

Generación Automática de Aplicaciones Web Móviles Mediante Componentes Configurables

Pablo Vera, Claudia Pons, Carina González,
Rocío Rodríguez, Daniel Giulianelli

Universidad Nacional de La Matanza
Departamento de Ingeniería e Investigaciones Tecnológicas
Florencio Varela 1903, San Justo, Buenos Aires, Argentina. Tel: 4480-8900 (int 8630)
pvera@ing.unlam.edu.ar, rrodriguez@ing.unlam.edu.ar, dgiulian@ing.unlam.edu.ar

Universidad Nacional de la Plata
LIFIA – Laboratorio de Investigación y Formación en Informática Avanzada
Calle 50 y 150, La Plata, Buenos Aires, Argentina. Tel: 221-422-8252
cpons@lifia.info.unlp.edu.ar

Universidad de la Laguna
Departamento de Ingeniería de Sistemas y Automática y Arquitectura y Tecnología de Computadores. Área de Arquitectura y Tecnología de Computadores
La Laguna, España. Tel: 922318284
cjgonza@ull.es

Resumen

Teniendo como objetivo final la generación completa del código fuente de una aplicación, se ha desarrollado CBDHM (Metodología de Modelado Hipermedia Basada en Componentes).

Esta metodología está basada en UML (Lenguaje Unificado de Modelado) y permite diseñar y construir aplicaciones web móviles utilizando dos modelos: el modelo de datos y el modelo de interfaz de usuario. El modelo de datos está basado en el diagrama de clases UML extendido para poder generar la base de datos y sus relaciones. El modelado de la interfaz de usuario se realiza mediante componentes predefinidos configurables. Para poder realizar la configuración, el diagrama de componentes es extendido mediante: estereotipos para identificar cada tipo de componente y valores etiquetados para especificar su funcionalidad de forma clara y concisa pero a su vez conteniendo toda la información

necesaria y suficiente para permitir generar una aplicación completa y funcional. Esta metodología se enmarca en el ámbito de MDA (Arquitectura Dirigida por Modelos) y plantea una primera transformación de modelo a modelo y luego una segunda transformación de los modelos al código fuente y script de la base de datos. Como soporte a la metodología se ha desarrollado una herramienta que permite realizar los modelos y las transformaciones.

Palabras clave: MDA, UML, Hipermedia Móvil.

Contexto

Este proyecto es financiado por la Universidad Nacional de La Matanza, cuenta con asesoría y colaboración de la Universidad Nacional de La Plata y la Universidad de La Laguna (España). Se corresponde a una línea de investigación y desarrollo dentro del área de MDA.

Introducción

CBHDM (Component Base Hypermedia Design Methodology) es una metodología de modelado de aplicaciones web móviles que permite generar código fuente mediante el enfoque MDA [1] (Arquitectura Dirigida por Modelos).

CBHDM necesita dos modelos: el modelo de datos (también denominado modelo conceptual) y el modelo de la interfaz de usuario (incluyendo navegación).

El modelo de datos está basado en el diagrama de clases de UML (Lenguaje Unificado de Modelado) y el modelo de la interfaz en el diagrama de componentes de UML. Ambos diagramas son extendidos mediante estereotipos y valores etiquetados, generando una extensión conservativa de UML. Al tratarse de una extensión conservativa el modelado podría realizarse con cualquier herramienta existente pero para evitar tener que recordar cómo escribir la configuración válida en cada valor etiquetado, se ha creado una herramienta de soporte que facilita la creación y configuración de los modelos y al mismo permite generar las transformaciones automáticas que la metodología prevé.

Los pasos de la metodología son los siguientes:

1. **Modelo Conceptual o de Datos:** El usuario confecciona el diagrama de clases UML, especificando ciertas propiedades definidas en la extensión del modelo para identificar el identificador de la clase y su descriptor. También hace uso de tipos de dato especialmente definidos para aplicaciones móviles.
2. **Transformación de Modelo a Modelo:** Basado en el diagrama de clases se generan en forma automática un modelo de Navegación e Interfaz con los componentes básicos para poder administrar la información de la base de datos. Por cada clase se crea un componente para listar los registros de la tabla, su correspondiente pantalla de edición y un menú apuntando a cada uno de los listados generados. También se genera en forma automática la navegación entre los componentes según lo definido por el W3C [2] como estándar para navegación en aplicaciones web móviles.
3. **Modelo de Navegación e Interfaz:** el usuario completa y/o modifica el diagrama generado en forma automática en el paso 2 para adaptarlo a la aplicación particular que está modelando. Los componentes disponibles en este modelo son:
 - a. **List:** para mostrar listados de información
 - b. **Search:** para mostrar listados de información permitiendo aplicar filtros de búsqueda.
 - c. **Login:** para ingreso al sistema
 - d. **UpdateView:** permite actualización de ciertas propiedades del objeto dejando otras de solo lectura
 - e. **CRUD:** creación y modificación de objetos
 - f. **Menu:** Menú con links para llegar a las distintas partes de la aplicación
4. **Transformación de Modelo a Código:** con los distintos modelos generados se genera en forma automática es script de la base de datos y el código fuente de la aplicación.

Herramienta de soporte al modelado

Se construyó una herramienta para poder modelar más fácilmente con esta metodología. La herramienta es una

aplicación web programada en C# sobre ASP.NET 4.5 con bases de datos SQL Server.

Incluye soporte para múltiples usuarios y múltiples proyectos.

Soporte para el modelo de datos

La herramienta permite definir las clases que pertenecen al modelo de datos permitiendo agregar clases estándar o clases del tipo enumeración. La figura 1 muestra un ejemplo de un modelo de datos donde se puede apreciar la presencia de botones para agregar nuevas clases y enumeraciones.

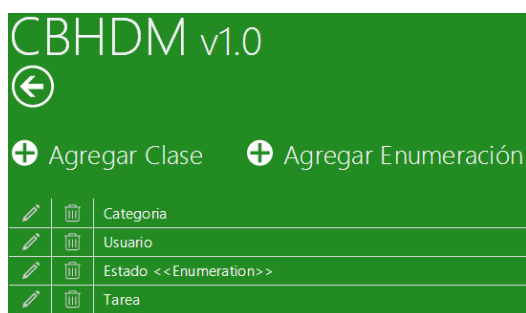


Figura 1: Clases y enumeraciones de un modelo de ejemplo.

Al momento de agregar una clase una enumeración la herramienta nos va a solicitar las propiedades estereotipadas que son necesarias para el modelo, es decir el identificador y el descriptor de la clase. De forma automática creará una propiedad del tipo entero para el identificador y una propiedad del tipo string para el descriptor. Luego permitirá agregar propiedades adicionales para el caso de las clases o valores de enumeración si se trata de una clase de ese tipo. La figura 2 muestra la pantalla para editar una clase del modelo, donde pueden verse las propiedades antes mencionadas y luego un listado de propiedades propias para la clase en cuestión.

✎ ✕	FechaCreacion :	Fecha / Hora
✎ ✕	FechaModificacion :	Fecha / Hora
✎ ✕	Usuario :	Usuario

Grabar y Cerrar

Figura 2: Pantalla de edición de una clase.

Soporte para el modelo de la interfaz de usuario

El sistema permite agregar los distintos componentes de la interfaz de usuario y además en cada uno de ellos existe un control particular que ayuda a la configuración del comportamiento de cada sección del componente.

La figura 3 muestra un ejemplo de componentes generados para una aplicación.

Componentes...			
		Identificador	Tipo
✎ ✎ ✕		cpnCRUDCategoria	CRUD
✎ ✎ ✕		cpnLstCategoria	List
✎ ✎ ✕		cpnCRUDUsuario	CRUD
✎ ✎ ✕		cpnLstUsuario	List
✎ ✎ ✕		cpnCRUDTarea	CRUD
✎ ✎ ✕		cpnTareasPendientes	List
✎ ✎ ✕		cpnLogin	Login
✎ ✎ ✕		cpnBusquedaTareas	Search
✎ ✎ ✕		cpnMainMenu	Menu

Figura 3: Componentes de un modelo de ejemplo

La figura 4 muestra un ejemplo de configuración de un valor etiquetado en particular de un componente, en este caso la ayuda para poder configurar los links que pueden ser parte por ejemplo del menú principal de la aplicación o los links incluidos en la barra de navegación de un componente particular.

Figura 4: Ayuda para configuración de los links.

Soporte para transformaciones

La herramienta permite realizar las dos transformaciones definidas por la metodología. La primera transformación que genera los componentes partiendo del diagrama de clases y la segunda que genera el código fuente y el script de la base de datos. La herramienta hace un seguimiento de las modificaciones realizadas en los modelos y da avisos al usuario para que realice las transformaciones necesarias para tener actualizado su proyecto. A modo de ejemplo, puede observarse en la figura 5 que se ha realizado una modificación en las clases del modelo conceptual pero no se han generado los componentes por eso se da una aviso al usuario de que los componentes están desactualizados.

En esta figura también puede observarse el menú principal de un proyecto con acceso a las clases, componentes y las transformaciones.



Figura 5: Menú con opciones para realizar las transformaciones y aviso de componentes desactualizados.

Generación de la aplicación

Una vez completado el modelo se realiza la generación de la aplicación, lo que genera por un lado es script de la base de datos y por el otro el código fuente completo de la aplicación. A modo de ejemplo en la Figura 6 se muestran pantallas funcionales generadas con la metodología.

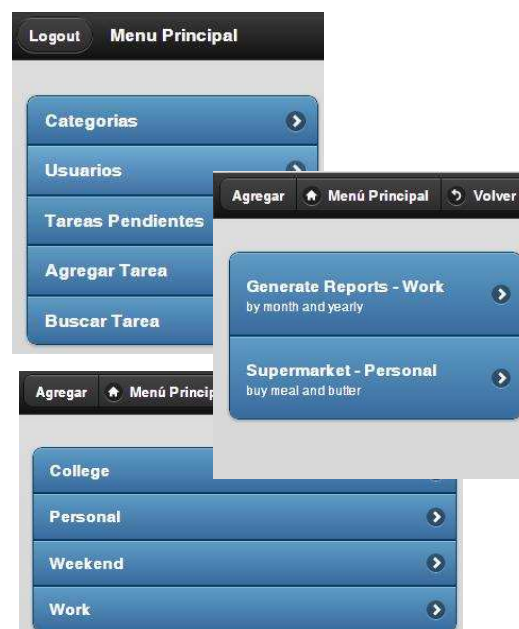


Figura 6: Ejemplos de pantallas de una aplicación funcional generadas con CBHDM

Líneas de Investigación, Desarrollo e Innovación

Los temas que se desarrollan dentro de esta línea son los siguientes:

- Desarrollo Dirigido por Modelos
- Modelado de hipermedia móvil
- Modelado basado en componentes
- Perfiles de UML
- Generación automática de Código

Resultados y Objetivos

El objetivo principal de esta línea de Investigación y Desarrollo es definir una metodología de modelado de aplicaciones web móviles que permita la generación de código automático.

Actualmente este proyecto se encuentra en un estado avanzando por lo que la metodología ya ha sido completamente definida y se está trabajando en la finalización de la herramienta de soporte. La herramienta actualmente soporta:

- Modelado de clases
- Modelado de componentes
- Transformación de clases a componentes
- Generación del script de base de datos
- Generación del código fuente de los componentes tipo Menú y List.

Como trabajo futuro se planea:

- Finalizar la generación del código para los componentes restantes
- Agregar templates para poder generar código fuente en distintos lenguajes.
- Agregar interoperabilidad a la herramienta incorporando la posibilidad de importar y

exportar modelos al formato XMI [3].

Desde el inicio del proyecto se han realizado distintas publicaciones académicas mostrando los avances del mismo tanto en congresos nacionales como internacionales. Las publicaciones más recientes son: [4], [5] y [6].

Formación de Recursos Humanos

En esta línea de investigación participan 6 docentes y 2 alumnos en formación.

Este trabajo tiene relación directa con una tesis Doctoral en desarrollo por uno de los docentes del grupo de investigación

Referencias

- [1] Kleppe A., Warmer J., Bast W. "MDA explained: the model driven architecture: practice and promise". Addison-Wesley Professional (2003)
- [2] W3C: Mobile Web Best Practices 1.0, <http://www.w3.org/TR/mobile-bp/>
- [3] OMG, "MOF 2 XMI Mapping", Version 2.4.1, <http://www.omg.org/spec/XMI/>
- [4] Vera, Pons, Gonzales, Rodríguez, Giulianelli, "Tool for Developing Mobile Web Application from UI Models –Based on CBHDM Methodology", SETECEC 2014
- [5] Vera, Pons, Gonzales, Giulianelli, Rodríguez, "Modeling Complex Mobile Web Applications from UI Components - Adding Different Roles and complex Database Design", CACIC 2013
- [6] Vera, Pons, Gonzales, Giulianelli, Rodríguez, "Metodología de Modelado de Aplicaciones Web Móviles Basada en Componentes de Interfaz de Usuario", JAIIO 2013