

## Herramienta Scaffold para desarrollos Web

Zilman, Christian Julio Cesar

christianzilman@gmail.com

Cátedra: Programación Web

Profesor: Ing. Correa, German

Universidad Tecnológica Nacional - Facultad Regional de Tucumán

[www.frt.utn.edu.ar](http://www.frt.utn.edu.ar)

**Resumen.** Herramienta Scaffold para desarrollos Web es una aplicación que ayuda en la etapa de construcción frente a cualquier desarrollo web de usuarios que utilizan el framework MVC Codeigniter. Combina la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo la menor cantidad de código posible, con un mínimo de configuración, siendo eficaz para las entregas en tiempo. Emplea el concepto de la meta-programación, donde el programador escribe una especificación que describe cómo la base de datos de la aplicación va a ser utilizada. La herramienta usa esta especificación para generar el código que la aplicación necesita para gestionar los datos de las tablas o vistas relacionadas.

La herramienta contempla el procedimiento de gestión: encargado de gestionar los datos de tablas relacionadas a través de las operaciones: agregar, modificar, eliminar, filtro, reporte y validaciones.

Un procedimiento de consulta permite visualizar los datos de las vistas de la base de datos a través de formularios y reportes. Indicando el procedimiento y cuáles son las tablas o vistas que desea crear, la herramienta efectúa la compilación y procede a realizar las construcciones de las clases modelos, las clases controladores, y las clases vistas. A la vez incorpora funcionalidades tales como: la administración de plantilla (para cambiar el diseño web), navegabilidad (cuales son las funcionalidades que puedo usar), administración de bases de datos (para cambiar de gestor de bases, base de dato y el driver con el cual debe trabajar). Con el empleo de esta herramienta se logra una rápida evolución del sistema y flexibilidad para adaptarse a cambios.

**Palabras Claves.** Scaffold, herramienta, desarrollos web, flexibilidad.

## 1 Introducción

Internet ha irrumpido en nuestras vidas y es ya una parte de ella. Internet es en sí mismo es Innovación, es un cambio de paradigmas, una revolución que acaba de empezar.

Así hoy las empresas privadas y las administraciones públicas han cambiado de forma radical, en muchos aspectos, la forma de administrar sus datos, han revolucionado nuestros conceptos, nuestra forma de trabajar, de comunicarnos, de hacer negocios, de ofrecer servicios. Internet es sinónimo de Innovación y es necesario promover la, a través de desarrollos webs que hagan uso del mismo.

Scaffold es un método meta-programación de construcción de la base de datos backend de aplicaciones de software. Es una técnica apoyada por algunos modelos de controlador de vista- marcos, en los que el programador puede escribir una especificación que describe cómo la base de datos de aplicación puede ser utilizada. El compilador utiliza esta especificación para generar el código que la aplicación puede utilizar para crear, leer, actualizar y eliminar las entradas de la base de datos, así también el tratamiento efectivo de la plantilla de diseño web como un "Scaffold" sobre la cual construir una aplicación potente y flexible. El Scaffold es una evolución de los generadores de código de base de datos de los entornos de desarrollo anteriores, como caso de ejemplo el generador Oracle, y muchos otros 4GL cliente-servidor de productos de desarrollos de software.

La Herramienta Scaffold creada tomó como base la estructura de la librería Scaffold[2] de Codeigniter, para crear su propia herramienta. La misma se basa en el paradigma de programación generativa, con especificar la estructura de datos genera el código de sus modelos, vistas, controladores adaptándose a la plantilla web.

Este trabajo tuvo iniciativas en el año 2010 en la cátedra programación web. En el año 2011 se continuó con mejoras en su funcionamiento, y fue donde se pudo comenzar la etapa de prueba con un caso de estudio real. De manera que la herramienta actualmente cuenta con las siguientes funciones enumeradas:

- Wizard: asistente para guiar usuarios en el uso de la herramienta.
- Base de datos: permite trabajar con los gestores de bases de datos como: MySQL [8], PostgreSQL [7], ODBC, SQL Server [9], SQLite [10].
- Validaciones: validar que los datos no sean nulos, el tipo de dato ingresado sea el correcto, limpieza de inyección de sql, limpieza de cross-site script.
- Ajax: utiliza esta técnica para crear los desarrollos webs interactivos, sin necesidad de tener muchas recargas, aumentando velocidad y usabilidad.
- Procedimiento Gestión: crea la estructura de forma dinámica para poder realizar la manipulación de los datos a través de las operaciones filtro, eliminar, agregar, modificar, reporte.

- Procedimiento Consulta: crea la estructura de forma dinámica a través de las vistas de la base de datos para poder visualizar los datos y permitir dar reportes más específicos.
- Estructura de páginas estáticas: crea contenidos estáticos que no necesitan conexión a bases de datos.
- Administración de plantilla: poder cambiar el diseño del sitio

## 2 Metodología de desarrollo

La metodología utilizada es la de Craig Larman [13], la cual se basa en UML. La justificación de la elección de esta metodología se funda en el hecho que es un método bastante flexible para introducir o eliminar elementos de la notación UML. Además, el autor propone un Ciclo de vida iterativo y creciente, el cual consideramos apropiado para el desarrollo de la herramienta. A continuación se detallan cuáles son las instancias de la metodología que se siguió:

- Análisis
  - Especificación de Requisitos del Software (ERS- Std IEEE830).
- Diseño
  - Elaboración de interfaz gráfica de usuario
  - Diagramas de clases
  - Arquitectura de la herramienta
- Construcción
- Pruebas / Testing

## 3 Especificación de Requisitos del Software

Se desarrolló una ERS que tiene una descripción completa del comportamiento de la herramienta. A continuación se describe algunos requerimientos y restricciones:

- El sistema deberá guiar al usuario a través de un asistente Wizard para el uso la herramienta.
- El sistema deberá generar la estructura de gestión de datos de una entidad con sus relaciones foráneas, con sus tipos de datos, validaciones y operaciones.
- El sistema deberá generar la estructura de gestión de datos de una entidad con sus relaciones foráneas y sus entidades débiles.
- El sistema deberá cambiar el diseño web definido de forma dinámica.
- El sistema podrá configurar la navegabilidad para usar las funciones.
- El sistema brindará flexibilidad ante los cambios continuos de los requerimientos.
- El sistema generará las interfaces de consultas de las vistas que tienen la base de datos
- El sistema permitirá trabajar con los motores de base de datos: Postgresql, mysql, Sqlite, SqlServer.

- La programación de la herramienta deberá desarrollarse en un Framework MVC PHP
- La configuración de la herramienta mínima de cada dispositivo sobre el cual el desarrollo web debe funcionar es Google Chrome, Mozilla Firefox 1.0 o Superior, Internet Explorer.

#### 4 Modelo Lógico de clases

Para generar la estructura de gestiones y consultas se estructuró los siguientes diagramas de clases.

Para el requerimiento de una sola entidad con sus relaciones foráneas se utilizó el siguiente diagrama:

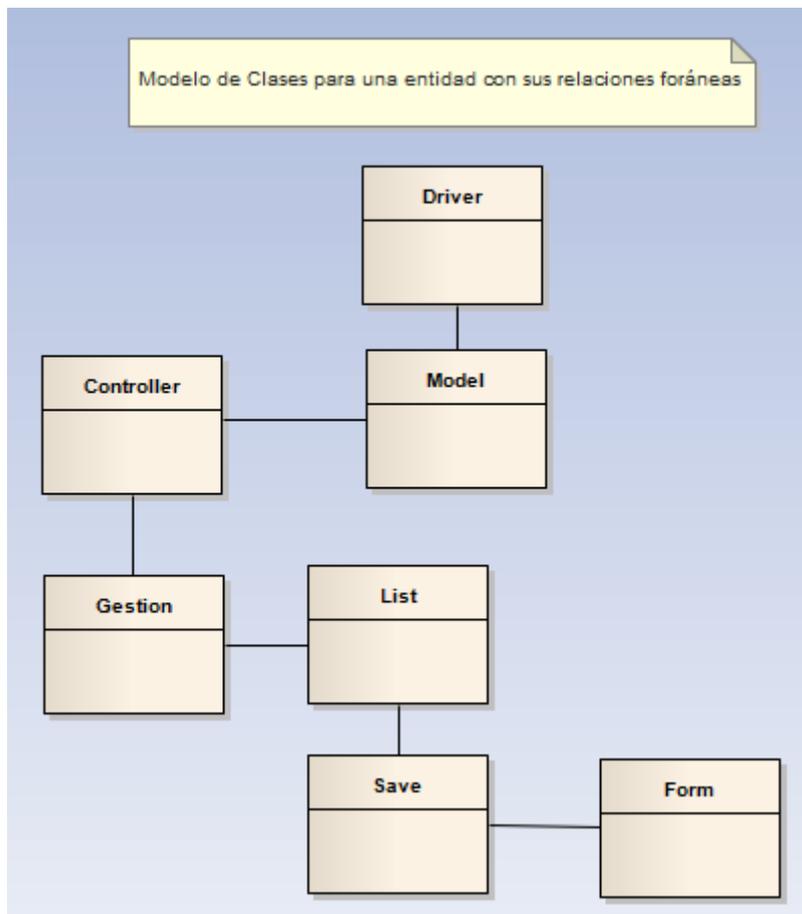


Figura 1: Modelo de clases 1

Para el requerimiento de una entidad con sus relaciones foráneas y entidades débiles se utilizó el siguiente diagrama:

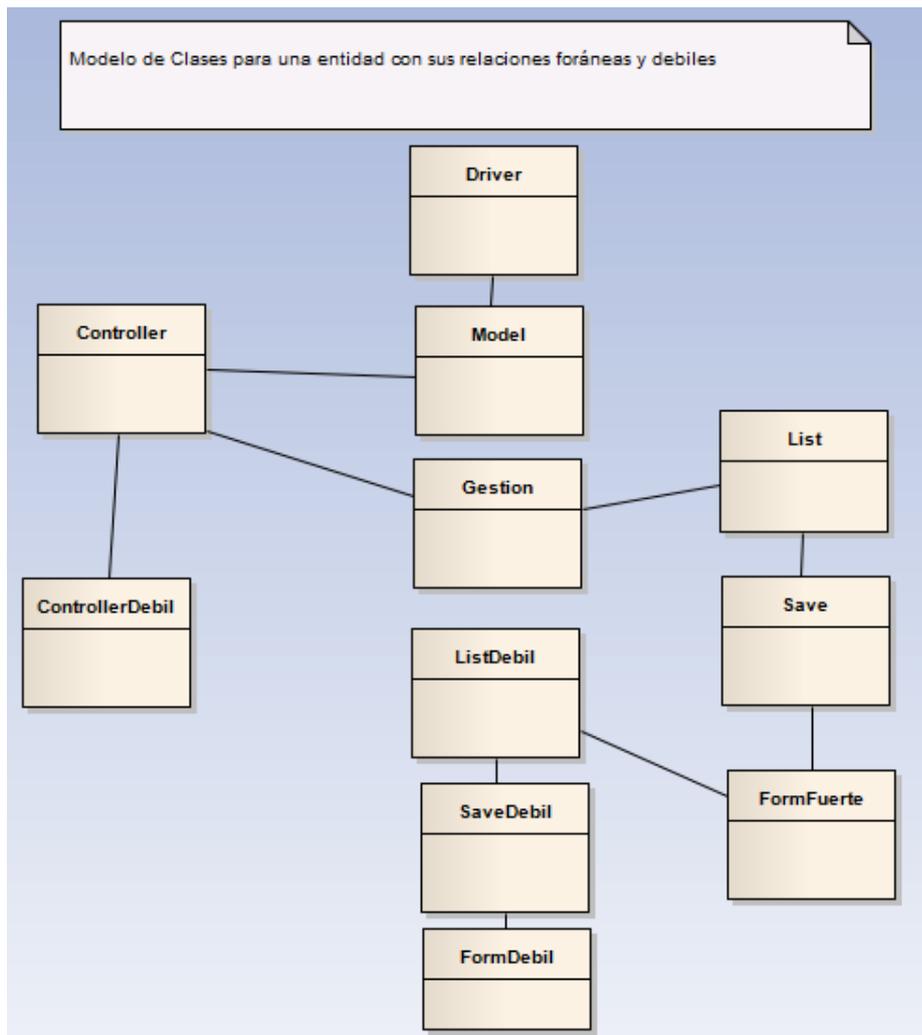


Figura 2: Modelo de clases 2.

Controller: se encarga de generar la clase controlador con toda la lógica de negocio para el caso de gestiones y consultas.

ControllerDebil: clase encargada de procesar la lógica de negocio de las entidades débiles que la clase controller utiliza.

Driver: es la clase que permite conectarse a los distintos motores de base de datos.

Model: encargado de definir los atributos y los métodos de persistencia de cada entidad a trabajar.

Vista: las clases encargadas de generar las vista que manipulara el usuario del sitios son gestión, list, save, form, listDebil, saveDebil, formDebil.

## 5 Arquitectura MVC

La arquitectura que se describe a continuación es la que genera la herramienta para que el desarrollo web sea funcional. Modelo - Vista - Controlador es un patrón de desarrollo, un estilo de arquitectura de software que separa el código fuente de las aplicaciones en tres grupos:

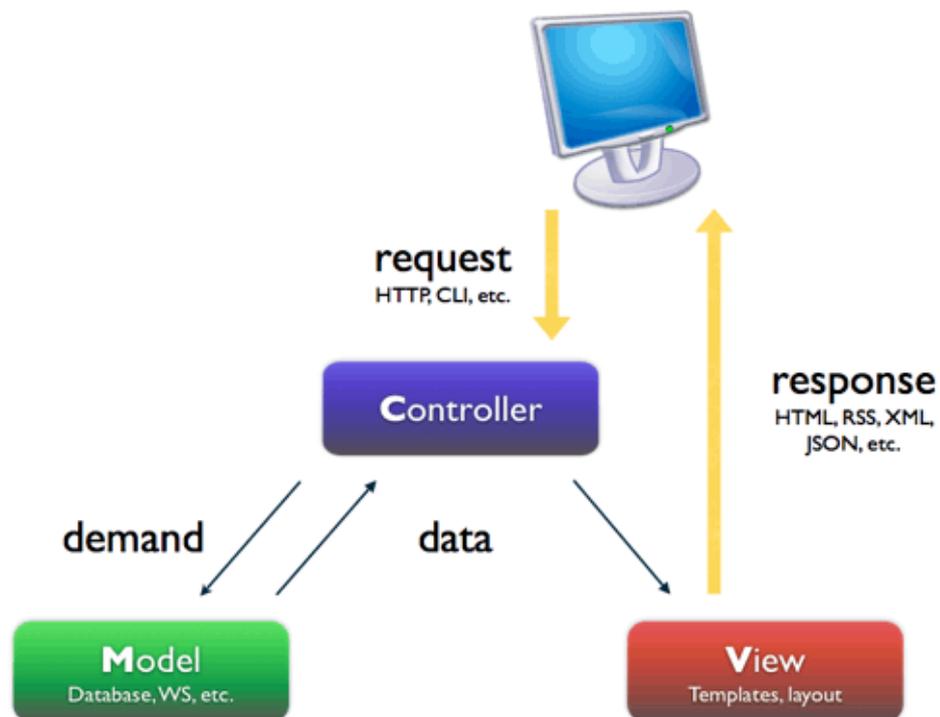


Figura 3: Arquitectura MVC

Modelo: todo el código que tiene que ver con el acceso a base de datos. En el modelo mantendremos encapsulada la complejidad de nuestra base de datos y simplemente crearemos funciones para recibir, insertar, actualizar o borrar información de nuestras tablas. Al mantenerse todas las llamadas a la base de datos en un mismo código, desde otras partes del programa podremos invocar las funciones que necesitemos del modelo y éste se encargará de procesarlas. En el modelo nos podrán preocupar cosas como el tipo de base de datos con la que trabajamos, o las tablas y sus relaciones, pero desde las otras partes del programa simplemente llamaremos a las funciones del modelo sin importarnos qué tiene que hacer éste para conseguir realizar las acciones invocadas.

Vista: la vista codifica y mantiene la presentación final de nuestra aplicación de cara al usuario. Es decir, en la vista colocaremos todo el código HTML, CSS, Javascript, etc. que se tiene que generar para producir la página tal cual queremos que la vea el usuario. En la práctica la vista no sólo sirve para producir páginas web, sino también cualquier otra salida que queramos enviar al usuario, en formatos o lenguajes distintos, como pueden ser feeds RSS, archivos JSON, xml, etc.

Controlador: el controlador podríamos decir que es la parte más importante, porque hace de enlace entre el modelo, la vista y cualquier otro recurso que se tenga que procesar en el servidor para generar la página web. En resumen, en el controlador guardamos la lógica de nuestras páginas y realizamos todas las acciones que sean necesarias para generarlas, ayudados del modelo o la vista.

## 6 Descripción del Funcionamiento de la Aplicación Scaffold

Asistente: al ingresar accedemos al menú el cual contiene 5 funciones donde al elegir una de ellas seremos direccionados a una nueva página web con el procedimiento correspondiente.

A continuación describiremos cada una de las opciones de nuestro Menú:

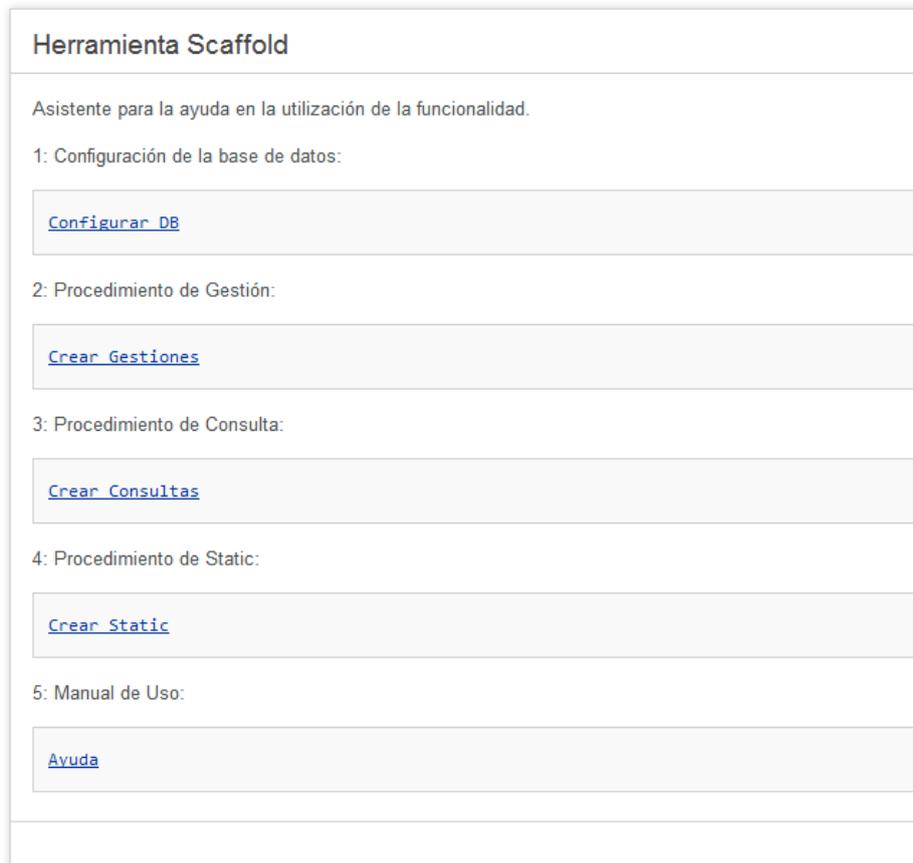


Figura 4: Menú

A- Configuración de la Base de datos: En esta ventana elegiremos el gestor de base de datos con el que va a trabajar la herramienta, y así mismo elegiremos la base de datos.



Figura 5: Configuración de la base de datos

B- Procedimiento Gestión: en esta opción el sistema reconoce automáticamente las entidades de la base de datos que agregamos y configuramos para poder administrar las relaciones (foráneas y débiles) y el menú de navegación. Al apretar el botón Generar nos crea el código con la estructura y funcionalidades necesarias para el sistema en construcción con las configuraciones realizadas.

Herramienta Scaffold

---

**Procedimiento Gestión**

realacion multiple
  simple  
 Eliminación Logica
  Eliminación Física

Entidad: 
 Débiles:
  Seleccione
  alumno
  curso
  distincion
  fecha
  funcionalidadtipo
  item
  localidad
  menu
  pais
  provincia
  prueba
  pruebadebil
  submenu
  tipo
  usuario

Menu de Navegación:

[Ir al Menú](#)

Figura 6: Procedimiento gestión

C- Procedimiento Consulta: En este caso debemos seleccionar las vistas que están cargadas en la DB que queramos que nos aparezca en el menú de navegación así la Herramienta Scaffold nos genere el código para el sistema en elaboración.

Herramienta Scaffold

Procedimiento Consulta

Vistas:

Menu de Navegación:

[Ir al Menú](#)

Figura 7: Procedimiento consulta

D- Procedimiento static: En este apartado nos genera las páginas que comúnmente no suelen tener cambios y el cliente/visitante de la página no puede manipular el contenido como por ejemplo "Historia de la Empresa" las cuales son sencillas y mayormente solo llevan unos cuantos párrafos e imágenes.

Herramienta Scaffold

Procedimiento Static

Título:

Contenido:

Menu de Navegación:

[Ir al Menú](#)

Figura 8: Procedimiento static

E- Manual de Uso: Nos muestra un documento de comunicación técnica que busca brindar asistencia a los sujetos que usan la Herramienta Scaffold en el cual apelamos a un lenguaje ameno y simple para llegar a la mayor cantidad posible de receptores.

Administración de la plantilla: Nos referimos al uso de plantillas de diseño para sistema en desarrollo donde podremos administrarlas de modo tal que nos aparezca dicho diseño en todo el código generado en los otros ítems (fig. 6, 7, 8).

## 7 Herramientas, lenguajes de programación y librerías utilizadas:

Lenguaje de Programación propuesto:

- Php
- Framework MVC Codeigniter[3] para trabajar con Php

Librerías utilizadas

- Librería Scaffold: se utilizó como base esta librería para la construcción de mi marco de trabajo.
- JQuery[3]: datatable, development-bundle, jquery-validate.password.
- Tcpdf[11]: para generar reportes.

Base de datos:

- MySQL
- PostgreSQL

Herramientas Utilizadas:

- XAMPP[5]
- Eclipse[6]

## 8 Caso de estudio : “Sistema de Control de Gestión Estratégico de Organizaciones Educativas (SCGEOE)”

EL SCGEOE es una herramienta web para ayudar al seguimiento y desarrollo educativo tanto de los alumnos, como a los profesores y demás roles que se cumplan en la institución aplicada. Como así también brinda una solución eficaz y eficiente para las gestiones comunes administrativas (asignatura, inscripciones, plan de estudio). Integra un Subsistema estratégico que permite generar informes de calidad para ayudar al directivo en la toma de decisión.

Este proyecto utilizó la herramienta Scaffold y contribuyó con un 50% para la finalización del proyecto final. El mismo fue desarrollado por los Egresado Universitarios Christian Julio Cesar Zilman, Maria Pia Luna. Fue aprobado por la facultad el día 2 de Marzo del año 2012.

Configuración de la Herramienta Scaffold:

[http://www.youtube.com/watch?v=PWmUw-Q\\_Mkw&feature=youtu.be](http://www.youtube.com/watch?v=PWmUw-Q_Mkw&feature=youtu.be)

Utilización de la herramienta en un caso de estudio:

[http://www.youtube.com/watch?v=WV\\_hPyFVYI0&feature=youtu.be](http://www.youtube.com/watch?v=WV_hPyFVYI0&feature=youtu.be)

## 9 Conclusiones

La utilización inicial de la herramienta fue exitosa alcanzando los objetivos planteados al comenzar el trabajo con el proyecto SCGEOE.

Se ha podido constatar que tiene un funcionamiento eficaz en el desarrollo web. Como beneficio disminuir el tiempo de desarrollo para permitir al desarrollador enfocarse en otros temas. Acortar los periodos de entregas de funcionalidades. Fácil adaptación a los cambios de requerimientos.

## 10 Agradecimientos

Agradecemos a las cátedras Programación Web impartida por el Ing. Germán Correa por habernos brindado la oportunidad de encarar un proyecto bajo su coordinación y apoyo.

## 11 Referencias

- [1] Codeigniter: <http://codeigniter.com/>
- [2] Scaffold: <http://code.google.com/p/ci-scaffold/>
- [3] JQuery: <http://jquery.com/>
- [4] Php: <http://www.php.net/>
- [5] XAMPP: <http://www.apachefriends.org/>
- [6] Eclipse: <http://www.eclipse.org/>
- [7] PostgreSQL: <http://www.postgresql.org/>
- [8] MySQL: <http://www.mysql.com/>
- [9] SQL Server: <http://www.microsoft.com/sqlserver/en/us/default.aspx>
- [10] SQLite: <http://www.sqlite.org/>
- [11] TCPDF: <http://www.tcpdf.org/>
- [12] SOMMERVILLE 2005: Sommerville Ian, Ingeniería del Software, Séptima Edición, Pearson Educación, Madrid, 2005.
- [13] LARMAN 2003: Larman Craig, UML y patrones Una introducción al análisis y diseño orientado a objetos y al proceso unificado, Segunda Edición, Prentice Hall, Pearson, Pearson Educación, Madrid, 2003.