente no contiene conocimiento de cómo entró en existencia.

Una página del servidor puede construir múltiples páginas del cliente, pero una página del cliente sólo puede ser construida a través de una página del servidor.

**Redirect:** un «redirect» la relación es una asociación unidireccional con otra página Web. Puede dirigirse los dos al cliente y páginas del servidor. Si la relación origina de un «página del servidor» entonces indica que el proceso de la demanda de la página puede continuar adelante con la otra página. Esto siempre indica esa página del destino participa en el construcción de una página del cliente. Esta relación en particular no es completamente estructural, desde la invocación real de un funcionamiento del redireccionamiento debe hacerse programáticamente en el código de la página originado.

Si la relación se origina de un «página del cliente» entonces esto indica que la página del destino será pedida automáticamente por el browser, sin la entrada del usuario. Un valor de retraso de tiempo puede ponerse que específica un retraso (en segundos) antes de la segunda página que se pide. Este uso de redireccionamiento corresponde a la etiqueta de META y HTTP-EQUIV valor de "Refresh".

### Modelado [3]

Modelar es muy importante, nos ayuda a manejar la complejidad. Un sistema puede representarse en diferentes formas, con modelos consistentes ya que cada modelo tiene un propósito específico y público. Al modelar es importante que se capture el nivel apropiado de abstracción y el modelo de los artefactos.

### Páginas [3]

El componente fundamental de una aplicación Web es la página. Browsers piden páginas (o las páginas conceptuales) de los servidores. Los servidores Web distribuyen páginas de información albrowsers. La composición y organización de una página Web en esencia está constituida por la interfaz de usuario para la aplicación.

### Servidor Scripting [3]

Es importante notar que la conexión entre el cliente y el servidor sólo existe durante una petición de la página. Una vez que la petición se cumple la conexión se rompe. Toda

actividad en el servidor (como efectuado por el usuario) ocurre durante la petición de la página. Esto representa una distinción muy significante entre las aplicaciones de servidor de cliente tradicionales. La lógica comercial en el servidor sólo es activada por la ejecución de scripts dentro de las páginas pedidas por el browser.

Dependiendo en el artefacto del scripting específico, las páginas escritas pueden contener al usuario que define variables, sub-rutinas y funciones. Algunos artefactos del scripting incluso permiten la definición e interacción de objetos.

El último resultado de este servidor a procesar es:

* + - 1. Ponga al día el estado comercial del servidor.
      2. Prepara un HTML estructurar página (interfaz del usuario) para el browser.

Una parte importante y sutil parte de la aplicación Web es entienda y acomodando de este paradigma de cliente y interacción del servidor. Los objetos comerciales no siempre son accesibles al manejar peticiones de interfaz de usuario individuales.

### Cliente Scripting [3]

El servidor no es el único componente en una aplicación Web que ejecuta scripts. El propio browser puede ejecutar código escrito en una página. Cuando el browser ejecuta un script, sin embargo, no tiene acceso directo a los recursos del servidor. Típicamente scripts que corren en el cliente aumentan la interfaz del usuario como oponerse a definir y llevar a cabo la esencia de la lógica comercial.

### Estereotipos de Páginas [3, 12]

Una mejor forma de modelar una página es separando en dos clases de estereotipos:

* + - 1. Páginas del Servidor
      2. Páginas del Cliente

Cualquier página Web dada en una aplicación Web que tiene funcionalidad en el servidor, así como de lado del cliente puede representarse en el modelo como dos clases separadas, aunque su implementación esté en el mismo archivo (o componente). En esta situación los métodos del servidor de una página Web y el alcance de variables de la página son todos contenidos en una clase en el modelo estereotipado «server page» («página del servidor»). Los métodos de esta clase representan el servidor de la página del lado del script; subrutinas y funciones. Una página del servidor puede tener relación a otros componentes que existen en el servidor.

La representación de las páginas del cliente, son semejantes en el diagrama de clases estereotipadas: «client page» («página del cliente»). Los atributos de página del cliente son los alcances de variables de la página y funciones que se ejecutan en el browser del cliente. Las páginas del cliente tienen métodos que se ejecutan solamente del lado del cliente, como por ejemplo, Java Applets y controles ActiveX, y elementos propios del DOM (Modelo de Objeto de Documento) es una forma de representar documentos estructurados (tales como una página Web HTML o un documento XML) que es independiente de cualquier lenguaje orientado a objetos.

Su finalidad es definir el conjunto de objetos que pueden componer documentos HTML (páginas Web) o XML, así como las estructuras que se definen dentro de él, sus propiedades y sus métodos, independientemente del lenguaje de programación utilizado, con el fin de evitar problemas de compatibilidad entre navegadores.

Hay una relación fundamental entre el servidor y el cliente que son estereotipos de una página Web. Una página del servidor finalmente establece página del cliente resultante.

Esta es una relación unidireccional, desde que una página de HTML completada tiene acceso pequeño a la interfaz del objeto de la página de servidor construida.

El estereotipo «builds» («construir») se aplica a las asociaciones y siempre es arrastrado en el modelo como una asociación unidireccional de una página del servidor a una página del cliente. Indica qué página del servidor es responsable para construir una página del cliente. Por ejemplo (Ver la figura 4.1):

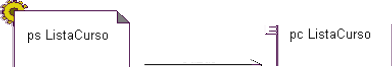


Figura 4.1 Las páginas del servidor construyen las páginas del cliente

Algunas páginas del servidor podrían redireccionar ciertas solicitudes de procesamiento a otras páginas servidoras (una especie de IF). Permitir modelar estas situaciones es útil para la reutilización. Para esto se utiliza el estereotipo <<redirects>>. Por ejemplo (Ver la figura 4.2):

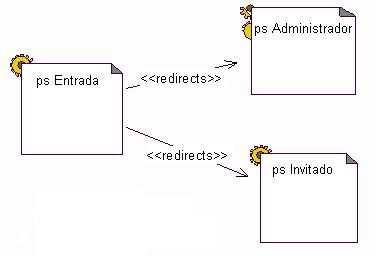


Figura 4.2 La Página del servidor es la que delega funcionalidad

Una parte fundamental entre la relación de las páginas del cliente y las páginas del servidor están en el diagrama de implementación. Los componentes en el diagrama de

implementación representan piezas distribuibles del sistema. Para estos estereotipos es la página Web. Un componente en un diagrama de implementación (vista del componente en Rational Rose) representa un archivo actual que es una demanda por el servidor Web, y qué comprende una página del servidor o página del cliente por lo menos. La figura 4.3, muestra conceptualmente esta relación.

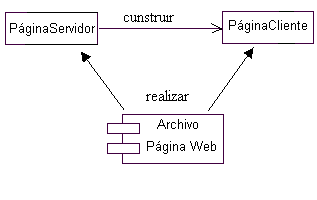


Figura 4.3 Unos componentes de página Web comprenden servidor y páginas del cliente

Una relación adicional que puede ser de importancia en el diseño de aplicación Web es el vinculo (link). Las páginas del cliente contienen a menudo vínculos (links o anchors) que unen a otras páginas Web. Estas otras páginas Web o pueden ser servidor o páginas del cliente desde que finalmente es el componente que es pedido por el browser del cliente.

Si el componente pedido comprende una página del servidor (a lo sumo uno) entonces la página del servidor se procesa para conseguir una página del cliente resultante para cumplir la demanda del browser.

El estereotipo: «links » se define para las asociaciones entre las páginas del cliente y otras páginas (servidor o cliente). Ver la figura 4.4:

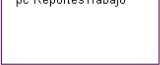


Figura 4.4 Página cliente vinculada a otras páginas clientes

La decisión de modelar todos links en páginas del cliente queda al diseñador, sin embargo, un bueno diseño debe modelar todos links relevantes, para la función de la aplicación. No puede ser necesario modelar hyperlinks a páginas Web fuera del sistema. La asociación de un «links» puede ser una asociación bi-direccional. La relación de un «links» en una página del servidor no tiene sentido. Si el link incluye parámetros, ellos son modelados como atributos del link fuera de la asociación, Ver la figura 4.5:

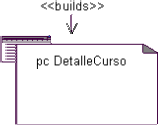
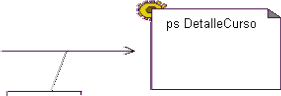
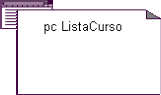


Figura 4.5 Uniéndose con parámetros

### Componentes [3]

Componentes en el sentido de interfaces disponible a los objetos en la aplicación Web como controles ActiveX y también DLLs, Java Applets o executables un estereotipo en la extensión Web. Simplemente con componentes de las páginas que identifican como ejecutarse en la máquina del servidor o en la máquina del cliente. Los estereotipos

«server component» (componente del servidor) y «client component» (componente del cliente) pueden ser aplicados a las clases en el diseño del modelo para distinguir disponibilidad. Con certeza un componente de acceso de banco de datos en el servidor no es directamente accesible por scritps del cliente que corren en un browser.

### Forms (Formularios) [3, 12]

Se definen estereotipos adicionales para separar y elaborar Form con el uso de HTML. Los formularios contienen atributos adicionales que no pueden ser apropiados en el contexto de la página del cliente. También es posible tener formularios múltiples en una sola página, cada targeting en una página de acción diferente. Esto puede ser planeado creando una nueva clase estereotipada para representar un solo formulario de HTML;

«form».

Una clase del formulario tiene como atributos los elementos del campo. Los métodos en una página del cliente tienen acceso a todos los atributos de los formularios contenidos dentro de una página. La relación apropiada entre una página del cliente y un formulario es su contenido. Las páginas del cliente contienen formularios.

Un formulario identifica una página Web específica (casi siempre con un estereotipo de página de servidor) aceptar y procesar datos sometidos con el formulario. Un «submits»

es el estereotipo de la asociación que representa la relación entre un formulario y la página Web que lo procesan, Ver la figura 4.6.

La asociación es bi-direccional desde que la página del proceso tiene acceso a los atributos del formulario que se someten cuando la asociación se comprende durante el runtime.

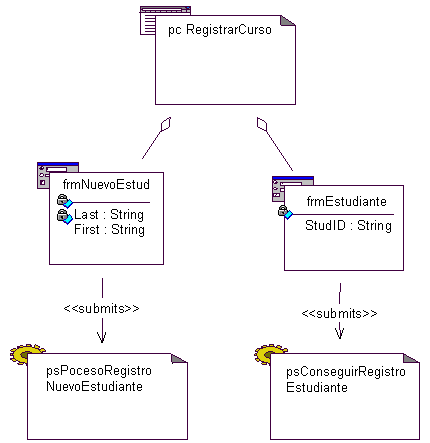


Figura 4.6 Los formularios hacen un submit a la página del servidor

### Framesets [3, 12]

Una interfaz de usuario adicional (y elemento del diseño) disponible en aplicaciones Web es el frame. Si se usa en una aplicación, representa una habilidad de presentar páginas Web múltiples al mismo tiempo. Típicamente estas páginas concurrentes que están juntas relacionadas representan una sola interfaz de usuario.

Los frames se implementan en HTML usando un frameset. Un frameset especifica y opcionalmente los nombres de los frames separados en los que pueden darse páginas Web. La implementación de un frameset está en una página HTML. Una página Web de frameset contiene normalmente estructura y contenido informativo que sólo se ve en el browsers másviejo. Esto nos lleva a modelar framesets como una página del cliente, pero uno especializado, y de un nuevo estereotipo: "frameset". En un modelo de diseño de clases estereotipar un frameset pueden tener todas las asociaciones que una página del cliente puede tener, con la comprensión que éstos son sólo apropiados para browsers más viejos.

Típicamente, los framesets contienen páginas del cliente múltiples. Cualquier página del cliente puede ser contenida por un frameset. ¡Puesto que un frameset es una especialización de una página del cliente, también puede contenerse en un frameset!

Actividad coordinada entre las páginas en frames (u otras ventanas) requiere la habilidad de referenciar páginas dentro de los frames.

El Target es el término usado cuando una referencia de página de cliente a otra página Web activa o frame. Desde que los targets representan un elemento muy diferente de un frameset, y considerando las páginas Web pueden también referenciar targets que están en otro browsers abierto, otro estereotipo de la clase se define; "target". Un target no tiene ninguna propiedad o atributos, es meramente un recipiente referenciable para una página del cliente. Una clase del frameset puede contener un target, o un target puede existir independientemente (como en el caso de una ventana del browser separada).

Las páginas Web contenidas en un frame se llaman targets. El estereotipo <<targeted link>> hace referencia a páginas que van a ser cargadas en un frame distinto del que contiene la página que tiene el link. Ver la figura 4.7.

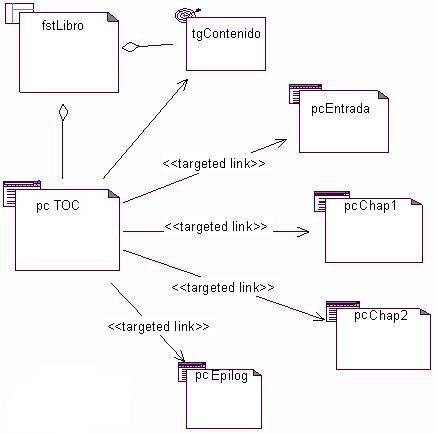


Figura 4.7 Usando framesets y targets.

# 5. PROTOTIPO DE UNA TIENDA VIRTUAL

### Descripción de la aplicación Web (Tienda Virtual) a modelar en UML

Dentro de los modelos de negocios que existen en el comercio electrónico se presenta la Tienda Virtual dentro del paradigma de Negocios a Clientes (b2c) [8].

El comercio electrónico es una metodología moderna en el proceso de comercialización, ayudada por la tecnología de punta como una nueva maniobra para el desarrollo de una mejor ventaja competitiva.

**Negocio a Cliente [8]:** el cliente puede comparar con la venta al detalle de manera electrónica. Esta categoría ha tenido gran aceptación y se ha ampliado sobre manera gracias al WWW, ya que existen diversos centros comerciales por todo Internet ofreciendo toda clase de bienes de consumo.

Lugar Comercial el cual funge la función de vender bienes y servicios, a través del Web, por lo cual está disponible las 24 horas al día, con un alcance global con la habilidad de relacionar y proporcionar información al cliente, así como órdenes de compra.

La Tienda Virtual en su naturaleza se muestra como un servicio dado a través de una entidad comercial o empresarial en los modelos del comercio electrónico es donde se presentan procesos definidos y capaces de ser automatizados. Y por si fuera poco es la unidad administrativa y elemental de los demás modelos de negocios del comercio electrónico [8].

Una Tienda Virtual va más allá de ser un almacén electrónico de los productos de ésta, representa una estrategia de negocio, pues las aplicaciones para una mercadotecnia en línea (marketing on-line) son innumerables, desde la generación de estadísticas de compra y venta, realización de análisis de comportamiento de mercado, análisis del cliente, de sus hábitos de consumo, así como la retroalimentación de los clientes y su autosuficiencia monetaria a través de publicidad externa la hace una excelente opción de desarrollo y sobre todo si se busca diseñar la aplicación que la gente, administre y presente.

Por lo anterior podemos definir una Tienda Virtual como el modelo de negocios de comercio electrónico que consta de aplicaciones de administración de servicios y procesos de mercadotecnia en línea con la función de vender bienes y servicios.

### Modelando la parte del usuario de la Tienda Virtual

El usuario es la persona que va a interactuar con la tienda virtual como lo describen los siguientes pasos:

* + 1. Ve las categorías disponibles en la tienda y sus productos.
    2. Elige los productos que desea.
    3. Agrega los productos al carrito de compras.
    4. Puede eliminar productos que no desee.
    5. Concreta la compra mediante la introducción de sus datos.
    6. Si no está inscrito se da de alta.
    7. Realiza el pago

Casos de Uso del Usuario:

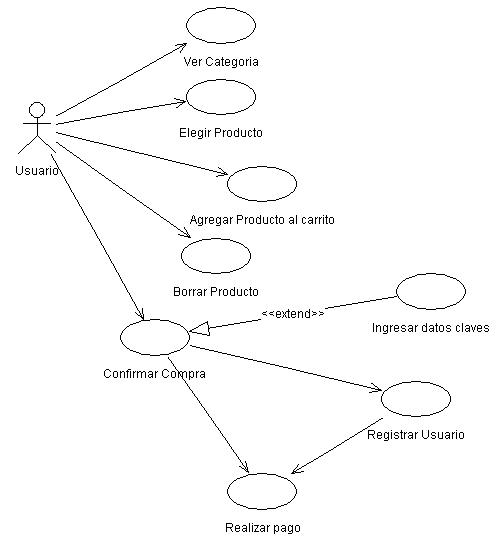


Figura 5.1 Caso de Uso del Usuario de la Tienda Virtual

Diagrama de Secuencia de Usuario:

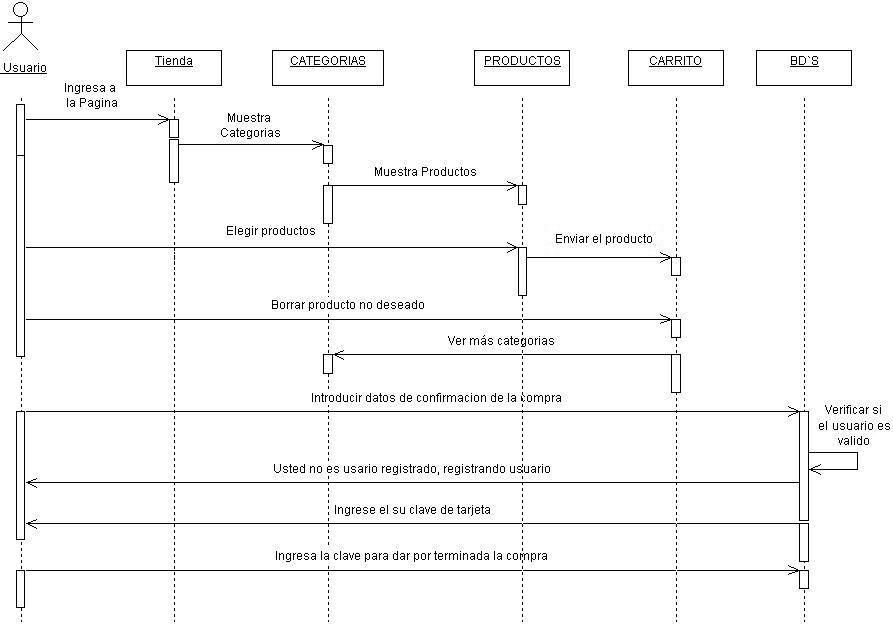


Figura 5.2 Diagrama de Secuencia del Usuario de la Tienda Virtual

Diagrama de Colaboración del Usuario:

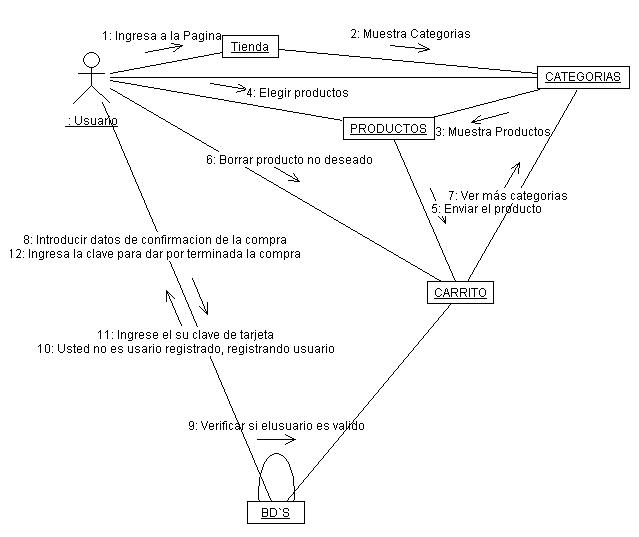


Figura 5.3 Diagrama de Colaboración del Usuario de la Tienda Virtual

Diagrama de los estereotipos del Usuario:

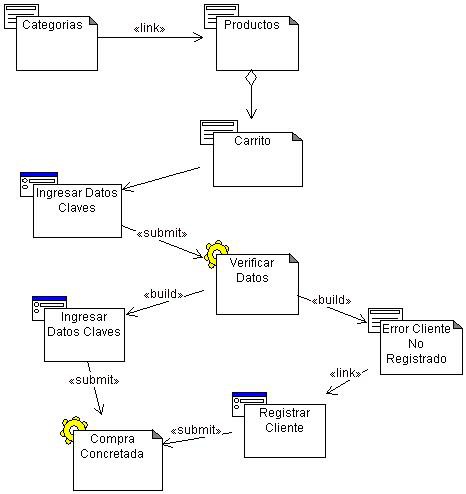


Figura 5.4 Diagrama de estereotipos del Usuario.

### Modelando la parte del administrador de la Tienda Virtual

El administrador es otro usuario, el cual se encargará de dar de alta, eliminar y modificar las categorías y productos como lo describen los siguientes puntos:

* + 1. Registra Categorías y sus productos
    2. Elimina Categorías y sus productos
    3. Modifica Categorías y sus productos

Diagrama de Caso de Usos del Administrador:

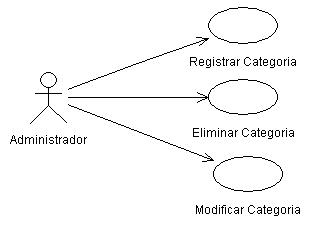


Figura 5.5 Caso de Uso del Administrador de la Tienda Virtual Diagrama de Secuencia del Administrador:

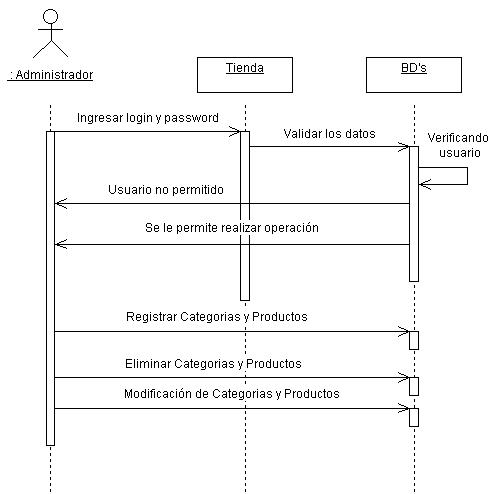


Figura 5.6 Diagrama de Secuencia del Administrador de la Tienda Virtual

Diagrama de Colaboración del Administrador:

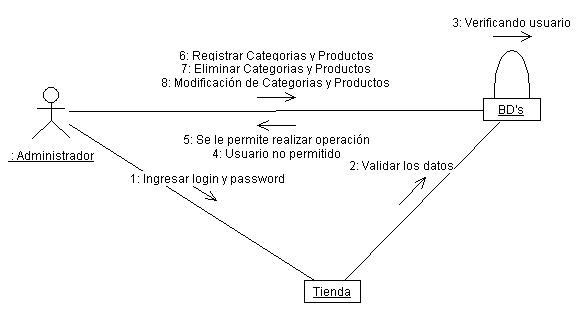


Figura 5.7 Diagrama de Colaboración del Administrador de la Tienda Virtual Diagrama de los estereotipos del Administrador:

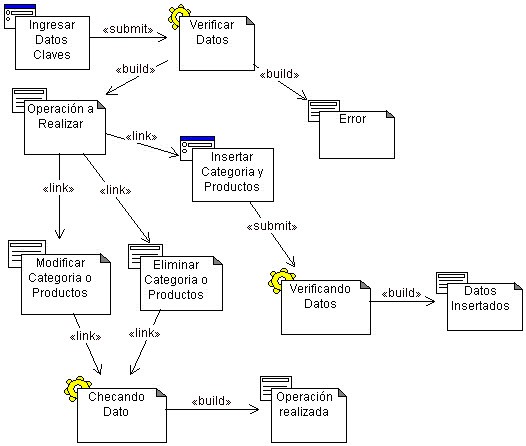


Figura 5.8 Diagrama de estereotipos del Administrador.

La figura 5.9 sería la presentación de la tienda virtual, en la cual podemos identificar en la parte azul, las siguientes ligas:

1. Inicio
2. Administrador
3. Confirmar la entrada

En la parte superior de color negro podemos visualizar las categorías que existen en la tienda virtual, las cuales también la podemos ver en parte central en la cual tiene el siguiente mensaje “Selección la categoría que desees:”

En la parte derecha de la pantalla podemos ver un producto el cual si se desea se puede compra dando click en “Agregar Artículo”.



Figura 5.9 Página principal

Al seleccionar alguna de las categorías que tiene la tienda virtual no envía a la página de productos (Ver figura 5.10) en la cual podemos seleccionar cualquiera de los productos y seleccionar la cantidad que se desee de ese artículo y dar click en “Agregar Artículo” y enviarlos al carrito de compras.



Figura 5.10 Página donde se visualizan los productos.

Cuando se envían los productos al carrito de compras podemos ver la figura 5.11 en la cual contiene todos los productos que el usuario ha deseado, el costo unitario de los productos y la suma de los productos la cual se presenta como un subtotal; si algún producto ya no se desea comprar existe la posibilidad de eliminar el producto y se reduce el subtotal. Si ya no se desea comprar ningún producto podemos dar click en “Carro Vació” el cual eliminará todos los productos y el carrito de compras quedará vacío; pero si el usuario quiere comprar los productos dar click en “Caja”.



Figura 5.11 Página del carrito de compras

Después de haber dado click en caja debemos confirmar la entrada para confirmar la compra de los productos, al seleccionar la primera opción es porque ya estamos registrados como usuarios en la tienda virtual, y debemos de ingresar los datos claves que nos pide los cuales son el E-mail y el Password (Ver figura 5.12). Sin el usuario no

se ha registrado debe de seleccionar la segunda opción para darse de alta para las dos opciones debe dar click en “Registrar”.



Figura 5.12 Confirmación de datos

La figura 5.13 muestra un formulario el cual debe llenarse para darse de alta como usuario de la tienda virtual y poder realizar compras de los productos, al dar click en continuar y si no existe algún error en el llenado del formulario podrá ir a la siguiente página.

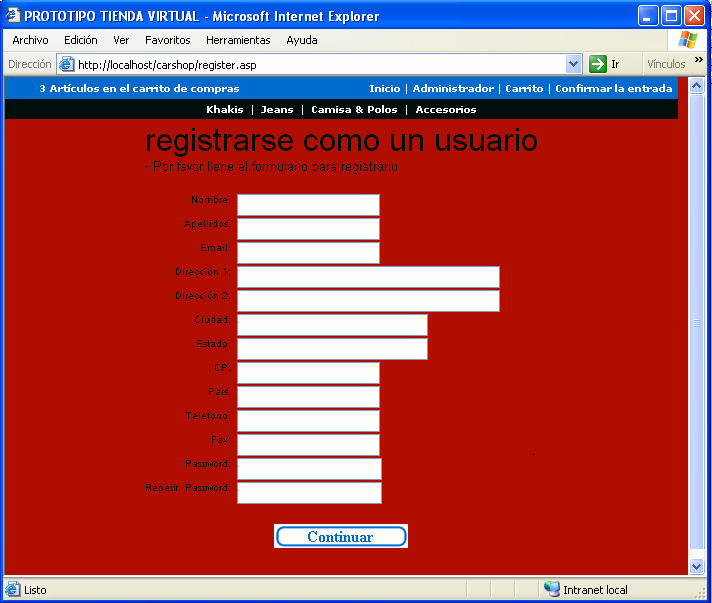


Figura 5.13 Página de registro de usuario

Después de haber sido registrado se concretará la compra, los datos de la siguiente página (ver figura 5.14) deben ser revisado por el cliente para dar por finalizada la compra.

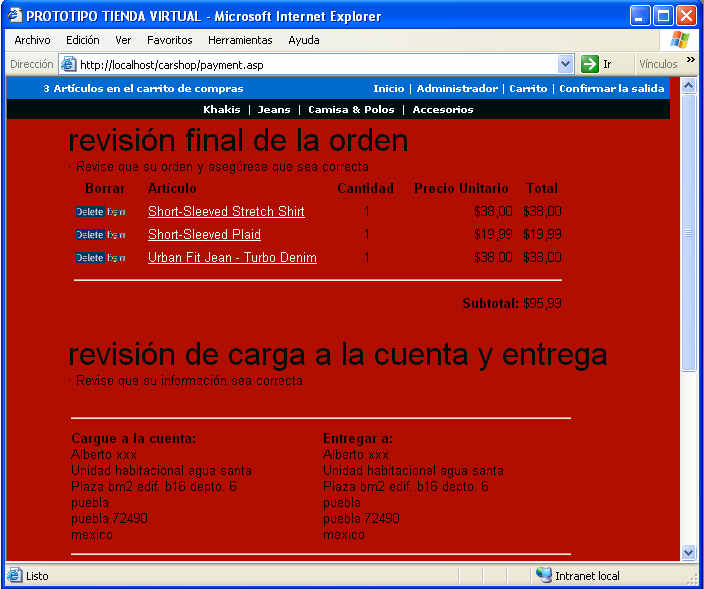


Figura 5.14 Finalización de la compra

La figura 5.15 de la página es exclusivamente para el administrador en la cual debe ser llenada correctamente, de no ser así no se le dará acceso para modificar la tienda virtual.

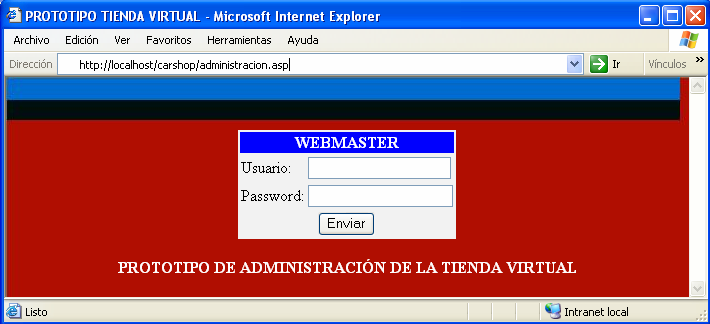


Figura 5.15 Administración

Después de haber ingresado los datos requeridos, no envía a la siguiente página (ver figura 5.16) desde la cual podemos elegir la operación que deseamos realizar (insertar, modificar o eliminar)

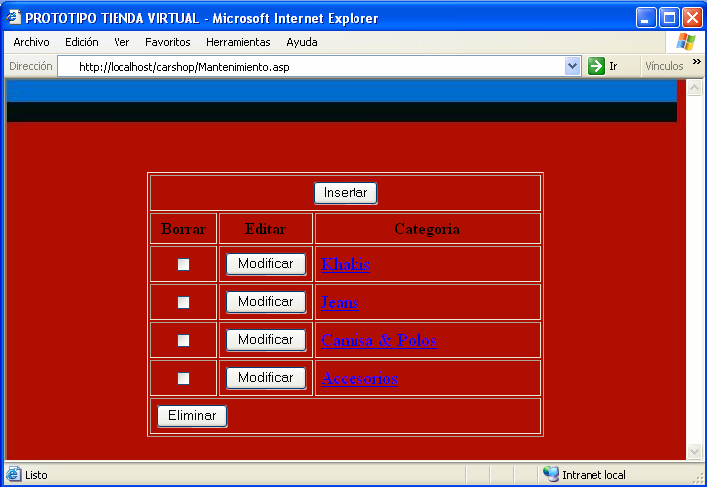


Figura 5.16 Mantenimiento de la tienda virtual

# 6. CONCLUSIONES Y PERSPECTIVAS.

## CONCLUSIONES

En este trabajo se hace una descripción de cada una de las metodologías utilizadas para realizar aplicaciones Web, haciendo un análisis de la Extensión de Aplicaciones Web para UML (WAE), es una de las metodologías más completas por el hecho de la claridad con la que cuenta sus estereotipos (clases) en el diseño de estas aplicaciones, además de los diagramas habituales de UML.

Es por eso que las aplicaciones Web como en los sistemas de información es necesario aplicar metodologías, esto es para realizar de una mejor manera y eficientemente éste tipo de aplicaciones. Es por ello que se pretende que se conozca y se utilice la metodología WAE dando como resultado aplicaciones de calidad.

El prototipo de la tienda virtual está realizado con la ayuda de la metodología WAE, en primer lugar, se definió lo que es una tienda virtual y en segundo lugar los diferentes usuarios que intervienen en el prototipo y como interactúan; esto es primordial porque no todos los usuarios interactúan de la misma forma.

La interacción del usuario de la tienda virtual es elección de la categoría que se muestran en la tienda y se eligen los productos, los cuales son llevados a un carrito de compra donde se contabiliza los productos y el costo que se debe de pagar.

La parte en la que el administrador interactúa con la tienda es la parte del mantenimiento (insertar, modificar o eliminar) de ésta.

El prototipo se realizó con los diagramas que son interactivos (diagrama de casos de usos, diagrama de secuencia y el diagrama de colaboración) dentro de UML y el diagrama de estereotipos de Jim Conallen, ya que como sabemos que las aplicaciones Web tienen que ser interactivas con los diferentes usuarios que consultan este tipo de aplicaciones.

El prototipo nos da una pauta para dar como satisfactorio el objetivo general de ésta tesis el cual fue: investigar la manera en la que UML puede ser aplicado al desarrollo de aplicaciones Web.

## PERSPECTIVAS

La perspectiva del prototipo es realizar la parte del pago de la compra, el envió del pedido y conexión que se debe realizar con los bancos, la cual quedaría de la siguiente forma:

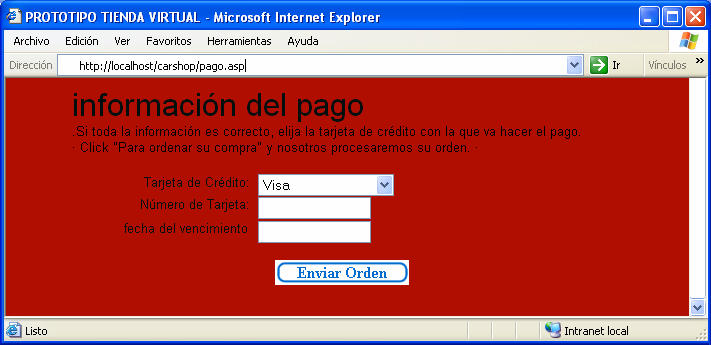


Figura 6.1 Pago de la orden.

UML ha sido desarrollado para responder a los requerimientos del desarrollo de aplicaciones orientadas a objetos tradicionales. Las aplicaciones Web presentan un cierto número de particularidades que demandan ciertas adaptaciones de UML con la finalidad de modelar la arquitectura.

La perspectiva de la tesis es desarrollar una metodología para adaptar UML al desarrollo de aplicaciones Web, con la colaboración de metodologías ya existentes.