**CAPITULO 6**

**Conclusiones y trabajos futuros**

En este trabajo de fin de carrera se llevó a cabo una investigación acerca de las RIA, analizando sus características y las tecnologías utilizadas para su implementación. Se ha visto que dentro de las características más relevantes, se encuentra el uso del lado del cliente para mejorar la interactividad de las aplicaciones para con los usuarios y de esa forma, obtener aplicaciones Web similares a una aplicación de escritorio, con pocos retardos en las recargas de página. Dentro de las tecnologías RIA existentes, las basadas en librerías *Javascript* son las más populares y la más utilizada actualmente es *jQuery*. Con las extensiones a *jQuery*, *jQueryUI* y *JQuery validation plug-in* es posible explotar el lado del cliente agregando *widgets* interactivos como los *richAccordion*, *richTabs, richAutoSuggest, richToolTip, richDatePicker* y llevar a cabo validaciones locales de campos.

Hoy en día, la ingeniería de software basada en modelos, cumple un papel importante en el desarrollo del software. He allí que existen muchas metodologías Web con este enfoque y que dan cierta cobertura a características de las RIA. En base a una investigación del estado del arte de las principales metodologías Web basadas en modelos, se concluyó que ninguna de ellas logra dar cobertura a todas las características RIA. He allí la necesidad de extender a las metodologías Web existentes con nuevas características RIA o bien proponer nuevas metodologías que ofrezcan una mayor flexibilidad de extensión. En este trabajo se propuso a la aproximación MoWebA que está basada en los estándares propuestos por la OMG, como la alternativa apropiada, debido a su adecuada separación en capas y baja cohesión, que le brinda la posibilidad de llevar a cabo extensiones RIA a cualquiera de las capas sin afectar a las otras. En este trabajo se extendió con características RIA a la capa de Presentación de MoWebA que abarca a los metamodelos de Contenido y Estructura.

Precisamente con este trabajo de fin de carrera se alcanzó aportar lo siguiente:

* Un estudio detallado de las principales características y tecnologías de las RIA
* Una investigación del estado del arte de las principales metodologías Web basadas en MDD y MDA que ofrecen cobertura a las RIA.
* Un análisis de los elementos de interfaz enriquecidos (*widgets*) más utilizados en base a un análisis efectuado a portales Web de uso extendido como Facebook, Youtube, Gmail y Amazon.
* Una extensión al metamodelo de contenido de MoWebA agregando los siguiente:
  + Una reestructuración y clasificación de los elementos de interfaz con los que cuenta la aproximación MoWebA, separando a los distintos componentes de interfaz en elementos de entrada, salida y control.
  + El agregado del patrón *composite*, para definir una jerarquía entre los distintos elementos de interfaz simples y compuestos
  + *Widgets interactivos* comunes en las aplicaciones RIA, precisamente richAccordion, richTabs, richAutoSuggest, richDatePicker y richToolTip y la validación de diversos campos de entrada.
* Una extensión al metamodelo de estructura Layout, para permitir la definición de las posiciones de los elementos en pixeles o en porcentajes.
* Para la definición de la sintaxis concreta de la presentación, se agregaron los nuevos widgets y la validación de campos al perfil de Contenido, como así también la nueva definición de coordenadas al perfil de Layout de MoWebA.
* Un análisis de las principales herramientas de transformación de modelo a texto (M2T) basado en plantillas.
* Con la herramienta de transformación (M2T) *Acceleo* se implementaron las siguientes plantillas:
  + **La plantilla de presentación**, la cual genera código para cada uno de los elementos definidos en el perfil de contenido de MoWebA a partir de los PIM de entrada. Para los *widgets* se genera código para la plataforma destino *jQueryUI* y *jQuery validation plug-in*.
  + **La plantilla de estructura**, la cual genera código CSS con las posiciones establecidas en el PIM de entrada
* Una ilustración evaluativa en la cual se presentan los aportes realizados a la capa de presentación de MoWebA.

Finalmente, en la se presentan las características RIA y a las metodologías web contempladas en el estados del arte junto a la aproximación MoWebA luego de las extensiones RIA.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Características versus metodologías | | OOHDM-RIA | OOH4RIA | WebML - RIA | Patrones con UWE | Patrones OOWS | UsiXML | UWE-R | Espacios interactivos con UML | UWE + RUX | MoWebA |
| Almacenamiento en el lado del cliente | | - |  | si | - | - | - | - | - | - | - |
| Lógica de negocio en el lado del cliente | Operaciones complejas | - | - | si | - | - | - | - | - | - | - |
| Operaciones específicas del dominio | - | - | - | - | - | - | si | - | - | - |
| Validación local | si | si | si | - | - | - | - | - | si | si |
| Presentaciones enriquecidas | Manejo de eventos en el lado cliente | - | - | si | si | si | - | si | si | si | - |
| Widgets | si | si | - | si | si | si | si | si | si | si |
| Paradigma de página única | si | si | si | si | - | si | - | - | si | si |
| Contenido multimedia | - | si | - | - | - | si | si | - | si | si |
| Comunicación cliente servidor | Sincronización de datos | - | - | si | - | - | - | si | - | si | - |
| Obtención de actualizaciones parciales de página | si | si | si | si | si | - | si | si | si | - |
| Push y Pull | - | - | si | - | - | - | si | - | - | - |

Tabla 1 Tabla ilustrativa presentada en el capítulo 2 junto a la aproximación MoWebA luego de las extensiones RIA.

Como trabajos futuros se podrían llevar a cabo lo siguientes extensiones RIAs a MoWebA:

* Agregar más propiedades a los *widgets* que forman parte de la extensión al metamodelo de contenido (*richDatePicker*, *richAutoSuggest, richAccordion, richTabs, richToolTip* y las validaciones locales de los campos, que ofrecen las plataformas jQueryUI*[[1]](#footnote-1)* y jQuery Validation Plugin[[2]](#footnote-2).
* Agregar más *widgets* interactivos al metamodelo de contenido como por ejemplo el *dialog*, *menú,* *progressbar*, *selectmenú*, *slider* y *spinner* que también forman parte de jQueryUI.
* Agregar a la capa de navegación extensiones para ofrecer una interacción asíncrona entre los lados cliente y servidor para cubrir el refrescado parcial de las páginas.
* Separar en el modelo de dominio las entidades, que pueden ser alojadas en lado cliente y servidor para lograr persistencia de datos en el lado cliente.
* Validar la propuesta llevando a cabo transformaciones a otras plataformas destino.
* Efectuar la validación de la propuesta de extensión RIA a la capa de presentación de MoWebA, con un caso de estudio formal, detalladamente planeado, en la que participe una población más grande de personas. Esto sería interesante ya que se minimizarían en gran medida las amenazas a la validez existentes en la ilustración de caso de estudio actual.

PI6: ¿Qué tan independiente de la plataforma destino son los PIM presentados en el método B con respecto al método A?

1. **jQuery UI 1.11 API Documentation:** <http://api.jqueryui.com/> 2015 [↑](#footnote-ref-1)
2. **jQuery Validation Plugin:** <http://jqueryvalidation.org/> 2015 [↑](#footnote-ref-2)