

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA



TESIS DE GRADO
USABILIDAD EN APLICACIONES WEB 2.0

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS

Postulante : Univ. Elizabeth Sofia Aróstegui Huanca

Tutor : Lic. Roberto Vargas Blacutt

Revisor : M. Sc. René Casilla Gutierrez

LA PAZ - BOLIVIA

2010

*A mis amados padres Alberto Aróstegui Mamani
y Eulogia Huanca Vargas que han apoyado constantemente
mis sueños y metas profesionales con sacrificio y dedicación
Papá, Mamá muchas gracias!*

AGRADECIMIENTOS

Ante todo quiero agradecer al Padre Celestial que guía mis pasos en todo momento y llena mi existencia de bendiciones.

Quiero agradecer a los docentes:

M. Sc. René Casilla Gutierrez, por el interés, guía y comprensión que ha mostrado desde el inicio de esta investigación y fue crucial para su emprendimiento. Agradezco sus buenos consejos en aquellos momentos de incertidumbre.

Lic. Roberto Vargas Blacutt docente ejemplar, por las sugerencias, confianza y apoyo prestados en la conclusión de este trabajo. Sin su sincero apoyo este trabajo no se hubiera llevado a cabo.

Quiero agradecer al amor de mi vida Nelson por el apoyo y confianza durante esta etapa crucial en mi vida.

eliz.infart@gmail.com

RESUMEN

La usabilidad es un atributo que muestra la calidad de una aplicación con respecto al fácil aprendizaje del usuario. Web 2.0 es una forma de diseñar aplicaciones web basada en la participación y acción del usuario. Ambos conceptos se interrelacionan en un solo aspecto importante: el usuario. Los actuales métodos de modelado de datos no ofrecen una perspectiva cómoda con respecto al modelado de la interfaz, que es la parte del software que tiene relación directa con él. Diseñadores y desarrolladores trabajan de forma separada y toman a la usabilidad como una característica de calidad secundaria que puede ajustarse al final. Por este motivo, el presente trabajo busca incorporar aquellas guías y estándares en un modelo completo que no solo ofrezca sugerencias sino métodos y artefactos concretos ajustables a una metodología para el modelado de datos que el proyecto necesite. Pilon es un modelo de usabilidad que se acomoda a la fase de diseño de cualquier metodología y posee cinco fases fundamentales en la construcción de cualquier aplicación web. Cada fase ofrece un formato de registro que es el producto obtenido y sirven de comunicación entre los distintos actores del sistema.

ÍNDICE GENERAL

CAPÍTULO I GENERALIDADES

1.1	Introducción.....	1
1.2	Antecedentes.....	3
1.3	Planteamiento del Problema... ..	5
1.4	Hipótesis.....	6
1.5	Objetivos.....	8
1.5.1	Objetivo General.....	8
1.5.2	Objetivos Específicos.....	8
1.6	Justificación.....	8
1.7	Alcances.....	9

CAPÍTULO II MARCO TEÓRICO

2.1.	Usabilidad.....	11
2.2.	Web 2.0.....	13
2.3.	Metodología de desarrollo UML basado en Ingeniería Web – UWE.....	17
2.4.	Métodos de Evaluación de Usabilidad.....	29
2.5.	Metodología de Evaluación de Calidad de Sitios Web – Web-site QEM.....	30

CAPÍTULO III MODELO PILÓN

3.1	Introducción.....	32
3.2	Fases del Modelo.....	33
3.2.1	Análisis.....	36
3.2.2	Diagramación.....	43
3.2.3	Diseño.....	52
3.2.4	Visualización.....	75
3.3	Construcción del Prototipo.....	83
3.3.1	Descripción.....	83
3.3.2	Plataforma tecnológica.....	83
3.3.3	Aplicación del modelo.....	84
3.4	Prueba de Hipótesis.....	89

CAPÍTULO IV CONCLUSIONES

4.1 Conclusiones.....	98
4.2 Recomendaciones.....	99

ANEXOS

ANEXO A: Árbol de Problemas

ANEXO B: Árbol de Objetivos

ANEXO C: Glosario

ANEXO D: Conceptos fundamentales del modelo Pilon

ANEXO E: Características de un logo

ANEXO F: Ficha Técnica de Elementos del Armazón Web

ANEXO G: Ficha Técnica de Estructuras del Armazón Web

ANEXO H: Plan de Pruebas

ANEXO I: Clasificación de la jerarquía visual

ANEXO J: Hoja de evaluación

ANEXO K: Análisis de la Marca

ANEXO L: Plantillas de Composición

ANEXO M: Armazones Web

ANEXO N: Guía de estilo visual

ANEXO O: Diagramas UWE



ÍNDICE DE FIGURAS

1.1	Crecimiento de los Sitios Web.....	3
2.1	Proceso de UWE para una versión.....	18
2.2	Iteraciones de los Flujos de Trabajo.....	19
2.3	Elementos de un Modelo de Casos de Uso	20
2.4	Modelo de Casos de Uso.....	21
2.5	Clase con variantes de compartimiento adicionales	22
2.6	Modelo Conceptual	22
2.7	Modelo de Usuario	23
2.8	Clase Navegación	24
2.9	Nodo externo	24
2.10	Modelo de Espacio de Navegación	25
2.11	a) Clase Índice y b) su notación abreviada	25
2.12	a) Clase Visita Guiada y b) su notación abreviada	26
2.13	a) Clase Consulta y b) sus notaciones abreviadas	26
2.14	a) Clase Menú y b) su abreviación.....	27
3.1	Fases del Modelo Pilón	33
3.2	Sub-fases de la fase de Diseño	34
3.3	Ejemplo de sitios web con marca definida y sus características	39
3.4	Eslogan de la aplicación <i>Burstcast</i>	40
3.5	Partes de la rejilla de Diagramación	44
3.6	Composición dividida en: a) Espacios uniformes y b) No uniformes.....	45
3.8	Paso 1 de la creación de la rejilla de diagramación	47
3.9	Paso 2 de la creación de la rejilla de diagramación	47
3.10	Paso 3 de la creación de la rejilla de diagramación	48
3.11	Paso 4 de la creación de la rejilla de diagramación	48
3.12	Paso 5 de la creación de la rejilla de diagramación	48
3.13	Rejilla de diagramación de ejemplo	49
3.14	Ejemplos de diagramación para una página principal	50
3.15	Delimitación de espacios en un armazón web	51
3.16	Numeración de espacios en un armazón web	51
3.17	Ejemplos de numeración para una página principal	52

3.18	Clasificación de las estructuras de contenido personalizado	54
3.19	Ejemplo de formulario de la página <i>google.com</i>	55
3.20	Ejemplo de catalogo de la página <i>apple.com</i>	56
3.21	Ejemplo de panel de la página <i>flickr.com</i>	56
3.22	Estereotipo de la jerarquía visual	58
3.23	Clasificación de la Navegación del armazón	59
3.24	Navegación estructural de la página <i>apple.com</i>	60
3.25	Navegación adaptativa de la página <i>apple.com</i>	63
3.26	Navegación asociativa de la página <i>apple.com</i>	64
3.27	Ejemplo de etiquetado de objetos del armazón	67
3.28	Criterio de numeración para el etiquetado de objetos del armazón	67
3.29	Modelo de descripción de objetos del armazón	68
3.30	Ejemplo de descripción de objetos del armazón	68
3.31	Muestra de la estructura del código conciso	75
3.32	Página Principal del Prototipo HotPlanner	88
3.33	Página de tareas del Prototipo HotPlanner	88
3.34	Página de contactos del Prototipo HotPlanner	89
3.35	Árbol de Requerimientos de Calidad especificando la Usabilidad	89
3.36	Interpretación geométrica de la función CDG	93
3.37	Indicadores elementales de subcaracterísticas	94
3.38	Página principal de la aplicación <i>Remember the milk</i>	95
3.39	Indicadores elementales de subcaracterísticas	97

ÍNDICE DE TABLAS

1.1	Comparación entre aplicaciones Web 1.0 y Web 2.0.....	6
1.2	Descripción de variables independientes.....	7
2.1	Descripción de las diferencias entre la Web 1.0, Web 1.5 y Web 2.0.....	15
3.1	Ejemplo de usuarios reales: estudiantes, niños y mujeres.....	35
3.2	Ejemplo de registro de usuarios reales: mujeres, niños y estudiantes	38
3.3	Ejemplo de registro de atributos y servicios de la marca	40
3.4	Ejemplo de registro de la selección del nombre de la aplicación	41
3.5	Ejemplo de registro de colores de la marca	41
3.6	Ejemplo de Registro del Logo de la Marca	42
3.7	Dimensiones aproximadas para las partes de la rejilla	47
3.8	Formato de texto del almacén	53
3.9	Objetos de contenido central y secundario	54
3.10	Designación de objetos de contenido del almacén	57
3.11	Ejemplo de especificación de etiquetas para opciones de navegación.....	61
3.12	Ejemplo de jerarquía para las opciones de navegación	62
3.13	Designación de estructuras de navegación	67
3.14	Lista de eventos	69
3.15	Tipo de operaciones de la aplicación	70
3.16	Tipos de cuadro de diálogo de la aplicación	70
3.17	Tipos de objetos de colaboración de la aplicación.....	71
3.18	Relación entre operaciones y cuadros de dialogo de la aplicación.....	72
3.19	Tipos de zonas de carga	72
3.20	Relación entre zonas de carga y operaciones	73
3.21	Relación entre zonas de carga y cuadros de dialogo	74
3.22	Relación entre zonas de carga y objetos de colaboración	74
3.23	Ejemplo de registro de la paleta cromática	77
3.24	Ejemplo de registro de los elementos gráficos de la aplicación	78
3.25	Ejemplo de registro de los estilos de tipografía de la aplicación.....	80
3.26	Ejemplo de registro de la tipografía de la marca de la aplicación	81
3.27	Ejemplo de registro de los estilos de los vínculos de la aplicación	82

3.28	Ejemplo de registro de la paleta cromática	83
3.29	Plataforma tecnológica del Prototipo	83
3.30	Dimensiones aproximadas para las partes de la rejilla	85
3.31	Estructuración de un formulario	86
3.32	Estructuración de objetos de un formulario.....	86
3.33	Requerimientos de calidad incluidos los criterios elementales	90
3.34	Valores de las muestras obtenidas del test de usabilidad	91
3.35	Obtención de indicadores de preferencia elemental	92
3.36	Cálculo de indicadores de preferencia global por subcaracterística	93
3.37	Obtención de indicadores de preferencia elemental	95
3.38	Cálculo de indicadores de preferencia global por subcaracterística	96





CAPÍTULO I

GENERALIDADES

“Un camino de mil kilómetros comienza con un primer paso.”

Lao-Tsé

1.1 Introducción

Con el masivo avance de las tecnologías web han surgido diversos puntos de vista acerca de la llegada o no de una nueva etapa en su evolución. Algunos autores la denominan web 2.0 y la definen como aquellas aplicaciones que obtienen su valor de las acciones y participación de los usuarios.

Aunque el término es ampliamente utilizado, su denominación es quizás menos importante que los proyectos envueltos bajo esas características conceptuales y técnicas. El mercado de ideas novedosas rápidamente se torna obsoleto y la web 2.0 es un emergente histórico relacionado al darwinismo digital¹, más que un momento especial y diferente de la evolución de las aplicaciones web.

Sin embargo no se debe dejar de lado el gran impacto que este concepto ha despertado en los usuarios nativos de la *World Wide Web*, que encuentran en el nuevo paradigma un medio de conocer y darse a conocer al mundo.

Hasta ahora, al llegar a una página la mayoría de los usuarios saben cómo interactuar: si ven un enlace saben que es para ir a otra página, entienden que deben pulsar un botón para realizar una acción. Las nuevas aplicaciones poseen una interacción mucho mayor, sin embargo los usuarios no tienen un modelo mental claro

¹ Darwinismo Digital es un principio elaborado por Evan Schwartz en su libro *Digital Darwinism. 7 breakthrough business strategies for surviving in the cutthroat web economy* (1999. Broadway Books, New York) que explica que en el mercado de aplicaciones Web sólo sobreviven las más aptas en función de su capacidad de adaptación a las demandas del medio, ya que nacen muchos más individuos (o productos en este caso) que los que la especie y el entorno pueden solventar, como planteó Darwin en su Teoría de la Evolución de 1859.

de su funcionamiento, se les introduce novedades que les suelen generar problemas. No son intuitivas y no siguen las convenciones actuales, por lo que no está claro qué deben hacer los usuarios para interactuar, lo que les exige un proceso de aprendizaje, que puede ser difícil para algunos.

Un ejemplo claro del mal uso de estas tecnologías puede observarse en algunas empresas orientadas a la venta electrónica de productos. Es mortal para los sitios de *e_commerce* cuando los usuarios no pueden operar el carro de compras, por eso usualmente lo mejor es apegarse a planes de *shopping-cart* simples que todos entienden [Nielsen, 2007]. Es en este tipo de sitios donde podría hacerse uso de tecnologías Web 1.0 sencillas y con mejores resultados.

Según Jakob Nielsen [2007], considerado el principal exponente de investigación en el campo de la usabilidad, la web 2.0 puede ser considerada peligrosa porque estas tecnologías generan más complejidad de la que pueden suprimir.

Según O'Reilly, principal promotor de esta noción, los principios constitutivos de ésta son siete y presentan a las personas como principales protagonistas de esta etapa.

Siendo el principal objetivo de esta tendencia los usuarios, es importante encarar el desarrollo de una aplicación desde ese punto, es por eso que el presente trabajo plantea el desarrollo de un modelo que apoye en el análisis y diseño de aplicaciones web 2.0 usables debido a que la mayor parte de las organizaciones empresariales e incluso diseñadores independientes están desarrollando aplicaciones muy atractivas pero con serios problemas de diseño usable, lo que repercute en pérdida de usuarios por fallas en la navegabilidad o contenido tedioso y confuso, al no contar con un modelo de construcción específica que sirva de orientación.

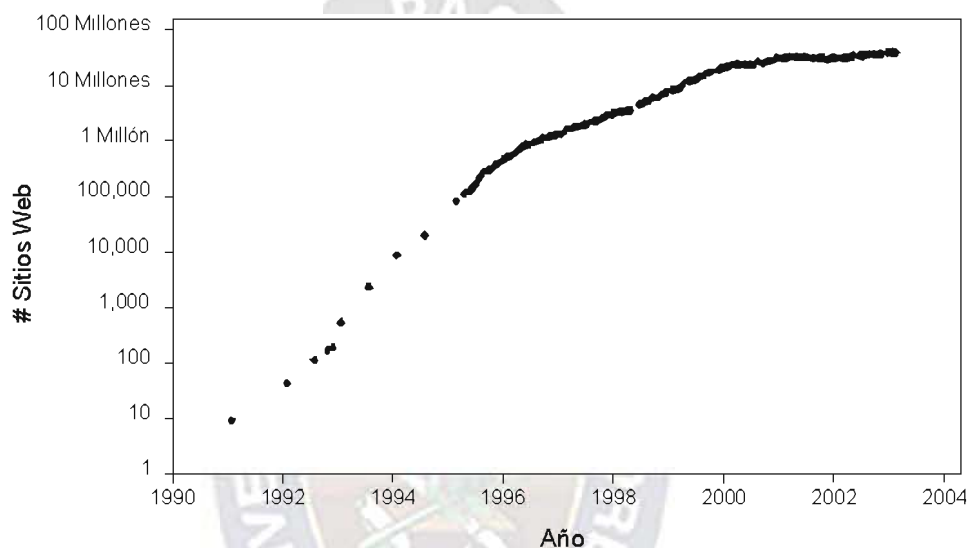
Cada fase del modelo ofrece un artefacto que sirve de comunicación entre los diferentes trabajadores de la aplicación y pueden ser comprendidos fácilmente por ellos, además de proporcionar un registro permanente que ofrezca comunicación, orden y uniformidad en la etapa de implementación.

1.2 Antecedentes

Del trabajo

En 10 años el número de sitios web ha crecido dramáticamente de 100 alrededor de 45 millones (Ver Figura 1.1).

Figura 1.1: Crecimiento de los Sitios Web



Fuente: [Murugesan & Ginine, 2005]

Antes de la web 2.0, internet estuvo marcado por dos etapas:

- La web 1.0 corresponde hasta finales de los años 1990 con la instalación de las redes y a la adopción de los estándares de la comunicación;
- La web 1.5 marca la llegada de un Internet comercial, principios del año 2000.

El término web 2.0 se acuñó por Dale Dougherty en la sesión de *brainstorming* con Tim O'Reilly en una conferencia el 2004. Aunque este término no se había definido perfectamente, la influencia que tubo fue significativa y hacía pensar en la próxima generación de aplicaciones de web [Stauffer, 2008].

La web 2.0 no solo es un grupo de tecnologías, sino una nueva perspectiva de negocios de software, el marketing a través del soporte [Musser & O'Reilly, 2006].

Podemos comparar servicios web que marcan claramente la evolución hacia la web 2.0 con una nueva forma de hacer las cosas como se observa en la Tabla 1.1:

Tabla 1.1: Comparación entre aplicaciones Web 1.0 y Web 2.0

WEB 1.0	WEB 2.0
Doubleclick →	Google AdSense (Servicios Publicidad)
Ofoto →	Flickr (Comunidades fotográficas)
Akamai →	BitTorrent (Distribución de contenidos)
mp3.com →	Napster (Descargas de música)
Britannica Online →	Wikipedia (Enciclopedias)
Sitios personales →	Blogs (Páginas personales)
Especulación con dominios →	Optimización en motores de búsqueda SEO
Páginas vistas →	Costo por click
CMSs →	Wikis (Administradores de contenidos)
Categorías/Directorios →	Tags

Fuente: [Vander, 2005]

Trabajos similares

Un estudio anterior acerca de la usabilidad y la web 2.0 fue el realizado por Ketharaman [2007]. En este artículo comenta el aumento en las exigencias de usabilidad que pueden solucionarse con una combinación de las tecnologías como AJAX y una mayor empatía con el usuario de la comunidad de desarrollo de software.

Otro trabajo que relaciona los conceptos de usabilidad y web 2.0 es el realizado por Javier Velasco [2007]. Este artículo describe el concepto de web 2.0 y explica cómo la usabilidad ha sido un factor clave en su proliferación.

Un trabajo sin duda muy interesante es el que plantea Jakob Nielsen [2007]. En su artículo “*Web 2.0 Can be dangerous*” critica fuertemente las tecnologías y los conceptos relacionados con esta temática, planteando una serie de inconvenientes muy acertados que surgen de su mal uso.

En el campo de la usabilidad, un trabajo destacable es el de Ben Hunt que en su libro “*Save the Pixel*” propone las bases teóricas para construir un sitio web usable.

1.3 Planteamiento del Problema

El avance acelerado de la tecnología y la constante incertidumbre acerca del éxito o fracaso de las nuevas tendencias tiene un efecto muy potente sobre la posición de las empresas u organizaciones con respecto al uso, sin conocimiento previo, de estas en las aplicaciones web que son desarrolladas.

Es así que se han identificado cuatro categorías de problemas frecuentes de usabilidad en aplicaciones web 2.0:

- **Diseño complejo**, no hay jerarquía o categorización de elementos, el ambiente en línea agrava el problema: a las personas les agrada usar un sitio web segundos o minutos, rechazando las interfaces complicadas de difícil interpretación [Usolab, 2005];
- **Difícil navegación**, las estructuras y elementos de navegación de un sitio web están mal utilizadas, no existe una jerarquía ni una construcción adecuada. Otro problema es el *Grope Navigation* o navegación al meter mano describe la navegación que no se muestra hasta que el puntero de ratón cambia al pasar sobre una zona sensitiva. Además la mayor parte de los sitios web 2.0 no incluyen una herramienta de búsqueda en el sitio para los usuarios;
- **Mala construcción** de los objetos que operan la información de la aplicación, generalmente se prioriza en la navegación pero el objetivo central de cada página no se detalla, por ello estas aplicaciones son difíciles de comprender;
- **Baja colaboración** al usuario son pocas las aplicaciones que cuentan con elementos de colaboración o ayuda para el usuario. Tampoco ayudan los cuadros de dialogo,

pues en su mayoría los diseñadores de estos sitios no los han implementado de manera adecuada;

- **Diseño no especializado**, esto repercute en el tipo de usuarios, se advirtió que el público objetivo de la web 2.0 son los jóvenes, excluyendo a personas mayores y de la tercera edad. Por otra parte, la última generación de aplicaciones está orientada a soportar cualquier tipo de usuario, sin darles un espacio propio ni un tratamiento adecuado.

El planteamiento del problema central viene del análisis en el árbol de problemas (Ver Anexo A) con resultado en la siguiente cuestionante:

¿Cómo hacer que las aplicaciones diseñadas con las tendencias web 2.0 utilicen de manera adecuada los principios de la usabilidad en el diseño de sus interfaces?

De esta manera los esfuerzos del presente trabajo estarán enfocados en el desarrollo de una solución para el problema planteado. Se realizará una revisión a los conceptos de usabilidad y web 2.0 para poder generar especificaciones de mejores prácticas en cuanto a la aplicación de los conceptos anteriormente mencionados.

1.4 Hipótesis

La introducción de un modelo de usabilidad en el diseño de aplicaciones web 2.0 logra incrementar el grado de interacción de los usuarios con estas aplicaciones.

Para la confirmación de la hipótesis se debe proceder a medir los factores influyentes:

Variable dependiente: $Y(x)$ Optimización de aplicaciones web 2.0

Variable independiente: x

x_1 : Facilidad de aprendizaje

x_2 : Operabilidad

x_3 : Navegabilidad

x_4 : Robustez

x_5 : Satisfacción

En la Tabla 1.2 se explica cada una de las variables anteriormente mencionadas.

Tabla 1.2: Descripción de variables independientes

Nombre de la variable	Definición Conceptual	Definición Instrumental	Definición Operacional
Facilidad de aprendizaje	Proceso con el que se adquieren habilidades, destrezas, conocimientos por medio de la experiencia, instrucción u observación	Característica que permite un aprendizaje rápido y significativo en el uso de sus componentes	Diseñar una interfaz de usuario coherente y simple que facilite la interacción del usuario con la aplicación
Operabilidad	Capacidad de ser operable o maniobrable	Característica que permite una fácil comprensión y uso de los recursos de la interfaz	Elaborar métodos para la construcción de los recursos de manejo de información de la interfaz
Navegabilidad	Capacidad de un sistema informático que permite la traslación entre sus interfaces	Característica que permite un fácil desplazamiento entre interfaces de la aplicación	Organizar los recursos de navegación para conseguir un resultado óptimo en la orientación del usuario
Robustez	Capacidad de hacer frente a variaciones en un ambiente operativo con una alteración o pérdida mínima de funcionalidad	Característica que permite responder a ciertas acciones del usuario	Estructuración de elementos que colaboren y mantengan informado al usuario
Satisfacción	Cumplir o llenar ciertos requisitos, exigencias, deseos o gustos	Característica que permite satisfacer a usuarios y temática definidos	Limitar los tipos de usuarios y objetos de la aplicación en ámbitos específicos acordes a un objetivo

1.5 Objetivos

El planteamiento del objetivo general y objetivos específicos viene de un análisis formulado en el árbol de objetivos (Ver Anexo B):

1.5.1 Objetivo General

Diseñar un modelo que organice el diseño de aplicaciones web 2.0 usables.

1.5.2 Objetivos Específicos

- Estandarizar objetos de diseño que simplifiquen la interfaz de la aplicación;
- Ajustar las estructuras de navegación denotando su correcto uso;
- Construir métodos para el diseño de recursos de contenido de la aplicación;
- Estructurar elementos que colaboren con las acciones de los usuarios;
- Analizar el tipo de usuarios y objetivos de la aplicación.

1.6 Justificación

Científica

Mediante el desarrollo de un modelo adecuado que solucione los problemas de los usuarios, se puede lograr un aporte científico que conlleve al equilibrio en el uso desmedido de las tecnologías Web 2.0 con el desarrollo de un modelo de Usabilidad que se encuentre dentro de la ingeniería Web orientado a estas tendencias.

Tecnológica

El modelo que se va a plantear intentará acortar la brecha que existe entre los usuarios y las aplicaciones que hacen uso de las tecnologías Web 2.0, brindando una serie de guías de diseño que los desarrolladores deben aplicar para este objetivo.

Además, mediante el uso de la tecnología AJAX que nos servirá de ejemplo para realizar un prototipo con el modelo planteado, se logrará una base tecnológica.

Social

El concepto de Web 2.0 con respecto al planteamiento de nuevos modelos de negocio esta cambiando completamente los enfoques tradicionales de marketing y ventas. Por eso la realización de esta investigación se enmarca en mejorar el uso de estos conceptos y tecnologías para ampliar la población de usuarios y llegar a aquellos que abandonan su uso por carencia de conocimientos o confusión en este campo.

1.7 Alcances

El prototipo en el que se aplicará los conceptos del modelo investigado abarca la construcción de un Organizador de Proyectos, categoría perteneciente a la clasificación de aplicaciones Web 2.0 explicada en el capítulo 2. Específicamente, se desarrollarán los módulos de intercambio y búsqueda de información, basado en *tags*, XML y AJAX, no con la intención de crear un sistema mucho más complejo.

El desarrollo del prototipo tendrá como base la Metodología UWE², esta abarca diagramas de modelado para la correcta documentación. UWE comprende un proceso de desarrollo con un conjunto de flujos de trabajo centrales que son: la captura de requerimientos, análisis y diseño, e implementación, los cuales se abarcarán en su totalidad para el desarrollo del presente trabajo. Sin embargo los flujos de trabajo de apoyo de esta metodología no se verán completamente. Estos son los flujos de trabajo de la Gestión de Proyectos como la gestión de riesgo, planeación de la iteración y evaluación de la iteración; y los flujos de trabajo de la Gestión de Calidad que comprende la validación de los requerimientos y la verificación de los artefactos; el flujo de trabajo de pruebas si se tomará en cuenta ya que es crucial para el análisis del resultado que nos proveerá el modelo. En cuanto a los modelos que propone la metodología se recurrirá al Modelo de Casos de Uso, Modelo Conceptual, Modelo de Usuario, Modelo de Espacio de Navegación y Modelo de Estructura de Navegación debido a que son de relevancia en el presente trabajo.

La metodología utilizada para la cuantificación de los resultados obtenidos de la evaluación del prototipo será la Metodología de Evaluación de Calidad de Sitios Web –

² UWE, del acrónimo *UML- Based Web Engineering* (UML basado en Ingeniería Web), es un lenguaje que colabora en la representación y modelado de la información de aplicaciones de hipermedia.

Web-site QEM³ planteado por Luis Olsina y recomendada por la Metodología UWE. Es bastante robusta y no solo permitirá realizar una evaluación de Organizadores de Proyectos que están actualmente en funcionamiento en la Web, sino también permitirá la comparación del prototipo con los mismos. Es necesario aclarar que la evaluación comprenderá el campo de la Usabilidad y no así las otras características propuestas por la metodología.



³ Web-site QEM, del acrónimo *Web – site Quality Evaluation Methodology* (Metodología para la Evaluación de la Calidad de Sitios Web),



CAPÍTULO II
MARCO TEÓRICO

*“La facultad de decidir es el camino hacia la originalidad.
La decisión más importante es la de hacer cada día algo que os acerque a vuestra meta.”*

Andrew Loomis

2.1. Usabilidad

El experto y gurú de la ergonomía en entornos web, Jakob Nielsen [2003] definió la usabilidad como un atributo de calidad que evalúa cuán fácil de usar es una interfaz. Nielsen también propone una definición del término en cinco componentes de calidad:

- Facilidad de aprendizaje, ¿Cuán fácil es para los usuarios lograr tareas básicas la primera vez que ellos se encuentran con el sistema?;
- Eficacia de uso, una vez que el usuario haya aprendido a utilizar el sistema, ¿Cuán rápido puede realizar tareas?;
- Facilidad del sistema para ser recordado, si un usuario ha utilizado el sistema anteriormente, ¿Recuerda lo bastante como para utilizarlo con más eficacia la próxima vez o no?;
- Frecuencia y severidad de errores, ¿Cuántos errores cometen los usuarios?, ¿Cuán serios son estos errores? y ¿Cuán fácil es para ellos recuperarse de estos errores?;
- Satisfacción, ¿Al usuario le gusta utilizar el sistema?, ¿está satisfecho?

Principios Heurísticos de la Usabilidad

Los Principios Heurísticos de Usabilidad fueron planteados por Jakob Nielsen [2005] debido a la gran necesidad de los diseñadores por encontrar una guía de Usabilidad en la cual establecerse. Son fundamentos generales de estructuración y

diseño cuyo seguimiento es altamente recomendable para cualquier sitio web, ya que se basan sobre hechos prácticamente irrefutables de lo que funciona y no durante la inspección del sitio por cualquier usuario. A continuación se citan los diez principios:

- Visibilidad del estado del sistema, el usuario siempre debe saber exactamente donde está, donde ha estado y donde puede ir. Es por eso que el sistema siempre debe mantener al usuario informado sobre su posición y lo que está pasando, a través de una respuesta apropiada dentro de un tiempo razonable;
- Acoplamiento entre el sistema y el mundo real, el sistema debe hablar el idioma de los usuarios, con palabras, frases y conceptos familiares a este, en lugar de términos orientados por el sistema;
- Máximo control y libertad del usuario, con frecuencia se escogen funciones del sistema equivocadas y se necesitará una clara y marcada "salida de la emergencia" para dejar el estado no deseado sin tener que atravesar por un extenso diálogo;
- Consistencia y estándares, los usuarios no deben preguntarse por el significado de diferentes palabras, situaciones, o acciones que signifiquen la misma cosa. Seguir las convenciones de la plataforma;
- Prevención de errores, aunque la construcción de un buen mensaje de error requiere de un diseño cuidadoso que impida la ocurrencia de un problema se debe, en primer lugar, eliminar las condiciones que son propensas a errores o, si no existe otra opción, verificarlas y presentarlas a los usuarios con la opción de confirmación correspondiente, antes de que ellos realicen la acción;
- Reconocer en lugar de recordar, el usuario tiene poca memoria para recordar qué información encontró en qué sitio. Por esto se debe minimizar la carga de memoria del usuario haciendo a los objetos, acciones, y opciones visibles. El usuario no debe tener que recordar la información de una parte del cuadro de diálogo a otra. Las instrucciones para el uso del sistema deben ser visibles o fácilmente recuperables siempre que esto sea apropiado;

- Flexibilidad y eficiencia de uso, los aceleradores, inadvertidos por el usuario principiante, pueden a menudo mejorar la velocidad de interacción para un usuario experto. Permitir a los usuarios adaptarse a las acciones frecuentes;
- Diseño estético y minimalista, los cuadros de diálogos no deben contener información irrelevante o poco necesitada. Cada unidad extra de información en un cuadro de diálogo compite con las unidades pertinentes de información y disminuye su visibilidad relativa;
- Ayudar a los usuarios a reconocer, diagnosticar y recuperar errores, deben expresarse los mensajes de error en un idioma llano y no en códigos, indicar de manera precisa el problema, y constructivamente sugerir una solución;
- Ayuda y documentación, aunque es bueno que el sistema pueda usarse sin documentación, esto puede ser necesario. Cualquier información debe ser fácil de buscar, enfocar las tareas del usuario y listar los pasos concretos a ser llevados a cabo, los cuales no deben ser demasiado largos.

2.2. Web 2.0

El concepto de web 2.0 es ampliamente utilizado en diversos ámbitos de la informática, y mal interpretado en la mayoría. Web 2.0 no es una nueva Internet, no es una tecnología, no es una herramienta, no es un *framework*⁴, no es AJAX. Son aquellas aplicaciones que obtienen su valor de las acciones y las participaciones de los usuarios. Es una actitud, una filosofía [Saavedra, 2007]. Aceptando que desde sus orígenes Internet propició la conectividad, es en un entorno de software social cuando ésta resulta especialmente significativa. Ahora bien: no existe una ruptura con el modelo de la web estática, se trata sólo de la emergencia de una nueva práctica, que provoca una convivencia de los dos tipos de aplicaciones [Cobo & Pardo, 2007].

⁴ *Framework* (base) es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Principios constitutivos de los sitios Web 2.0

- **La web como plataforma**, las nuevas compañías ofrecen software gratuito, utilizando a la web como plataforma. El concepto opuesto al de *desktop*, *el webtop*, es apropiado para explicar este fenómeno comercial [Cobo & Pardo, 2007];
- **Aprovechando la inteligencia colectiva**, a medida que los usuarios agregan nuevo contenido, y sitios web nuevos, se enlazan con la estructura de la web gracias a otros usuarios que descubren el contenido y enlazan con él [O'Reilly, 2005];
- **La gestión de la base de datos como competencia básica**, lo valioso de las aplicaciones web 2.0 son los datos. Así es crucial obtener una masa crítica de usuarios que produce un volumen de datos de gran valor [Cobo & Pardo, 2007];
- **El fin del ciclo de las actualizaciones de versiones del software**, los usuarios deben ser tratados como co-desarrolladores, considerando las prácticas de desarrollo del software libre [O'Reilly, 2005];
- **Los Modelos de Programación Ligeros**, las aplicaciones web híbridas denominadas *Mashups*⁵ permiten obtener lo mejor de cada aplicación. Un ejemplo son las aplicaciones existentes alrededor de *Google Maps* [Cobo & Pardo, 2007];
- **El software no limitado a un solo dispositivo**, los teléfonos móviles de tercera generación constituyen una plataforma de entretenimiento, gestión y generación de contenidos y fortalecimiento de redes sociales [Cobo & Pardo, 2007];
- **Experiencias enriquecedoras del Usuario**, las RIA ⁶ proporcionan no sólo contenido multimedia sino también experiencias de aplicación del estilo GUI. Conjuntamente *Second Life*⁷ y el uso escalable de avatares es una enriquecedora experiencia de usuario [Cobo & Pardo, 2007].

⁵ *Mashup* integra las ideas en inglés *mix* (combinar) y *match* (hacer coincidir). Es un punto de conexión entre aplicaciones web diferentes que permiten obtener lo mejor de cada una.

⁶ RIA, del acrónimo *Rich Internet Applications* (Aplicaciones Ricas de Internet), son un tipo de aplicaciones con una combinación de las ventajas que ofrecen las aplicaciones Web y las aplicaciones tradicionales.

⁷ *Second Life* (segunda vida) trata de un mundo/videojuego virtual con una estructura escalable de avatares. Creada por *Linden Lab*, representa un ejemplo de complejidad a nivel de comunidades en línea

Diferencias entre Web 1.0, Web 1.5 y Web 2.0

En la Tabla 2.1 se explica las principales diferencias entre la Web 1.0, Web 1.5 y Web 2.0.

Tabla 2.1: Descripción de las diferencias entre la Web 1.0, Web 1.5 y Web 2.0

	Web 1.0	Web 1.5	Web 2.0
Tipo de Web	Estática	Dinámica	Colaborativa
Período	1994-1997	1997-2003	2003 - hoy
Tecnología asociada	HTML, GIF	DHTML, ASP, CSS	Ajax, DHTML, XML, Soap
Características	Las páginas web son documentos estáticos que jamás se actualizaban	Las páginas web son construidas dinámicamente a partir de una o varias bases de datos	Los usuarios se convierten en contribuidores. Publican la información y hacen cambios en los datos

Fuente: [Paz, 2005]

Clasificación de aplicaciones Web 2.0

La estructura propuesta busca ordenar la web 2.0 en cuatro ejes que evidencian algunos de los principales desarrollos de Internet en su fase más reciente y son: las Redes Sociales, la Organización Social de Contenidos, Organización Social e Inteligente de la Información y las Aplicaciones, Servicios y *Mashups*.

Redes Sociales

Esta categoría describe a todas aquellas herramientas diseñadas para la creación de espacios que promuevan o faciliten la conformación de comunidades e instancias de intercambio social. Estas herramientas, en su mayoría gratuitas y de

fácil uso, ofrecen un espacio virtual para escribir y compartir contenidos multimedia con personas de intereses similares y que contribuyen a fortalecer aquellas redes sociales débiles [Cobo & Pardo, 2007].

Organización Social de Contenidos

Esta parte hace referencia a aquellas herramientas que favorecen la lectura y la escritura en línea, así como su distribución e intercambio. Bajo esta perspectiva se ha logrado sistematizar esta categoría en diez sub categorías [Cobo & Pardo, 2007]:

- Software de *Weblogs* o *blogware*;
- *Blogging*;
- Sistemas Gestión de Contenidos o *Content Management Systems*;
- *Wikis*;
- Procesador de Textos en Línea;
- Hojas de Cálculo en línea;
- Plataformas para almacenar, publicar, compartir y editar fotografías digitales;
- Plataformas para almacenar, publicar, compartir y editar Videos/TV;
- Plataformas para Calendario;
- Presentación de Diapositivas.

Organización social e inteligente de la información

Son herramientas y recursos que ayudan a etiquetar, syndicar e indexar, y facilitan el orden y almacenamiento de la información, así como de otros recursos disponibles en la Red. Se ha dividido esta categoría en tres sub categorías [Cobo & Pardo, 2007]:

- Buscadores;
- Lectores de RSS⁸ y Agregadores *Feeds*⁹;
- Marcadores Sociales de Favoritos o *Social Bookmark* y Nubes de *Tags*.

⁸ RSS, del acrónimo *Really Simple Syndication* (Sindicación muy simple), es un formato de documento para la sindicación de contenidos de páginas web, basado en el XML conforme especificaciones publicadas por el W3C.

⁹ *Feed* (fuente) es un formato de datos para suministrar información actualizada frecuentemente. Los interesados pueden usar un programa agregador para acceder a sus fuentes suscritas desde un mismo lugar. Técnicamente, se emplea para denominar a los documentos con formato RSS o *Atom*, basados en lenguaje XML.

Aplicaciones, Servicios y *Mashups*

Dentro de esta clasificación se incluye un sinnúmero de herramientas, software, plataformas en línea y un híbrido de recursos creados para ofrecer servicios de valor añadido al usuario final. Se ha logrado dividir esta categoría en cuatro sub categorías [Cobo & Pardo, 2007]:

- Organizador de Proyectos, ofrecen herramientas para el trabajo, facilitan la organización de equipos que trabajan de manera sindicada, apoyados en el uso de Internet, escritura colaborativa, intercambio de archivos, calendario/agenda, servicio de correo electrónico, VoIP y otros. Sitios destacados: *rememberthemilk*, *planner.zoho* y *basecampHQ*;
- *Webtop*;
- Almacenamiento en la Web;
- Reproductores y agregadores de Música, son y búsqueda de audios (*podcasts*¹⁰).

2.3. Metodología de Desarrollo UML basado en Ingeniería Web - UWE

UWE cubre todos los aspectos que el desarrollo de Aplicaciones de Hipermedia Adaptativa¹¹ requiere y está basado en el Proceso Unificado y RUP [Koch, 2000].

Proceso de Desarrollo

La vida de una aplicación es dividida por UWE en ciclos, cada ciclo concluye en una versión. UWE no establece el número de iteraciones requeridas en un ciclo de vida de una aplicación de hipermedia adaptativa, pero distingue varias fases.

Fases del Proceso de Desarrollo

Las primeras cuatro fases son las mismas que en el Proceso Unificado: Comienzo, Elaboración, Construcción y Transición. El proceso se extiende para incluir

¹⁰ *Podcast* es una serie de audio o video media digital, distribuido por Internet mediante una descarga sindicada (*syndicated download*), a través de *Web feeds* a reproductores media portátiles y a computadoras personales.

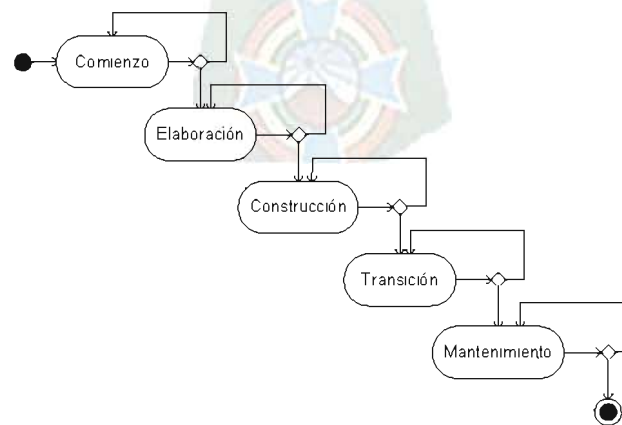
¹¹ Las Aplicaciones de Hipermedia Adaptativa son aplicaciones que adaptan el contenido o presentación de sus nodos y/o vínculos al usuario. Un nodo es una unidad que contiene texto y/o elementos multimedia, como imágenes, video, audio, o animaciones. Un vínculo normalmente es direccionable y conecta dos nodos: el nodo fuente y el nodo destino.

la fase de Mantenimiento, debido a que son una parte importante del ciclo de vida de las aplicaciones hipermedia [Koch, 2000].

- La Fase de Comienzo empieza con la definición de objetivos de desarrollo, así como una primera aproximación a la arquitectura del sistema, una estimación de costos y un plan de horarios [Koch, 2000];
- Durante la Fase de la Elaboración la arquitectura del sistema y un juego de modelos de diseño son definidos. Una arquitectura estable así como el control de riesgos son prerequisites para la siguiente fase [Koch, 2000];
- La Fase de Construcción se enfoca en el desarrollo del sistema, aunque se realizan requisitos adicionales de extracción y cambios menores en el diseño de la arquitectura durante este paso [Koch, 2000];
- La Fase de Transición cubre el período durante el cual el sistema es probado como una versión completa (normalmente llamada versión *beta*) por un grupo reducido de usuarios [Koch, 2000];
- La Fase de Mantenimiento empieza cuando la primera versión se entrega y se extiende hasta que el sistema ya no se use [Koch, 2000].

El diagrama de actividad mostrado en la Figura 2.1 presenta el proceso de ingeniería de software iterativo que incluye estas cinco fases para cada versión.

Figura 2.1: Proceso de UWE para una versión



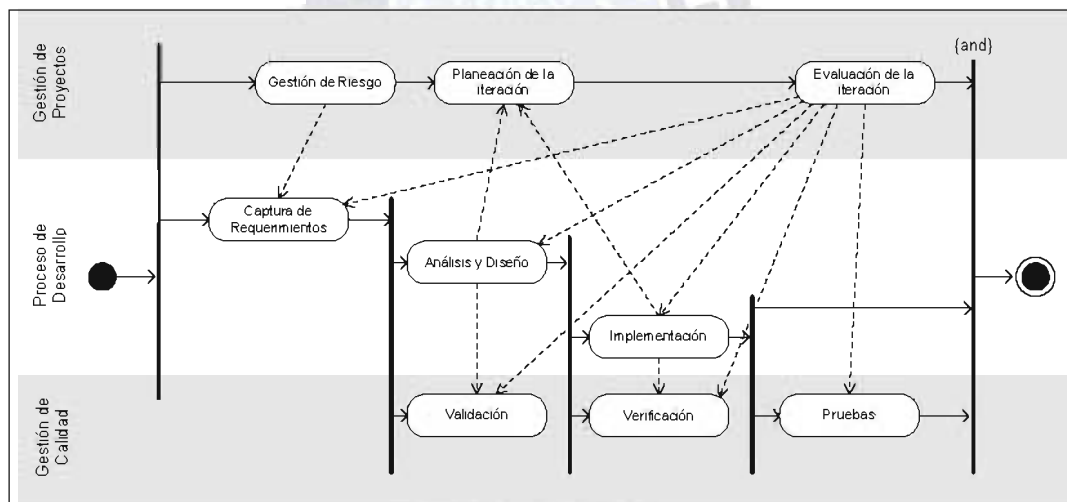
Fuente: [Koch, 2000]

Flujos de Trabajo

Es importante señalar que cada Fase del proceso de desarrollo está constituida por Flujos de Trabajo: Captura de Requerimientos, el Análisis y Diseño, e Implementación.

Un diagrama de actividad UML se usa en este trabajo para representar las Iteraciones de los Flujos de Trabajo (Ver Figura 2.2). Este representa el flujo de control (líneas continuas) y dependencias (líneas punteadas) entre flujos de trabajo. Se usan líneas transversales para indicar los diferentes grupos de flujos de trabajo: gestión de proyectos, proceso de desarrollo y gestión de calidad [Koch, 2000].

Figura 2.2: Iteraciones de los Flujos de Trabajo



Fuente: [Koch, 2000]

Captura de Requerimientos

La captura de requerimientos es el proceso de determinar o descubrir que aplicación será desarrollada. Un requerimiento es una condición o capacidad a la que una aplicación debe ajustarse. Dos categorías de requerimientos pueden ser distinguidas [Koch, 2000]:

- Requerimientos funcionales son las acciones que el sistema podrá representar, es decir son usados para describir el comportamiento del sistema teniendo en cuenta las ciertas condiciones de entrada.

- Requerimientos no funcionales especifican las propiedades de un sistema, así como el entorno y restricciones de implementación, rendimiento, confiabilidad, etc.

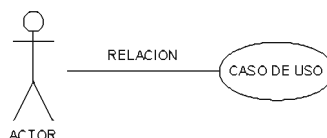
Las actividades a ser realizadas en este flujo de trabajo son:

- Extracción de las necesidades de Información;
- Extracción de Requerimientos Adicionales;
- Encontrar actores y Casos de Uso;
- Priorizar los Casos de Uso;
- Detallar los Casos de Uso;
- Estructurar Casos de Uso;
- Extracción de necesidades de Navegación;
- Extracción de necesidades de la UI;
- Prototipar la UI.

Modelo de Casos de Uso

Para describir los requerimientos funcionales de una aplicación se puede usar un modelo de casos de uso (Ver Figura 2.3 y Figura 2.4). El modelo de casos de uso está conformado por dos elementos de modelado principales, llamados casos de uso y actores [Koch et al, 2001]. Un caso de uso es una unidad coherente de funcionalidad provista de aplicaciones que interactúan con uno o más actores externos de la aplicación. Un actor es el rol que un usuario puede desempeñar con respecto a un sistema o una entidad, tales como otro sistema o una base de datos.

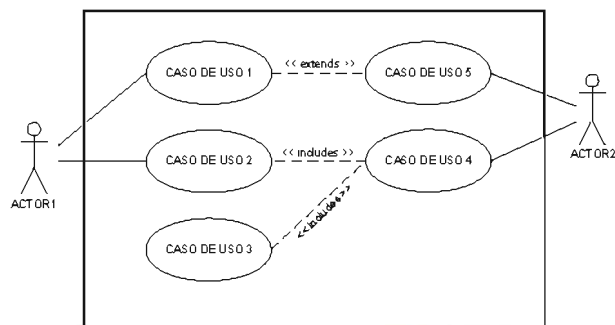
Figura 2.3: Elementos de un Modelo de Casos de Uso



Fuente: [Modificado de Koch & Kraus, 2002]

El número de casos de uso identificados se empaqueta en un juego de vistas del modelo de caso de uso o, en otros términos, la división del modelo de casos de uso en paquetes.

Figura 2.4: Modelo de Casos de Uso



Fuente: [Modificado de Koch & Kraus, 2002]

Análisis y Diseño

El propósito del análisis y diseño es traducir la descripción de requerimientos en una especificación que describe cómo implementar la aplicación de hipertexto adaptativa. El análisis se concentra en los requisitos funcionales de la aplicación, haciendo caso omiso de requisitos no funcionales y restricciones de implementación. Durante el diseño los resultados de análisis son adaptados a las condiciones impuestas por los requisitos no funcionales [Koch, 2000]. El diseño comprende las siguientes actividades:

- Diseño conceptual;
- Diseño del modelo de usuario;
- Diseño de navegación;
- Diseño de presentación;
- Diseño de adaptación;
- Diseño de arquitectura;
- Definición de los subsistemas e interfaces.

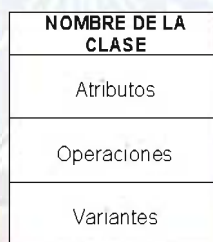
Modelo Conceptual

El diseño conceptual incluye a los objetos involucrados en la interacción entre el usuario y la aplicación, especificado en los casos de uso.

Los principales elementos usados para el modelo conceptual son las clases y asociaciones. Si el modelo conceptual consiste de varias clases se recomienda que se agrupen usando elementos de modelado de paquetes de UML [Koch & Kraus, 2002].

Una clase es descrita por un nombre, atributo, operaciones y variantes como se ve en la Figura 2.5 [Koch, et al, 2001].

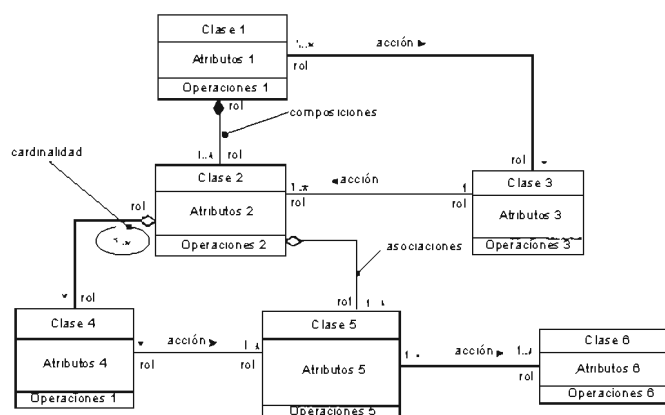
Figura 2.5: Clase con variantes de compartimiento adicionales



Fuente: [Koch, et al, 2001]

La construcción de modelos de clase (Ver Figura 2.6) se enfoca en el correcto análisis de los casos de uso [Koch, et al, 2001].

Figura 2.6: Modelo Conceptual

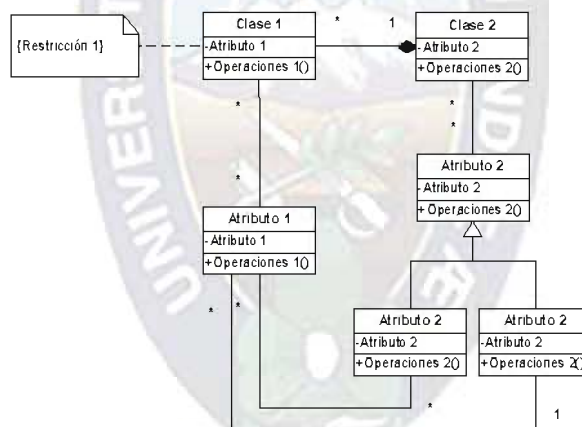


Fuente: [Modificado de Koch & Kraus, 2002]

Modelo de Usuario

El modelo de usuario se diseña con el propósito de establecer qué atributos del usuario se escogerán para elaborar un perfil de usuario, determinar cómo estos atributos se relacionan a nosotros y cómo ellos se relacionan a los elementos del dominio. Un diagrama de la clase se usa para la representación de un modelo de usuario estático. Los elementos de modelado usados en el modelo del usuario son: clases y asociaciones. Si el modelo de usuario se compone de muchas clases, se recomienda agruparlos usando el elemento de modelado paquete de UML. También se puede crear los siguientes sub-modelos: conocimiento de dominio dependencia, conocimiento de fondo y modelo cognoscitivo. Cada sub modelo puede entonces representarse como un paquete [Koch, 2000].

Figura 2.7: Modelo de Usuario



Fuente: [Modificado de Koch, 2000]

Los requisitos capturados como los casos del uso pueden ser útiles para identificar las clases para el modelo de usuario (Ver Figura 2.7).

Modelo de Navegación

Este modelo se comprime en el modelo de espacio de navegación y el modelo de estructura de navegación. El primero especifica qué objetos pueden ser visitados mediante una navegación a través de la aplicación. El segundo define cómo estos objetos son alcanzados [Koch, et al, 2001].

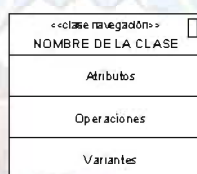
Modelo de Espacio de Navegación

En el proceso de construir el modelo de espacio de navegación el desarrollador toma decisiones cruciales de diseño, tales como qué vista del modelo conceptual es necesaria para la aplicación y cuáles serán los caminos de navegación requeridos para el aseguramiento de la funcionalidad.

Se usan tres elementos de modelado para la construcción del modelo de espacio de navegación: las clases navegación, los nodos externos y las asociaciones de navegación que expresan la navegabilidad directa [Koch, 2000].

La clase de navegación modela una clase cuyas instancias son visitadas por usuarios durante la navegación (Ver Figura 2.8) [Koch et al, 2001].

Figura 2.8: Clase Navegación



Fuente: [Koch et al, 2001]

Un nodo externo modela un objetivo perteneciente a otra aplicación hipertexto, es decir el nodo no es parte de la aplicación que está siendo modelada, pero puede alcanzarse por medio del nodo fuente de la aplicación (Ver Figura 2.9) [Koch, 2000].

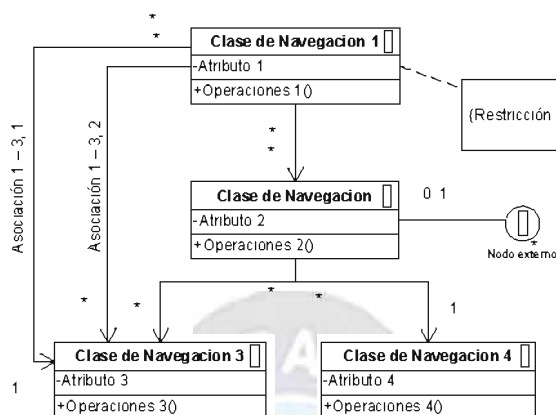
Figura 2.9: Nodo externo



Fuente: Koch [2000]

El modelo de espacio de navegación (Ver Figura 2.10) es construido con las clases de navegación y las asociaciones de navegación y están representadas gráficamente por un diagrama de clases de UML.

Figura 2.10: Modelo de Espacio de Navegación



Fuente: [Modificado de Koch, 2000]

Modelo de Estructura de Navegación

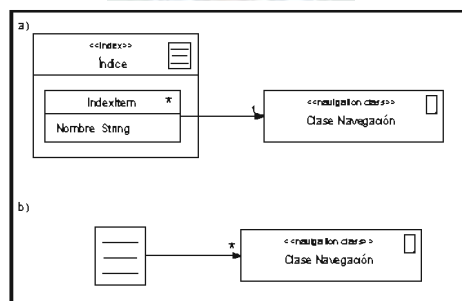
El modelo de estructura de navegación describe cómo la navegación es soportada por elementos de acceso tales como índices, visitas guiadas, preguntas y menús [Koch et al, 2001].

Primitivas de Acceso

Las primitivas de acceso son nodos de navegación adicionales requeridas para acceder a objetos de navegación. Estas son definidas como estereotipos UML: índices, visitas guiadas, consultas y menús [Koch et al, 2001].

Los índices (Ver Figura 2.11) permiten el acceso directo a las instancias de la clase de navegación [Koch et al, 2001].

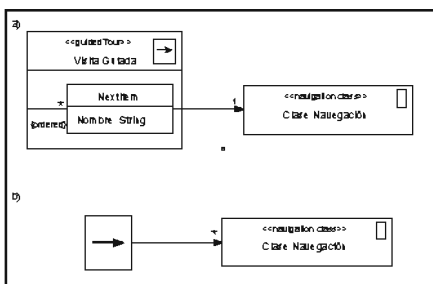
Figura 2.11: a) Clase Índice y b) su notación abreviada



Fuente: [Koch et al, 2001]

Las visitas guiadas (Ver Figura 2.12) proveen acceso secuencial a las instancias de una clase navegación. Las visitas guiadas deben ser controladas por el usuario o por el sistema [Koch et al, 2001].

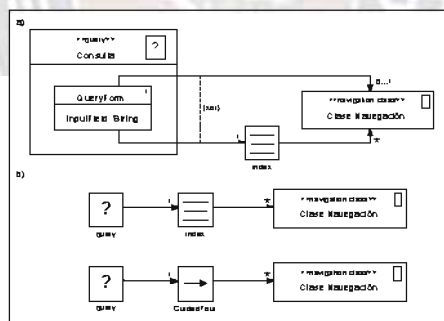
Figura 2.12: a) Clase Visita Guiada y b) su notación abreviada



Fuente: [Koch et al, 2001]

Una consulta (Ver Figura 2.13) es modelada por una clase que tiene una serie de preguntas como atributo. Cualquier clase consulta es la fuente de dos asociaciones dirigidas relacionadas por la restricción {xor} [Koch et al, 2001].

Figura 2.13: a) Clase Consulta y b) sus notaciones abreviadas



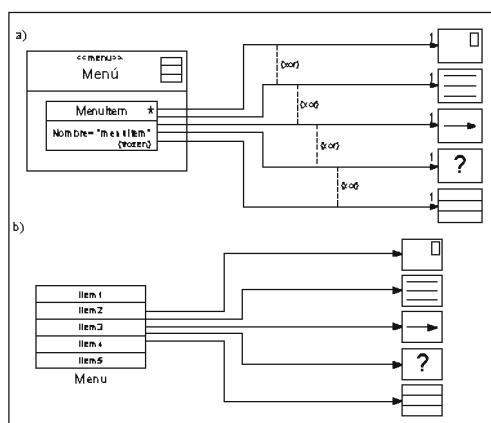
Fuente: [Koch et al, 2001]

Adición de Menús

En este paso, las primitivas de acceso de tipo menú son añadidas al modelo de estructura de navegación [Koch, 2000].

El elemento de modelado menú (Ver Figura 2.14) es una primitiva de acceso adicional que puede ser añadida a la lista presentada en el paso previo. Un menú es un conjunto de elementos heterogéneos, tales como índices, visitas guiadas, consultas, una instancia de una clase navegación u otro menú [Koch et al, 2001].

Figura 2.14: a) Clase Menú y b) su abreviación



Fuente: [Koch et al, 2001]

Implementación

Este flujo de trabajo consiste en la transformación de los resultados de la fase de diseño, subsistemas e interfaces en un sistema implementado en términos de componentes, por ejemplo código fuente, scripts, ejecutables, etc. Cuestiones de implementación, como conversión de documentos, generación de plantillas o *templates* y/o la generación dinámica de páginas tienen que ser considerados en este flujo de trabajo [Koch, 2000].

Las actividades que pertenecen a este flujo de trabajo son:

- Generación de componentes;
- Ensamblando los subsistemas e interfaces;
- Integración de software.

Los típicos componentes a ser producidos son [Koch, 2000]:

- Componentes media, como texto, imágenes, video, audio y animación (estos constituyen el contenido de la aplicación hipertexto);
- Bases de datos para almacenar el contenido;
- Componentes de estructura requeridos por el paradigma hipertexto, como menús, índices, visitas guiadas y vínculos;

- Componentes de interfaz de usuario, como ventanas, *frames*, botones, logotipos, formularios y banners;
- Componentes para la generación de la página dinámica, como *servlets* de *Java*;
- Motores de búsqueda;

Pruebas

Este es un flujo de trabajo considerado de apoyo por la metodología UWE y no forma parte de los flujos de trabajo centrales. Sin embargo su desarrollo es crucial para la correcta implementación de cualquier trabajo.

La etapa de pruebas es un proceso que implica verificar la correcta implementación de los resultados de un sistema puesto en marcha. También se verifica si el software producido es correcto, es decir si trabaja sin fallas [Koch, 2000].

Las actividades de la fase de prueba son [Koch, 2000]:

- Planificación de la prueba, consiste en coleccionar información para planear las pruebas y crear un plan de prueba apropiado basado en esta información. Por esto se debe llevar a cabo los siguientes pasos:
 - Identificar los requisitos para las pruebas;
 - Valorar los riesgos;
 - Desarrollar la estrategia de pruebas;
 - Identificar los recursos necesarios para realizar las pruebas;
 - Generar el Plan de pruebas;
- Diseño de la prueba, los siguientes pasos deben ser llevados a cabo para la definición de los casos de prueba y los procedimientos de prueba:
 - Definición de las condiciones de prueba;
 - Identificación de los casos de uso a probar para concentrándose en la funcionalidad adaptativa;

- Preparación de los datos de prueba apropiados;
- Especificación de los resultados esperados en la prueba;
- Implementación de la prueba, implementar una prueba casi siempre significa escribir o reutilizar código, también llamados *scripts* de prueba. El ambiente de prueba tiene que ser preparado incluyendo los datos, *hardware*, *software*, herramientas, etc.
- Realización de la prueba, los procedimientos de prueba son ejecutados durante esta actividad a mano o automáticamente. La actividad constar de:
 - Ejecutar los *scripts* de prueba;
 - Evaluar los resultados;
 - Determinar acciones posteriores: si los resultados son como era de esperar, ninguna acción es necesaria, si los resultados son inesperados, la causa del problema debe ser determinada y resuelta.

La calidad de sitios web puede ser cuantificada por métodos como la Metodología de Evaluación de Calidad de Sitios Web – Web-site QEM. Es un método prescriptivo y descriptivo basado en la evaluación y comparación de características de calidad de y atributos en las diferentes fases del ciclo de hipermedia [Koch, 2000].

Se puede añadir que en la actividad de Diseño de la prueba es posible señalar los métodos o técnicas que se utilizarán, por ejemplo en el caso de la Usabilidad se puede especificar métodos como el Método de Pensamiento en voz alta o el Método de Observación.

2.4. Métodos de Evaluación de Usabilidad

Método de pensamiento en voz alta (*Thinking Aloud Method*)

Durante el transcurso de la prueba en la que el participante está realizando una “tarea” como parte de un escenario de uso, se solicita que exprese en voz alta sus pensamientos, sensaciones y opiniones mientras interactúa con el producto [Cortés,

2000]. El escuchar los pensamientos de un usuario facilita la comprensión de las acciones del mismo (por qué hace lo que hace) esta información es inestimable en la prueba. La mayor desventaja resulta en la sensación de anormalidad que experimentan los usuarios, además es difícil para usuarios expertos expresarse verbalmente [Nielsen, 1993];

Método de Observación

Supone visitar uno o más usuarios y hacer tan poco como sea posible para no entrometerse en su trabajo. Por supuesto, el observador puede tomar notas (discretamente), y si es posible grabar en video en algunos ambientes. De vez en cuando, puede ser necesario interrumpir al usuario para pedir una explicación en alguna actividad que es imposible que el observador comprenda, pero tales preguntas a los usuarios deben mantenerse a un mínimo [Nielsen, 1993]. La desventaja de este método recae en la difícil toma de apuntes por parte del observador, además no se puede tener control de las actividades que se espera evaluar del usuario;

2.5. Metodología de Evaluación de Calidad de Sitios Web – *Web-site QEM*

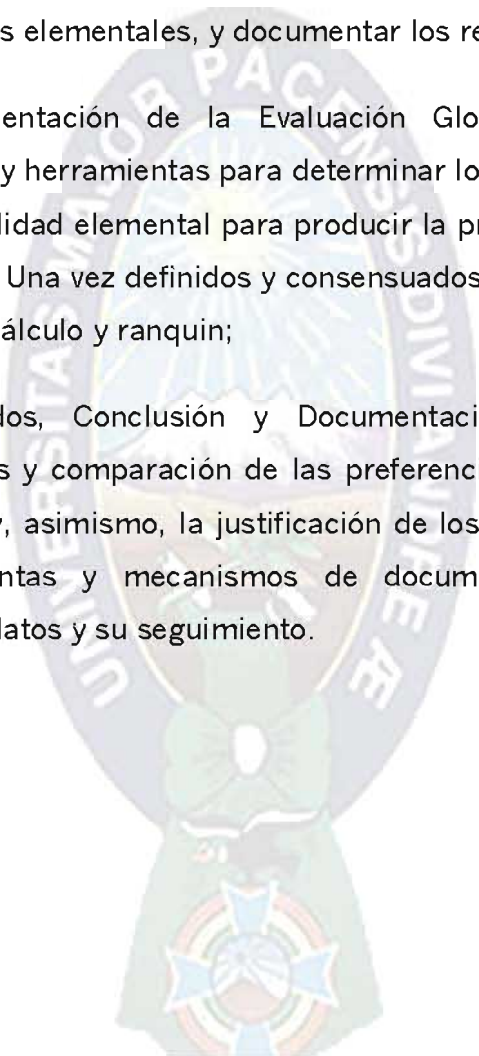
Para la cuantificación de resultados se procederá al aplicación de la Metodología Evaluación de Calidad de Sitios Web o, metodología Web-site QEM planteado por Luis Olsina. Este es un enfoque sistemático, disciplinado y cuantitativo que se adecue a la evaluación, comparación y análisis de calidad de sistemas de información centrados en la Web (más o menos complejos) [Olsina, 1999].

Las fases de la metodología son [Olsina, 1999]:

- Planificación y Programación de la Evaluación de Calidad, esta contiene actividades y procedimientos de soporte, con el fin de determinar objetivos estratégicos, tácticos y operativos;
- Definición y Especificación de Requerimientos de Calidad, el proceso de determinación de requerimientos, realizado en una mezcla de estrategias prescriptivas y descriptivas, culmina con un documento que jerárquicamente

especifica a todas las características y atributos cuantificables que modelan a la calidad según las necesidades del usuario;

- Definición e Implementación de la Evaluación Elemental, esta fase trata con actividades, modelos, técnicas y herramientas para determinar métricas y criterios de evaluación para cada atributo cuantificable. Una vez definidos y consensuados los criterios se debe ejecutar el proceso de recolección de datos, computar las métricas y preferencias elementales, y documentar los resultados;
- Definición e Implementación de la Evaluación Global, esta fase trata con actividades, modelos, y herramientas para determinar los criterios de agregación de las preferencias de calidad elemental para producir la preferencia global, para cada sistema seleccionado. Una vez definidos y consensuados los criterios, se debe llevar a cabo el proceso de cálculo y ranquin;
- Análisis de Resultados, Conclusión y Documentación, la misma trata con actividades de análisis y comparación de las preferencias de calidad elementales, parciales y globales, y, asimismo, la justificación de los resultados. Por otra parte, se utilizan herramientas y mecanismos de documentación para facilitar la interpretación de los datos y su seguimiento.



The background features a large, faded watermark of the University of the Pacific logo. The logo is an oval shape containing a sunburst at the top, a shield in the center, and a cross at the bottom. The text "UNIVERSITAS MAJOR PACENSIS DIVINAE" is visible around the perimeter of the oval.

CAPÍTULO III

MODELO PILON

“La simplicidad no es la meta, es el subproducto de una buena idea y de expectativas modestas.”

Paul Rand

3.1. Introducción

El diseño de una aplicación web involucra mucho más que una perspectiva gráfica y estética. Hablar en términos de estética solo es centrarnos en una parte del diseño. Diseñar es el proceso de convertir las necesidades y requerimientos del usuario en servicios y atributos de la aplicación, es decir la creación visual con un propósito. El diseño involucra brindar al usuario una experiencia satisfactoria al comunicarle el objetivo de la aplicación. Su creación no debe ser sólo estética sino también funcional mientras refleja una identidad visual.

El Modelo Pilón, consiste en un proceso de diseño orientado a aplicaciones Web 2.0 que hace énfasis en la usabilidad. La parte central es la construcción del *Armazón Web*, fundamentado en la información recolectada de la metodología seleccionada y se acopla a la fase de diseño de la misma, en el caso del diseño del prototipo UWE.

Antes de la aplicación del modelo es importante que el desarrollador conozca conceptos fundamentales previos sobre espacio, estructura y elemento, ya que se utilizarán durante todo el proceso de desarrollo. Estos se encuentran en el Anexo D del presente trabajo.

Cada etapa proporciona artefactos que sirven como un medio de comunicación entre las fases y apoyan en el registro de los resultados obtenidos.

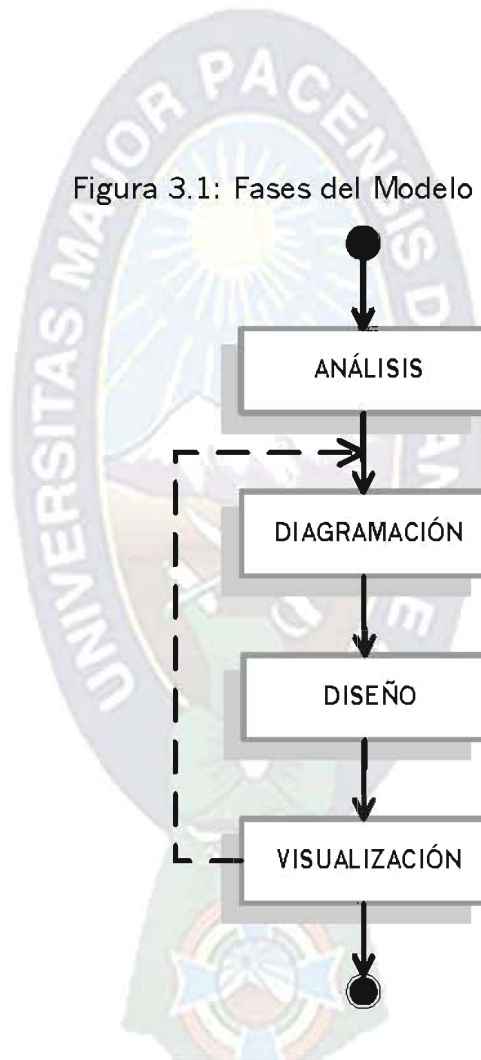
Para la prueba del modelo se ha procedido a la construcción de un prototipo explicado de manera posterior.

3.2. Fases del Modelo

El Modelo consta de seis fases (Ver Figura 3.1):

- Análisis;
- Diagramación;
- Diseño;
- Visualización.

Figura 3.1: Fases del Modelo Pílon



Cada una de estas fases representa un estado del modelo, en el que se llevará a cabo actividades específicas y darán como producto un conjunto de artefactos denominados Formato de Registro.

La fase de Análisis constituye un previo estudio de los elementos básicos que forman parte de la identidad de la aplicación dando un breve vistazo a la imagen

definiendo un estilo adecuado en base al grupo de usuarios identificados para la aplicación.

La fase de Diagramación se fundamenta en el diseño de la rejilla y la delimitación de espacios de las plantillas de composición pero no abarca estructuras ni elementos.

La fase de Diseño es la única que se subdivide en tres sub-fases específicas (Ver Figura 3.2):

- Contenido;
- Navegación;
- Robustez.

Figura 3.2: Sub-fases de la fase de Diseño



Las sub-fases de Contenido, Navegación y Robustez se centran en la explicación de los distintos objetos que pueden componer el armazón web y la relación de estas con el usuario. La sub-fases de contenido y navegación comparten información para lograr un acomodamiento adecuado de los objetos del armazón, ambas deben completarse paralelamente y para ingresar a la sub-fase de robustez. La sub-fase de Robustez completa la interacción de los objetos obtenida de cada armazón. Las tres sub-fases producen el armazón web.

La fase de Visualización produce un conjunto de métodos para la construcción de elementos visuales y reglas que deben instaurarse en la construcción de los nuevos de acuerdo a cada aplicación. Ayudan a crear la imagen gráfica casi en su totalidad, esta fase depende de una parte de la información que se obtiene en la fase de análisis.

En el caso de hacer hecho algún cambio en el diseño de la aplicación o es necesario hacer una modificación en algunas estructuras y/o elementos es posible retroalimentar el modelo en una nueva iteración y efectuar los cambios necesarios para ajustarlo a los resultados esperados.

Cada fase cuenta con actividades a ser realizadas secuencial y ordenadamente. En el caso de la fase de Diseño las sub-fases cuentan con estas actividades.

Se debe aclarar que en el transcurso del modelo no se toma en cuenta el diseño de la base de datos, o el mapa de navegación ya que son elementos que se obtienen al desarrollar la metodología seleccionada. El presente modelo se acopla a esa metodología y hará uso de esos artefactos para su correcto desarrollo.

La Tabla 3.1 muestra un resumen de los objetivos, actividades y artefactos que conforman cada fase

Tabla 3.1: Ejemplo de usuarios reales: estudiantes, niños y mujeres

FASE	OBJETIVO	ACTIVIDADES	ARTEFACTO
ANÁLISIS	Identificar a los usuarios reales para diseñar el ambiente adecuado a ellos	<ul style="list-style-type: none"> • Identificar a los usuarios reales • Selección de las características descriptivas de la marca • Crear los elementos de la marca 	Formato de Registro de Análisis de la Marca
DIAGRAMACIÓN	Lograr precisión, eficiencia y orden entre los objetos del armazón	<ul style="list-style-type: none"> • Crear la rejilla de diagramación • Delimitar espacios • Numeración de espacios 	Formato de Registro de las Plantillas de Composición

DISEÑO	CONTENIDO	Organizar los objetos de contenido de acuerdo a un orden y función	<ul style="list-style-type: none"> • Determinar tipo de contenido • Designar objetos de contenido • Asignar jerarquía visual 	Formato de Registro de los Armazones Web
	NAVEGACIÓN	Construir la navegación optima para un cómodo desplazamiento del usuario	<ul style="list-style-type: none"> • Determinar tipo de navegación • Designar objetos de navegación 	
	ROBUSTEZ	Registrar y representar las funciones de los objetos y su interacción con los usuarios	<ul style="list-style-type: none"> • Etiquetar objetos • Determinar acciones de cada objeto • Registro de código conciso 	
VISUALIZACIÓN		Construir y estandarizar los elementos gráficos de la aplicación	<ul style="list-style-type: none"> • Complementar paleta cromática • Determinar requerimientos de elementos gráficos • Estandarización de tipografía • Predefinir estilo de vínculos • Establecer tipo de jerarquía visual 	Guía de Estilo Visual

3.2.1. Análisis

Para el desarrollo de esta fase en necesaria la captura de requisitos relacionada a los usuarios y al contenido de la aplicación. El objetivo de esta es identificar a los **usuarios reales** y diseñar la **marca** de la aplicación, conceptos que ayudan a mantener unidad visual y un nivel de comunicación adecuado.

Usuarios reales son aquellos que utilizan directamente la aplicación para lograr un determinado objetivo u objetivos que la incumben. Cabe señalar que un usuario real es distinto de un actor. La diferencia incide en que un actor representa un usuario en general, mediante una ingeniería de requerimientos ayuda a identificar los procesos de la aplicación. Sin embargo un usuario real ayuda a delimitar el estilo, contenido y grado de dificultad de la aplicación. Algunos usuarios pueden realizar las mismas operaciones, es decir, ser un solo actor. En tal caso, cada usuario constituye una instancia de un actor.

Una **marca** es la suma de todo lo que las personas perciben, como ser opiniones previas, prejuicios, y lo que ha sido dicho acerca del sitio web, que puede ser reforzado por su experiencia personal, directa o indirectamente. Esta marca ayuda a definir la primera impresión del sitio, y se construye en base a dos aspectos: el usuario y el propósito de la aplicación.

Artefactos

El artefacto obtenido durante esta fase es el Formato de Registro de Análisis de la Marca. Este puede encontrarse completo en el Anexo K, con las correspondientes indicaciones de uso.

Actividades

Se debe considerar las siguientes actividades en esta fase:

- Identificar a los usuarios reales;
- Selección de las características descriptivas de la marca;
- Crear los elementos de la marca.

Identificar a los usuarios reales

Cuando se procede a diseñar un sistema, las personas menos adecuadas para ser tomadas en cuenta como usuarios de la aplicación son los desarrolladores del sistema y/o los ejecutivos que soliciten el proyecto, ellos pueden sugerir ideas para el

desarrollo de la misma pero sus opiniones no serán exactas, y en el caso de los desarrolladores, el probar el sistema con estas personas puede dar incluso una idea de falsa usabilidad dada su amplia experiencia en uso de sistemas.

Por otra parte es importante especializar al tipo de usuario que utilizará de la aplicación web, ya que no es factible diseñarla para todas las personas que hacen uso de internet, aún sabiendo que la Web 2.0 tiene carácter social. El diseño debe priorizar a los usuarios que tengan un nivel adecuado y necesiten del tipo de aplicación que se pretende desarrollar. Por eso se puede concluir que una aplicación debe aceptar un máximo de dos tipos de usuario relacionados.

La Tabla 3.2 muestra un ejemplo del modo de registro de las características de los usuarios reales de la aplicación. Los ejemplos que se muestran en ella solo describen a un tipo de usuario por aplicación.

Tabla 3.2: Ejemplo de registro de usuarios reales: mujeres, niños y estudiantes

Tipo de usuario	Edad	Nivel
mujeres	18 - (?)	Experto
niños	6 - 12	Novato
estudiantes	18 - 24	No definido

En la columna Tipo de usuario se ha dispuesto ejemplos de usuario por **género** (mujeres), **edad** (niños) y **nivel académico** (estudiantes), aunque el campo está abierto para otras categorizaciones que puedan ser encontradas en el desarrollo. La columna Edad denota el intervalo que será útil para cotejar los requerimientos de diseño que existiesen. La columna Nivel establece el grado de conocimiento en manejo de controles de la interfaz por parte de los usuarios identificados. Para un nivel de manejo de interfaz **experto** es posible desarrollar controles sofisticados y diversos servicios, sin embargo un nivel **novato** como el de niños de nivel primario requiere una interfaz limpia y sencilla. Un sistema que admita a usuarios de un nivel **no definido** puede aceptar usuarios expertos, aficionados y novatos, pero debe basarse en los últimos, pues los expertos pueden hacer uso de controles simples, pero los novatos no pueden utilizar adecuadamente controles complejos.

Selección de las características descriptivas de la Marca

Al ingresar a la aplicación se produce una gama de percepciones denominadas *look and feel*. Este puede ser percibido como agradable o desagradable para el usuario. El ambiente que el usuario perciba puede ser armonioso, cómodo y apropiado si concuerda con sus expectativas, necesidades y experiencias.

Por este motivo es necesario realizar un análisis previo, para reconocer las características de la marca una vez identificados los usuarios reales, y de esta manera ofrecer un ambiente adecuado en relación con sus características y necesidades. En consecuencia es necesario considerar los siguientes puntos:

- Listar palabras que podrían describir las características de la marca. Por ejemplo: técnico, extravagante, elegante, moderno, académico, popular, divertido, serio, etc. Estas palabras definen como se espera que el usuario perciba la aplicación. La Figura 3.3 muestra una colección de sitios web con sus características;

Figura 3.3: Ejemplo de sitios web con marca definida y sus características



- Listar los atributos y servicios que la aplicación ofrece y la hacen diferente de las demás. Los atributos son cualidades o propiedades de una aplicación. Tratan de expresar lo positivo que se espera que el usuario diga de ella. Como ejemplos de atributos de la aplicación se puede citar la velocidad de interacción, la simplicidad entre otros. Los atributos deben ser siempre positivos, aún transformándolos en su opuesto, por ejemplo el opuesto de velocidad es lentitud, lo cual no significa algo

malo del todo, ya que algunos usuarios no son veloces interactuando con la interfaz. Los servicios son un conjunto de funciones con las que cuenta una aplicación. Un ejemplo de servicio puede observarse en las redes sociales *Facebook* y *Hi5*. Una de las diferencias entre ambas es la capacidad que ofrece *Hi5* al permitir la personalización del aspecto del entorno del usuario, servicio del que *Facebook* carece. La Tabla 3.3 muestra ejemplos de atributos y servicios que podrían representar a una marca determinada, en el formato de registro correspondiente.

Tabla 3.3: Ejemplo de registro de atributos y servicios de la marca

Atributos		Servicios
Normal	Opuesto	
Velocidad de Interacción	Lentitud de Interacción	<ul style="list-style-type: none"> • Edición de Fotografías • Recordatorios vía correo electrónico y móvil • Administración de listas de productos favoritos por usuario
Simple	Complejo	

Crear los elementos de la marca

Las características descriptivas de la marca ayudan a diseñar sus elementos. Se procede la descripción de los mismos forma secuencial:



- **Nombre**, la elección del nombre adecuado para la aplicación depende de sus objetivos. Sin embargo se puede observar que algunos sitios escogen nombres extravagantes con el propósito de llamar la atención del usuario. En síntesis, el nombre debe transmitir las características de la marca, debe ser fácil de asociar y recordar, porque no se podrá cambiar en tanto la aplicación este en línea. La Tabla 3.4 muestra el modo adecuado para registrar la selección del nombre de la aplicación Web. En la columna Etapa 1 anotar una lista de posibles nombres que pueda tener la aplicación, la cantidad no es importante pero se sugiere un número par entre diez y dieciséis. En la columna Etapa 2 seleccionar la mitad de los nombres escritos en la columna de Etapa 1 de acuerdo a preferencias del equipo de desarrollo. Finalmente en la columna Final registrar el nombre seleccionado. Este no podrá cambiarse;

Tabla 3.4: Ejemplo de registro de la selección del nombre de la aplicación

Etapa 1		Etapa 2	Final
Tipografía Online	TypeOnline	Typedia	Typedia
BuscaTipos	1001fonts	Letras Virtuales	
Typedia	FontManager	TypeBook	
Letras Virtuales	FontBook	TypeOnline	
TypeBook	BigFonts	BigFonts	

- **Color**, es fundamental a la hora de definir la identidad visual de una determinada marca. Para esto se sugiere una previa investigación en Psicología del Color asociada a cada tipo de usuario, así por ejemplo las mujeres se asocian a colores rojos, rosados y pasteles, pero este no es un estándar. Por otra parte es imprescindible la reproducción de estos colores con total fidelidad, evitando cualquier tipo de variación. Así, es necesario registrar el código cromático de cada color, que será respetado en cada artefacto gráfico referente a la marca. El máximo número de colores, estéticamente aceptables para la representación de la marca son tres. Es necesario señalar que estos colores no pueden modificarse una vez seleccionados. La Tabla 3.5 muestra el formato de registro adecuado para listar los colores seleccionados de la marca. Cabe notar que es necesario registrar tanto el código Hexadecimal del color como sus equivalentes en valores RGB y CMYK, ya que algunos programas gráficos utilizan el código RGB o Hexadecimal, y el CMYK es útil en caso de realizar materiales de impresos. Los valores de la Tabla son un ejemplo obtenido de la Reunión Regional de Ramas hecha en Bolivia del IEEE;

Tabla 3.5: Ejemplo de registro de colores de la marca

Color	Hexadecimal	RGB	CMYK
	#1F5FA6	R: 31 G: 95 B: 166	C: 91 M: 67 Y: 4 K: 1
	#FFFFFF	R: 255 G: 255 B: 255	C: 0 M: 0 Y: 0 K: 0

- **Logo**, debe estar íntimamente relacionado con el nombre y el color seleccionados, y debe contar con las siguientes características de diseño: forma, peso, contraste y

presencia (Ver Anexo E). Al igual que el color, una vez diseñado este no puede modificarse. La Tabla 3.6 muestra formato de registro adecuado para el logo que representa la marca. La casilla Variaciones del logo muestra variantes aceptadas del logo seleccionado que podrán reproducirse en diferentes medios. Al crear variantes del logo es posible utilizarlas en distintas partes de la aplicación y así no crear incongruencias. El máximo número de variantes es tres. Los logos que se muestran son un ejemplo obtenido de la Reunión Regional de Ramas hecha en Bolivia del IEEE;

Tabla 3.6: Ejemplo de Registro del Logo de la Marca



- **Eslogan**, es necesario dar una descripción de lo que realiza el sitio, de manera sencilla y corta, para que el usuario se haga una idea de lo que puede encontrar. La Figura 3.4 muestra el eslogan de la aplicación *Burstcast*, puede observarse como en seis palabras se ha logrado explicar el objetivo del sitio.

Figura 3.4: Eslogan de la aplicación *Burstcast*



Es necesario observar que el diseño de algunos de estos elementos pueden omitirse si se hubieran diseñado con anterioridad, en tal caso diseñar los que haga falta. Estos se ven con mejor claridad en la página principal, que debe ser diseñada con cuidado por ser la puerta de ingreso a la aplicación. Por otra parte apoyan la unidad visual del diseño en todas las páginas, ya que es un punto distintivo en ellas.

3.2.2. Diagramación

El objetivo de la fase de Diagramación recae en lograr precisión, eficiencia y orden entre las estructuras y elementos del armazón, mediante la construcción de una rejilla y la delimitación de espacios. Para poder lograr una diagramación adecuada a la aplicación es necesario conocer los elementos que podrían ingresar en cada armazón, es decir el diagrama de clases y el de estructura de navegación o mapa de navegación, además del usuario al que va dirigido para determinar el grado de complejidad, y la primera fase del presente modelo ya desarrollada. Cada aplicación es diferente y por ello requiere una diagramación que se ajuste a sus necesidades.

Artefactos

El artefacto obtenido durante esta fase es el Formato de Registro de las Plantillas de Composición. Estas poseen una significación parecida a la de boceto en este trabajo, ya que nos ayuda a delimitar, proporcionar, estandarizar e individualizar los espacios de una composición (ver Anexo L).

Actividades

El desarrollo de la fase de Diagramación consta de las siguientes actividades:

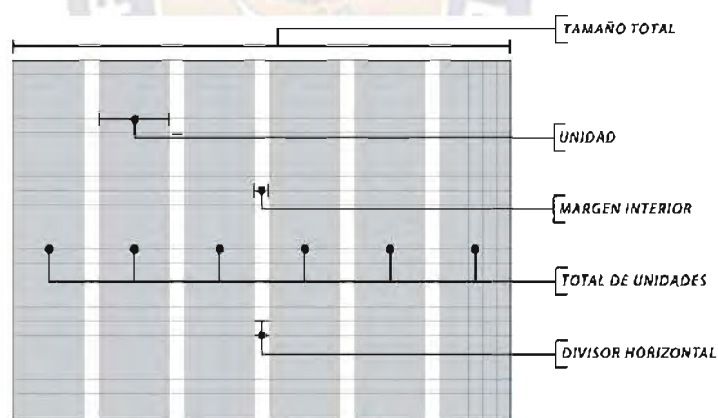
- Crear la rejilla de diagramación;
- Delimitar espacios;
- Numeración de espacios.

Crear la rejilla de diagramación

Cuando se comienza a diseñar una página web una tarea morosa es la especificación de tamaños y distancias entre los objetos. El diseñador se ve siempre absorto en el trabajo de aumentar o disminuir, según su criterio, los elementos y estructuras de la composición, casi siempre con poco acierto. El no acomodar de manera adecuada estos elementos provoca una sensación de desorden y desequilibrio que el usuario advierte.

Una rejilla de diagramación provee claridad, proporción, continuidad y orden en la delimitación de espacios y posicionamiento de estructuras y elementos del armazón y de ninguna manera debe limitar las oportunidades de composición. Esta no aparecerá en el diseño final pero su influencia será evidente. La rejilla de diagramación ocupa todo el espacio de la composición y está compuesta por las partes mostradas en la Figura 3.5:

Figura 3.5: Partes de la rejilla de Diagramación

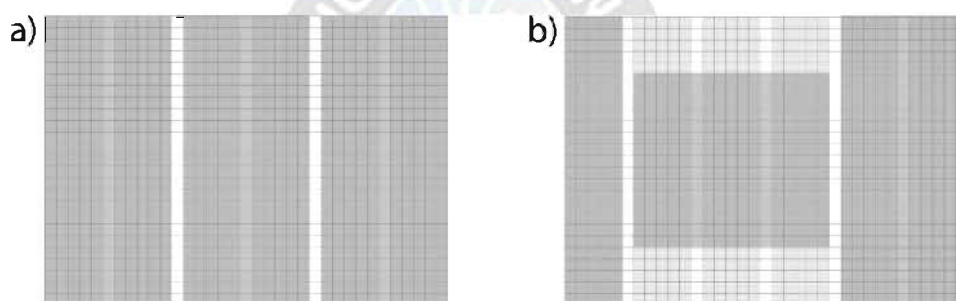


El **tamaño total** de la composición que puede definirse como el tamaño de la página web de forma horizontal, el tamaño vertical no es tomado en cuenta pues este puede acomodarse a las necesidades de cada composición. Las **unidades** son bloques rectangulares de una misma dimensión que dividen la composición y pueden combinarse para formar los espacios que determinarán la estructura visual del armazón. El **margen interior**, que es el espacio entre las unidades, sirve para separar a los espacios de tal manera que ofrezcan soltura al diseño. El **total de unidades** es la cantidad de unidades en las que se divide la composición, el total de márgenes (no

detallado en la Figura 3.5) podría definirse como el total de unidades menos uno. Los **divisores horizontales** son líneas horizontales que asegurarán orden y alineación y están separados por la misma distancia del margen interior.

Como se hubo señalado anteriormente, las unidades se combinan para formar espacios. La Figura 3.6 muestra una composición dividida en seis unidades que posteriormente se combinan en tres espacios uniformes y posteriormente en tres no uniformes (uno de una unidad, otro de tres y un último de dos unidades).

Figura 3.6: Composición dividida en: a) Espacios uniformes y b) No uniformes



Una delimitación como la mostrada en la Figura 3.10 inciso b es posible, siempre y cuando obedezca a un motivo mucho más práctico que estético.

Para la construcción de la rejilla es necesario considerar los siguientes pasos:

El **primer paso** es analizar la estructura básica a la que se acomodarán las necesidades específicas de la aplicación. Para hallar las proporciones adecuadas entre las partes de la rejilla se debe tomar en cuenta la siguiente fórmula:

$$TT = U \cdot TU + M(TU - 1) \quad (1)$$

Siendo:

- TT = Tamaño Total
- U = Unidad
- TU = Total de Unidades
- M = Margen interior

Interpretando la fórmula (1) tenemos que el tamaño total es igual a la suma de todas las unidades más la suma de todos los márgenes interiores, el total de márgenes es igual al total de unidades menos uno. Entonces el diseño de la rejilla

depende de tres factores fundamentales: el tamaño total de la composición, la cantidad de unidades (total de unidades) en las que se pretende dividir dicha composición, el margen interior y el tamaño de cada unidad.

El tamaño total de la composición es un valor que depende del tamaño de la aplicación, por ejemplo para páginas web informativas de empresas y servicios, dada la poca cantidad de elementos que contienen, se adopta un tamaño máximo de 720 px para la disponibilidad en todas las resoluciones de monitor. Sin embargo, las aplicaciones web, por la cantidad de servicios y operaciones que contienen trabajan en un tamaño de 960 px. Se puede observar que no es un número tan elevado como para no interactuar con resoluciones de 800 x 600 px, que es la resolución mínima.

El total de unidades determina el grado de precisión que se desea tener en cuanto a la composición. Es aconsejable usar un total de unidades que tenga un amplio número de divisores, por ejemplo el 12 (es divisible entre 2, 3, 4 y 6), es decir, la posibilidad de agrupar las unidades para formar dos, tres, cuatro y hasta seis espacios proporcionales de forma horizontal.

El margen interior define separación entre las unidades. Algo que se debe tomar en cuenta es que cada unidad debe ser divisible entre el margen, así el tamaño de cada unidad esta dado por la multiplicación del margen interior por un x:

$$U = x \cdot M \quad (2)$$

Siendo: $x = \text{Numero entero positivo}$

Las unidades son la incógnita a descubrir.

Para la resolución de la ecuación (1) es necesario tener el total de unidades y el tamaño del margen adecuado. Otro punto importante es especificar el intervalo de valores en los que se encontrará el Tamaño Total de la composición. Este está dado por un intervalo debido a que la suma de de las variables puede sobrepasar o ser menor a un valor que se considere determinado.

La Tabla 3.7 muestra un conjunto de valores ideales que es posible asignar a las variables dependiendo del tamaño de la aplicación. Es necesario aclarar que estos valores no son obligatorios y el diseñador puede decir si dispone o no de los mismos.

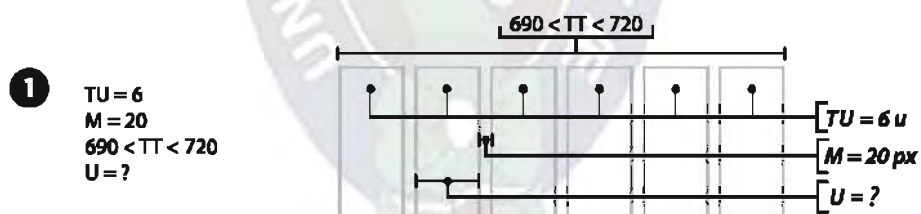
Tabla 3.7: Dimensiones aproximadas para las partes de la rejilla

Medida/Variable	Tamaño Total	Margen Interior	Total de Unidades
Mínimo	800 px -> (690 px a 720 px)	$10 < x < 15$	6
Medio	1024 px -> (930 px a 960 px)	$16 < x < 20$	12
Máximo	aún no determinado	$21 < x < 30$	18

Tomando como ejemplo el diseño de una aplicación orientada a la venta de equipo electrónico orientada a la Web 2.0 para aumentar las ventas mediante las recomendaciones de los compradores. La rejilla no debe ser de una precisión estricta por lo tanto tomando dimensiones aproximadas de la Tabla 3.7 es posible establecer un total de unidades de 6, un margen medio de 20 px y debido a su condición de tienda virtual debe estar disponible para resoluciones mínimas por eso se escoge el tamaño total mínimo de 800 px con un intervalo de (690 px a 720 px).

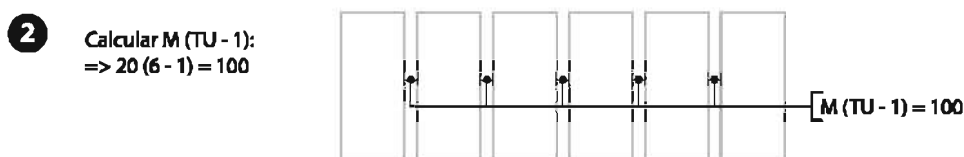
De esta manera, se aplica el primer paso para el ejemplo en la Figura 3.8, para una aplicación con poca cantidad de información.

Figura 3.8: Paso 1 de la creación de la rejilla de diagramación



El **segundo** paso es calcular la distancia que todos los márgenes ocupan, dado el valor que se ha asignado. La parte de la fórmula (1) que encuentra ese valor es **M(TU-1)**. Para entender este paso es necesario observar el ejemplo formulado en la Figura 3.9.

Figura 3.9: Paso 2 de la creación de la rejilla de diagramación



El **tercer paso** es calcular la distancia que todas las unidades ocupan, sin embargo la unidad es la incógnita. La fórmula (2) puede ayudar a simplificar el proceso de cálculo, reemplazándola en la parte de la fórmula (1) que encuentra ese valor: **$U \cdot TU$** . El resultado queda en función de x . Observar la continuación del ejemplo formulado en la Figura 3.10.

Figura 3.10: Paso 3 de la creación de la rejilla de diagramación

$$\begin{array}{ll} \text{3} & \text{Calcular } U \cdot TU: \\ & \text{pero: } U = x \cdot M \\ & \text{donde } x = \text{entero} \end{array} \quad \begin{array}{l} \Rightarrow x \cdot M \cdot TU \\ \text{reemplazando los datos:} \\ \Rightarrow x \cdot 20 \cdot 6 = x \cdot 120 \end{array}$$

El **cuarto paso** es calcular el valor de x , mediante una serie de aproximaciones, de tal manera que la suma entre ese valor y el resultado obtenido en el segundo paso (**$M(TU-1) = 100$**) no salga del intervalo determinado para el tamaño total. Observar la continuación del ejemplo formulado en la Figura 3.11.

Figura 3.11: Paso 4 de la creación de la rejilla de diagramación

$$\begin{array}{l} \text{4} \quad \text{Calcular } x \text{ de tal manera que:} \\ \quad 690 < x \cdot 120 + 100 < 720 \\ \text{para } x = 4 \Rightarrow 4 \cdot 120 + 100 = 580 \quad \times \\ \text{para } x = 5 \Rightarrow 5 \cdot 120 + 100 = 700 \quad \checkmark \\ \text{para } x = 6 \Rightarrow 6 \cdot 120 + 100 = 820 \quad \times \end{array}$$

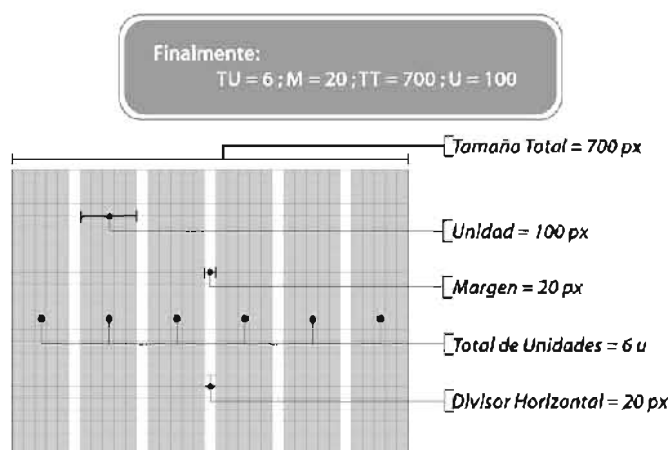
El **quinto paso** es hallar el valor de cada unidad reemplazando los valores hallados en la fórmula (2) como muestra el ejemplo de la Figura 3.12.

Figura 3.12: Paso 5 de la creación de la rejilla de diagramación

$$\begin{array}{l} \text{5} \quad \text{Calcular } U: \\ \quad U = x \cdot M \\ \quad U = 5 \cdot 20 \\ \quad U = 100 \end{array}$$

Finalmente graficar la rejilla con los valores encontrados, como muestra el ejemplo de la Figura 3.13. Como se puede observar, los divisores horizontales están separados el uno del otro por la misma distancia del margen interior.

Figura 3.13: Rejilla de diagramación de ejemplo



La rejilla final puede extenderse verticalmente de manera indefinida, sin embargo existen restricciones en cuanto al tamaño que esta adquiera, todo depende de la delimitación de espacios y el contenido que estos pudieran tener.

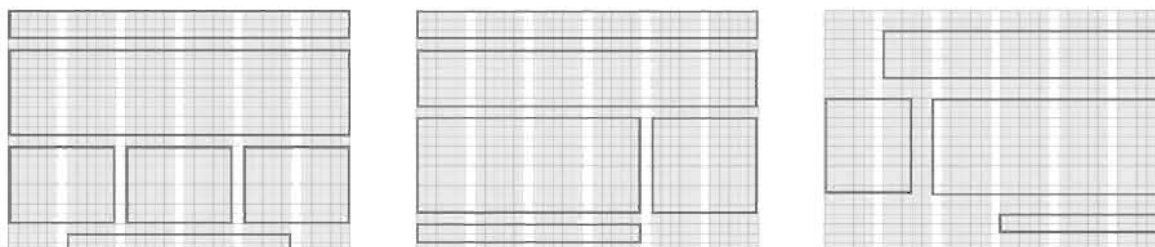
Delimitar Espacios

Al contar con una rejilla de diagramación, el siguiente paso es delimitar los espacios del armazón. Para esto es necesario aplicar conceptos de composición y jerarquía concreta detallados en el Anexo I. Posteriormente es necesario seguir los siguientes pasos:

Primero determinar si la pagina principal tendrá una distribución distinta a las demás páginas. En tal caso delimitar los espacios necesarios para su contenido. La página principal es la única en la que existe una carencia de espacios fijos y es diagramada en una plantilla determinada debido a su importancia, ser la puerta de entrada a la aplicación. Por otra parte, también es permisible que la diagramación ajustada a alguna plantilla ya realizada, todo depende de la aplicación y del diseñador. Los objetos que mantendrán la unidad entre la página principal y las demás son los elementos de la marca diseñados en la fase anterior, estos deben estar presentes en el diseño de la misma, además las estructuras de entrada a la aplicación, cambio de contraseña, los servicios que ofrece (para que sirve) y de ser posible a quien está orientada la aplicación. En las aplicaciones web 2.0 no es recomendable la integración de un *intro* debido al carácter utilitario de las mismas, el usuario quiere lograr sus

objetivos, esa característica puede acoplarse a páginas empresariales o relativas a eventos. La Figura 3.14 muestra un conjunto de posibles diagramaciones para una página principal.

Figura 3.14: Ejemplos de diagramación para una página principal



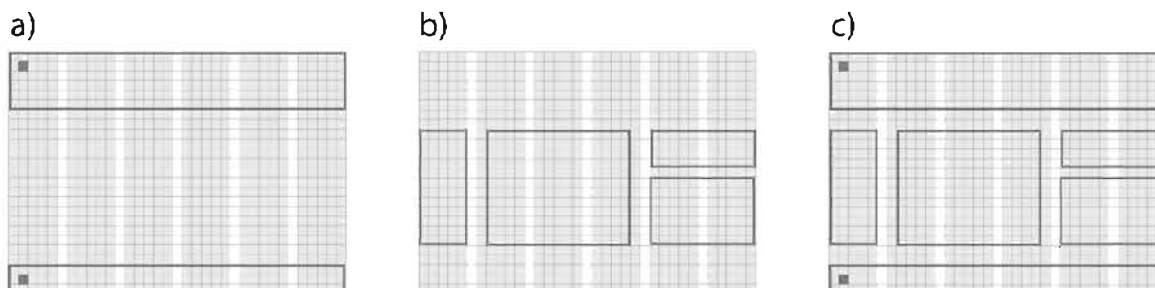
Segundo, delimitar los espacios fijos. Estos ayudan a encontrar unidad visual y puntos de referencia en la aplicación, delimitar al menos un espacio fijo refuerza estos conceptos, y es indispensable. Se mantendrán de igual forma en cada plantilla. Se puede reconocer como componentes de estos espacios a los elementos de la marca (logo, eslogan), la navegación global (navegación principal, y otra navegación global secundaria en el *header* y/o *footer*), indicadores de estado (por ejemplo, "Entró al sistema como..."), funciones globales o de alto nivel (buscar, salir del sistema, registro) y banners, aunque estos pueden ser variables en su contenido pero no en el espacio que ocupen. El estereotipo para identificar los espacios fijos en una composición es un cuadrado rectangular negro como se muestra en la Figura 3.15 inciso a.

Tercero, delimitar los espacios variables. Un armazón debe acomodarse a una determinada plantilla que se ajuste a sus necesidades. Generalmente se volverán a delimitar en cada una, sin embargo el conocer el posible contenido que tiene cada armazón mediante la metodología utilizada ayuda a realizar la tarea de manera consciente y no al azar. Estos espacios no poseen ningún estereotipo y se representan como muestra la Figura 3.15 inciso b.

Para el ejemplo de la tienda virtual la Figura 3.15 inciso c muestra la diagramación de espacios fijos y variables, además se puede advertir que el espacio

en blanco compuesto por el margen entre espacios proporciona holgura a la composición.

Figura 3.15: Delimitación de espacios en un armazón web

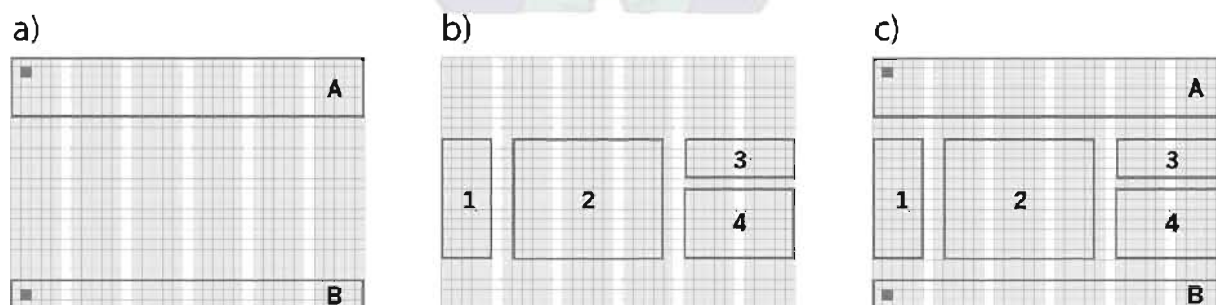


Numeración de espacios

Con los espacios ya delimitados, el último paso corresponde a su enumeración. Este paso es necesario para la fase de Robustez, ayuda en la identificación de los diferentes elementos y estructuras de cada espacio, de esta manera se puede controlar la posición de cada objeto.

Los espacios fijos se proceden a enumerar con letras, sin embargo los espacios variables se enumeran con números naturales positivos.

Figura 3.16: Numeración de espacios en un armazón web

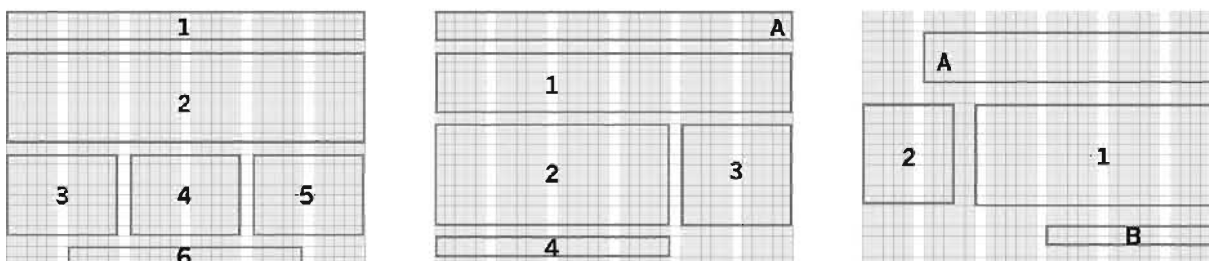


La Figura 3.16 incisos a y b muestra la enumeración correspondiente a los espacios fijos y variables de la Figura 3.15. La Figura 3.16 inciso c muestra la plantilla de composición completa que debe ser añadida al formato de registro.

En la página principal se debe proceder a enumerar con números positivos si no se conserva ningún espacio fijo, pero si se mantiene alguno o todos proceder a

enumerarlos con las letras que les corresponde como muestran las imágenes de la Figura 3.17. Luego registrar la composición en el formato de registro de las plantillas de composición.

Figura 3.17: Ejemplos de numeración para una página principal



El uso capas o tablas para el diagramado que realice el programador es indistinto y no se toma en cuenta para este trabajo.

3.2.3. Diseño

El objetivo de la fase de diseño es proyectar el armazón web de cada página de la aplicación.

La fase de Diseño requiere de las plantillas de composición y la rejilla de la anterior fase, además del modelo de estructuras de navegación o, de no contar con el mismo, el mapa de navegación de la aplicación y el diagrama de clases.

Es necesario detallar que los pasos de cada sub fase son cíclicos para construir cada objeto del armazón.

Artefactos

El artefacto obtenido durante esta fase es la parte del Formato de Registro de los Armazones Web (ver Anexo M). De esta manera se concluye el diseño de los armazones web.

3.2.3.1. Contenido

El objetivo de esta sub fase es organizar los objetos de contenido de acuerdo a un orden y función que se ajuste al objetivo de cada página.

El contenido es el objetivo de la página, es el motivo por el que esta ha sido construida, de esta manera la jerarquía debe establecerse en esta sub-fase.

Es necesario detallar el formato el formato del texto que se registrará en el almacén, para ello se puede observar la Tabla 3.8, en la que se detalla los diferentes formatos que pueden estar involucrados.

Tabla 3.8 Formato de texto del almacén

Formato	Descripción
< texto >	El texto detallado entre signos de mayor y menor representa a datos obtenidos de la base de datos.
n <i>k</i> <u>s</u> a A	El grosor (negrita) y el estilo de las fuentes (itálica, subrayado, minúscula o mayúscula) se respeta en el diseño del almacén y se mantendrá en el diseño final
✓ • ✕ ★ 1 a	Las viñetas o numeraciones se mantienen en el almacén pero pueden mejorarse visualmente en el diseño final

Actividades

Para el desarrollo de la sub-fase de Contenido se debe considerar las siguientes actividades:

- Determinar tipo de contenido;
- Designar objetos de contenido;
- Asignar jerarquía visual.

Determinar tipo de contenido

Como se había señalado en el Anexo D, las estructuras de contenido se clasifican en predefinido y personalizado. Una estructura de contenido predefinido es aquella elaborada por otros diseñadores pero puede reutilizarse para seguir un mismo propósito, por ejemplo: el menú tipo carrusel (ver Anexo G). Una estructura de contenido personalizado es aquella construida por el mismo diseñador de la aplicación y, aunque generalmente no se pueda reutilizar, sirve para una tarea específica, por

ejemplo la estructura destinada a la autenticación de usuarios que ingresan a la aplicación.

Las estructuras de contenido personalizado se dividen pueden ser de cuatro tipos como muestra el esquema de la Figura 3.18.

Figura 3.18: Clasificación de las estructuras de contenido personalizado



Los objetos de contenido pueden ser utilizados de forma central, es decir que ser el objetivo de la página, o de modo secundario, es decir que apoyan al objeto central. La Tabla 3.9 muestra los tipos de estructuras que pueden ser centrales y secundarias, los estereotipos provienen de los objetos clasificados en los Anexos F y G.

Tabla 3.9 Objetos de contenido central y secundario

CONTENIDO	ESTRUCTURA		ELEMENTO
	PREDEFINIDA	PERSONALIZADA	
CENTRAL	<ul style="list-style-type: none"> Reproductor de audio o video 	<ul style="list-style-type: none"> Formulario Catálogo Panel 	Estáticos: <ul style="list-style-type: none"> Texto Imagen Diagrama Tabla Animación
SECUNDARIO	<ul style="list-style-type: none"> Control Deslizante 	<ul style="list-style-type: none"> Formulario Catálogo 	Estáticos: <ul style="list-style-type: none"> Texto Imagen Diagrama Tabla Animación

Formulario

Un formulario utiliza elementos de dinámicos y de acción (Ver Anexo F). Los formularios pueden ser de autenticación (*login*), aceptación (de acuerdo con los

términos de referencia) o de datos (formularios normales). En síntesis, cualquier estructura que haga uso de elementos dinámicos, es un formulario. Algunas veces los formularios son pequeños como es el caso de los de autenticación y aceptación, pero sin importar su tamaño los pasos a seguir para realizar su construcción se mantienen y son los siguientes:

- Asignar título y grupos de datos al formulario, para identificarlo y conocer el objetivo del mismo. Si el formulario es muy extenso agruparlo en secciones de acuerdo a relación de datos;
- Seleccionar campos, se obtiene los campos del modelo de clases que sean necesarios y relevantes al formulario;
- Etiquetar campos, una vez obtenidos los campos se procede a etiquetar los campos con nombres adecuados al lenguaje del usuario.

Un ejemplo de catálogo se muestra en la Figura 3.19 de la página *google.com*.

Figura 3.19: Ejemplo de formulario de la página *google.com*

The image shows the Google Advanced Search interface. At the top, it says 'Google Búsqueda Avanzada' and 'Experimentos de Búsqueda y Configuración de Google'. Below this, there are several sections for refining search results:

- Buscar en las páginas:** Includes options for 'contenido en páginas', 'con la misma extensión', 'sin algunas de las palabras', and 'sin las palabras'.
- idioma:** 'Buscar páginas escritas en' with a dropdown menu.
- Región:** 'Buscar páginas escritas en' with a dropdown menu.
- Formato de archivo:** 'Buscar páginas con estos formatos de archivo' with a dropdown menu.
- Fecha:** 'Buscar las páginas con estos rangos de fechas' with a dropdown menu.
- Presencia:** 'Buscar resultados en los que me interesa estar presente' with a dropdown menu.
- Dominio:** 'Buscar resultados de dominios o sitios Web' with a dropdown menu.
- Derechos de autor:** 'Buscar resultados que' with a dropdown menu.
- SafeSearch:** 'Si: Sí No' and 'Filtrar contenido'.

Catálogo

Un catálogo es una colección de estructuras simples o elementos que se repiten, diferentes únicamente por los datos individuales que poseen, este también puede ser un listado de datos sin incluir imágenes con las correspondientes opciones. Los pasos para su construcción son los siguientes:

- Asignar nombre de catálogo, para identificarlo y conocer su objetivo;
- Determinar número máximo de ítems, un catálogo no puede sobrepasar los veinte ítems por la sobrecarga que esto implica;

- Asignar funciones a ítems, determinar qué elementos realizarán las opciones individuales de cada ítem, por ejemplo cerrar, comprar, texto de información del ítem.

Un ejemplo de catálogo se muestra en la Figura 3.20 de la página *apple.com*.

Figura 3.20: Ejemplo de catalogo de la página *apple.com*



Panel

Un panel es una colección de operaciones que se realizan sobre uno o un conjunto de objetos y los modifican de forma general.

- Determinar propósito del panel, que es lo que hará y cuantos objetos contendrá;
- Determinar funciones del panel, listar las posibles funciones que el panel pueda ofrecer y que hará cada una de ellas.

Un ejemplo de panel se muestra en la Figura 3.21 de la página *flickr.com*.

Figura 3.21: Ejemplo de panel de la página *flickr.com*



Designar objetos de contenido

Los objetos de pueden manejar el contenido de la aplicación se detallan en la Tabla 3.10.

Tabla 3.10 Designación de objetos de contenido del armazón

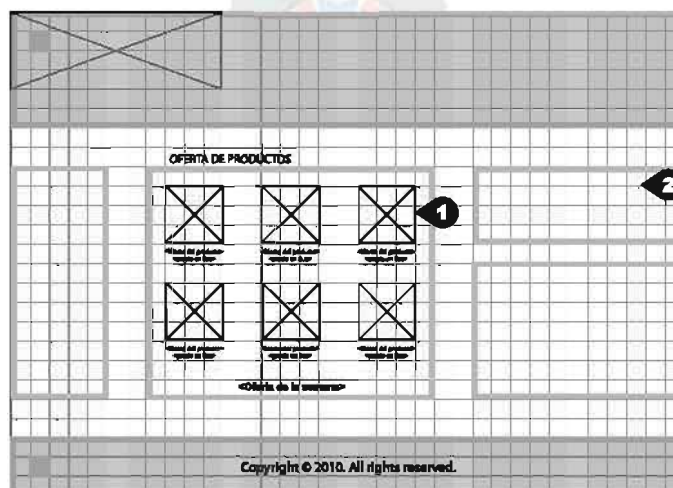
Estructura personalizada	Estructuras que utiliza	
	Estructura predefinida	Elementos
Formulario	<ul style="list-style-type: none"> Control Deslizante 	Dinámicos: <ul style="list-style-type: none"> Entrada de texto Contraseña Campo de Entrada de texto Lista de Selección Lista de Selección Múltiple Casilla de Verificación Colección de Casillas de Verificación Botón de opción Colección de Botones de Opción Acción: <ul style="list-style-type: none"> Botón Vínculo Icono
Catálogo		Estáticos: <ul style="list-style-type: none"> Texto Imagen Diagrama Dinámicos: <ul style="list-style-type: none"> Lista de Selección Casilla de Verificación Colección de Casillas de Verificación Botón de opción Colección de Botones de Opción Acción: <ul style="list-style-type: none"> Botón Vínculo Icono

Panel	<ul style="list-style-type: none"> • Reproductor de audio o video 	<p>Estáticos:</p> <ul style="list-style-type: none"> • Texto • Imagen • Diagrama <p>Dinámicos:</p> <ul style="list-style-type: none"> • Lista de Selección • Casilla de Verificación • Colección de Casillas de Verificación • Botón de opción • Colección de Botones de Opción <p>Acción:</p> <ul style="list-style-type: none"> • Botón • Vínculo • Icono
-------	--	--

Asignar jerarquía visual

Posteriormente es necesario señalar los espacios que serán el centro atención del armazón web. Cabe recalcar que debe escogerse un máximo de dos espacios, siendo el estereotipo 1 el que tendrá mayor jerarquía visual que el 2. Se especifica los medios de énfasis visual en la Guía de Estilo Visual mejor detallado en la fase de Visualización. Es importante señalar que la jerarquía está determinada por el objetivo de cada página. La Figura 3.22 muestra un ejemplo del estereotipo utilizado para la jerarquía visual del armazón.

Figura 3.22: Estereotipo de la jerarquía visual



3.2.3.2. Navegación

La mejor navegación es la que no se nota en absoluto, es decir, la que está disponible todo el tiempo pero no entorpece los objetivos del usuario. Los objetos de navegación deben estar disponibles sin entorpecer el objetivo central de cada página web. El diseño de una adecuada navegación para la aplicación es importante ya que provee acceso a la información, muestra la ubicación del usuario (donde está, donde puede ir, que puede hacer), sugiere el “acerca de” de una aplicación, refleja a la marca, afecta la credibilidad de la aplicación y por último afecta a la decisión final del usuario. Una marca se devalúa con pobre navegación, los usuarios continúan usando una aplicación en la que pueden ubicarse.

El objetivo de la fase de Navegación es construir una navegación adecuada que ayude al usuario a desplazarse sin dificultades a través de la aplicación.

Actividades

Para el desarrollo de la fase de Navegación se debe considerar las siguientes actividades:

- Determinar tipo de navegación;
- Designar objetos de navegación;

Determinar tipo de navegación

Las estructuras de navegación estructural y elementos sueltos utilizados con este fin pueden clasificarse en tres categorías (ver Figura 3.23).

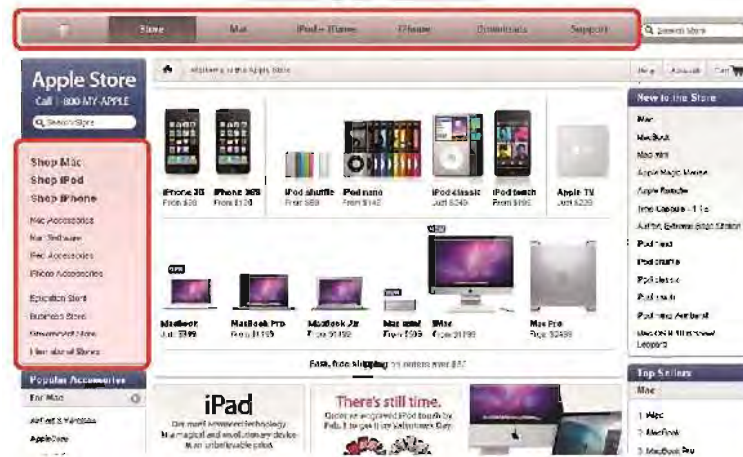
Figura 3.23 Clasificación de la Navegación del armazón



Navegación estructural

Representa una **jerarquía** de vínculos que generalmente sigue la estructura de la aplicación web. Permite a los usuarios movilidad a través de los diferentes puntos de la jerarquía. La Figura 3.24 muestra a la página de *apple.com* en la que se resalta la navegación estructural del sitio;

Figura 3.24: Navegación estructural de la página *apple.com*



Para diseñar la navegación estructural de nivel uno o dos de un sitio deben considerarse los siguientes pasos:

- **Definir las opciones a mostrar**, qué opciones mostrar, y cuales promover, debe depender del contexto de la página. La metodología seleccionada y la primera fase del presente modelo son aspectos que ayudan a determinar el tipo de aplicación, sus partes y los usuarios a los que va dirigida. De este modo la navegación permanente se enfoca al objetivo de sitio, la marca, el tipo de usuario, los objetivos del usuario y el conjunto de clases encontradas.

La cantidad de opciones también es crucial. Es necesario seleccionar los vínculos necesarios que puedan ser mantenidos constantemente y evitar las páginas “en construcción” ya que los usuarios no retornan a una aplicación que no ofrece aquello que no se encuentra. El número de opciones debe aproximarse a un número de doce por la sobrecarga visual.

Tomando el ejemplo de la fase de diagramación respecto a la tienda virtual, las opciones que deben permanecer siempre disponibles son: Volver a la página de inicio, un campo de búsqueda de datos, oferta de los productos, descarga del software necesario de los equipos, apoyo técnico en caso de encontrar fallas, ayuda sobre el sitio, y por supuesto ingreso a la cuenta del cliente.

- **Especificar etiquetas para las opciones**, Para etiquetar las opciones de manera clara se debe tomar en cuenta que las palabras utilizadas en cada opción de navegación construyen una imagen de lo que se puede encontrar, y a donde es posible ir, y forma parte de la experiencia de la marca. De esta manera, lo que se resuelve decir en cada opción comunica parte de la esencia de la marca.

Los dos errores más comunes en la creación de etiquetas son: Primero, demasiada generalidad, opciones como Información General solo levantan más cuestionantes en la mente del usuario, se debe puntualizar de manera adecuada. Segundo, el uso de verbos transitivos sin objeto. Los verbos transitivos son los que involucran hacer algo para alguien, es decir, requieren que un objeto dé sentido. Los verbos intransitivos no necesitan un objeto, por ejemplo "Buscar". Sin embargo Explora, Experimenta, y Contacta son verbos transitivos, no dicen mucho a menos que se refieran a un objeto como Contáctanos, el nosotros queda comprendido implícitamente y así el significado está completo. Volviendo al ejemplo de la aplicación para ventas de equipo electrónico del anterior punto es posible refinar las etiquetas de las opciones anteriormente citadas como muestra la Tabla 3.11:

Tabla 3.11 Ejemplo de especificación de etiquetas para opciones de navegación

Volver a la página de inicio ->	Inicio
Campo de búsqueda ->	Buscar
Oferta de los productos ->	Tienda
Descarga de software ->	Descargas
Apoyo técnico ->	Soporte
Ayuda sobre el sitio ->	Ayuda
Ingreso a la cuenta de usuario ->	Entrar al sistema

- **Jerarquizar las opciones**, esto depende de una combinación de lógica y objetivos de la marca. Usando lógica, si una página es leída de izquierda a derecha entonces las opciones de navegación más cercanas al origen (esquina superior - izquierda) tendrán superioridad espacial sobre otras opciones, por ende se espera lo más relevante o importante. Los objetivos del sitio, o la imagen de la marca pueden influir en el orden de las opciones. Si se observa que alguna de las opciones no forma parte del grupo es necesario considerar la posibilidad de crear otro grupo de opciones. Para entender este punto de mejor manera y retornando al ejemplo de actividades anteriores, la jerarquía adecuada de las opciones de navegación se muestra en la Tabla 3.12:

Tabla 3.12 Ejemplo de jerarquía para las opciones de navegación

Inicio	
Tienda	
Descargas	
Soporte	
Buscar	
	Ayuda
	Entrar al sistema

Debido a que el centro de las actividades de esta aplicación es la venta de equipo vía *online*, la opción de la tienda debe ser la primera, solo antecedida por el botón de volver al inicio. La opción de soporte se muestra en penúltimo lugar por el siguiente motivo, se espera que el usuario solicite soporte después de que ha encontrado el producto requerido. Se puede observar que las opciones de ayuda y entrar al sistema no forman parte de la colección de opciones original, por lo que deben separarse y formar un nuevo conjunto.

Navegación adaptativa

Es un tipo especial de una navegación, los enlaces son generados en un proceso de **filtrado colaborativo o social**. El proceso depende de una clasificación algorítmica basado en el comportamiento del usuario. La Figura 3.25 muestra navegación adaptativa de una de las secciones de la página *apple.com*.

Figura 3.25: Navegación adaptativa de la página *apple.com*

Para diseñar la navegación adaptativa de un sitio deben considerarse los siguientes pasos:

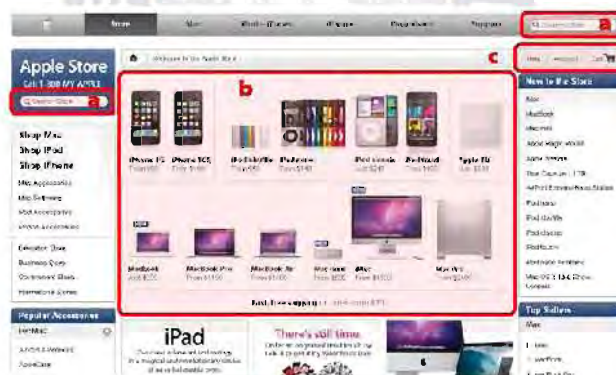
- Determinar criterio de adaptación**, la navegación adaptativa siempre se observa desde el punto de vista de los usuarios, y puede enfocarse de dos maneras: para **personalizar** o para **optimizar**. Ambas se centran en el concepto de metadatos (datos de los datos), para lograr sus objetivos. La personalización se ajusta a las necesidades de un usuario en específico, para determinar el grado de personalización es necesario observar los objetivos del usuario y de la aplicación. Una estructura muy útil en este punto es la *nube de tags* en la que se incrementa de tamaño automáticamente los vínculos más usados. La optimización intenta mejorar la navegación de la aplicación por completo. Las aplicaciones web 2.0 se caracterizan por la elevada cantidad de información que poseen. La cuestionante es como ayudar al usuario a encontrar de manera óptima dicha información. Para solucionar este percance surge el concepto **promocionar**, poner los vínculos más importantes, recientes o recomendados que puedan facilitar su búsqueda. La Figura 3.23 muestra navegación adaptativa de una de las secciones de la página *apple.com* con el concepto de promoción. Ambas maneras de ver la navegación adaptativa deben ser pensadas por el diseñador y analista de forma concreta, no permitiendo ciertas libertades al usuario, ya que el usuario no sabe lo que otros usuarios piensan o ven y de ninguna manera sabe que es lo mas optimo para él y los demás;

- **Determinar cantidad de elementos**, cuando se ha determinado el criterio de adaptación adecuado, es necesario saber qué cantidad de elementos ingresarán en el almacén.

Navegación asociativa

Conecta diferentes páginas de temas similares sin importar la ubicación en la estructura. La Figura 3.26 resalta la navegación asociativa de una de las secciones de la página *apple.com*. Esta puede ser representada por estructuras de navegación asociativa (Ver Anexo G) como la búsqueda como muestra de la Figura 3.26 bloque a, también representarse embebida en texto como en títulos, también puede ser imágenes como muestra la Figura 3.26 bloque b, también puede representarse por iconos sueltos como el carrito de compras o por vínculos agrupados en estructuras como muestra la Figura 3.26 bloque c;

Figura 3. 26: Navegación asociativa de la página *apple.com*



La navegación asociativa puede clasificarse de tres maneras:

- Asociación estructural, relacionada a la estructura de la misma aplicación, por ejemplo el botón ir a inicio;
- Asociación de citación, referencia a personas, productos, organizaciones y lugares, y definiciones de una palabra en una página interna o externa;
- Asociación abierta, referente a imágenes o vínculos sueltos por ejemplo como publicidad.

La navegación asociativa ostenta un alto grado de libertad a diferencia de la estructural en la que el orden rige su construcción. Es ilógico y no usable poner los vínculos asociativos en cualquier parte de la página sin una aparente relación con el contexto ya que estos aparentarían ser nodos sueltos.

El tratamiento gráfico de iconos se detallada en la fase de visualización, pero su posición debe obedecer al carácter del contexto en el que se halla. Por ejemplo la Figura 3.24 bloque c muestra la sección de tienda de la pagina *apple.com*, por ende es lógico encontrar cerca de los productos el ícono de carro de compras.

Los vínculos asociativos también pueden estar embebidos en el contenido que el usuario comparta, en tal caso es imposible representarlos en el armazón, la única solución es predefinir un estilo y formato adecuado que los visitantes puedan identificar con facilidad.

Designar objetos de navegación

Para la navegación estructural las mejores estructuras de navegación son aquellas que muestran todas las opciones que se ha especificado, la jerarquía también está implícita por ello las estructuras predefinidas de navegación estructural son adecuadas. En el Anexo G se puede encontrar a navegación de árbol, las barras de navegación, las lengüetas de navegación o simplemente usar menús de vínculos. Sin embargo es necesario recalcar que cada una de estas estructuras posee un límite visual reconocible por el usuario. La sobrecarga de opciones supone un error grave. Seleccionada la estructura de navegación lo siguiente es acomodarla a la rejilla de diagramación. Esta tarea se facilita habiendo definido la rejilla adecuada. La Tabla 3.13 muestra los objetos que deben utilizarse para la representación de cada categoría de navegación en el armazón web.

Tabla 3.13: Designación de estructuras de navegación

NAVEGACIÓN	ESTRUCTURA	ELEMENTO
ESTRUCTURAL	<ul style="list-style-type: none"> • Navegación de Árbol • Lengüeta de Navegación • Barra de Navegación 	Ninguno

ADAPTATIVA	<ul style="list-style-type: none"> • Nube de Tags • Barra de Navegación vertical 	Ninguno
ASOCIATIVA	<ul style="list-style-type: none"> • Paginación • Rastro de Migas • Carrusel • Búsqueda • Barra de Navegación vertical 	Acción: <ul style="list-style-type: none"> • Vínculo • Ícono • Imagen Vínculo

3.2.3.3. Robustez

Cuando el armazón web ha tomado forma mediante los elementos y estructuras que lo componen el siguiente paso es determinar los eventos que pueden desencadenarse después de alguna acción del usuario.

Actividades

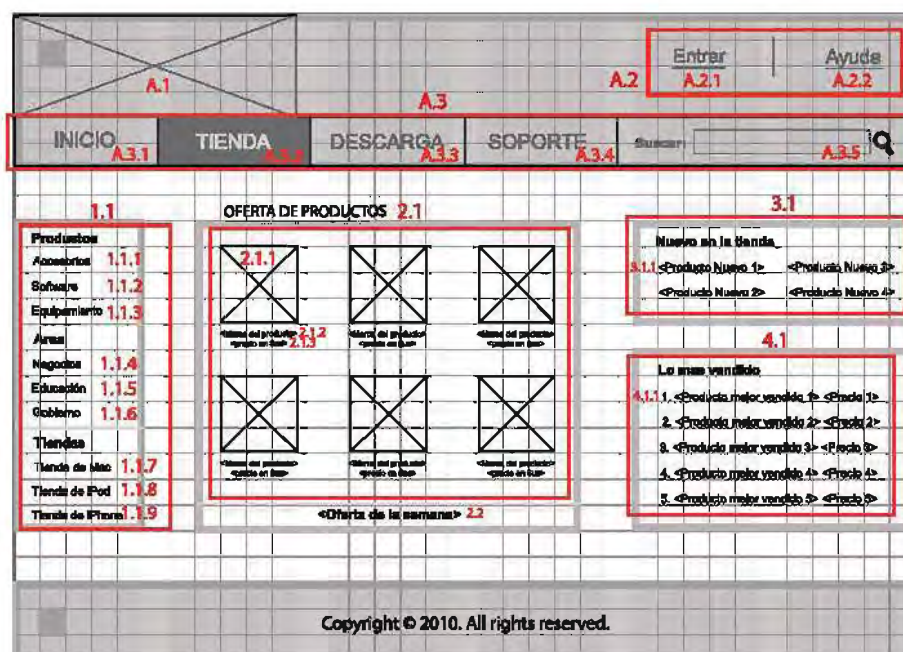
Para el desarrollo de esta fase se debe proceder a realizar las siguientes actividades:

- Etiquetar objetos;
- Determinar acciones de cada objeto;
- Registro de código conciso.

Etiquetar objetos

Mediante la numeración de espacios determinada en la fase de diagramación ahora se procede a etiquetar cada elemento que realice alguna acción o que necesite una descripción. Dado el ejemplo de la tienda virtual planteado en las anteriores sub-fases es posible realizar el etiquetado de objetos como muestra la Figura 3.27. Para la estructura catálogo se puede etiquetar la primera estructura y luego las demás solo figuran como muestra de posición, lo mismo para navegación adaptativa y estructuras que involucren repetición por obtención de datos de la base de datos.

Figura 3.27: Ejemplo de etiquetado de objetos del armaz3n



La numeraci3n debe realizarse como muestra la Figura 3.28 a un nivel estructuras y elementos m3ximo de tres, por eso es importante no construir estructuras anidadas una en la otra a tal grado.

Figura 3.28: Criterio de numeraci3n para el etiquetado de objetos del armaz3n

Ejemplo Esterotipo Objeto	A . 1 letra o n3mero n3mero espacio . elemento
Ejemplo Esterotipo Objeto	A . 1 . 1 letra o n3mero n3mero n3mero espacio . estructura . elemento
Ejemplo Esterotipo Objeto	A . 1 . 1 . 1 letra o n3mero n3mero n3mero n3mero espacio . estructura . estructura . elemento

Determinar acciones de cada objeto

Cada objeto de la p3gina posee acciones que deben ser analizadas. Si el objeto de an3lisis tiene un evento asociado a 3l, por consecuencia tendr3 una acci3n y una zona de carga, sin embargo algunos elementos son **est3ticos**, en tal caso es posible **obviar** todas las tareas excepto la descripci3n del objeto.

- **Descripción del objeto**, Para la correcta descripción de una función es necesario detallar la etiqueta del objeto, el tipo de objeto que se detalla, realizar una descripción textual breve, y por último establecer pre y post condición pero solo para estructuras enteras y no para simples elementos. Las Figuras 3.29 y 3.30 muestran el modelo de descripción de los objetos del armazón y ejemplos asociados;

Figura 3.29: Modelo de descripción de objetos del armazón

<etiqueta de objeto>, <estereotipo de elemento o estructura> (Nombre de objeto si corresponde)
Descripción: <descripción>
pre-condición: <pre-condición solo estructuras>
post-condición: <post-condición solo estructuras>

Figura 3.30: Ejemplo de descripción de objetos del armazón

A.1, Imagen

Descripción: *Logotipo del sitio*

2.1, Catálogo

Descripción: *Oferta de productos recientes de la tienda virtual*

pre-condición: *Ingresar a la sección de la tienda*

post-condición: *Muestra producto individual que se seleccione*

- **Identificación de eventos**, los elementos y las estructuras poseen eventos que deben analizarse para establecer el modo y medio de respuesta adecuada. La Tabla 3.14 muestra una lista de posibles eventos y una descripción detallada de la forma en cómo se desencadenan;

Tabla 3.14: Lista de eventos

Evento	Descripción
<i>onclick</i>	Se produce cuando se da una pulsación o clic al botón del ratón sobre un elemento de la página, generalmente un botón o un enlace
<i>ondragdrop</i>	Se produce cuando un usuario suelta algo que había arrastrado sobre la página web
<i>onfocus</i>	El evento <i>onfocus</i> es lo contrario de <i>onblur</i> . Se produce cuando un elemento de la página o la ventana ganan el foco de la aplicación

<i>onkeydown</i>	Este evento se produce en el instante que un usuario presiona una tecla, independientemente que la suelte o no. Se produce en el momento de la pulsación
<i>onkeypress</i>	Ocurre un evento <i>onkeypress</i> cuando el usuario deja pulsada una tecla un tiempo determinado. Antes de este evento se produce un <i>onkeydown</i> en el momento que se pulsa la tecla
<i>onkeyup</i>	Se produce cuando el usuario deja de apretar una tecla. Se produce en el momento que se libera la tecla
<i>onload</i>	Este evento se desata cuando la página o imágenes han terminado de cargarse
<i>onmousedown</i>	Se produce el evento <i>onmousedown</i> cuando el usuario pulsa sobre un elemento de la página. <i>Onmousedown</i> se produce en el momento de pulsar el botón, se suelte o no
<i>onmouseout</i>	Se desata un evento <i>onmuoseout</i> cuando el puntero del ratón sale del área ocupada por un elemento de la página
<i>onmouseover</i>	Este evento se desata cuando el puntero del ratón entra en el área ocupada por un elemento de la página
<i>onmouseup</i>	Este evento se produce en el momento que el usuario suelta el botón del ratón, que previamente había pulsado
<i>onresize</i>	Evento que se produce cuando se redimensiona la ventana del navegador, o el <i>frame</i> , en caso de que la página los tenga
<i>onreset</i>	Este evento está asociado a los formularios y se desata en el momento que un usuario hace clic en el botón de <i>reset</i> de un formulario
<i>onselect</i>	Se ejecuta cuando un usuario realiza una selección de un elemento de un formulario
<i>onsubmit</i>	Ocurre cuando el visitante apreta sobre el botón de enviar el formulario. Se ejecuta antes del envío propiamente dicho
<i>onunload</i>	Al abandonar una página, ya sea porque se pulse sobre un enlace que nos lleve a otra página o porque se cierre la ventana del navegador, se ejecuta el evento <i>onunload</i>

- **Seleccionar tipo de tarea**, una tarea es la respuesta de la aplicación a un evento desencadenado por el usuario. Entre las tareas que es posible identificar se ha identificado tres categorías y su respectiva descripción en código conciso:

- Operación, son acciones que se ejecutan cuando un evento es desencadenado, entre las más importantes es posible señalar las de la Tabla 3.15;

Tabla 3.15: Tipo de operaciones de la aplicación

Operación	Descripción	Estereotipo
Búsqueda	Mediante la introducción de un criterio realiza la búsqueda de datos	Oper.busca(<i>criterio</i>)
Modificación	Permite la modificación de la información que el usuario está operando	Oper.modifica(<i>objeto</i>)
Eliminación	Permite la eliminación de un registro que el usuario escoja	Oper.elimina(<i>objeto</i>)
Verificación	Verifica el ingreso de datos antes de su envío o modificación	Oper.verifica (<i>objeto</i>) →1. (<i>criterio1</i>)→ acción →2. (<i>criterio1</i>)→ acción ... →n. (<i>criterio1</i>)→ acción
Envío	Envía los datos pertenecientes a un objeto a un destino (página u objeto)	Oper.envia(<i>objeto</i> o <i>página</i>)
Navegación	Detalla la ruta de navegación de un objeto	Oper.navegacion(<i>página</i>)

- Cuadro de dialogo, son ventanas secundarias que permite que los usuarios reciban respuestas de la aplicación y pueden aparecer antes o después de una operación. Se detallan en la Tabla 3.16;

Tabla 3.16: Tipos de cuadro de dialogo de la aplicación

Cuadro	Descripción	Estereotipo
Error	Alertan sobre problemas que han ocurrido	Dial.error (<i>mensaje</i>)
Advertencia	Alertan sobre condiciones que podrían causar problemas en el futuro	Dial.advert (<i>mensaje</i>)
Información	Informa o muestra datos sobre eventos relacionados a la actividad en curso. También sobre operaciones exitosas	Dial.inform (<i>mensaje</i>)

Confirmación	Pregunta al usuario si desea proseguir con una acción	Dial.confirm (<i>mensaje</i>) (afirmación)→tarea (negación)→tarea
Solicitud	Solicita el ingreso o selección de datos	Dial.solicitud (<i>datos</i>)

- Colaboración, son objetos que aparecen automáticamente de acuerdo a las necesidades del usuario. Se detallan en la Tabla 3.17.

Tabla 3.17: Tipos de objetos de colaboración de la aplicación

Colaboración	Descripción	Estereotipo
Etiqueta	Son notas de no más de cuatro palabras que aparecen al pasar el puntero sobre un objeto	<etiq:(<i>detalle</i>)> Detalle: Nota de la etiqueta
Burbuja	Son notas de no más de quince palabras que aparecen al pasar el puntero sobre un objeto y ayudan a explicar las partes más complicadas	<bub:(<i>detalle</i>)> Detalle: Nota de la burbuja
Autocompletar	Sugerencias del sistema que aparece en un campo de texto adyacente a una caja de entrada de texto que el usuario este tratando	<auto:(<i>detalle</i>)> Detalle: criterio para autocompletar por ejemplo: <i>e-mail</i>
Cursor	Cambio del cursor de acuerdo al lugar por donde este pase en la pantalla: texto, puntero de botón.	<cursor:(<i>tipo de cursor</i>)>
Calendario selector	Botón o parte de la pagina que muestra un calendario de ayuda en la entrada de fechas	<calendar>

Los cuadros de dialogo de información y solicitud son construidos de manera similar a la de una página web y se relacionan con las operaciones de la siguiente manera (ver Tabla 3.18).

Tabla 3.18: Relación entre operaciones y cuadros de dialogo de la aplicación

Cuadro de Diálogo	Error	Advertencia	Información	Confirmación	Solicitud
Operación					
Búsqueda	✓ ₂				
Modificación	✓ x ₂		✓ ₂	✓ ₁	✓ ₁
Eliminación	✓ x ₂		✓ ₂	✓ ₁	
Verificación	✓ ₂	✓ x ₂	✓ ₂		
Envío	✓ ₂		✓ ₂	✓ ₁	✓ _{1 o 2}
Navegación		✓ x ₁			

✓ = uso adecuado ✓~~x~~ = uso en casos extremos
 1 = antes de la operación 2 = después de la operación

- **Establecer zona de carga**, todas las tareas (cuadros de dialogo, operaciones o colaboración) tienen una zona de carga, es decir el lugar en el que se mostraran o efectuaran sus funciones. La Tabla 3.19 muestra la categorización de zonas de carga disponibles y el estereotipo para su desarrollo como código conciso;

Tabla 3.19: Tipos de zonas de carga

Zona	Tipo	Descripción	Estereotipo
Interna (Ocurre en la misma página)	Intermedia	Ocurre sobre la misma página y detiene las funciones de los demás objetos	Zona: In.medio
	Espacio	Utiliza el mismo espacio de la página al que pertenece el objeto u otro espacio circundante, no detiene el funcionamiento de los demás objetos	Zona: In.espacio(<i>espacio</i>)
	Objeto	Utiliza el área circundante al objeto ampliando el tamaño de este, no detiene el funcionamiento de los demás objetos	Zona: In.objeto(<i>objeto</i>)

	Sobre	Ocurre sobre la misma página encima del objeto pero no detiene las funciones de los demás objetos	Zona: In.sobre(<i>objeto</i>)
Externa (Ocurre en otra página)	Total	Recarga la misma página lo que significa que detiene los procesos de todos los demás objetos	Zona: Ex.total
	Otra	Utiliza un espacio de una nueva página lo que deja a los demás objetos efectuar sus funciones	Zona: Ex.otra(<i>página, espacio</i>)
	Blanco	Utiliza una página en blanco, permite a los objetos efectuar sus funciones	Zona: Ex.blanco

En el caso de las operaciones la zona de carga representa el lugar en el que se mostrarán los resultados obtenidos (ver Tabla 3.20).

Tabla 3.20: Relación entre zonas de carga y operaciones

Operación		Búsqueda	Modificación	Eliminación	Verificación	Envío	Navegación
Zona							
I	Intermedia						
	Espacio	✓	✓	✓			✗
	Objeto				✓	✗	
	Sobre				✓		
E	Total	✓	✓	✓		✓	✓
	Otra	✓	✓	✓		✓	✓
	Blanco	✗				✗	

✓ = uso adecuado ✗ = uso en casos extremos

En el caso de los cuadros de diálogo la zona de carga representa el lugar en el que es visualizado el diálogo (ver Tabla 3.21).

Tabla 3.21: Relación entre zonas de carga y cuadros de dialogo

Cuadro de Diálogo		Error	Advertencia	Información	Confirmación	Solicitud
Zona						
I	Intermedia	✓✗	✓✗		✓	✓✗
	Espacio	✓	✓	✓		✓
	Objeto	✓	✓	✓		
	Sobre	✓	✓	✓		✓
E	Total			✓		
	Otra			✓		
	Blanco			✓✗		

✓ = uso adecuado

✓✗ = uso en casos extremos

En el caso de los objetos de colaboración la zona de carga representa el lugar en el que es visualizado el objeto de colaboración. Se observa que estas tareas ocurren en zonas internas de la misma página web (ver Tabla 3.22).

Tabla 3.22: Relación entre zonas de carga y objetos de colaboración

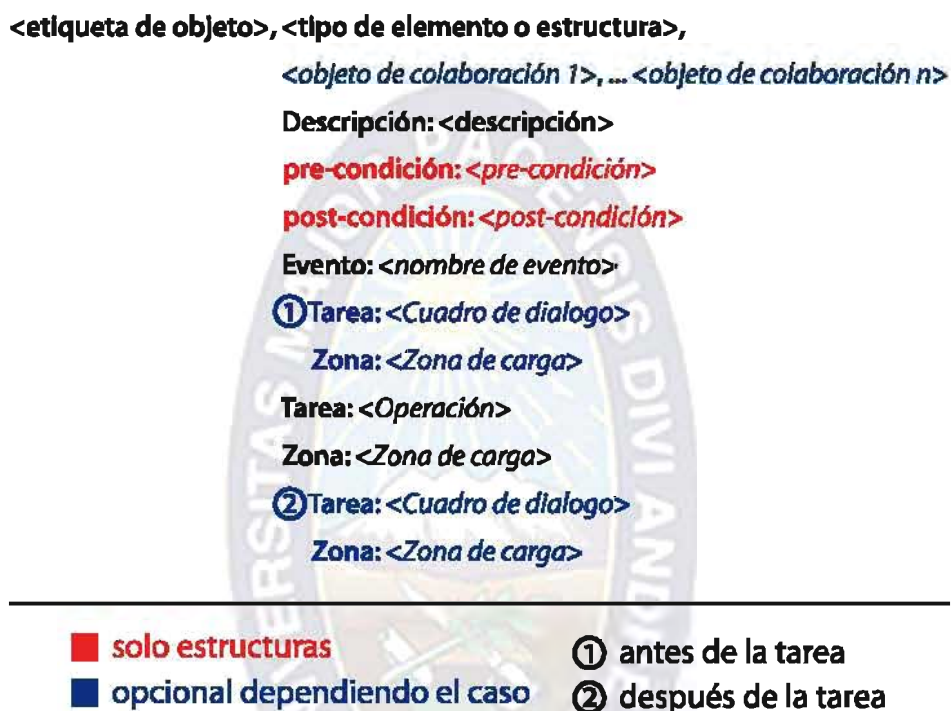
Colaboración		Etiqueta	Burbuja	Autocompletar	Cursor	Calendario selector
Zona						
I	Intermedia					
	Espacio		✓			✓
	Objeto	✓	✓	✓		✓
	Sobre	✓	✓		✓	

✓ = uso adecuado

Registro de código conciso

El formato de registro de los Armazones Web cuenta con una sección exclusiva para el código conciso denominada acción/respuesta. El código se estructura como muestra la Figura 3.31.

Figura 3.31: Muestra de la estructura del código conciso



Visualización

La última fase está dedicada al desarrollo de la guía de registro y tratamiento de los elementos gráficos. Por este motivo y siendo los Armazones Web bocetos de las páginas Web, debe procederse al desarrollo de la Guía de Estilo que contendrá la colección de elementos visuales y formas de jerarquía visual que se usarán para lograr la satisfacción y la atención del usuario.

Para el desarrollo de esta fase es necesario el diseño de los armazones web, porque estos determinan el tipo, tamaño y cantidad de artefactos gráficos que son necesarios para mantener el balance de la composición, además del documento de análisis de la marca desarrollado en la fase de análisis, que nos ofrece los lineamientos generales del diseño y usuarios a los que está enfocada.

Artefactos

El artefacto obtenido durante esta fase es la Guía de Estilo Visual (ver Anexo N). Su estructura obedece al orden de las actividades anteriormente expuestas para esta fase.

Actividades

- Complementar paleta cromática
- Determinar requerimientos de elementos gráficos
- Estandarización de tipografía
- Predefinir estilo de vínculos
- Establecer tipo de jerarquía visual

Complementar paleta cromática

Se debe observar al color de dos maneras. Por el significado que se le pueda asignar y por la estética asociada a la marca. La segunda se ha construido durante la fase de análisis, sin embargo la primera se asocia al modo en que serán vistos ciertos colores cuando interactúen con el usuario durante la ejecución de funciones o respuestas de la aplicación.

De esta manera, es importante tener un conocimiento básico acerca de la percepción que los usuarios reales puedan tener sobre un determinado color. Es necesario guardar una selección de colores especiales “distinta a los de la marca” y de preferencia básicos para usarlos en la aplicación. La Tabla 3.23 muestra un formato que sirve de referencia para el registro del significado de los colores seleccionados con respecto a la aplicación. Los datos que se observan sirven como ejemplo pero es recomendable no variar en demasía los colores propuestos referentes a los errores (rojo), advertencias (amarillo) y resultados exitosos (azul o verde) ya que son un estándar en el mundo occidental. Debe hacerse referencia también a los colores de la marca, obtenidos en la fase de análisis, y detallarse en primer lugar. También se

observa que es posible escoger un o dos colores color secundario a los de la marca que armonice con ellos y ofrezca otra alternativa de diseño visual en la interfaz.

A diferencia de la Tabla 3.5 registro de colores de la marca, se observa que el código CMYK ya no se incluye en esta tabla debido a que estos colores son exclusivos de la interfaz y no se requiere trabajos de impresión para ellos.

Tabla 3.23 Ejemplo de registro de la paleta cromática

Color	Hexadecimal	RGB	Significado
	#006699	R: 0 G: 88 B: 147	Color de la Marca
	#665665	R: 56 G: 61 B: 66	Color de la Marca
	#006699	R: 0 G: 88 B: 147	Tipografía para títulos
	#665665	R: 56 G: 61 B: 66	Tipografía para texto general
	#969696	R: 150 G: 150 B: 150	Tipografía para comentarios
	#FFC0FF	R: 255 G: 192 B: 255	Color secundario a la marca
	#FF0000	R: 255 G: 0 B: 0	Error, Atención, Restricción, Alto
	#FFFF00	R: 255 G: 255 B: 0	Pregunta, Advertencia
	#0000FF	R: 0 G: 0 B: 255	Resultado exitoso
	#FF9600	R: 255 G: 150 B: 0	Introducción de datos

Determinar requerimientos de elementos gráficos

Los armazones web proporcionan una colección de requerimientos en elementos gráficos que debe resolverse y deben respetar las **palabras descriptivas** de la marca. Estos pueden clasificarse en iconos, botones y artefactos gráficos.

Los **iconos**, el método para el diseño de un icono no se referencia en el presente trabajo, sin embargo no por ello se desmerece su importancia. Un ícono es una representación pictórica de una sección, objeto o función destinado a la interacción

del usuario con la aplicación. Para una construcción adecuada de estos elementos se recomienda recurrir a la Tesis “Comunicación iconográfica en la interfaz gráfica de usuario” de Gustavo Bautista disponible en la biblioteca de la Carrera de Informática de la Universidad Mayor de San Andrés. Utilizar iconos bajados de internet no es muy recomendable debido a que poseen derechos de autor y se debe pagar un monto para adquirirlos y los gratuitos generalmente no se ajustan a las necesidades de cada aplicación. Los iconos pueden dividirse en dos tipos: iconos de función, que representan a funciones u objetos de sistema y los iconos de símbolo, que muestran un signo o señal de la aplicación.

Los **botones**, son diferentes de los vínculos o iconos ya que generalmente son usados en formularios y representan el envío de información. En la actualidad es posible el diseño de elementos gráficos que aparenten ser botones mediante un efecto bisel o 3d para salir del estándar de botones tradicionales.

La Tabla 3.24 muestra un ejemplo del modo de registro de los íconos y botones de la aplicación. Los iconos deben tener una colección de los mismos iconos en tres diferentes dimensiones, una menor, una mediana y una grande, para su uso en diferentes partes de la aplicación y para asegurar su nitidez en esas tres. Además se observa los dos tipos de iconos que existen. Por otra parte también se observan iconos en dos diferente colores para diferenciarlos cuando el cursor este sobre. El diseño de los botones se registra posteriormente de acuerdo a las dimensiones que se especifiquen en el armazón y de preferencia deben seguir un mismo estilo, los mostrados únicamente sirven como ejemplo.

Tabla 3.24: Ejemplo de registro de los elementos gráficos de la aplicación

Elemento	Muestras		Dimensiones
	Iconos de Función	Iconos de símbolos	
Iconos			Menor: 20 x 20 px
			Mediano: 40 x 40 px

		Grande: 60 x 60 px
Botones		

Los **artefectos gráficos** son construidos generalmente por un diseñador y pueden clasificarse en:

- **Banners**, a pesar de que son los objetos que el cliente espera ver en un inicio, los banners deben diseñarse con mucho cuidado porque están destinados a llamar la atención como publicidad, cabecera, o vínculos, y se sabe que el mal diseñado de los mismos abruma, confunde y distrae al usuario de su objetivo principal;
- **Imágenes**, una imagen adecuada puede representar un conjunto de atributos de la Marca. Además puede comunicar un mensaje de diferentes maneras de forma rápida, conteniendo todo lo que se quiere decir de la aplicación de tal manera que el usuario lo perciba instantáneamente. Sin embargo no se debe abusar del uso de las mismas ya que su mayor desventaja recae en el tedio que involucra el verlas continuamente;
- **Gráficos**, pueden representar datos estadísticos, esquemas o infografías. Estas sirven para esquematizar o sintetizar un conjunto de información de modo ameno y en poco espacio;
- **Animaciones**, no deben exceder los cinco segundos para exponer su objetivo, ya que el redundar solo causa fastidio.

Estandarización de tipografía

Aunque la selección de una tipografía¹² adecuada no es tomada en cuenta de manera crucial en una aplicación es recomendable seleccionar un conjunto de letras adecuadas que conserven el estilo y la unidad en la aplicación.

¹² Arte dedicado a la construcción de letras.

Los factores que se deben considerar en la selección de los tipos son el **tamaño** (medido en puntos o em), el **grosor** (*bold, light, regular*), el **estilo** (itálica, subrayada, tachada) y el **formato de texto** (mayúscula, minúscula, tipo oración).

Existen cuatro categorías de tipos que deben tomarse en cuenta para el diseño de una aplicación:

- Tipografía de la marca, tratada en la primera fase durante el diseño del logo, esta tipografía es generalmente grande, llamativa y está integrada en elementos gráficos, por lo que no se debe estudiar su disponibilidad en un navegador;
- Tipografía para títulos, es de tamaño mucho mayor que las siguientes categorías. Debe resaltar con el grosor y un estilo subrayado en algunos casos;
- Tipografía para texto general, el tamaño establecido para la web 2.0 es 14 pt como mínimo por la idea de alta legibilidad que tiene como principio;
- Tipografía para comentarios, generalmente de tamaño menor a la del texto general y con tendencia itálica, muy útil en etiquetas y cuadros de ayuda, es de tamaño menor que la tipografía para texto general.

La Tabla 3.25 muestra un ejemplo del modo de registro de las familias tipográficas de la aplicación. Pueden existir hasta dos familias tipográficas distintas para la marca, pero su tamaño no se determina debido al uso en elementos gráficos.

Tabla 3.25: Ejemplo de registro de los estilos de tipografía de la aplicación

Tipografía	Familia	Tamaño	Grosor	Estilo	Formato
Marca	Mouser	(no determinado)	Regular	Normal	Minúscula
Títulos	Arial	18 pt	Bold	Subrayada	Mayúscula
General	Arial	14 pt	Regular	Normal	Tipo oración
Comentarios	Times New Roman	11 pt	Light	Itálica	Tipo oración

La Tabla 3.26 muestra un ejemplo del modo de registro de las muestras de las familias tipográficas de la aplicación. Es posible registrar hasta dos familias tipográficas totalmente distintas. La muestra debe incluir las letras en mayúscula, minúscula, números y símbolos disponibles y deben agruparse en una imagen.

Tabla 3.26: Ejemplo de registro de la tipografía de la marca de la aplicación

Familia	Muestra
Mouser	<p>a b c d e f g h i j k l m n o p q r s t u v w x y z</p> <p>A B C D E F G H I J K L M N O P Q R S T U V W X Y Z</p> <p>1 2 3 4 5 6 7 8 9 0 ! ? \$ % & / () = * + - ; : , .</p>

Es preferible mantener el mismo estilo y familia en la tipografía de títulos y texto general, para denotar uniformidad, sin embargo la de títulos debe tener un tamaño y grosor superiores.

Por otra parte la tipografía que se escoja en el texto de títulos, comentarios y texto general debe ser legible y estar disponible en cualquier plataforma para no ocasionar errores en el diseño, entre las familias disponibles podemos encontrar la *Arial*, *Courier New*, *Times New Roman* y *Verdana*.

Predefinir estilo de vínculos

Los vínculos de la aplicación deben tener un tratamiento especial, ya que cuando el usuario ha utilizado alguno es necesario dar una pista de que ya lo hubo utilizado. Además el usuario debe saber diferenciar entre un vínculo y texto estático.

Se reconocen cuatro tipos de estados en los vínculos:

- **Reposo**, antes de que el vínculo sea utilizado;
- **Sobre**, cuando el ratón pasa por encima del vínculo;
- **Presionado**, cuando se mantiene pulsado el botón del ratón sobre el vínculo;
- **Visitado**, cuando ya se ha utilizado el vínculo.

La Tabla 3.27 muestra un ejemplo del modo de registro de los estilos de los vínculos de la aplicación en el texto de los títulos y texto general. Para diferenciar los vínculos del texto es necesario marcar una diferencia en relación al estilo, el grosor y el color. El color que se muestra en los ejemplos refleja el estándar que se ha respetado hasta ahora en la páginas HTML, sin embargo es posible modificarlo para hacerlo mucho más estético y en **relación** a los colores de la **marca**.

Tabla 3.27: Ejemplo de registro de los estilos de los vínculos de la aplicación

Texto	Estado de Vínculo	Estilo	Grosor	Color	Hexadecimal
Título	Reposo	Subrayada	Bold		#008ACF
	Sobre	Subrayada	Bold		#0000FF
	Presionado	normal	Bold		#0000FF
	Visitado	normal	Bold		#4C0099
General	Reposo	Itálica	normal		#0000FF
	Sobre	Itálica	normal		#8282FF
	Presionado	Itálica	normal		#8282FF
	Visitado	Itálica	normal		#780078

Establecer tipo de jerarquía visual

Jerarquía visual es el control de la importancia visual de los elementos en la composición. La jerarquía está implícita en las diferentes etapas de interacción del usuario con la aplicación. Cuando el usuario ingresa a la aplicación puede observarse una etapa, cuando busca información otra etapa, cuando cambiar de sección se observa otra etapa, estas etapas no tienen un orden estricto para cada usuario pero es posible observar una jerarquía implícita en cada una de ellas.

De acuerdo a la estructuración del armazón y la identificación de la jerarquía visual en cada uno de ellos el diseñador debe proceder a escoger el tipo de jerarquía que corresponda. Los tipos de jerarquía visual y su orden de prioridad se detallan en el Anexo I.

La Tabla 3.28 muestra el modo de registro de las jerarquías visuales de cada armazón, se debe tener en cuenta la prioridad.

Tabla 3.28: Ejemplo de registro de la paleta cromática

ID	1	2
1	Movimiento	Contrastado
2	Imagen	Contrastado
3	Imagen	Brillante o luminoso

3.3. Construcción del Prototipo

3.3.1. Descripción

El objetivo principal del prototipo es el desarrollo de una aplicación orientada a la administración de tareas y facilitan la organización de equipos que trabajan de manera distribuida, apoyados en el uso de Internet en la Web, pertenece a la categoría de Aplicaciones, Servicios y *Mashups* descrita en el capítulo dos. Constituye tal vez el tipo de aplicación más completa, ya que integran una amplia gama de instrumentos de organización y gestión.

3.3.2. Plataforma tecnológica

Par el desarrollo del prototipo se ha procedido a la utilización de los programas y tecnologías mostrados en la Tabla 3.29.

Tabla 3.29: Plataforma tecnológica del Prototipo

PROGRAMAS	FRAMEWORK	LENGUAJES
<i>WampServer</i> <i>Adobe Dreamweaver</i>	<i>jQuery</i>	PHP HTML <i>Javascript</i> XML AJAX CSS

- **WampServer**, WAMP es el acrónimo usado para describir un sistema de infraestructura de internet que usa Windows (como sistema operativo), Apache (como servidor web), MySQL (como gestor de bases de datos) y PHP, Perl, o Python (como lenguajes de programación). Su uso permite servir páginas HTML a internet, además de poder gestionar datos en ellas, al mismo tiempo un WAMP, proporciona lenguajes de programación para desarrollar aplicaciones web;
- **Adobe Dreamweaver**, es una aplicación en forma de estudio enfocada a la construcción y edición de sitios y aplicaciones web basadas en estándares. Creado inicialmente por *Macromedia* y actualmente producido por *Adobe Systems*. Es el programa de este tipo más utilizado en el sector del diseño y la programación web, por sus funcionalidades, su integración con otras herramientas como *Adobe Flash* y, recientemente, por su soporte de los estándares del *World Wide Web Consortium*;
- **jQuery**, es una biblioteca o *framework* de *Javascript*, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la tecnología AJAX a páginas web;
- **AJAX**, acrónimo de JavaScript asíncrono y XML (*Asynchronous JavaScript And XML*), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA. Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

3.3.3. Aplicación del modelo

Fase de Análisis

- **Identificar a los usuarios reales de la aplicación Web**, los organizadores de tareas centran su atención en la organización de tareas y administración seria de la

información, por ende se ha escogido dos grupos de usuarios reales que podrían hacer uso y además tener un conocimiento adecuado en el manejo de este tipo de aplicaciones: Estudiantes y Profesionales;

- **Selección de las características descriptivas de la marca**, para tal grupo de usuarios se debe ofrecer una imagen moderna, simple, clara, fresca y activa;
- **Crear los elementos de la marca**, cada uno de estos elementos se detallan en el Anexo K de este documento, siguiendo las reglas de psicología de color.

Fase de Diagramación

- **Crear la rejilla de diagramación**, debido a que el objetivo de esta aplicación de es informar si no el manejo de información se ha determinado las condiciones de diseño detalla das en la Tabla 3.30.

Tabla 3.30: Dimensiones aproximadas para las partes de la rejilla

Medida/Variable	Tamaño Total	Margen Interior	Total de Unidades
Medio	1024 px -> (930 px a 960 px)	20	12

La rejilla final se detalla en el Anexo L siguiendo los pasos establecidos para esta fase;

- **Delimitar espacios y numerar espacios**, se ha formado una composición holgada que no complique la visión de diferentes aspectos y partes de la aplicación, en total han sido necesarias tres plantillas de composición, la numeración y especificación de espacios fijos obedece a una composición tradicional de cabecera y pie (ver Anexo L).

Fase de Diseño

Sub-Fase de Contenido

- **Determinar tipo de contenido**, las estructuras de contenido a ser utilizadas con prioridad son los catálogos y formularios, por su importancia en la introducción y

muestra de datos. A continuación el ejemplo de la construcción del formulario de ingreso a la aplicación esquematizado en la Tabla 3.31.

Tabla 3.31: Estructuración de un formulario

Asignar título y grupos de datos	Regístrate y se parte de nosotros		
Seleccionar campos	Diagrama de clases (Anexo O): Nombre, apellido, género, fecha_nac, email, nick, passw		
Etiquetar campos	nombre	Nombre:	ORDEN FINAL Nombre: Apellido: Género: Fecha de Nacimiento: Nombre de usuario: Contraseña: Correo electrónico:
	apellido	Apellido:	
	género	Género:	
	fecha_nac	Fecha de Nacimiento:	
	email	Correo electrónico:	
	nick	Nombre de usuario:	
	passw	Contraseña:	

- **Designar objetos de contenido**, se procede a la designación de los elementos del formulario (ver Tabla 3.32).

Tabla 3.32: Estructuración de objetos de un formulario

Nombre:	<<Entrada de texto>>	Cadena
Apellido:	<<Entrada de texto>>	Cadena
Género:	<<Lista de Selección>>	Valor binario
Fecha de Nacimiento:	<<Entrada de texto>>	Fecha
Nombre de usuario:	<<Entrada de texto>>	Cadena
Contraseña:	<<Contraseña>>	Cadena
Correo electrónico:	<<Entrada de texto>>	Cadena

- **Asignar jerarquía visual**, la página principal tiene un alto grado de importancia debido su importancia.
- **Registro de objetos**, se puede observar el Anexo M en el que se detallan los objetos de contenido registrados.

Fase de Visualización

El Anexo N muestra el artefacto completo de Guía visual de la aplicación.

- **Complementar paleta cromática**, para esto se ha escogido una paleta de colores neutros con excepción de los de la marca y los de comunicación;
- **Determinar requerimientos de elementos gráficos**, el armazón ha detallado tres iconos de función y tres iconos de símbolo;
- **Estandarización de tipografía**, se hizo uso de tipografía *Arial* por ser estándar en la mayor parte de equipos informáticos. La tipografía de la marca es un poco más llamativa;
- **Predefinir estilo de vínculos**, los colores escogidos en esta etapa son mas bien comunes para evitar confusiones y continuar con el estándar;
- **Establecer tipo de jerarquía visual**, se ha definido no utilizar animaciones en la aplicación por su alto grado distractivo. La mayor jerarquía utilizada es la de imagen.

Por último una vez compuesta e implementada la aplicación se procede a mostrar las interfaces finales del prototipo, pagina principal (Figura 3.32), pagina de tareas (Figura 3.33) y pagina de contactos (Figura 3.34).

Figura 3.32: Página Principal del Prototipo HotPlanner



Figura 3.33: Página de tareas del Prototipo HotPlanner



Figura 3.34: Página de contactos del Prototipo HotPlanner



3.4. Prueba de Hipótesis

Para probar la hipótesis se ha recurrido a la metodología de Evaluación de Calidad de Sitios Web – *Web-site* QEM planteada por Luis Olsina, y para la recogida de datos se ha usado los métodos de Pensamiento en voz alta y observación ya mencionados en el capítulo dos.

Primeramente se debe definir el árbol de requerimientos de calidad pero enfocado a aplicaciones web. De esta manera se lo describe en la Figura 3.35.

Figura 3.35: Árbol de Requerimientos de Calidad especificando la Usabilidad

1. Usabilidad
 - 1.1. Facilidad de aprendizaje
 - 1.1.1. Entendimiento del objetivo de la aplicación
 - 1.1.2. Reconocimiento de secciones
 - 1.1.3. Rememoración de tareas
 - 1.2. Operabilidad
 - 1.2.1. Eficacia en el uso de objetos
 - 1.2.2. Encontrabilidad de objetos
 - 1.3. Navegabilidad
 - 1.3.1. Mecanismo de búsqueda
 - 1.3.2. Orientación en la aplicación
 - 1.4. Robustez
 - 1.4.1. Recuperación de errores
 - 1.4.2. Información al usuario
 - 1.4.3. Objetos de colaboración
 - 1.5. Satisfacción
 - 1.5.1. Compatibilidad de estilo con el usuario
 - 1.5.2. Relación entre el estilo y el propósito de la aplicación

Luego para cada atributo se define el criterio de calidad elemental mostrado en la Tabla 3.33.

Tabla 3.33: Requerimientos de calidad incluidos los criterios elementales

Subcaracterística	Atributo			Criterio
	Código	Nombre	Descripción	
1.1	1.1.1	Entendimiento del objetivo de la aplicación	Medida en la cual el usuario ha entendido el propósito de la aplicación	$CrE(X_i) = \{(0,0), (1,100)\}$
	1.1.2	Reconocimiento de secciones	Tiempo que tarda un usuario en reconocer las áreas de la aplicación	$CrE(t_i) = \{(0,15), (1,4)\}$
	1.1.3	Rememoración de tareas	Tiempo de realización de tareas anteriormente realizadas	$CrE(t_i) = \{(0,20), (1,6)\}$
1.2	1.2.1	Eficacia en el uso de objetos	Tiempo de realización de tareas	$CrE(t_i) = \{(0,30), (1,10)\}$
	1.2.2	Encontrabilidad de objetos	Tiempo en el que el usuario encuentra los objetos de navegación de la aplicación	$CrE(t_i) = \{(0,15), (1,4)\}$
1.3	1.3.1	Mecanismo de búsqueda	Facilidad para encontrar y usar el mecanismo de búsqueda	$CrE(X_i) = \{(0,0), (1,100)\}$
	1.3.2	Orientación en la aplicación	Tiempo en el que el usuario encuentra los objetos de la aplicación	$CrE(t_i) = \{(0,15), (1,4)\}$
1.4	1.4.1	Recuperación de errores	Porcentaje en el que los mensajes de error ayudaron en la solución de un problema	$CrE(X_i) = \{(0,0), (1,100)\}$
	1.4.2	Información al usuario	Medida en la que la aplicación mantuvo informado al usuario	$CrE(X_i) = \{(0,0), (1,100)\}$
	1.4.3	Objetos de colaboración	Cantidad de objetos de colaboración utilizados por el usuario	$CrE(X_i) = \{(0,0), (1,3)\}$
1.5	1.5.1	Compatibilidad de estilo con el usuario	Medida de agrado de la aplicación	$CrE(X_i) = \{(0,0), (0.5,50), (1,100)\}$
	1.5.2	Relación entre el estilo y el propósito de la aplicación	Medida de relación que el usuario cree que tiene el propósito y la imagen visual de la aplicación	$CrE(X_i) = \{(0,0), (0.5,50), (1,100)\}$

Posteriormente se procedió a la prueba del test (ver Anexo J) a una muestra de quince usuarios en base a los criterios para cada atributo de la subcaracterística de la usabilidad, los valores obtenidos se detallan en la Tabla 3.34.

Tabla 3.34: Valores de las muestras obtenidas del test de usabilidad

Código		Preguntas	Valores de las muestras a 15 usuarios							
1.1	1.1.1	1.a	1							
	1.1.2	4.a, 5.a, 6.a, 8.a, 10.a	0,87	0,95	0,87	0,81	0,77	0,9	0,8	
			0,77	0,87	0,8	0,87	0,9	0,87	0,8	
	1.1.3	7.a	1	0,86	1	1	0,89	0,9	0,95	0,8
			0,89	0,9	1	1	0,87	0,89	1	
1.2	1.2.1	3.a, 4.b, 5.b, 6.b, 7.b, 8.b, 9.b, 10.b	0,93	0,86	1	0,95	0,8	0,9	0,95	0,8
			0,89	0,88	1	0,9	0,75	0,8	1	
	1.2.2	3.b, 4.c, 5.c, 6.c, 7.c, 8.c, 9.b, 10.c	1	0,9	1	0,9	0,7	0,9	0,9	0,8
			0,89	0,88	1	0,9	0,8	0,9	0,9	
1.3	1.3.1	10.d	1							
	1.3.2	4.d, 5.d, 6.d, 8.d, 10.2	1	0,95	1	0,89	0,8	0,9	0,9	
			0,88	0,9	0,88	1	1	0,8	1	
						0,9				
1.4	1.4.1	2.a, 3.c, 4.e, 5.e, 6.e, 7.d, 8.e, 9.c, 10.f	1	0,98	1	0,98	0,95	1	0,982	1
			0,94	0,96	1	1	0,93	1	1	
	1.4.2	2.b, 3.d, 4.f, 5.f, 6.f, 7.e, 8.f, 9.d, 10.g	1							
	1.4.3	3.e, 4.g, 6.g, 7.f, 8.g, 9.e, 10.h	0,92	0,94	0,93	0,93	0,85	0,88	0,9	
			0,87	0,7	0,96	0,89	0,9	0,7	0,8	
						0,9				
1.5	1.5.1	11.a	0,8	1	0,9	1	0,9	0,9	1	0,9
			0,9	0,7	0,9	0,9	0,8	0,9	1	
	1.5.2	11.b	0,75	1	1	0,9	0,95	0,9	1	
			0,95	0,9	0,85	0,9	0,95	0,85	0,9	
						1				

Ahora, en base a estos datos se procede al cálculo de los indicadores de preferencia elemental (IE_i) obtenido de los valores promediados de todas las muestras, mostrando en resultado de la Tabla 3.35.

Tabla 3.35: Obtención de indicadores de preferencia elemental

Subcaracterística		Atributo		Indicadores de preferencia elemental (IE_i)
1.1	Facilidad de Aprendizaje	1.1.1	Entendimiento del objetivo de la aplicación	1
		1.1.2	Reconocimiento de secciones	0.84
		1.1.3	Rememoración de tareas	0.93
1.2	Operabilidad	1.2.1	Eficacia en el uso de objetos	0.89
		1.2.2	Encontrabilidad de objetos	0.89
1.3	Navegabilidad	1.3.1	Mecanismo de búsqueda	1
		1.3.2	Orientación en la aplicación	0.92
1.4	Robustez	1.4.1	Recuperación de errores	0.98
		1.4.2	Información al usuario	1
		1.4.3	Objetos de colaboración	0.87
1.5	Satisfacción	1.5.1	Compatibilidad de estilo con el usuario	0.90
		1.5.2	Relación entre el estilo y el propósito de la aplicación	0.92

Conocidos estos valores procedemos al cálculo del indicador de preferencia global (IG_i) para las subcaracterísticas, obtenido de los indicadores elementales de cada atributo. Este indicador se obtiene aplicando la fórmula (1).

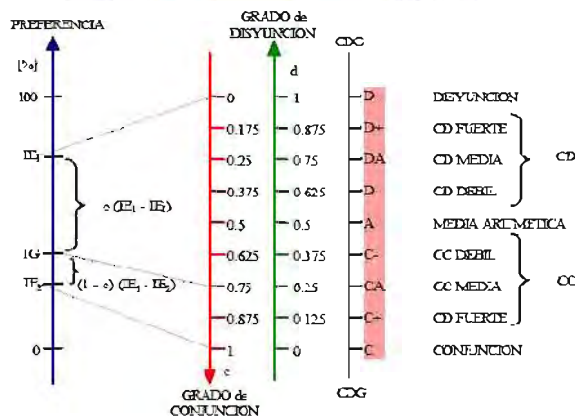
$$IG_i = c \cdot \text{Min}(IE_1, \dots, IE_m) + d \cdot \text{Max}(IE_1, \dots, IE_m) \quad (1)$$

Donde: $c = 0.75$

$d = 0.25$

Los parámetros ajustables c y d toman esos valores debido a que los requisitos de calidad son de tipo mandatorio, es decir que es obligatorio que cada uno de los atributos sea medido por que son necesarios en este trabajo. Tales valores obedecen a un grado de disyunción obtenido de una cuasi-conjunción (CC) media como muestra la Figura 3.36.

Figura 3.36: Interpretación geométrica de la función CDG



De esta manera se procede al cálculo de valores para indicadores globales de cada subcaracterísticas en la Tabla 3.36.

Tabla 3.36: Cálculo de indicadores de preferencia global por subcaracterística

Subcaracterística		Indicadores de preferencia global (IG_i)
1.1	Facilidad de Aprendizaje	$IG_{1,1} = 0.75 \times \text{Min}(1, 0.84, 0.93) + 0.25 \times \text{Max}(1, 0.84, 0.93)$ $= 0.63 + 0.25 = 0.88$
1.2	Operabilidad	$IG_{1,2} = 0.75 \times \text{Min}(0.89, 0.89) + 0.25 \times \text{Max}(0.89, 0.89)$ $= 0.6675 + 0.2225 = 0.89$
1.3	Navegabilidad	$IG_{1,3} = 0.75 \times \text{Min}(1, 0.92) + 0.25 \times \text{Max}(1, 0.92)$ $= 0.69 + 0.25 = 0.94$

1.4	Robustez	$IG_{1.4} = 0.75 \times \text{Min}(0.98, 1, 0.87) + 0.25 \times \text{Max}(0.98, 1, 0.87)$ $= 0.65 + 0.25 = 0.90$
1.5	Satisfacción	$IG_{1.5} = 0.75 \times \text{Min}(0.90, 0.92) + 0.25 \times \text{Max}(0.90, 0.92)$ $= 0.675 + 0.23 = 0.90$

Por último se procede al cálculo del indicador de calidad global, tomando como indicadores elementales a los indicadores globales encontrados como muestra la Figura 3.37.

Figura 3.37: Indicadores elementales de subcaracterísticas

1 Usabilidad

↳ 1.1 0.88 IE₁

↳ 1.2 0.89 IE₂

↳ 1.3 0.94 IE₃

↳ 1.4 0.90 IE₄

↳ 1.5 0.90 IE₅

Aplicando una vez más la cuasi-conjunción y los valores de los parámetros c y d se obtiene el valor final para el indicador global del atributo de calidad usabilidad.

$$\begin{aligned}
 IG_U &= c \times \text{Min}(IE_1, IE_2, IE_3, IE_4, IE_5) + d \times \text{Max}(IE_1, IE_2, IE_3, IE_4, IE_5) \\
 &= 0.75 \times \text{Min}(0.88, 0.89, 0.94, 0.90, 0.90) + 0.25 \times \text{Max}(0.88, 0.89, 0.94, 0.90, 0.90) \\
 &= 0.75 \times 0.88 + 0.25 \times 0.94 \\
 &= 0.895 \cong 0.9
 \end{aligned}$$

Entonces es posible afirmar que el prototipo desarrollado con el **Modelo Pilón** ha obtenido un **0.9** sobre 1 de calidad en usabilidad.

Posteriormente se procedió a la prueba del test a una muestra de quince usuarios en base a los criterios anteriormente utilizados a una aplicación web 2.0 que comparte los mismos objetivos que el prototipo desarrollo pero no ha sido diseñado con el modelo. La aplicación *Remember the milk* es un organizador de tareas como muestra la Figura 3.38.

Figura 3.38: Pagina principal de la aplicación *Remember the milk*

Ahora, en base a estos datos se procede al cálculo de los indicadores de preferencia elemental (IE_i) obtenido de los valores promediados de todas las muestras, mostrando en resultado de la Tabla 3.37.

Tabla 3.37: Obtención de indicadores de preferencia elemental

Subcaracterística		Atributo		Indicadores de preferencia elemental (IE_i)
1.1	Facilidad de Aprendizaje	1.1.1	Entendimiento del objetivo de la aplicación	0.64
		1.1.2	Reconocimiento de secciones	0.72
		1.1.3	Rememoración de tareas	0.80
1.2	Operabilidad	1.2.1	Eficacia en el uso de objetos	0.70
		1.2.2	Encontrabilidad de objetos	0.74
1.3	Navegabilidad	1.3.1	Mecanismo de búsqueda	0.84
		1.3.2	Orientación en la aplicación	0.76

1.4	Robustez	1.4.1	Recuperación de errores	0.71
		1.4.2	Información al usuario	0.72
		1.4.3	Objetos de colaboración	0.76
1.5	Satisfacción	1.5.1	Compatibilidad de estilo con el usuario	0.72
		1.5.2	Relación entre el estilo y el propósito de la aplicación	0.82

Conocidos estos valores procedemos al cálculo del indicador de preferencia global (IG_i) para las subcaracterísticas, obtenido de los indicadores elementales de cada atributo. Este indicador se obtiene aplicando la fórmula (1).

$$IG_i = c \cdot \text{Min}(IE_1, \dots, IE_m) + d \cdot \text{Max}(IE_1, \dots, IE_m) \quad (1)$$

Donde: $c = 0.75$
 $d = 0.25$

Los parámetros ajustables c y d toman esos valores debido a que los requisitos de calidad son de tipo mandatorio, es decir que es obligatorio que cada uno de los atributos sea medido por que son necesarios en este trabajo. Tales valores obedecen a un grado de disyunción obtenido de una cuasi-conjunción (CC) media.

De esta manera se procede al cálculo de valores para indicadores globales de cada subcaracterísticas en la Tabla 3.38.

Tabla 3.38: Cálculo de indicadores de preferencia global por subcaracterística

Subcaracterística		Indicadores de preferencia global (IG _i)
1.1	Facilidad de Aprendizaje	$IG_{1.1} = 0.75 \times \text{Min}(0.64, 0.72, 0.80) + 0.25 \times \text{Max}(0.64, 0.72, 0.80)$ $= 0.48 + 0.2 = 0.68$
1.2	Operabilidad	$IG_{1.2} = 0.75 \times \text{Min}(0.7, 0.74) + 0.25 \times \text{Max}(0.7, 0.74)$ $= 0.525 + 0.185 = 0.71$

1.3	Navegabilidad	$IG_{13} = 0.75 \times \text{Min}(0.84, 0.76) + 0.25 \times \text{Max}(0.84, 0.76)$ $= 0.57 + 0.21 = 0.78$
1.4	Robustez	$IG_{14} = 0.75 \times \text{Min}(0.71, 0.72, 0.76) + 0.25 \times \text{Max}(0.71, 0.72, 0.76)$ $= 0.5325 + 0.19 = 0.72$
1.5	Satisfacción	$IG_{15} = 0.75 \times \text{Min}(0.72, 0.82) + 0.25 \times \text{Max}(0.72, 0.82)$ $= 0.54 + 0.205 = 0.745$

Por último se procede al cálculo del indicador de calidad global, tomando como indicadores elementales a los indicadores globales encontrados como muestra la Figura 3.39.

Figura 3.39 Indicadores elementales de subcaracterísticas

1 Usabilidad

↳	1.1	0.68	IE ₁
↳	1.2	0.71	IE ₂
↳	1.3	0.78	IE ₃
↳	1.4	0.72	IE ₄
↳	1.5	0.745	IE ₅

Aplicando una vez más la cuasi-conjunción y los valores de los parámetros c y d se obtiene el valor final para el indicador global del atributo de calidad usabilidad.

$$\begin{aligned}
 IG_U &= c \times \text{Min}(IE_1, IE_2, IE_3, IE_4, IE_5) + d \times \text{Max}(IE_1, IE_2, IE_3, IE_4, IE_5) \\
 &= 0.75 \times \text{Min}(0.68, 0.71, 0.78, 0.72, 0.745) + 0.25 \times \text{Max}(0.68, 0.71, 0.78, 0.72, 0.745) \\
 &= 0.75 \times 0.68 + 0.25 \times 0.78 \\
 &= 0.705 \cong 0.71
 \end{aligned}$$

Finalmente se puede observar que la otra aplicación evaluada ha obtenido **0.71** como indicador global de calidad con respecto a usabilidad, lo que involucra un **incremento en el grado de usabilidad** dado que mediante la aplicación del modelo se ha conseguido un indicador de **0.9**



CAPÍTULO IV
CONCLUSIONES

“Lo importante es no dejar de hacerse preguntas.”

Albert Einstein

4.1 Conclusiones

De acuerdo a la interrogante y supuestos que orientaron la realización del presente trabajo de investigación, después de la formalización del modelo de usabilidad se derivan las siguientes conclusiones:

- La ventaja del aplicar el modelo de usabilidad radica en la facilidad de diseño y comunicación ágil entre diseñadores y desarrolladores, logrando un producto de alta calidad que llega al usuario final;
- El modelo desarrollado propone tareas para el desarrollo de cada una de las estructuras propuestas y paneles de uso correcto de elementos colaborativos que son óptimos no tan solo de modo visual si no usables;
- Un buen análisis de usuarios reales disminuye la cantidad de problemas de usabilidad, a menor experticia de los usuarios mayores serán los objetos de colaboración y cuadros de dialogo que se tenga que implementar;
- Un beneficio extra del modelo es el registro de todos los objetos visuales que proporcionan uniformidad para diseños futuros, y de esta manera mantener la identidad corporativa de la aplicación;
- El determinar la jerarquía visual, un buen diagramado y composición de jerarquías concretas apoya la continuidad en la búsqueda de objetos por parte del usuario;

- Los cuadros de dialogo y elementos de colaboración estructurados con el modelo ayudan de forma eficiente en el proceso de ensayo-error que conlleva el aprendizaje de la aplicación;
- El modelo ha resultado eficiente no solo en el ámbito de aplicaciones web, sino también de sistemas y páginas web informativas.

4.2 Recomendaciones

La Fase de Análisis puede mejorar su desempeño con el desarrollo de las siguientes actividades adicionales:

- Entrevistar a usuarios reales;
- Identificar las necesidades de los usuarios;
- Determinar capacidades de los usuarios;
- Registrar preferencias de usuarios.

La Fase de Diagramación comprende la construcción de una rejilla por columnas o unidades, sin embargo hay una gama de tipos de rejillas de las que se puede realizar un método de construcción, su alcance e impacto en el diseño de una aplicación web. Entre los métodos a desarrollarse podría investigarse el proceso de diseño de una rejilla manuscrita, modular y jerárquica.

La fase de Visualización puede ser ampliada con el desarrollo de un método que permita la construcción gráfica de cada uno de los componentes que señala como la paleta cromática, elementos gráficos, la tipografía, y la Jerarquía visual.

Otro factor que no se ha tocado en el modelo es un componente que sugiere Nielsen y otros expertos y es la **facilidad del sistema para ser recordado**. Añadiendo este factor al modelo como fase posterior a la de visualización y denominada **Verificación** es posible realizar un estudio más completo de que objetos y como son utilizados por el usuario antes de la implementación del sistema.

BIBLIOGRAFÍA

- [Cobo & Pardo, 2007] Cobo, C., Pardo, H., 2002: "Planeta Web 2.0 - Inteligencia colectiva o medios fast food". Ed: Grup de Recerca d'Interaccions Digitals, Universitat de Vic. Flacso México, Barcelona / México DF.
- [Cortés, 2000] Cortés, A., 2000: "Métodos de Usabilidad" Disponible: [<http://www.sidar.org/recur/desdi/traduc/es/visitable/Herramientas.htm>]. Visitado: [05-10-08]
- [Ketharaman, 2007] Ketharaman, S., 2007: "Usability in the Web 2.0 World". Oracle Financial Services Software Limited - Pune, India.
- [Koch, 2000] Koch, N., 2000: "Software Engineering for Adaptive Hypermedia Systems - Reference Model, Modeling Techniques and Development Process". Tesis Doctoral, Universidad de Munich, Alemania.
- [Koch et al, 2001] Koch, N., Kraus, A., Hennicker, R., 2001: "The Authoring Process of the UML-based Web Engineering Approach". Instituto de Ciencias de la Computación Ludwig-Maximilians, Universidad de Munich, Alemania.
- [Koch & Kraus, 2002] Koch, N., Kraus, A., 2002: "The Expressive Power of UML-based Web Engineering". Instituto de Ciencias de la Computación Ludwig-Maximilians, Universidad de Munich, Alemania.
- [Murugesan & Ginine, 2005] Murugesan, S., Ginige, A., 2006: "Web Engineering: Introduction and Perspectives". Ed: Woojong Suh, Estados Unidos de América.

- [Musser & O'Reilly, 2006] Musser, J., O'Reilly, T., 2006: "Web 2.0 Principles and Best Practices". Ed: O'Reilly Radar, Estados Unidos de América.
- [Nielsen, 2007] Nielsen, J., 2007: "Web 2.0 Can Be Dangerous..." Disponible: [<http://www.useit.com/alertbox/web-2.html>]. Visitado: [05-05-08]
- [Nielsen, 2006] Nielsen, J., 2006: "F-Shaped Pattern For Reading Web Content" Disponible: [http://www.useit.com/alertbox/reading_pattern.html]. Visitado: [05-05-08]
- [Nielsen, 2005] Nielsen, J., 2005: "Ten Usability Heuristics" Disponible: [http://www.useit.com/papers/heuristic/heuristic_list.html]. Visitado: [20-08-07]
- [Nielsen, 1993] Nielsen, J., 1993: "Usability Engineering". Ed: Morgan Kaufmann, Estados Unidos de América.
- [O'Reilly, 2005] O'Reilly, T., 2005: "What Is Web 2.0 - Design Patterns and Business Models for the Next Generation of Software". Disponible: [<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html?page=1>]. Visitado: [05-05-08]
- [Olsina, 1999] Olsina, L., 1999: "Metodología Cuantitativa para la Evaluación y Comparación de la Calidad de Sitios Web". Tesis Doctoral, Facultad de Ciencias Exactas Universidad Nacional de La Plata, Argentina.
- [Paz, 2005] Paz, E. 2005: "Características principales de Web 1.0, Web 1.5 y Web 2.0". Disponible: [http://e-global.es/b2b-blog/2005/11/23/caracteristicas-principales-de-web-1_0-web-1_5-y-web-2_0/]. Visitado: [10-04-08]

- [Saavedra, 2007] Saavedra, A., 2007: "Redes sociales y la Web 2.0". CCBOL 2007, Universidad Mayor de San Andrés, Bolivia.
- [Stauffer, 2008] Stauffer, T., 2008: "How to do everything with your Web 2.0 Blog". Ed: McGraw-Hill Companies, Estados Unidos de América.
- [Usolab, 2005] Usolab, 2005: "Web 2.0: los nuevos desafíos de la interfaz de usuarios" Disponible: [<http://www.usolab.com/articulos/index.php>]. Visitado: [05-05-08]
- [Vander, 2005] Vander, C., 2005: "Qué es la Web 2.0". Disponible: [<http://www.maestrosdelweb.com/principiantes/glosario>]. Visitado: [10-04-08]
- [Val, 2000] Val, J., 2000: "Introducción a la lógica visual". Disponible: [http://www.imageandart.com/tutoriales/morfologia/logica_visual/index.html]. Visitado: [10-04-08]
- [Vander, 2005] Vander, C., 2005: "Qué es la Web 2.0". Disponible: [<http://www.maestrosdelweb.com/principiantes/glosario>]. Visitado: [10-04-08]
- [Wucius, 1995] Wucius, W., 1995: "Principios del diseño en color". Ed: Gustavo Gili, Barcelona.
- [Velasco, 2007] Velasco, J., 2007: "Web 2.0: Hija de la Usabilidad". Centro de Investigación de la Web - DCC, Universidad de Chile.

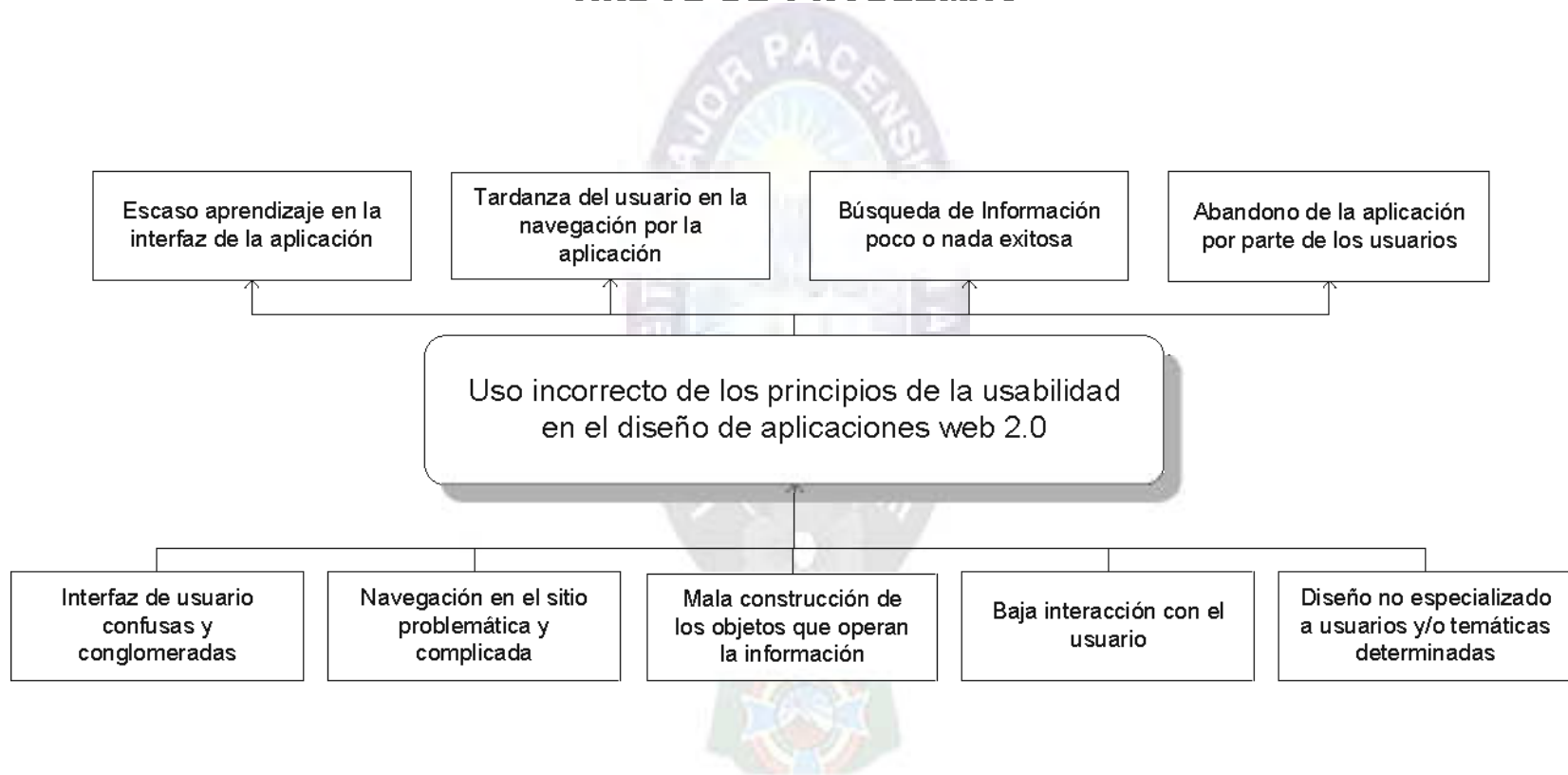


ANEXOS



ANEXO A

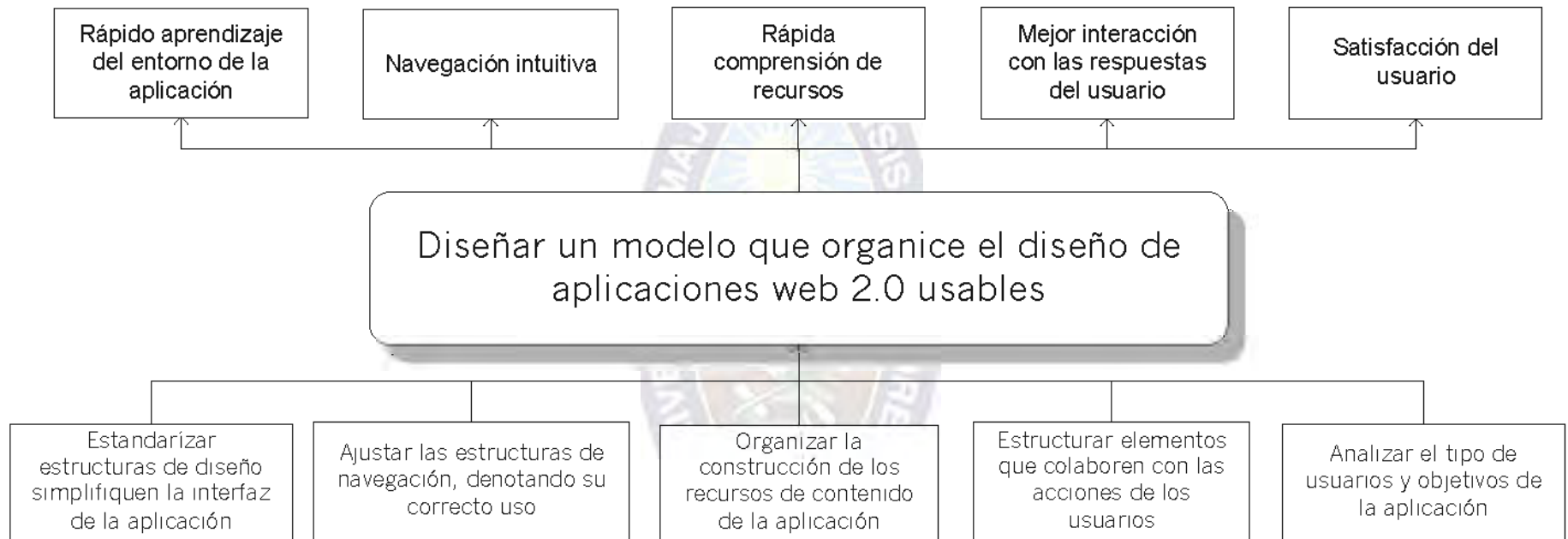
ÁRBOL DE PROBLEMAS





ANEXO B

ÁRBOL DE OBJETIVOS





ANEXO C

GLOSARIO

Acelerador, es una tecla o una combinación de teclas que invocan la acción de algún componente sin tener en cuenta la posición del cursor cuando el acelerador se presiona. Para que el usuario sepa que hay un acelerador asociado con un componente, el acelerador necesita ser desplegado.

Actor es cualquier entidad que intercambia información con el sistema. Un actor puede ser un usuario, equipo físico externo, u otro sistema. Define un coherente set de papeles que los usuarios del sistema pueden representar cuando interactúan con él.

Armazón Web (*Web Wireframe*) es una maqueta de la interfaz que indica la funcionalidad y el diseño de una ventana, pero no su apariencia final haciendo uso de segmentos de línea, controles, y texto, sin color o gráficos complicados.

Artefacto es un pedazo de información que: 1) es producido, modificado, o usado por un proceso, 2) define un área de responsabilidad, y 3) está sujeto al control de versiones. Un artefacto puede ser un modelo, un elemento del modelo, o un documento.

Banner es un formato publicitario en Internet. Consiste en incluir una pieza publicitaria dentro de una página web. Los *banners* se crean a partir de imágenes o animaciones diseñadas con la intención de atraer la atención, resultar notorias y comunicar el mensaje deseado.

Beta perpetuo es la versión de un producto software colocada en línea y siempre dispuesta a ser modificada, utilizando el método de ensayo y error para aprender del consumo de los usuarios. Así se mejora la herramienta en forma constante, sin costes adicionales para el consumidor y acelerando el proceso de mejoras a la aplicación.

Brainstorming (lluvia de ideas) podría asemejarse a las sesiones de mesa redonda en las cuales los participantes emiten diversos criterios.

CMYK, del acrónimo *Cyan Magenta Yellow Black* (Cian Magenta Amarillo Negro), es un sistema de color orientado a trabajos impresos, cuyo rango de valores va desde 0 a 100.

Diagramación es la distribución y organización de los elementos del contenido en un espacio bidimensional (papel, medios electrónicos) mediante criterios de jerarquización buscando la funcionalidad del mensaje bajo una apariencia estética agradable.

Estereotipo es una idea o imagen aceptada por la mayoría como patrón o modelo de cualidades o de conducta.

Estilo es un conjunto de características que individualizan o distinguen.

Flujo de trabajo (*Workflow*) es la representación de una secuencia de operaciones, declaradas como el trabajo de una persona, el trabajo de un mecanismo simple o complicado, el trabajo de un grupo de personas, el trabajo del personal de una organización, o máquinas.

Formulario (*Form*) es un conjunto de campos solicitados por un determinado programa, los cuales se almacenarán para su posterior uso o manipulación.

GUI, del acrónimo *Graphical User Interface* (Interfaz Gráfica de Usuario), es un tipo de interfaz de usuario que utiliza un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles.

Identidad visual es un conjunto de rasgos visuales propios que caracterizan un individuo o una colectividad frente a los demás.

Iteración es la repetición de una secuencia de instrucciones o eventos, resulta en uno o más paquetes atómicos y completos del trabajo del proyecto que pueda realizar alguna función tangible del negocio. Múltiples iteraciones contribuyen a crear un producto completamente integrado.

Look and feel (aspecto y percepción) de un objeto, trabajo gráfico, o programa de aplicación resulta de una percepción subjetiva del diseño. Representa la impresión que el objeto hace sobre el visor, el efecto que produce, el carácter que aparenta poseer, o la expresión que su exterior forma. En el diseño de software se usa con respecto a una interfaz de usuario gráfico y comprende los aspectos de su diseño, incluyendo elementos como los colores, las formas, el diseño, y los tipos de letras (en las que

recae el aspecto), así como el comportamiento de elementos dinámicos como botones, cajas y menús (en los que recae la percepción).

Psicología del color es la expresión de los colores desde el punto de vista psicológico. Según esta disciplina cada color posee un significado y expresa estados anímicos o emociones de muy concreta significación psíquica, e incluso acción fisiológica.

RGB, del acrónimo *Red, Green Blue* (Rojo Verde Azul), es un sistema de color utilizado en dispositivos electrónicos, cuyo rango de valores va desde 0 a 255.

Tag (etiqueta) es un nombre asignado a una estructura de datos como un campo, archivo, párrafo u otro objeto.

Tratamiento gráfico es una propiedad que permite incluir elementos visuales (color, gráficos, tipografías) a un boceto.

Unidad visual es una propiedad que permite mantener una misma línea de estilo, haciendo notar que pertenece a un todo.

Versión de un producto software, es el estado en el se encuentra en un momento dado en su desarrollo o modificación. Por ejemplo, una versión de un programa puede ser "Windows 3.1", indica que es la edición número (3.x) de Windows, con algunas mejoras (x.1). También puede especificarse con letras; por ejemplo, 1.0b indica que se encuentra en estado beta. Se debe aclarar que un principio de la Web 2.0 es el estado beta perpetuo.

VoIP, del acrónimo *Voice over Internet Protocol* (Voz sobre Protocolo de Internet), es un grupo de recursos que hacen posible que la señal de voz viaje a través de Internet empleando un protocolo IP (*Internet Protocol*).

Webtop, del acrónimo *Web desktop* (escritorio Web), es un escritorio virtual en la Web, que funciona tanto en un navegador, como en software. Los *webtop*, a menudo se caracterizan por un entorno similar al de *Windows*, *Mac* o *Linux*, pero ahora, al estar dependiendo de internet, se considera que tienen mucha más funcionalidad.

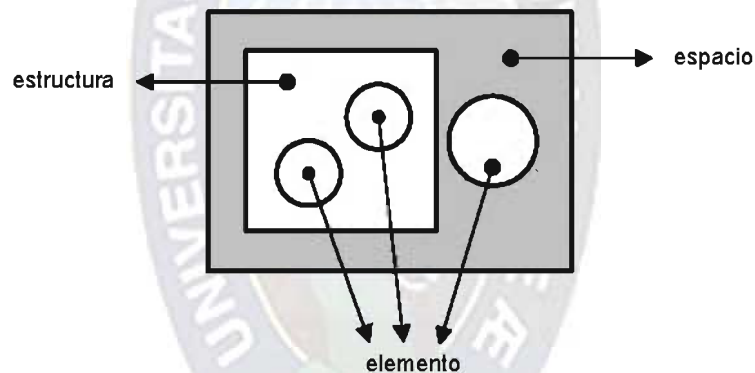


ANEXO D

CONCEPTOS FUNDAMENTALES DEL MODELO PILON

Un **Armazón Web** es la representación esquemática de una página web. Esta sin embargo mejor estructurado y completo que un simple boceto. Su objetivo es servir como herramienta de comunicación entre arquitectos de información, programadores, diseñadores y clientes, sin ocuparse de la estética, a su vez estos constituirán el esquema en el que se basara la programación del sitio. Puede realizarse a mano alzada sobre papel, sin embargo para mejorar la precisión en las dimensiones y estereotipado de elementos se debe hacer uso de programas de edición grafica como *Adobe Fireworks*, *Adobe Illustrator* o *Microsoft Visio*.

Durante el desarrollo del modelo también se hablará en términos de **espacio**, **estructura** y **elemento**



Espacio es un área de la aplicación que tiene un propósito diferente y puede contener estructuras y/o elementos. Además un espacio puede clasificarse en:

- Espacio **fijo** es aquel que no cambiara sus elementos, tamaño ni diseño visual en cada armazón web;
- Espacio **variable** es aquel que cambiará su presentación, tamaño, y elementos o estructuras que contenga;
- Espacio **en blanco** son áreas alrededor de los elementos que no poseen contenido y los ayuda a destacar o separarlos de los demás elementos. En un nivel micro se refiere a la configuración de espacio entre caracteres, y en un nivel macro al

espacio entre los elementos más grandes del armazón web. Sin el espacio en blanco, un diseño luce lleno, como una habitación llena de gente.

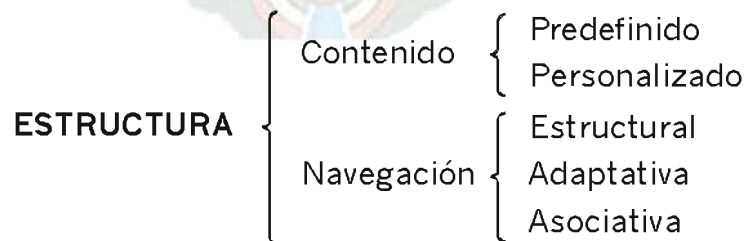
La Figura muestra un esquema de la clasificación de los espacios.



Estructura es un conjunto de elementos que brinda solución a un problema de diseño frecuente, de tal forma que podría usarse muchas veces y nunca del mismo modo. Por otra parte, una estructura puede clasificarse en dos categorías: de contenido y de navegación. Por el diseño se puede clasificar de la siguiente forma:

- Estructura de **contenido**, su propósito es el manejo y muestra de datos y archivos en la aplicación. Puede clasificarse en dos sub-categorías: predefinido y personalizado. Esta división se detalla en profundidad durante la sub-fase de contenido;
- Estructura de **navegación**, su objetivo es el desplazamiento a través de las páginas de la aplicación. Para tal propósito se dividen en tres sub-categorías: estructural, adaptativa y asociativa. Esta división se detalla en profundidad durante la sub-fase de navegación.

El Anexo G muestra una colección de estructuras y respectivos estereotipos. La Figura señala un esquema de la clasificación de las Estructuras mencionada anteriormente.



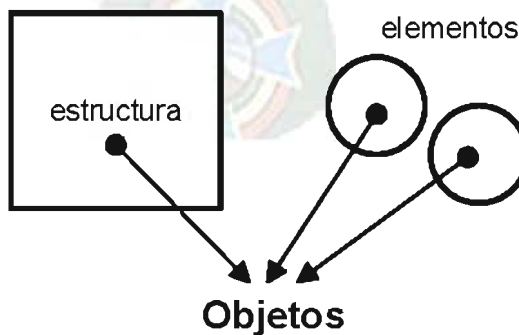
Elemento es la unidad de diseño más simple y carece de sentido si no se combina con otros elementos o estructuras. A diferencia de las últimas, estos no podrán servir para tareas específicas de Navegación, Contenido o Respuesta sin embargo son necesarios para representar objetos sueltos como imágenes, graficas, animación, texto, o vínculos embebidos. Los elementos se clasifican de la siguiente manera:

- Elemento **estático**, es aquel que ofrece información no editable al usuario, mediante el se puede informar o atraer la atención del usuario;
- Elemento **dinámico**, permite el ingreso o la selección de información;
- Elemento de **acción**, permite la ejecución de una operación, la activación de procesos o el enrutamiento entre Armazones web.

El Anexo F muestra una colección de elementos y respectivos estereotipos en que también se señala su clasificación. La Figura muestra un esquema de la clasificación de los elementos.



Por último es necesario aclarar que las estructuras y los elementos son vistos de manera general como objetos del armazón como muestra la Figura.



Para entender los estereotipos construidos para el armazón web descritos en los Anexos F y G, la ficha técnica de cada elemento o estructura cuenta con una serie de campos que se procede a describir en la Tabla.

Estereotipo	<Nombre del elemento o estructura>
Esquema Grafico	<Representación gráfica del elemento o estructura>
Descripción	<Breve explicación de las características del elemento o estructura>
Restricciones	<Limitaciones y condiciones que posee el elemento o estructura para su funcionamiento>
Valores	<Valores obligatorios que deben añadirse al elemento o estructura>
Acciones	< Solo en Estructuras, funciones propias de la estructura>
Cuando Usar	<Solo en Estructuras, modo correcto y sugerencias de uso>



ANEXO E

CARACTERISTICAS DE UN LOGO

Forma, un logo distintivo tienen una forma reconocible, aún en el bosquejo. El cerebro es capaz de identificar formas, ya que puede hacerlo rápidamente, por eso es importante también el espacio en blanco. La idea general debe ser simple y clara, pero no pobre. La siguiente imagen muestra un ejemplo claro de este punto;

El logo de Google es fácilmente identificable por su forma, aun en diversos estilos



Buen uso del espacio en blanco. La mente interpreta automáticamente el significado.

Peso, el logotipo no debe depender de muchas características para ser reconocible. Los logotipos que tienen mejor peso son reconocibles cuando son vistos al lado de otras imágenes fuertes. Demasiados colores, gradientes, efectos 3D y dibujos complicados pueden ser perjudiciales para el peso del logotipo. La siguiente imagen muestra un ejemplo;



Welcome to
TERENA

Mal balance del Peso,
demasiadas características,
baja legibilidad

ACQUA

Buen balance del Peso, uso
de elementos adecuados sin
complejidad.

Contraste, apoya al reconocimiento de la forma, haciendo denotar los bordes entre elementos con mayor claridad. A continuación un ejemplo.



Pobre contraste, el número 15 y la sigla UNLA se pierden en la cinta de color morado.



Alto contraste, las curvas esféricas no se superponen por el buen uso de color.

Presencia, se refiere a llenar gran parte del espacio disponible para el logo con formas que mantienen el significado del mismo, por ejemplo palabras o símbolos identificables. Esto mejora su definición, lo hace más claro, y por lo tanto más identificable.



Pobre Presencia, elementos que no facilitan la interpretación del significado del logo.



Buena Presencia, elementos que se relacionan con el significado del logo.







ANEXO F

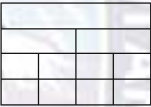
FICHA TÉCNICA DE ELEMENTOS DEL ARMAZÓN WEB

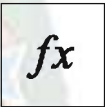
ELEMENTOS

Elementos Estáticos

Estereotipo	<<Texto>> ; <<Text>>	E001
Esquema Grafico	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p><Este es un texto de ejemplo, puede reemplazarse con el texto requerido de acuerdo al contenido.></p> </div>	
Descripción	<p>Un elemento de tipo <<Texto>> es usado para representar texto estático. Debe ingresarse el contenido que irá en el texto como muestra el ejemplo.</p> <p>Se representará este texto en un espacio de acuerdo con las medidas que se especifiquen en la rejilla.</p>	
Restricciones	Usar tipografía común como Arial o Verdana para homogeneizar el esquema.	
Valores Etiquetados	Ninguno	
Estereotipo	<<Imagen>> ; <<Picture>>	E002
Esquema Grafico		
Descripción	<p>Un elemento de tipo <<Imagen>> es usado para representar una imagen estática, esta puede ser: fotografía, fotomontaje o dibujo.</p>	
Restricciones	Ninguna	
Valores Etiquetados	<ul style="list-style-type: none"> - Nombre: Nombre de la Imagen a ser mostrada. - Extensión: Tipo de archivo de Imagen. 	

Estereotipo	<<Diagrama>> ; << <i>Chart</i> >>
Esquema Grafico	
Descripción	Un elemento de tipo <<Diagrama>> es usado para representar datos estadísticos, esquemas o infografías. Pueden ser una simple imagen estática u obtenerse de la Base de datos. Abstraen una gran cantidad de información.
Restricciones	Ninguna
Valores Etiquetados	- Nombre: Nombre del Diagrama.

Estereotipo	<<Tabla>> ; << <i>Table</i> >>
Esquema Grafico	
Descripción	Un elemento de tipo <<Tabla>> es una superficie donde pueden representarse los objetos y organizarse de mejor manera. Se representará en bloques rectangulares de acuerdo con las medidas que se especifiquen en la rejilla.
Restricciones	Ninguna
Valores Etiquetados	- Nombre: Nombre de la Tabla.

Estereotipo	<<Animación>> ; << <i>Animation</i> >>
Esquema Grafico	
Descripción	Un elemento de tipo <<Animación>> es usado para representar objetos con movimiento. Estas Animaciones pueden ser: <i>Gifs</i> Animados o Archivos Flash no interactivos.
Restricciones	Ninguna
Valores Etiquetados	- Nombre: Nombre de la Animación.

Elementos Dinámicos

Estereotipo	<<Entrada de texto>> ; << <i>TextInput</i> >>
Esquema Grafico	<Etiqueta> <input type="text"/>
Descripción	Un elemento de tipo <<Entrada de texto>> es usado para representar a un elemento que permita la introducción de datos por parte del usuario (texto, números, nombres de archivo). Solo permite la entrada de una fila de datos.
Restricciones	Ninguna
Valores Etiquetados	<ul style="list-style-type: none"> - Etiqueta: Nombre del campo de entrada de texto. - Valor: Posible valor de la entrada de texto.

Estereotipo	<<Contraseña>> ; << <i>Password</i> >>
Esquema Grafico	<Etiqueta> <input type="password" value="***"/>
Descripción	Un elemento de tipo <<Contraseña>> se comporta de manera similar a un elemento de tipo <<Entrada de texto>> excepto que enmascara los caracteres ingresados al campo.
Restricciones	Ninguna
Valores Etiquetados	<ul style="list-style-type: none"> - Etiqueta: Nombre del campo de contraseña.

Estereotipo	<< Campo de Entrada de texto>> ; << <i>TextInput Field</i> >>
Esquema Grafico	<Etiqueta> <input type="text"/>
Descripción	Un elemento de tipo <<Campo de Entrada de texto>> es usado para representar a un elemento que permita la introducción de datos por parte del usuario (texto, números, nombres de archivo). Permite la entrada de varias filas de datos.
Restricciones	Ninguna
Valores Etiquetados	<ul style="list-style-type: none"> - Etiqueta: Nombre del campo de entrada de texto. - Valor: Posible valor de la entrada de texto.

Estereotipo	<<Lista de Selección>> ; <<ComboBox>>
Esquema Grafico	<Etiqueta> <input type="text" value="- Selección 1 -"/> ▼
Descripción	Una <<Lista de Selección>> presenta al usuario la posibilidad de escoger un valor de un conjunto de opciones. A diferencia de la <<Colección de Botones de Opción>> este tipo de elemento permite la introducción de muchas opciones en un espacio reducido.
Restricciones	- Número de opciones que se muestren.
Valores Etiquetados	Etiqueta: Nombre de la Lista de Selección. Opciones: Nombre de las opciones de la colección. Valor: Posibles valores de las opciones.

Estereotipo	<<Lista de Selección Múltiple>> ; <<Multi ComboBox >>
Esquema Grafico	<Etiqueta> <input type="text" value="Item 1"/> Item 2 Item 3 Item 4
Descripción	Una <<Lista de Selección Múltiple>> presenta al usuario la posibilidad de escoger uno o más valores de un conjunto de opciones. A diferencia de la <<Colección de Casillas de Verificación>> este tipo de elemento permite la introducción de muchas opciones en un espacio reducido.
Restricciones	- Número de opciones que se muestren.
Valores Etiquetados	Etiqueta: Nombre de la Lista de Multiselección. Opciones: Nombre de las opciones de la colección. Valor: Posibles valores de las opciones.


Estereotipo	<<Casilla de Verificación>> ; <<CheckBox>>
Esquema Grafico	<input checked="" type="checkbox"/> <Etiqueta>
Descripción	Una <<Casilla de Verificación>> presenta al usuario la posibilidad de seleccionar una opción, o permitir una determinada acción.
Restricciones	Ninguna
Valores Etiquetados	Etiqueta: Nombre de la Casilla de Verificación. Valor: Posible valor de la opción.


Estereotipo	<<Colección de Casillas de Verificación>> ; << <i>CheckBox Array</i> >>
Esquema Grafico	<Etiqueta> <input type="checkbox"/> Opción 1 <input checked="" type="checkbox"/> Opción 3 <input checked="" type="checkbox"/> Opción 2 <input type="checkbox"/> Opción 4
Descripción	Una <<Colección de Casillas de Verificación>> presenta al usuario la posibilidad de escoger uno o más valores de un conjunto de opciones.
Restricciones	<ul style="list-style-type: none"> - Número de opciones que se muestren. - Debe existir al menos una opción seleccionada por defecto, generalmente la primera.
Valores Etiquetados	Etiqueta: Nombre de la Colección de Casillas de Verificación. Opciones: Nombre de las opciones de la colección. Valor: Posibles valores de las opciones.

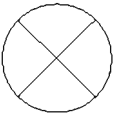
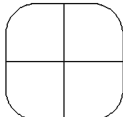
Estereotipo	<<Botón de opción>> ; << <i>RadioButton</i> >>
Esquema Grafico	<input checked="" type="radio"/> <Etiqueta>
Descripción	Un <<Botón de opción>> presenta al usuario la posibilidad de seleccionar una opción, o permitir una determinada acción.
Restricciones	<ul style="list-style-type: none"> - Debe estar seleccionada por defecto.
Valores Etiquetados	Etiqueta: Nombre del Botón de opción. Valor: Posible valor de la opción.

Estereotipo	<<Colección de Botones de Opción>> ; << <i>RadioButton Array</i> >>
Esquema Grafico	<Etiqueta> <input checked="" type="radio"/> Opción 1 <input type="radio"/> Opción 3 <input type="radio"/> Opción 2 <input type="radio"/> Opción 4
Descripción	Una <<Colección de Botones de Opción>> presenta al usuario la posibilidad de escoger un valor de un conjunto de opciones.
Restricciones	<ul style="list-style-type: none"> - Número de opciones que se muestren. - Debe existir al menos una opción seleccionada por defecto, generalmente la primera.
Valores Etiquetados	Etiqueta: Nombre de la Colección de Botones de Opción. Opciones: Nombre de las opciones de la colección. Valor: Posibles valores de las opciones.

Elementos de Acción

Estereotipo	<<Botón>> ; << <i>Button</i> >>
Esquema Grafico	
Descripción	Un elemento de tipo <<Botón>> es usado para iniciar una acción solicitada por el usuario.
Restricciones	Estos elementos solo funcionan si están asociados a un formulario.
Valores Etiquetados	<ul style="list-style-type: none"> - Etiqueta: Nombre del Botón. - Acción: Descripción de la acción que realiza el botón.

Estereotipo	<<Vínculo>> ; << <i>link</i> >>
Esquema Grafico	
Descripción	Un elemento de tipo <<Vínculo>> puede ser usado para iniciar una acción solicitada por el usuario, o como mecanismo de navegación.
Restricciones	Ninguna
Valores Etiquetados	<ul style="list-style-type: none"> - Etiqueta: Nombre del Vínculo. - Acción: Descripción de la acción que realiza.

Estereotipo	<<Icono>> ; << <i>Icon</i> >>
Esquema Grafico	 
Descripción	Un elemento de tipo <<Icono>> puede ser usado para iniciar una acción solicitada por el usuario, o como mecanismo de navegación. A diferencia de un Vínculo, este es representado por una imagen pero posee las mismas características que el primero. El esquema gráfico cuadrado con vértices redondeados viene dado para imágenes de mayor tamaño que actúan como vínculos.
Restricciones	Ninguna
Valores Etiquetados	<ul style="list-style-type: none"> - Imagen: Nombre de la Imagen a ser mostrada. - Etiqueta: Nombre del icono. Al diseñar la pagina en su modo visual la etiqueta debe desaparecer. - Acción: Descripción de la acción que realiza.





ANEXO G


FICHA TÉCNICA DE ESTRUCTURAS DEL ARMAZÓN WEB

ESTRUCTURAS DE NAVEGACIÓN


Estructuras de Navegación Estructural

Estereotipo	<<Navegación de Árbol>> ; << <i>TreeNavigation</i> >>
Esquema Grafico	
Descripción	Las estructuras de tipo <<Navegación de Árbol>> permite el acceso a estructuras jerárquicas.
Restricciones	Ninguna
Valores Etiquetados	Ítem: Nombre de los vínculos del árbol.
Cuando Usar	-Usar cuando se ha organizado la navegación en categorías.

Estereotipo	<<Lengüeta de Navegación>> ; << <i>NavigationTabs</i> >>
Esquema Grafico	
Descripción	Las estructuras de tipo <<Lengüeta de Navegación>> muestran la navegación de un sitio mostrando una señal clara de la ubicación del usuario en el sitio.
Restricciones	Estas estructuras no están disponibles para todos los navegadores. La cantidad de lengüetas oscila entre dos a diez. Nombres de los vínculos cortos.
Valores Etiquetados	Vínculos: Nombre de los vínculos de la lengüeta.
Cuando Usar	-Usar cuando se tiene que representar las opciones cómo una sección en un área delimitada.

Estereotipo	<<Barra de Navegación>> ; <<NavigationBar>>
Esquema Grafico	
Descripción	Las estructuras de tipo <<Barra de Navegación >> son un conjunto de vínculos que facilitan la navegación a través de una estructura jerárquica determinada.
Restricciones	Número de vínculos a ser mostrados para horizontal 12 y vertical hasta 20.
Valores Etiquetados	Vínculos: Nombre de los vínculos.
Cuando Usar	<ul style="list-style-type: none"> - Usar cuando se ha organizado la navegación en categorías. - También se puede agrupar vínculos asociativos de un tema sin un orden jerárquico aparente.

Estructuras de Navegación Adaptativa

Estereotipo	<<Nube de Tags>> ; <<TagCloud>>
Esquema Grafico	
Descripción	Las estructuras de tipo <<Nube de Tags>> son un conjunto de vínculos que facilitan la navegación a los temas más frecuentes de la aplicación.
Restricciones	Vínculos deben ordenarse alfabéticamente y crecer en base a una frecuencia de uso.
Valores Etiquetados	- Nombre: Nombre de la Nube de Tags.
Cuando Usar	- Usar cuando el usuario necesite un camino rápido para llegar a un tema.


Estructuras de Navegación Asociativa

Estereotipo	<<Paginación>> ; <<Paging>>
Esquema Grafico	
Descripción	Las estructuras de navegación de tipo <<Paginación>> se usan para listar un conjunto de datos en distintos bloques de páginas.
Restricciones	Ninguna
Valores Etiquetados	- Número de Elementos: Número de elementos mostrados por página.
Acciones	Previo, Siguiente, Primero, Ultimo
Cuando Usar	<ul style="list-style-type: none"> - En resultados de búsqueda. - Existen más resultados de los que pueda caber en una página

Estereotipo	<<Rastro de Migas>> ; <<BreadcrumbTrail>>
Esquema Grafico	<u><Etiqueta 1></u> > <u><Etiqueta 2></u> > ... > <u><Etiqueta N></u>
Descripción	Las estructuras de tipo <<Rastro de Migas>> son usados para permitir la navegación hacia la página raíz y tener entendimiento de dónde está en relación con el resto de la aplicación.
Restricciones	Ninguna
Valores Etiquetados	Vínculos: Nombre de los vínculos del rastro.
Cuando Usar	<ul style="list-style-type: none"> - Se está dentro de una jerarquía de páginas. - Si el usuario no puede navegar fácilmente por otros métodos de navegación locales.

Estereotipo	<<Carrusel>> ; <<Carousel>>
Esquema Grafico	
Descripción	Una estructura <<Carrusel>> permite que el diseñador utilice de manera óptima el espacio en una página Web mientras provee una experiencia cautivadora para la selección de objetos.

Restricciones	<ul style="list-style-type: none"> - No funcionan adecuadamente en todos los navegadores. - Cantidad de ítems a ser mostrados. - Debe existir un ítem en que este centrado el foco del Carrusel.
Valores Etiquetados	<p>Título: Nombre del Carrusel.</p> <p>Título de ítem: Etiqueta de los ítems del carrusel.</p>
Acciones	<ul style="list-style-type: none"> - Siguiete - Anterior - Ver todo - Ver ítem #
Cuando Usar	<ul style="list-style-type: none"> - Usar cuando el modelo mental de los usuarios acerca de un tema o un conjunto de objetos está íntimamente relacionado con las imágenes, y el espacio disponible en la pantalla es demasiado pequeño para acomodar todos los artículos. - Los Carruseles no son adecuados para comparaciones de elementos que estén uno al lado del otro, especialmente cuando estos no se pueden mostrar de inmediato. Para esto, usar una tabla.

Estereotipo	<<Búsqueda>> ; <<Search>>
Esquema	Bucar: <input type="text"/> 
Descripción	Las estructuras de tipo << Búsqueda >> sirven para la introducción de un criterio que el usuario necesite encontrar.
Restricciones	Ninguna
Valores Etiquetados	<p>Nombre: Nombre del archivo de video o audio.</p> <p>Criterio: Posible valor de la entrada de texto.</p>
Cuando Usar	-Usar cuando el usuario necesite buscar información al interior de la aplicación de manera rápida.

ESTRUCTURAS DE CONTENIDO

Estructuras de Contenido Predefinido

Estereotipo	<<Reproductor de audio o video>> ; <<Audio/Video Player>>	
Esquema Grafico	<p>Video</p> 	<p>Audio</p> 

Descripción	Las estructuras de tipo << Reproductor de Video>> o << Reproductor de Audio>> son usados para representar la posibilidad de reproducir ese tipo de contenido. La diferencia entre ambos es la pantalla de reproducción del video, la cual le da una mayor dimensión.
Restricciones	- Se recomienda que la longitud del Video no supere los 60 segundos, si es contenido que estará embebido en la página Web y pretende usarse como fuente de información en lugar de solo descarga.
Valores Etiquetados	- Nombre: Nombre del archivo de video o audio. - Ruta: Ruta URL donde se halla el archivo.
Acciones	Play, Pausa, Stop
Cuando Usar	- Usar estos elementos si se considera que se aprovechara su naturaleza dinámica, de lo contrario se recomienda desistir su uso.

Estereotipo	<<Control Deslizante>> ; <<Slider>>
Esquema Grafico	
Descripción	Una estructura <<Control Deslizante>> permite al usuario introducir un valor numérico limitado por un valor máximo y un valor mínimo. Mediante su utilización se eliminan errores de entrada de datos. Es posible mover el indicador o hacer click en algún valor. Actúa sobre un área o en toda la página. Puede modificar valores incluso del sistema operativo, como el sonido.
Restricciones	- No funcionan adecuadamente en todos los navegadores. - Cantidad de valores a introducir.
Valores Etiquetados	Etiqueta: Nombre del Control Deslizante. Opciones: Nombre de los valores del control. Valor: Valores de las opciones.
Cuando Usar	- Cuando se necesita un rango determinado en la introducción de datos que influyen en el contenido.



ANEXO H

HISTORIAL DE REVISIONES			
Fecha	Versión	Autor	Descripción
14/04/2010	1.0	Elizabeth Aróstegui	.

PLAN DE PRUEBAS

HotPlanner

Índice

- 1. Introducción**
 - 1.1. Propósito
 - 1.2. Ámbito
- 2. Requerimientos de las pruebas**
- 3. Estrategia de prueba**
 - 3.1. Tipos de pruebas y técnicas
- 4. Recursos**
 - 4.1. Recursos hardware
 - 4.2. Herramientas de soporte
 - 4.3. Configuración de entornos de prueba
 - 4.4. Recursos humanos
- 5. Actividades de prueba**
- 6. Resultados de las pruebas**
- 7. Tareas de la etapa de prueba**

1. Introducción

1.1. Propósito

Este documento describe el Plan de pruebas para un Organizador de Proyectos. En concreto define los siguientes objetivos específicos:

- Identifica los elementos que se van a probar;
- Describe la estrategia de pruebas que se va a seguir en el proceso de prueba;
- Identifica los recursos necesarios para llevar a cabo el proceso de;
- Lista los resultados que se obtienen de las actividades de prueba.

1.2. Ámbito

Este Plan de Pruebas describe las pruebas de usabilidad que se aplicarán a la aplicación desarrollada.

El objetivo es probar todos los requisitos definidos en la Definición y Especificación de Requerimientos de Calidad.

2. Requerimientos de la pruebas

- **Pruebas de interfaz de usuario:**
 - Verificar que los usuarios aprenden a manejar de manera eficiente la interfaz de usuario;
 - Verificar que la navegación a través de un conjunto de pantallas es fácil;
 - Verificar que todas las interfaces de usuario siguen la organización de sus elementos adecuada;
 - Verificar todas las funciones de ayuda e interacción.

3. Estrategia de prueba

Cada usuario deberá ser evaluado individualmente por un examinador y un observador, por ello el costo de ambiente y equipos es bajo.

Se examinara a cinco usuarios por ser un número adecuado según los estudios de evaluación de Usabilidad de Nielsen.

3.1. Tipos de pruebas y técnicas

3.1.1 Pruebas de interfaz de usuario

Las pruebas de interfaz de usuario verifican la interacción del usuario con la aplicación. El objetivo de esta prueba es asegurar que la interfaz de usuario permite al usuario acceder y navegar a través de toda la funcionalidad de la aplicación.

Objetivos de la prueba	Verificar los siguientes objetivos: <ul style="list-style-type: none">• El usuario es capaz de reconocer y aprender a manipular el entorno de la aplicación;• La navegación a través de la aplicación refleja adecuadamente la planificada;• Los elementos de la páginas de la aplicación están organizados claramente y permiten un adecuado entendimiento de las funciones de la aplicación;• El usuario recibe colaboración y ayuda frente a todas las entradas y acciones que le brinda la aplicación.
Técnicas	<ul style="list-style-type: none">• Método de Pensamiento en voz alta• Método de Observación
Criterios de finalización	Cada página se ha verificado con éxito y es consistente con la versión de referencia o con los estándares utilizados

4. Recursos

En esta sección describimos los recursos necesarios para realizar el proceso de prueba, sus principales responsabilidades y características.

4.1. Recursos hardware

Recurso	Cantidad	Nombre y Tipo
PC	1	Ejecución de las pruebas

4.2. Herramientas de soporte

Gestor de Base de Datos WampServer

4.3. Configuración del entorno de prueba

Instalación del Gestor;

Configuración de la Base de Datos y los archivos de conexión.

4.4. Recursos humanos

Rol	Cantidad	Responsabilidades específicas o comentarios
Gestor de prueba	1	<ul style="list-style-type: none">• Proporcionar una dirección técnica;• Adquirir los recursos apropiados;• Informar de la gestión.
Examinador	1	<ul style="list-style-type: none">• Ejecutar pruebas;• Documentar los resultados.
Observador	1	<ul style="list-style-type: none">• Registrar el comportamiento del usuario durante la prueba;• Evaluar el esfuerzo de prueba.

5. Actividades de prueba

Las actividades del proceso de prueba son:

Actividad	Fecha de comienzo	Fecha de finalización
Planificación de la prueba	28 de Abril	12 de Mayo
Diseño de la prueba	13 de Mayo	16 de Mayo
Implementación de la prueba	17 de Mayo	17 de Mayo
Ejecución de la prueba	24 de Junio	25 de Junio
Evaluación de la prueba	26 de Junio	28 de Junio

6. Resultados de las pruebas

Del proceso de prueba se obtienen los siguientes documentos:

Documento de desarrollo de software	Fecha
Plan de prueba	12 de Mayo
Reporte de las pruebas	28 de Junio
Modelo corregido	1 de Julio

7. Tareas de la etapa de pruebas

Las tareas que se realizan en cada una de las actividades son:

- **Planificación de las pruebas:**
 - Identificar los requisitos para las pruebas;
 - Valorar los riesgos;
 - Desarrollar la estrategia de pruebas;
 - Identificar los recursos necesarios para realizar las pruebas;
 - Generar el Plan de pruebas;
- **Diseño de las pruebas:**
 - Definición de las condiciones de prueba;
 - Identificación de los casos de uso a probar para concentrándose en la funcionalidad adaptativa;
 - Preparación de los datos de prueba apropiados;
 - Especificación de los resultados esperados en la prueba;
- **Implementación de las pruebas:**
 - Establecer el entorno de prueba;
 - Desarrollar los *scripts* de prueba y los datos de prueba;
- **Ejecución de las pruebas:**
 - Ejecutar los *scripts* de prueba;
 - Evaluar los resultados;
 - Determinar acciones posteriores: si los resultados son como era de esperar, ninguna acción es necesaria, si los resultados son inesperados, la causa del problema debe ser determinada y resuelta;
- **Evaluación de las pruebas:**
 - Analizar los defectos;
 - Determinar si se han alcanzado los criterios de las pruebas;
 - Crear los informes de evaluación de las pruebas.



ANEXO I

CLASIFICACIÓN DE LA JERARQUÍA VISUAL

(Fuente [Val, 2000])

Todo lo que nos rodea en sociedad está lleno de valores jerárquicos. Desde el semáforo que regula el tráfico al trato especial de que goza un jefe de gobierno. La naturaleza tampoco se libra de prioridades. Consideremos la forma de alimentación de los polluelos en el nido o el orden en la comida de un grupo de leones. Muchas de estas jerarquías se asocian a patrones visuales lógicos, que usamos sin apenas caer en la cuenta. Es lo que puede suceder al ceder el puesto en una mesa a la hora de comer o en la configuración que adoptamos cuando van a hacernos una fotografía. Veamos ahora algunos de estos elementos de lógica visual que afectan o son afectados por valores jerárquicos.

1. Jerarquía arriba / abajo

Se manifiesta como la prioridad de lo superior en el espacio o en el plano, que presenta una conexión directa con lo superior en el orden de las ideas o en el orden moral. La imagen reproducida más abajo establece una jerarquía entre las letras que tiene una de sus bases más relevantes en su posición relativa superior / inferior.



La letra M supera en visibilidad a las otras letras por el hecho de estar más alta y destacar. Para localizarla, el organismo gasta menor cantidad de energía. Su posición respecto los otros elementos le sirve de base de proyección.

2. Jerarquía grande / pequeño

Es una de las más utilizadas. En términos generales, abundan las situaciones en las que el valor de las cosas es proporcional a su tamaño físico, respecto de las cosas que hay a su alrededor. Un fenómeno que ya encontramos en el arte egipcio, en el que se nos muestran gigantescos faraones imperando sobre ejércitos de diminutas figurillas.



En la imagen superior, la letra F presenta ese dominio faraónico sobre el resto de las letras que la acompañan. Esa fuerza caracteriza el diseño y hace que éste sea leído en un orden determinado.

3. Jerarquía centrado / periférico

Esta jerarquía visual también es de las más utilizadas y ha sido denominada "fuerza del centro". Tiene su primer fundamento en la estructuración simétrica que presentan tantos seres y sucesos de la naturaleza. Una ley que captamos y proyectamos a numerosos hechos o cuestiones de trascendencia social.



En la imagen que presentamos, la línea central, a pesar de su menor tamaño, adquiere una fuerza extra a causa de la situación que ocupa. De la misma manera, una persona que ocupa el centro de una mesa presidencial en un acto social, toma para sí ese factor de jerarquía visual.

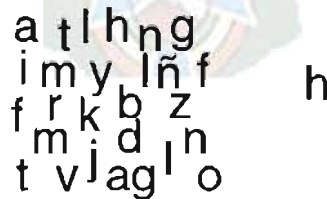
4. Jerarquía de lo contrastado

Este tipo de jerarquía visual actúa mediante un juego de choque perceptivo. Aquello que contrasta salta hacia nosotros sin necesidad de que nosotros vayamos hacia él. Su primitivo fundamento es el de un acto de defensa, ligado a la exploración del medio en que nos movemos.



5. Jerarquía de lo separado frente a lo grupal

Un viejo y desgastado chiste hacía decir a un vigilante en un fuerte del oeste americano: "¡vienen dos mil tres indios!". A la pregunta de cómo lo sabía, su respuesta era: "porque delante vienen tres y luego unos dos mil". Ciertamente, eso es lo que sucede en nuestra percepción: vemos lo separado y después el grupo. O si se prefiere, los grupos tienden a formar unidades de mayor nivel: tres árboles y un bosque.



En la imagen reproducida aquí arriba la letra h se beneficia de esa situación y salta hacia el espectador como un elemento con valor en sí mismo. El resto es "un" bloque, pero su complejidad formal requiere de nosotros mucho más tiempo de atención, que no siempre estaremos dispuestos a dar.

6. Jerarquía de lo brillante

En términos generales, todo aquello que irradia ha sido valorado por las más diversas culturas: desde el oro hasta los atributos iconográficos de la santidad, con sus aureolas luminiscentes. Toda radiación tiene fuerza visual. Y casi toda fuerza visual puede traducirse en fuerza física.



A pesar de la baja calidad de las imágenes en la red y del motivo general que llena toda la fotografía, se puede apreciar cómo la zona con mayor brillo llama nuestra atención a pesar de no tener un contenido particularmente importante en el contexto de la imagen.

Aplicaciones al diseño

Las jerarquías visuales básicas ponen de manifiesto aspectos diversos de lógica visual y son aplicables al diseño gráfico, web, industrial y arquitectónico, y de hecho son usadas muy frecuentemente, tanto de manera consciente como inconsciente, por los profesionales. Los lectores de publicaciones, las personas que ven la televisión o circulan por las calles de las ciudades, llevan en sí mismos estos criterios de lógica visual. De ahí que, al usarlos, el diseñador, por así decir, hable un lenguaje formal conocido por ellos. Y este es un hecho que, evidentemente, facilita la comunicación con el público al que se dirige el mensaje. Usar principios jerárquicos, a través de cualquiera de sus muchas variantes, multiplica la eficacia de la comunicación sin limitar la creatividad formal o conceptual de los diseñadores.

Sin embargo a esta clasificación de la jerarquía hace falta añadir la jerarquía de la imagen sobre el texto y la jerarquía del movimiento.

- Jerarquía de la imagen sobre el texto, las imágenes llaman la atención mucho más que el texto, estas literalmente levantan la página, sin embargo el abusar de ellas es negativo para la legibilidad de la página web;

Jerarquía del movimiento, los objetos con movimiento tienen un alto grado de jerarquía visual. Los videos y contenido flash son útiles pero deben ubicarse con mucho cuidado para que el usuario no pierda el objetivo.



ANEXO J

HOJA DE EVALUACIÓN

Nombre del Usuario:.....Genero:.....

Fecha de la Evaluación:..... Hora de inicio:..... Hora de fin:.....

1. Objetivo de la aplicación

a.

Código	Métrica	Respuesta Mala	Resultado	Respuesta buena
1 1 1	Pregunta al usuario: ¿Para qué sirve la aplicación?	0		100

2. Registro e ingreso a la aplicación

a.

Código	Métrica	Mensaje de error no sirvió	Resultado	Mensaje de error sirvió
1 4 1	Mensajes de error ayudaron en la recuperación de errores	0		100

b.

Código	Métrica	Mensaje no sirvió	Resultado	Mensaje sirvió
1 4 2	La aplicación mantuvo informado al usuario	0		100

3. Crear una tarea

a.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 1	Tiempo de realización de la tarea	30 s		10 s

b.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 2	Facilidad para encontrar objetos	15 s		4 s

c.

Código	Métrica	Mensaje de error no sirvió	Resultado	Mensaje de error sirvió
1 4 1	Mensajes de error ayudaron en la recuperación de errores	0		100

d.

Código	Métrica	Mensaje no sirvió	Resultado	Mensaje sirvió
1 4 2	La aplicación mantuvo informado al usuario	0		100

e.

Código	Métrica	# mínimo de objetos	Resultado	# máximo de objetos
1 4 3	Objetos de colaboración usados	0/3		3/3

4. Invitar a un contacto (correo sugerido: nel@hotmail.com)

a.

Código	Métrica	Máximo	Resultado	Mínimo
1 1 2	Reconocimiento de vínculos	15 s		4 s

b.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 1	Tiempo de realización de la tarea	30 s		10 s

c.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 2	Facilidad para encontrar objetos	15 s		4 s

d.

Código	Métrica	Máximo	Resultado	Mínimo
1 3 2	Orientación en la aplicación	15 s		4 s

e.

Código	Métrica	Mensaje de error no sirvió	Resultado	Mensaje de error sirvió
1 4 1	Mensajes de error ayudaron en la recuperación de errores	0		100

f.

Código	Métrica	Mensaje no sirvió	Resultado	Mensaje sirvió
1.4 2	La aplicación mantuvo informado al usuario	0		100

g.

Código	Métrica	# mínimo de objetos	Resultado	# máximo de objetos
1 4 3	Objetos de colaboración usados	0/3		3/3

5. Cambiar datos de perfil (Luego sugerir en los datos de zona: el alto)

a.

Código	Métrica	Máximo	Resultado	Mínimo
1 1 2	Reconocimiento de vínculos	15 s		4 s

b.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 1	Tiempo de realización de la tarea	30 s		10 s

c.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 2	Facilidad para encontrar objetos	15 s		4 s

d.

Código	Métrica	Máximo	Resultado	Mínimo
1 3 2	Orientación en la aplicación	15 s		4 s

e.

Código	Métrica	Mensaje de error no sirvió	Resultado	Mensaje de error sirvió
1 4 1	Mensajes de error ayudaron en la recuperación de errores	0		100

f.

Código	Métrica	Mensaje no sirvió	Resultado	Mensaje sirvió
1 4 2	La aplicación mantuvo informado al usuario	0		100

6. Volver a la sección de tareas e invitar a un contacto sugerido por el sistema

a.

Código	Métrica	Máximo	Resultado	Mínimo
1 1 2	Reconocimiento de vínculos	15 s		4 s

b.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 1	Tiempo de realización de la tarea	30 s		10 s

c.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 2	Facilidad para encontrar objetos	15 s		4 s

d.

Código	Métrica	Máximo	Resultado	Mínimo
1 3 2	Orientación en la aplicación	15 s		4 s

e.

Código	Métrica	Mensaje de error no sirvió	Resultado	Mensaje de error sirvió
1.4 1	Mensajes de error ayudaron en la recuperación de errores	0		100

f.

Código	Métrica	Mensaje no sirvió	Resultado	Mensaje sirvió
1.4 2	La aplicación mantuvo informado al usuario	0		100

g.

Código	Métrica	# mínimo de objetos	Resultado	# máximo de objetos
1 4 3	Objetos de colaboración usados	0/3		3/3

7. Crear una tarea

a.

Código	Métrica	Máximo	Resultado	Mínimo
1 1 3	Rememoración de tareas	20 s		6 s

b.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 1	Tiempo de realización de la tarea	30 s		10 s

c.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 2	Facilidad para encontrar objetos	15 s		4 s

d.

Código	Métrica	Mensaje de error no sirvió	Resultado	Mensaje de error sirvió
1.4 1	Mensajes de error ayudaron en la recuperación de errores	0		100

e.

Código	Métrica	Mensaje no sirvió	Resultado	Mensaje sirvió
1.4 2	La aplicación mantuvo informado al usuario	0		100

f.

Código	Métrica	# mínimo de objetos	Resultado	# máximo de objetos
1 4 3	Objetos de colaboración usados	0/3		3/3

8. Crear una lista de tareas

a.

Código	Métrica	Máximo	Resultado	Mínimo
1 1 2	Reconocimiento de vínculos	15 s		4 s

b.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 1	Tiempo de realización de la tarea	30 s		10 s

c.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 2	Facilidad para encontrar objetos	15 s		4 s

d.

Código	Métrica	Máximo	Resultado	Mínimo
1 3 2	Orientación en la aplicación	15 s		4 s

e.

Código	Métrica	Mensaje de error no sirvió	Resultado	Mensaje de error sirvió
1 4 1	Mensajes de error ayudaron en la recuperación de errores	0		100

f.

Código	Métrica	Mensaje no sirvió	Resultado	Mensaje sirvió
1 4 2	La aplicación mantuvo informado al usuario	0		100

g.

Código	Métrica	# mínimo de objetos	Resultado	# máximo de objetos
1.4 3	Objetos de colaboración usados	0/3		3/3

9. Crear una tarea en la nueva lista

a.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 1	Tiempo de realización de la tarea	30 s		10 s

b.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 2	Facilidad para encontrar objetos	15 s		4 s

c.

Código	Métrica	Mensaje de error no sirvió	Resultado	Mensaje de error sirvió
1 4 1	Mensajes de error ayudaron en la recuperación de errores	0		100

d.

Código	Métrica	Mensaje no sirvió	Resultado	Mensaje sirvió
1 4 2	La aplicación mantuvo informado al usuario	0		100

e.

Código	Métrica	# mínimo de objetos	Resultado	# máximo de objetos
1 4 3	Objetos de colaboración usados	0/3		3/3

10. Asignar etiquetas a tareas que el usuario escoja

a.

Código	Métrica	Máximo	Resultado	Mínimo
1 1 2	Reconocimiento de vínculos	15 s		4 s

b.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 1	Tiempo de realización de la tarea	30 s		10 s

c.

Código	Métrica	Máximo	Resultado	Mínimo
1 2 2	Facilidad para encontrar objetos	15 s		4 s

d.

Código	Métrica	No uso	Resultado	Si uso
1 3 1	Mecanismo de búsqueda	0		100

e.

Código	Métrica	Máximo	Resultado	Mínimo
1 3 2	Orientación en la aplicación	15 s		4 s

f.

Código	Métrica	Mensaje de error no sirvió	Resultado	Mensaje de error sirvió
1 4 1	Mensajes de error ayudaron en la recuperación de errores	0		100

g.

Código	Métrica	Mensaje no sirvió	Resultado	Mensaje sirvió
1 4 2	La aplicación mantuvo informado al usuario	0		100

h.

Código	Métrica	# mínimo de objetos	Resultado	# máximo de objetos
1 4 3	Objetos de colaboración usados	0/3		3/3

11. Satisfacción con la aplicación

a.

Código	Métrica	NO	REGULAR	SI	Resultado Obtenido
1 5 1	<i>Pregunta al usuario:</i> ¿Te agrada la aplicación?	0	50	100	

b.

Código	Métrica	NO	REGULAR	SI	Resultado Obtenido
1 5 2	<i>Pregunta al usuario:</i> ¿Crees que el estilo visual va de acuerdo a la aplicación?	0	50	100	



ANEXO K

HISTORIAL DE REVISIONES			
Fecha	Versión	Autor	Descripción
23/11/09	1.0	Elizabeth Aróstegui	Primera fase del modelo

Formato de registro

ANÁLISIS DE LA MARCA HOTPLANNER

1. USUARIOS REALES

Tipo de Usuario	Edad	Nivel
Estudiantes	16 - 24	Aficionado
Profesionales	24 - (¿?)	Experto

2. CARACTERÍSTICAS DE LA MARCA

a. Lista de palabras descriptivas de la Marca

- Moderno
- Simple
- Claro
- Fresco
- Activo

b. Lista de Atributos y Servicios



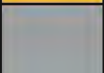
Atributos		Servicios
Normal	Opuesto	
Fácil	Difícil	<ul style="list-style-type: none"> • Adición de contactos. • Uso de etiquetas para la clasificación de tareas. • Capacidad de compartir tareas entre usuarios. • Ver el estado de tiempo para que el usuario pueda proyectar de manera adecuada sus tareas.
Rápido	Lento	
Gratuito	Con costo	

3. ELEMENTOS DE LA MARCA

a. Nombre

Etapa 1	Etapa 2	Final
Illimaniplanner Pandora Hotplanner Speedplanner Hotmanager Mastica tus tareas Organiza tu Chairo Chaskiplanner TaskManager Planius	Pandora Hotplanner Hotmanager Chaskiplanner Planius	Hotplanner

b. Color

Color	Hexadecimal	RGB	CMY K
	#EB6D1F	R: 235 G: 109 B:31	C: 0 M: 68 Y:94 K: 0
	#FBBE0E	R: 251 G: 190 B:14	C: 0 M: 29 Y:3 K: 0
	#9C9E9F	R: 156 G: 158 B:159	C: 0 M: 0 Y:0 K: 50

c. Logo



d. Eslogan

Recuerda, Apunta, Comparte

4. CONSIDERACIONES CREATIVAS EN LOS REQUERIMIENTOS

Requerimientos Observados
Ninguno



ANEXO L

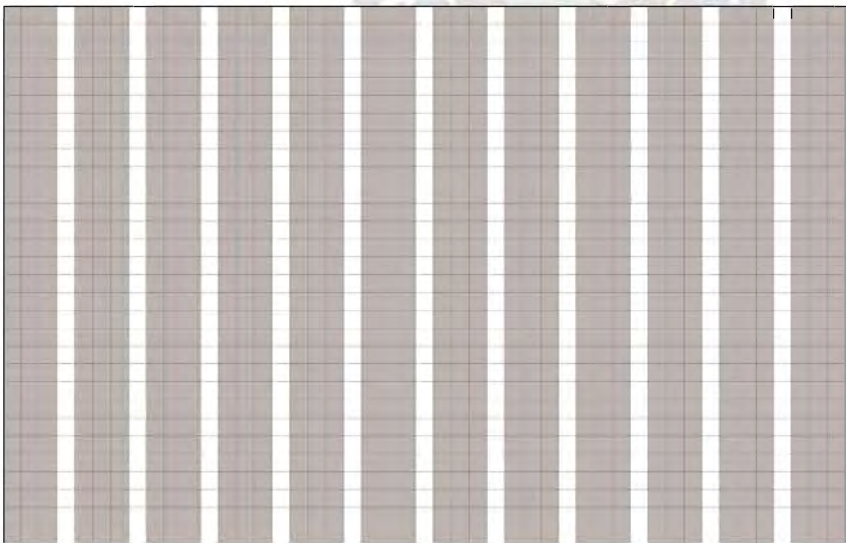
HISTORIAL DE REVISIONES			
Fecha	Versión	Autor	Descripción
15/06/10	1.0	Elizabeth Aróstegui	Segunda fase del modelo

Formato de registro

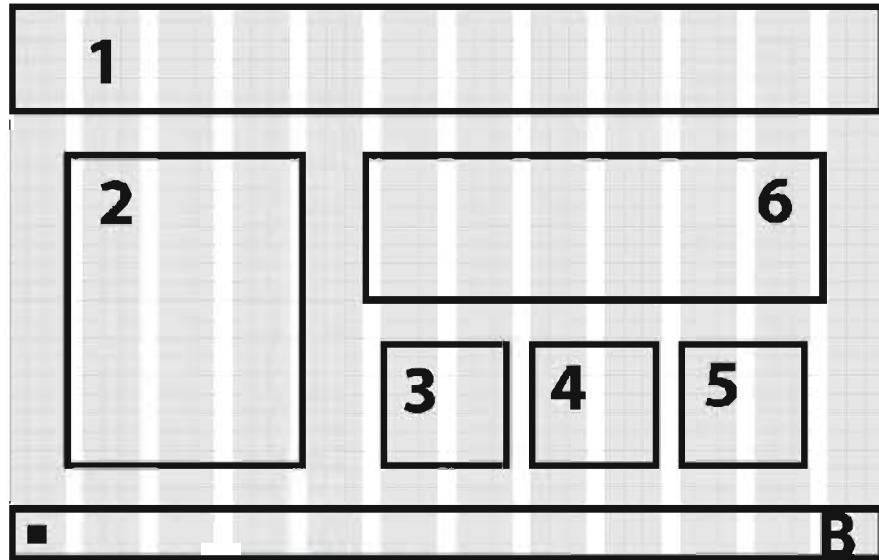
PLANTILLAS DE COMPOSICIÓN

HOTPLANNER

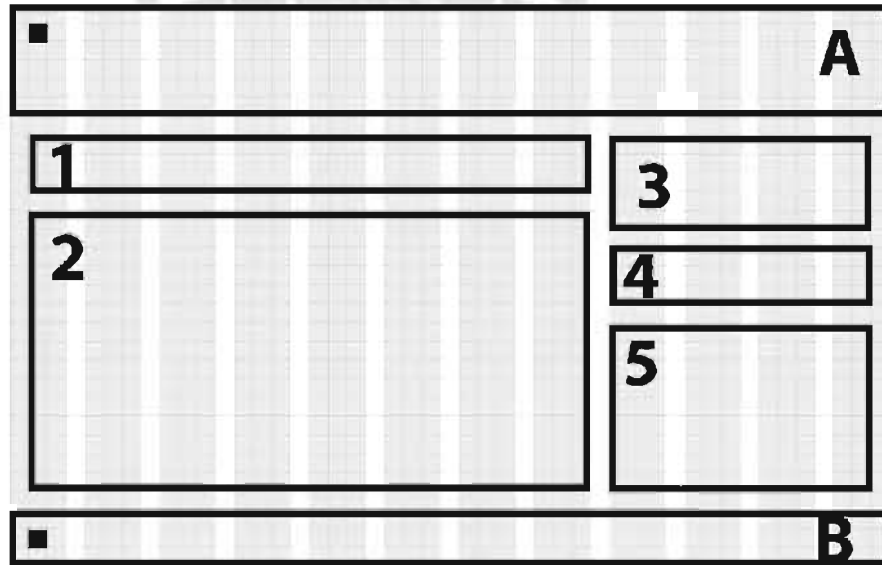
REJILLA DE DIAGRAMACIÓN

	<p>Dimensiones:</p> <p>TU = 12 U</p> <p>M = 20 px</p> <p>TT = 940 px</p> <p>U = 60 px</p>
--	--

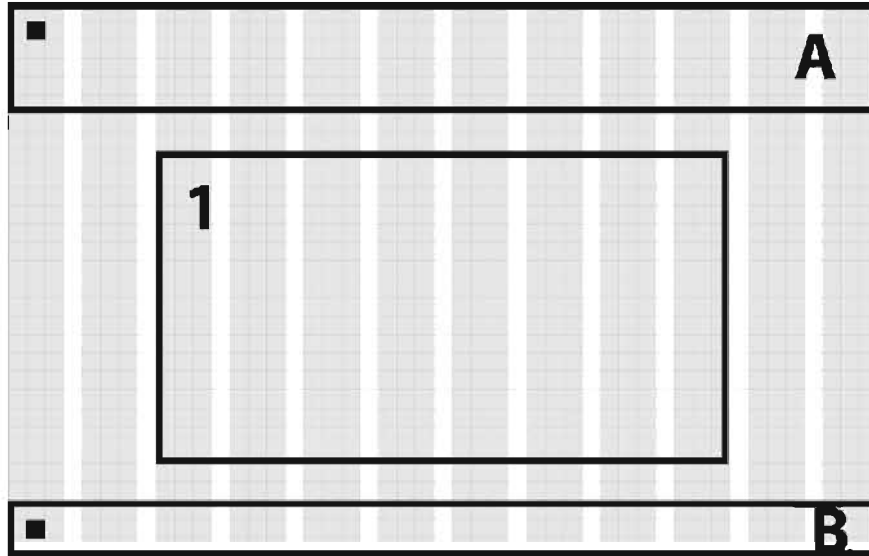
PCP



PCS-1



PCS-2



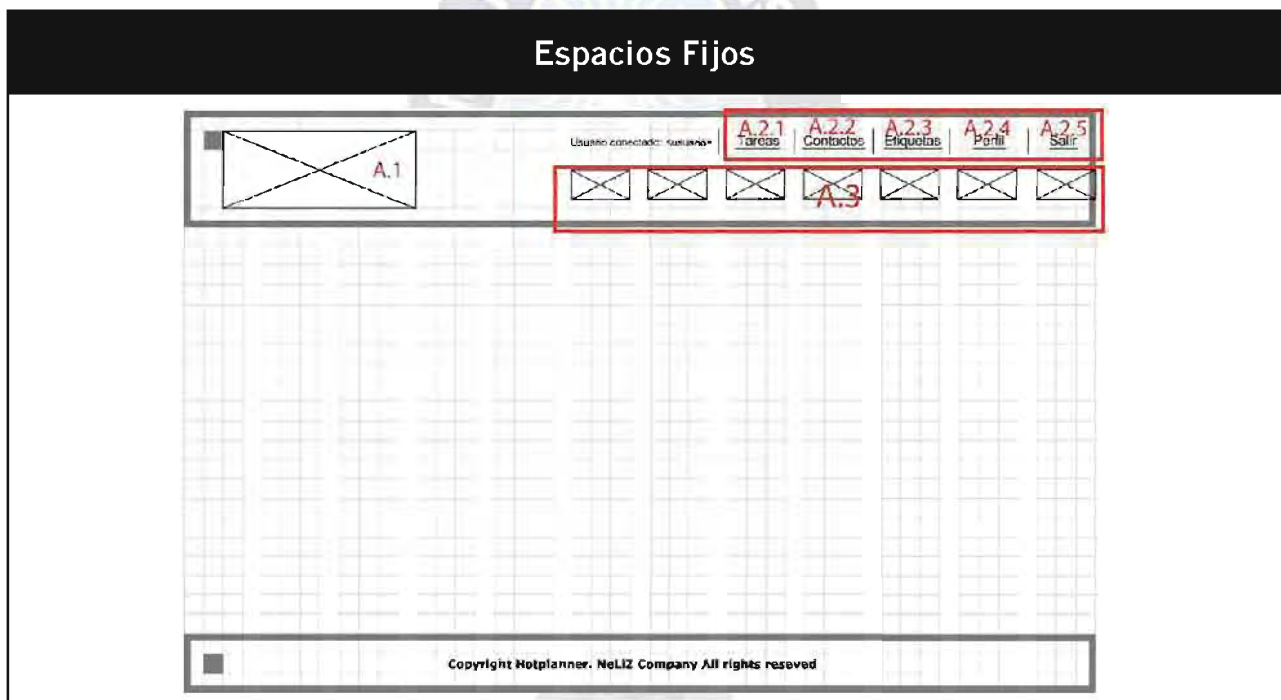


ANEXO M

HISTORIAL DE REVISIONES			
Fecha	Versión	Autor	Descripción
15/06/10	1.0	Elizabeth Aróstegui	Tercera fase del modelo

Formato de registro

ARMAZONES WEB HOTPLANNER



Acción / Respuesta		Información de Diagramación
<p>A.1, imagen Descripción: <i>Logotipo del sitio</i></p> <p>A.3, Catálogo Descripción: <i>Iconos del pronóstico de tiempo</i> pre-condición: <i>Ingreso al sistema</i> post-condición: <i>Información sobre estado del tiempo en las tareas</i></p>	<p>A.2, Barra de Navegación Descripción: <i>Barra de navegación principal</i> pre-condición: <i>Ingreso al sistema</i> post-condición: <i>Navegación a las secciones</i></p> <p>A.2.1, Vínculo Evento: <i>onclick</i> Tarea: <i>Oper.navegacion(Tareas)</i> Zona: <i>Ex.total</i></p> <p>A.2.2, Vínculo Evento: <i>onclick</i> Tarea: <i>Oper.navegacion(Contactos)</i> Zona: <i>Ex.total</i></p>	<p>■ A</p> <p>■ B</p>

ID	Nombre de la Página	Versión	Descripción	Des	Acción / Respuesta
1	Página principal	1.1		No	<p>1.1, Imagen Descripción: Logotipo del sitio</p> <p>3.1, 4.1, 5.1, 6.1, Imagen Descripción: Imágenes de promoción y explicación de la aplicación</p> <p>1.2, Formulario (login) Descripción: Formulario de ingreso al sistema pre-condición: Entrar a la página principal post-condición: Ingreso al sistema</p> <p>1.2.3, Botón (Buscar) Evento: onclick, onkeypress(enter) Tarea: Oper.envia(Tareas) Zona: Ex.total 2 Tarea: Dial.error (Usuario no registrado, revise tus datos) Zona: In.objeto(1.2)</p>
					<p>Información de Diagramación</p>

ID	Nombre de la Página	Versión	Descripción	Descartar	Acción / Respuesta
2	Tareas	1.1	Página dedica al manejo de las tareas del usuario	No	<p>5.1, Catálogo (Sugerencias para hacer contactos)</p> <p>Descripción: Contactos que le sistema sugiere</p> <p>pre-condición: Ingreso al sistema</p> <p>post-condición: Invitación o rechazo de sugerencia</p> <p>5.1.1, Botón</p> <p>Evento: onclick</p> <p>1 Tarea: DialConfirm (Debes invitar a este usuario)</p> <p>(acceptar) -> Oper.enviar(Tareas)</p> <p>2 Tarea: DialInform (Tu invitación fue enviada con éxito)</p> <p>Zona: Zona: Inuspacio(!)</p> <p>(cancelar) -> Oper.navegacion(Tareas)</p> <p>5.1.2, Botón</p> <p>Evento: onclick</p> <p>1 Tarea: DialConfirm (Estás seguro de ignorar esta sugerencia?)</p> <p>(acceptar) -> Oper.enviar(Tareas)</p> <p>2 Tarea: DialInform (Un usuario fue rechazado de la lista de sugerencias)</p> <p>Zona: Zona: Inuspacio(!)</p> <p>(cancelar) -> Oper.navegacion(Tareas)</p> <p>5.1.3, Imagen</p> <p>Descripción: Foto de contacto</p> <p>5.1.4, Texto</p> <p>Descripción: Nombre del contacto</p>



ANEXO N

HISTORIAL DE REVISIONES			
Fecha	Versión	Autor	Descripción
15/06/10	1.0	Elizabeth Aróstegui	Cuarta fase del modelo

Formato de registro

GUÍA DE ESTILO VISUAL HOTPLANNER

1. Paleta cromática

Color	Hexadecimal	RGB	Significado
	#EB6D1F	R: 235 G: 109 B:31	Color de la Marca
	#FBBE0E	R: 251 G: 190 B:14	Color de la Marca
	#9C9E9F	R: 156 G: 158 B:159	Color de la Marca
	#000000	R: 0 G: 0 B: 0	Tipografía para títulos
	#665665	R: 56 G: 61 B: 66	Tipografía para texto general
	#969696	R: 150 G: 150 B: 150	Tipografía para comentarios
	#FF0000	R: 255 G: 0 B: 0	Error, Atención, Restricción, Alto
	#FFFF00	R: 255 G: 255 B: 0	Pregunta, Advertencia
	#00C800	R: 0 G: 200 B: 0	Resultado exitoso

2. Elementos gráficos

Elemento	Muestras		Dimensiones
	Iconos de Función	Iconos de símbolos	
Iconos			Menor: 20 px
			Mediano: 40 px
			Grande: 80 px
Botones			

3. Tipografía

3.1. Estilos de Tipografía

Tipografía	Familia	Tamaño	Grosor	Estilo	Formato
Marca	Plastique	(no determinado)	Regular	Normal	Mayúscula
Marca	Berlin Sans FB	(no determinado)	Regular	Normal	Mayúscula
Títulos	Arial	18 pt	Bold	Subrayada	Mayúscula
General	Arial	14 pt	Regular	Normal	Tipo oración
Comentarios	Arial	11 pt	Light	Itálica	Tipo oración

3.2. Tipografía de la Marca

Familia	Muestra
Plastique	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 1 2 3 4 5 6 7 8 9 0 ! ? % & / () = * + - ; : , .
Berlin Sans FB	a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 1 2 3 4 5 6 7 8 9 0 ! ? % & / () = * + - ; : , .

4. Estilo de vínculos

Texto	Estado de Vínculo	Estilo	Grosor	Color	Hexadecimal
Título	Reposo	Subrayada	Bold		#000000
	Sobre	Subrayada	Bold		#0000FF
	Presionado	normal	Bold		#0000FF
	Visitado	normal	Bold		#4C0099
General	Reposo	Itálica	normal		#665665
	Sobre	Itálica	normal		#8282FF
	Presionado	Itálica	normal		#8282FF
	Visitado	Itálica	normal		#780078

5. Jerarquía visual

ID	1	2
1	Imagen	Contrastado
2	Imagen	Contrastado
3	Imagen	Contrastado
4	Imagen	Contrastado



ANEXO 0

DIAGRAMA DE CASOS DE USO DE LA APLICACIÓN HOTPLANNER

CASOS DE USO GENERALES



CASOS DE USO ESPECÍFICOS

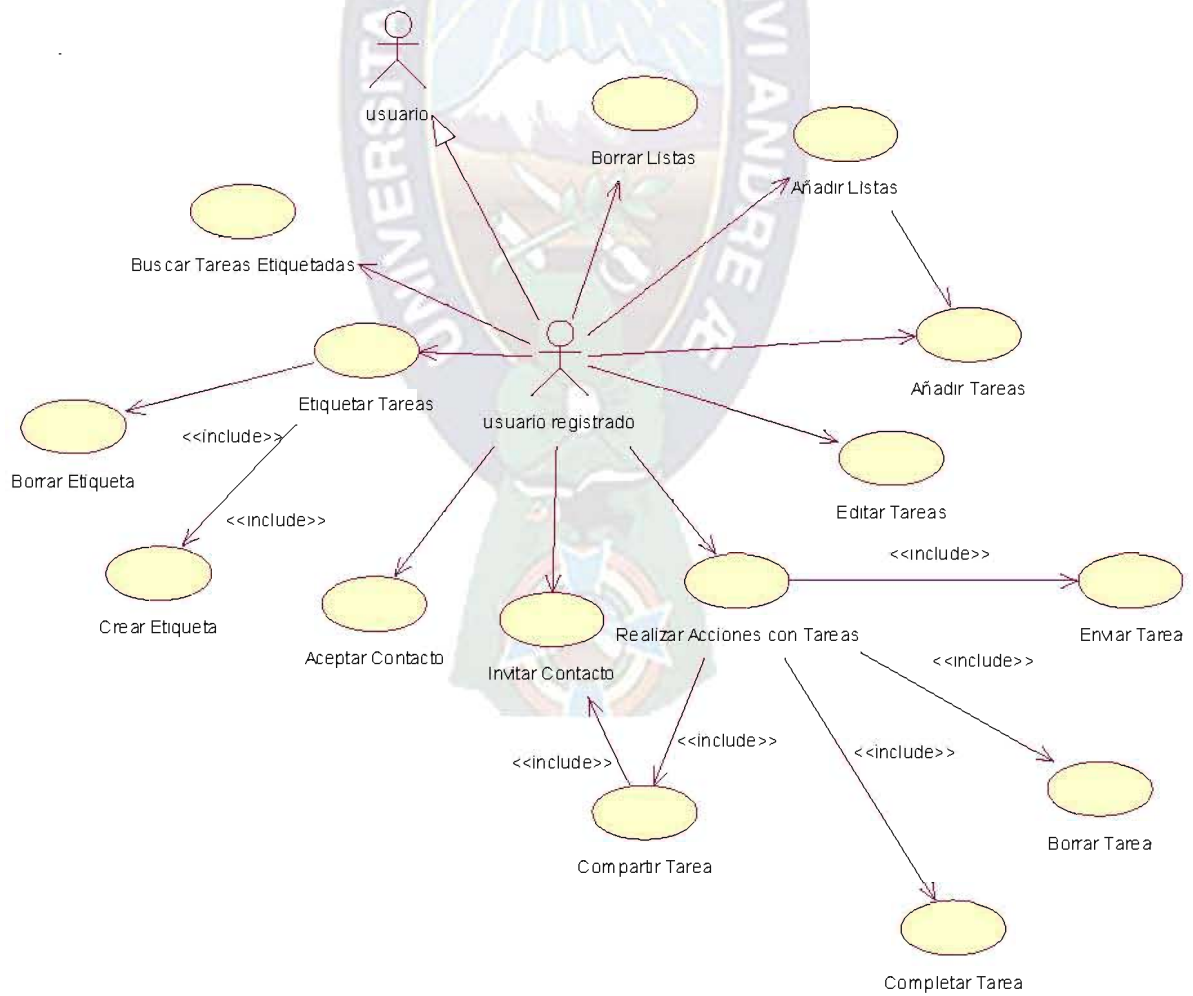
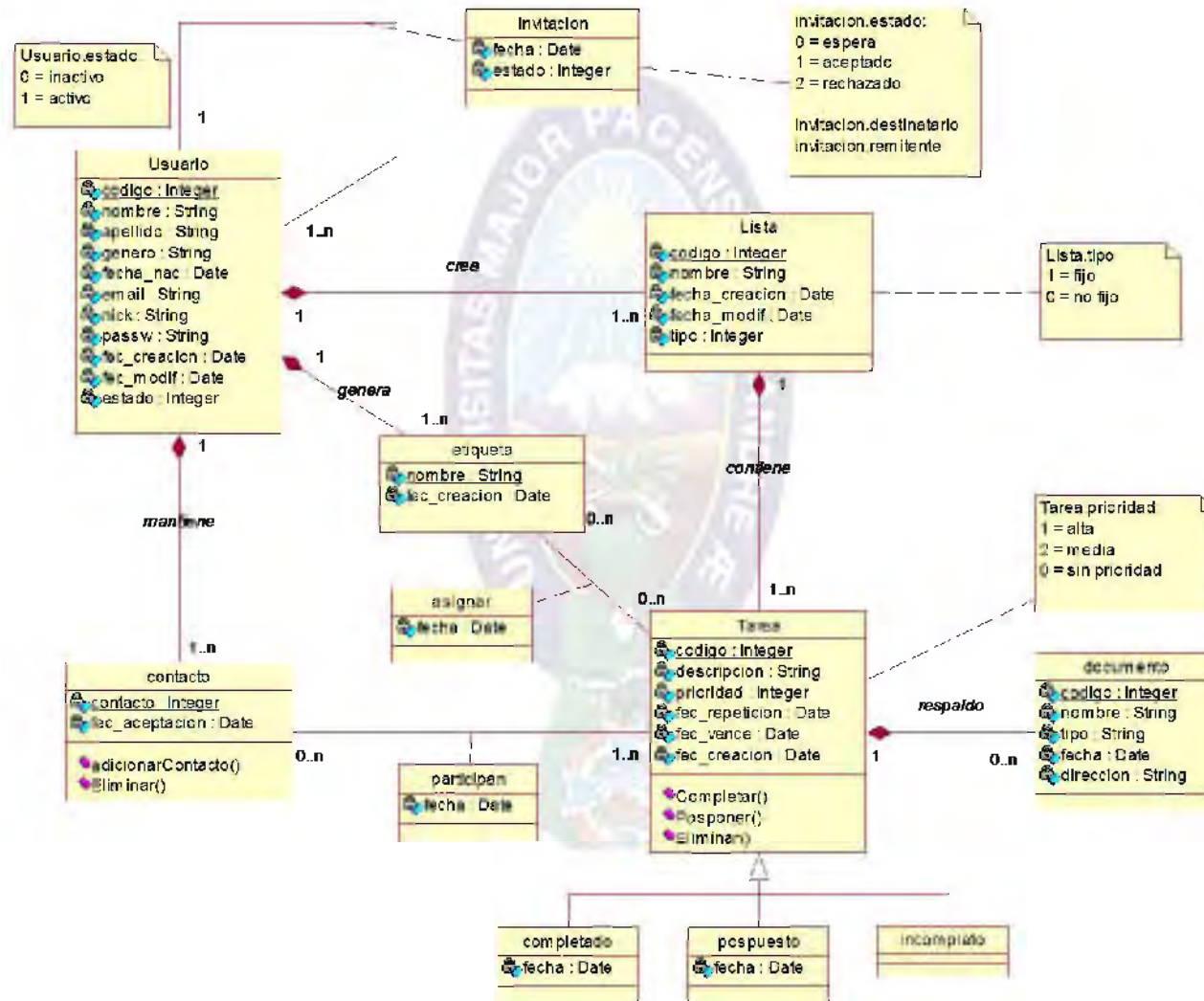
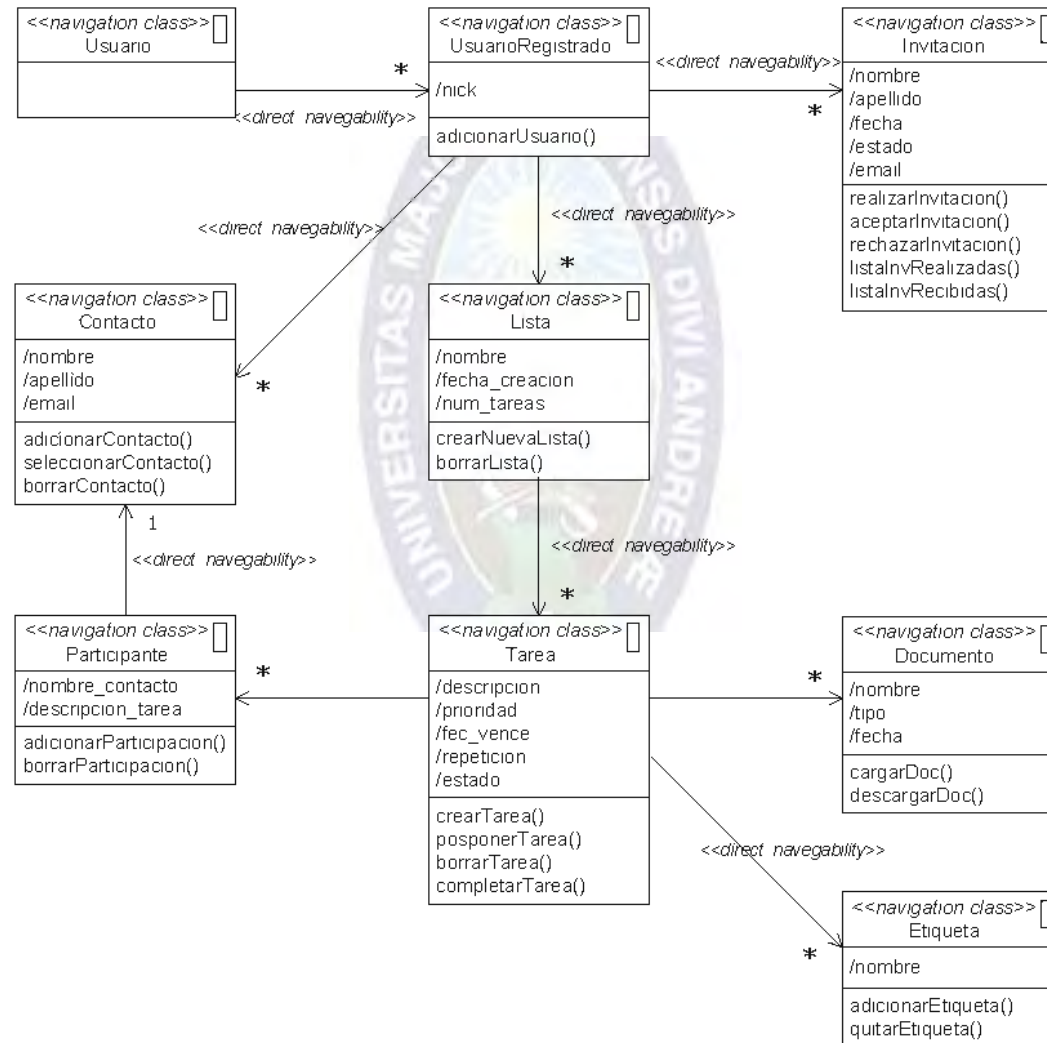


DIAGRAMA DE CLASES DE LA APLICACIÓN HOTPLANNER



MODELO DE ESPACIO DE NAVEGACIÓN DE LA APLICACIÓN HOTPLANNER



MODELO DE ESTRUCTURA DE NAVEGACIÓN DE LA APLICACIÓN HOTPLANNER

