

II ENCUENTRO DE JÓVENES INVESTIGADORES

“Consolidando espacios del quehacer científico en San Juan”

Propuesta:

ANTOLOGÍA DE PATRONES DE INTERACCIÓN APLICADOS A IMPLEMENTACIÓN DE REFERENCIA

Autor: *Marcuzzi, Cecilia. Becaria CICITCA 2011-2012.*

Institución: *Instituto de Informática – Fac. Cs. Exactas F. y Naturales –
Universidad Nacional de San Juan*

2013

Antecedentes

El presente trabajo es la consecución de un proyecto realizado dentro de una beca de investigación para estudiantes avanzados, en el marco del proyecto de investigación “Formalización de descripciones de Casos de Uso a través de metamodelos. ForCUPIDo” (E-883) y como trabajo final de la Licenciatura en Sistemas de Información. Inicialmente se analizaron las características de usabilidad en el diseño de una interfaz, dentro de la disciplina Interacción Humano-Computador, luego se detallaron y estudiaron los patrones de interfaz aplicados a casos de estudio. A partir del conocimiento adquirido, se extendió la plantilla CUPIDo (Plantilla Integradora para Documentar Casos de Uso v1.4) [1], incorporando un nuevo campo en la misma, para permitir identificar los patrones a aplicar en el diseño de la interfaz de ese caso de uso, obteniendo como resultado la versión 1.5 de la plantilla, denominada CUPIDo+Pi. Esto dio origen a la Antología de patrones de interfaz y a la necesidad del desarrollo de una aplicación que permita a los analistas y desarrolladores visualizar los diferentes patrones y aplicarlos en etapas tempranas del desarrollo de software, en la etapa de modelamiento con casos de uso.

Fundamentación

En los procesos de ingeniería de software se hace fuerte hincapié en detectar, en etapas tempranas del desarrollo de software, todas las necesidades del nuevo sistema... En un mercado o una sociedad actual cada vez más informatizados y más exigente, con tecnologías de avanzada, el usuario común no se conforma ya con un sistema que hace lo que él necesita, sino que su conformidad o satisfacción va acompañada con otras características adicionales, como la facilidad de uso, comodidad en el uso y aspectos de diseño visual del mismo [2]. Estos aspectos son conocidos, dentro de la disciplina de Interacción Humano-Computador, como características de ‘usabilidad’ del software y se considera un atributo de calidad del software [3].

En base a estos principios y con el objeto brindar al cliente un software que cubra todas sus necesidades y expectativas, surge la necesidad de identificar patrones de usabilidad para el diseño de interfaz, en etapas tempranas del desarrollo de software.

Los patrones son soluciones a problemas comunes y recurrentes en el diseño. Es buena costumbre implementarlos ya que ahorran tiempo y mejoran en efectividad y eficiencia la aplicación. Christopher Alexander [4] define: "*Cada patrón de diseño describe un problema que ocurre una y otra vez en nuestro entorno, así como la solución a ese problema, de tal modo que se puede aplicar esta solución un millón de veces, sin hacer lo mismo dos veces*", citado por Erich Gamma y otros [5]. Los diseñadores y desarrolladores históricamente no le han dado la suficiente importancia al aspecto de usabilidad dentro del proceso de diseño y además no existe una guía estandarizada que los ayude en el mismo [6].

Teniendo en cuenta las falencias en el diseño de interfaces desde la perspectiva de la usabilidad, se ve la necesidad de presentar una iniciativa que permita minimizar dicha problemática, con la definición de un marco de referencia de patrones HCI (Interacción Humano-Computador) para el diseño de interfaces, que sirva de guía a los diseñadores y desarrolladores en el proceso de diseño y construcción de interfaces en el marco del Desarrollo Dirigido por Modelos (MDD) [7], [8].

Objetivos

General:

Generar una Antología de patrones de diseño de Interfaces y una aplicación software que sirva de referencia para la utilización de los patrones de interfaz, en etapas tempranas del desarrollo de software.

Específicos:

- Identificar patrones y extender la antología de patrones de interacción y usabilidad.
- Desarrollar una aplicación software o prototipo de software que pueda ser usado como referencia y de ayuda al diseñador, para el uso y aplicación de patrones de interfaz.

Desarrollo

Primero se elaboró una antología, la cual se define como una recopilación de obras ó temáticas, que se seleccionan por algún motivo específico, por pertenecer a alguien o por un fin determinado [9].

Se consideraron los patrones de interacción de V. Welie y H. Trætteberg [10] para la generación de la antología en la cual se detallan la descripción del problema/solución, como así también las categorías y sub-categorías de patrones. La clasificación principal de los patrones de interacción consiste en: Necesidades del usuario (User Needs), Necesidades de las aplicaciones (Applications Needs) y Contexto del diseño (Context of Desing). A continuación se presenta una tabla con el resumen de algunos de los patrones estudiados, debido a que la clasificación de los mismos es muy extensa.

Tabla 1. Patrones de Interacción

USER NEEDS	
Basic Interactions	
Wizard	"Asistente". Guía al usuario en la realización de una tarea que necesita tomar varias decisiones, mostrarle cuales son los pasos que existen y cuáles han sido realizados a la vez que se le guía a lo largo de la tarea.
Stepping	"Paso a Paso". Permite a los usuarios ir al paso anterior y al siguiente de una tarea, es una de las formas más básicas para navegar.
Paging	"Pagar". Permite a los usuarios buscar algún tema de interés navegando a través de una lista de elementos. Este patrón presenta los resultados agrupados en páginas con un número fijo de artículos.
Slideshow	"Presentación". Permite a los usuarios ver una serie de imágenes o fotos, durante unos segundos. Posee controles para "anterior" y "siguiente".
Miscelleaneous	
News Box	"Cajas de noticias". Adiciona secciones de noticias breves, las cuales continuamente se actualizan. Es útil en aquellos sitios que poseen información reciente.
Footer Bar	"Barra inferior". Barra que aparece en la parte inferior de la página conteniendo los asuntos legales del sitio.
Hotlist	"Lista picante". Este patrón está conformado por una lista de elementos que son populares dentro de un conjunto. Es útil en los casos donde se debe elegir algún elemento pero existe una gran cantidad de ellos.

Making Choices	
Country/Region Selector	“Selector de región o país”. Permite seleccionar un país o ciudad específica de una lista desplegable organizada alfabéticamente.
Poll	“Encuesta”. Útil para realizar encuestas a los usuarios y luego relevar la información obtenida en el sitio.
APPLICATION NEEDS	
Feedback	
Input Error Message	“Mensaje de Error de Entrada”. Informa al usuario acerca de entradas de datos incorrectas, dónde se han producido y cómo resolverlas.
Progress indication	“Indicador de Progreso”. Muestra un indicador con el progreso real o estimado de que se está avanzando. Se utiliza en los casos que se debe esperar unos segundos a que se cargue alguna aplicación o página.
Drawing Attention	
Outgoing Links	“Enlaces de Salida”. Muestra al usuario los enlaces que le llevarán fuera del actual sitio web marcándolos con un icono después de su etiqueta.
Color Coded Section	“Sección codificada por color”. Permite que los usuarios reconozcan si están en el lugar correcto. Para ello, cada sección debe tener un color diferente que lo identifique.
Alternating Row Colors	“Color de fila que varía”. Se utiliza cuando los usuarios necesitan buscar información específica en una tabla. Para ello se confecciona una tabla con filas de colores alternados y de esta forma los datos son más legibles.
Simplifying Interaction	
Enlarged Click area	“Área de clickeo ampliada”. Este patrón permite ampliar las zonas de selección y es útil cuando los ítems son relativamente pequeños por lo que su selección resulta complicada.
Font Enlarger	“Amplificador de fuente”. Permite visualizar las páginas con un tamaño mayor de fuente. Los usuarios pueden incrementar o decrementar el tamaño de letra utilizando.
CONTEXT OF DESIGN	
Site Types	
Community Site	“Sitio comunitario”. Sitios de personas que poseen intereses comunes y actúan como un grupo para sí mismos y para ayudar a los demás.

Simplifying Interaction	
Enlarged Click area	“Área de clickeo ampliada”. Este patrón permite ampliar las zonas de selección y es útil cuando los ítems son relativamente pequeños por lo que su selección resulta complicada.
Font Enlarger	“Ampliador de fuente”. Permite visualizar las páginas con un tamaño mayor de fuente. Los usuarios pueden incrementar o decrementar el tamaño de letra utilizando.
CONTEXT OF DESIGN	
Site Types	
Community Site	“Sitio comunitario”. Sitios de personas que poseen intereses comunes y actúan como un grupo para sí mismos y para ayudar a los demás.
Travel Site	“Sitio de viajes”. Permite a los usuarios reservar un viaje o un vuelo a un destino determinado, a través de la búsqueda en una base de datos con las posibles reservas.
Experiences	
Shopping	“Comercial”. Se utiliza cuando los usuarios desean buscar productos o realizar compras en línea, más comúnmente denominado e-commerce.
Fun	“Entretenimiento”. Son sitios cuyo objetivo es la diversión y entretenimiento de los usuarios.
Page Types	
Printer-Friendly Page	“Página amigable para imprimir”. Permite imprimir el contenido de una página visualizada por el usuario.
Article Page	“Página de artículo”. Permite a los usuarios leer una gran cantidad de texto en un formato y estructura consistente, ubicado en la parte central de la página.

Una vez recopilados y analizados los patrones de interacción, se llevó a cabo la instancia de creación de la implementación de referencia, es decir el desarrollo del software de aplicación. Para lo cual, primero se realizó el diseño de prototipos de la implementación.

En la figura 1 es posible visualizar el menú de categorías principales y sub-categorías y un selector de patrones por cada una. Por ejemplo, se seleccionó la pestaña User Needs, opción Making Choices y se eligió de la lista el patrón Slideshow. Una vez elegido el patrón, aparecerán los campos para que el usuario ingrese los datos requeridos. Como resultado se obtiene una lista de imágenes que se deslizan en forma horizontal, acompañada de controles para desplazarse manualmente hacia

atrás, hacia adelante, hacer una pausa o reanudar la presentación cuando se interrumpe.

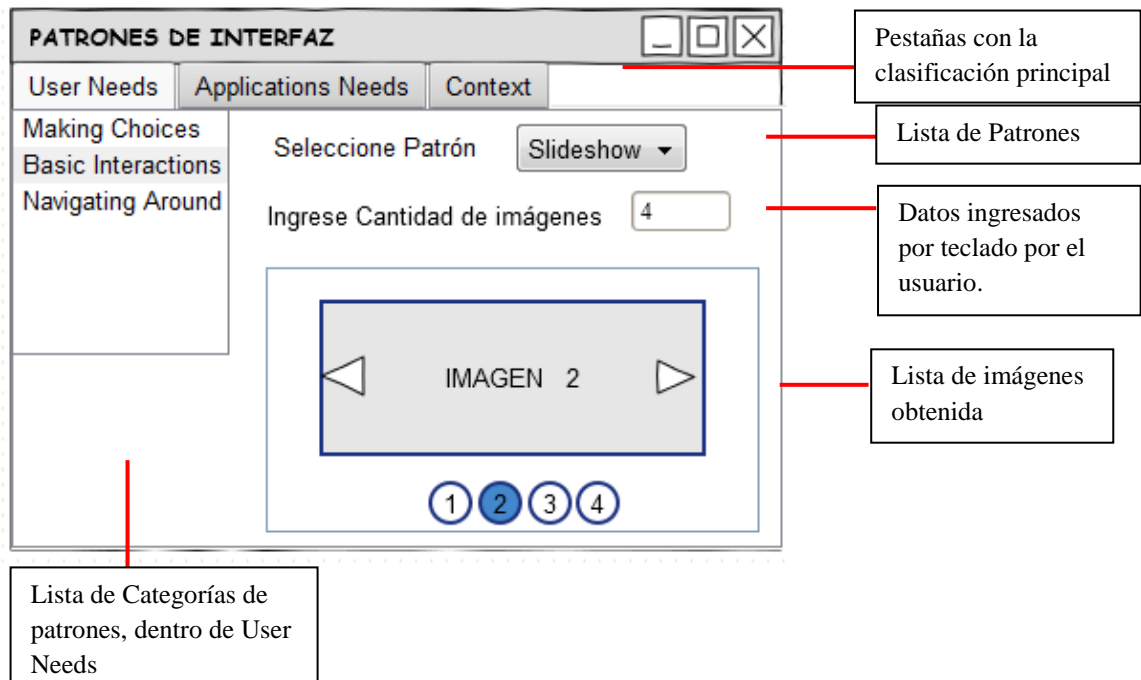


Fig. 1. Prototipo del patrón Slideshow

La implementación se codificó en lenguaje php [11] por tratarse de una herramienta open source. A continuación, en la figura 2 se muestra un fragmento del código del patrón slideshow.

```

<?php
    include("showslider.php");
    $show=new ShowSlider('showlider',618,246);
    $show->add_slide('imgs/foto1.jpg','','');
    $show-> add_slide ('imgs/foto2.jpg','','');
    $show->add_slide('imgs/foto3.jpg','','');
    $show->add_slide('imgs/foto4.jpg','','');
    ?>
<html>
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8" />
<?php $show->render_includes(); ?>
<title>Integrating ShowSlider with PHP</title>
</head>

<body>
<?php $show->render_slides() ?>
</body>
</html>

```

Fig. 2. Código del patrón slideshow

En la codificación de la figura 2 Primero que nada se debe incluir la clase *showslider*. Posteriormente crear un objeto, donde el primer parámetro es el nombre del directorio que contiene los archivos *showslider* y los parámetros siguientes indican el tamaño de las imágenes. Luego se utilizan diferentes métodos de la clase para mostrar el *slideshow*: *add_slide* para agregar imágenes, *render_includes* es el método que incluye archivos *css* y *js*, y el método *render_slides* que muestra el *showslider*.

Conclusiones

Poder realizar esta aplicación concreta permitió destacar la importancia del uso de patrones de interfaz en etapas tempranas de desarrollo de software, en la etapa de análisis del sistema, a fin de que el diseñador tenga en cuenta estas características y junto al diseño del sistema pueda generarse el diseño de la interfaz, reduciendo tiempos minimizando el re-trabajo posterior, ya que la prototipación permite mostrar al usuario el sistema, promoviendo la validación por parte del usuario.

Por medio de la implementación de una aplicación o prototipo de software de referencia de los patrones de interfaz, es posible mostrar al cliente o usuario los prototipos con la incorporación de patrones de interacción, lo cual es un valor agregado y sustancial que, no sólo permitirá a los usuarios vislumbrar el alcance del sistema, sino también familiarizarse con “look and feel” del estilo y diseño de la interfaz, minimizando el impacto de la implantación del nuevo sistema. Como trabajo a futuro se propone aplicar esta implementación en la asignatura Diseño de software, de las carreras de Licenciatura en Sistemas de Información y Ciencias de la Computación, de la Fac. de Cs. Exactas, Físicas y Naturales, para validar la usabilidad de la misma y de esa forma mejorar la aplicación. El uso de esta aplicación por parte de los alumnos se espera les permita alcanzar diseños más usables y eficientes, tanto en el ejercicio de sus actividades prácticas como también las profesionales.

Al ser una aplicación desarrollada con herramientas Open Source, se pretende dejar libre el acceso a la misma, para quien desee usarla, esto permitirá también la validación externa y la consiguiente retroalimentación para mejorarla.

Bibliografía

- [1] M. I. Lund, C. Ferrarini, L. N. Aballay, y E. Meni, “CUPIDo - Plantilla para Documentar Casos de Uso”, presented at the V Congreso de Tecnología en Educación y Educación en Tecnología, El Calafate, Santa Cruz, Argentina, 2010.
- [2] J. Nielsen, Designing web usability. New Riders, 2000.
- [3] J. Nielsen, «Heuristic evaluation», en Usability inspection methods, J. Nielsen y R. L. Mack, Eds. New York, NY, USA: John Wiley & Sons, Inc., 1994, pp. 25–62.
- [4] C. Alexander, S. Ishikawa, M. Silverstein, M. Jacobson, S. Angel, y I. Fiksdahl-King, El Lenguaje de Patrones. Gustavo Gili Editorial S.A., 1980.
- [5] E. Gamma, R. Helm, R. Johnson, y J. Vlissides, Design Patterns: Elements of Reusable Object-oriented Software. Pearson Education, 2004.
- [6] G. Ford y H. Gelderblom, «The effects of culture on performance achieved through the use of human computer interaction», en Proceedings of the 2003 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology, Republic of South Africa, 2003, pp. 218–230.
- [7] O. Pastor y J. C. Molina, Model-Driven Architecture in Practice: A Software Production Environment Based on Conceptual Modeling. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.
- [8] C. Atkinson y T. Kuhne, «Model-driven development: a metamodeling foundation», IEEE Software, vol. 20, n.o 5, pp. 36-41, 2003.
- [9] Antología - Wikipedia, la enciclopedia libre, <http://www.wikipedia.com>
- [10] M. van Welie y H. Trætteberg, “Interaction Patterns in User Interfaces”, in Proc. Seventh Pattern Languages of Programs Conference: PLoP 2000, pp. 13–16, 2000
- [11] Boronczyk, T., Naramore, E., Gerner, J., Scouarnec, Y., Stolz, J., Glass, M., “Desarrollo web con php 6, apache y mysql”, Multimedia/Wrox, pág.: 816, Oct. 2009.