



Facultad de Ingeniería
Carrera de Ingeniería Informática

Desarrollo del módulo **XAY** para Yii PHP Framework

**Trabajo de diploma para optar por el título de Ingeniería en
Informática**

Autor:

SantixaySichanh

Tutores:

Msc. Jorge Luis Mazaira Fernández

Ing. Gabriel Alejandro López López

Cienfuegos, Cuba

Junio, 2015

Declaración de autoría

Yo, SantixaySichanh, declaro que soy el único autor de este trabajo y autorizo al Departamento de Informática de la Facultad de Ingeniería en la Universidad de Cienfuegos para que hagan el uso que estimen pertinente con este trabajo de diploma.

Para que así conste firmo (firmamos) la presente a los _____ días del mes de _____ del _____.

SantixaySichanh

Nombre completo del autor

Nombre completo del primer tutor

Nombre completo del segundo tutor

Agradecimientos

- A mis padres Bandai y KamPhan por haberme apoyado y guiado durante toda mi vida para que me convirtiera en la persona que soy.
- A mi Hermana Chanvilay por haberme apoyado estos años.
- A toda mi familia por su apoyo y amor.
- A mis amigos de Laos y de extranjeros por haber compartido su amistad y su amor en estos años.
- A todos mis compañeros de aula que me acompañaron durante estos años.
- A todos mis profesores por los conocimientos y valores que inculcaron en mí.
- A mi tutor Gabriel que me acompañó durante todo este período, que tuvo la paciencia y dedicación necesaria para que este trabajo fuera posible.
- A Lachy por su cariño y gran apoyo en todo el tiempo que yo necesito.
- A mi oponente Alex por su apoyo y cariño.
- A mi tutor Jorge Luis por su apoyo.
- A todas las personas que contribuyeron al desarrollo de este trabajo.

Dedicatoria

A mi papá Bandai.

A mi mamá Kamphan.

A mi hermana Chanvilay.

Resumen

El presente trabajo aborda el tema de los generadores de código aplicados al desarrollo de aplicaciones web con el objetivo de agilizar la construcción de este tipo de productos. Tomando como base a Yii PHP Framework se hace un estudio de la herramienta Gii CRUD Generator y las extensiones GiiX, Awegen y YCM. A partir del análisis de las deficiencias encontradas en estas herramientas se decide implementar un módulo para la gestión de entidades de negocio en aplicaciones desarrolladas con Yii PHP Framework incorporando los aspectos positivos tanto de YCM como de GiiX. Además se implementan nuevas funcionalidades orientadas a los reportes y se realiza el cálculo de la factibilidad económica del desarrollo de la solución propuesta.

Palabras claves: Yii PHP Framework, Gii GRUD Generator, YCM, Modulo, generador de código.

Abstract

The present work makes a little emphasis on the subject about code generators applied to the development web applications with the objective of accelerating the construction of these types of products. Taking the Yii PHP Framework as a base, a study of the Gii CRUD Generator tool is made and so as other extensions like GiiX, Awegen and YCM. From the analysis of the deficiencies found in theses tool, a decision can be made to use a module for the management of the business entities using applications developed. Within the Yii PHP Framework, In-cooperating positive aspects both of the YCM and the GiiX. Furthermore the use of new report orientated functionalities is made, and a calculation of the economic feasibility of the development of the proposed solution is made.

Key words:Yii PHP Framework, Gii GRUD Generator, YCM, module, code generator.

Índice

Introducción.....	1
Capítulo 1: Generadores automáticos de código. Estudio de Yii PHP Framework	7
1.1. Introducción.....	7
1.2. Conceptos básicos del dominio.....	7
1.2.1. Metaprogramación.....	7
1.2.2. Generadores de código	8
1.2.3. Reflexión.....	9
1.2.4. Reflexión en PHP	9
1.3. Descripción del objeto de estudio	10
1.3.1. Flujo actual de los procesos y análisis crítico de la ejecución de estos.....	10
1.3.2. Sistemas automatizados existentes vinculados al campo de acción a nivel nacional e internacional	13
1.3.3. Análisis crítico de los sistemas existentes	14
1.4. Arquitectura de software	14
1.4.1. Modelo Vista Controlador (MVC).....	15
1.4.2. MVC en Yii.....	15
1.5. Metodologías de desarrollo de Software	16
1.5.1. Rational Unified Process (RUP)	17
1.6. Lenguajes de modelado/programación utilizados	18
1.6.1. Lenguaje de Modelación Unificado (UML).....	18
1.6.2. PHP	18
1.6.3. HTML.....	18
1.6.4. JavaScript.....	19

1.6.5. CSS	19
1.7. Herramientas utilizadas para el desarrollo del sistema	20
1.7.1. Visual Paradigm	20
1.7.2. Umbrello	20
1.7.3. PhpMyAdmin	20
1.7.4. NetBeans.....	21
1.7.5. PHP Unit.....	21
1.7.6. XAMPP	21
1.7.7. Yii PHP Framework	21
1.7.8. Yii Booster	22
1.8. Conclusiones del capítulo	22
Capítulo 2: Descripción y construcción de la solución propuesta.....	23
2.1. Introducción.....	23
2.2. Definición de las entidades y los conceptos principales.....	23
2.2.1. Aplicación	23
2.2.2. Modulo.....	23
2.2.3. Componente	24
2.2.4. Formulario	24
2.2.5. Gii	24
2.2.6. Plantilla de Gii.....	25
2.3. Reglas a considerar	25
2.4. Modelo del dominio	26
2.5. Requisitos	27
2.5.1. Actores del sistema a automatizar.....	27
2.5.2. Los requisitos funcionales	27

2.5.3. Los requisitos no funcionales	28
2.6. Diagrama de Casos de Uso del Sistema	29
2.7. Descripción de los casos de uso.....	29
2.7.1. Diagrama de clases del diseño.....	31
2.7.2. Principios de diseño	34
2.8.3. Diagrama de despliegue.....	36
2.9. Conclusiones del capítulo	36
Capítulo 3: Estudio de factibilidad	37
3.1. Introducción.....	37
3.2. Factibilidad.....	37
3.2.1. Cálculos de Puntos de Casos de Uso sin ajustar	37
3.2.2. Factor de peso de los actores sin ajustar (UAW)	38
3.2.3. Factor de peso de los casos de uso sin ajustar (UUCW)	39
3.2.4. Cálculo de puntos de casos de uso ajustados.....	40
3.2.5. Factor de complejidad técnica (TCF).....	40
3.2.6. Factor de ambiente (EF).....	41
3.2.7. Cálculo de costos	43
3.3. Beneficios tangibles e intangibles	43
3.4. Análisis de costos y beneficios.....	44
3.5. Conclusiones del capítulo	44
Conclusiones.....	45
Recomendaciones:.....	46
Referencias Bibliográficas.....	47
Bibliografía	50
Anexos	53

Anexo: A Descripción de los casos de uso.....	53
Anexo A1 Gestionar entidad de cada tipo de dato del sistema.....	53
Anexo A2 Gestionar entidades relacionadas	55
Anexo A3 Imprimir un listado de modelos.....	56
Anexo A4 Imprimir los detalles de una instancia de modelo	57
Anexo A5 Listar las entidades gestionables.....	58
Anexo A6 Exportar a PDF los detalle de una instancia de modelo	59
Anexo A7 Exportar a PDF un listado de instancias de modelo	60
Anexo A8 Exportar a Excel un listado de instancias de modelo	61
Anexo A9 Exportar a CSV un listado de instancias de modelo	62
Anexo A10 Consultar Google Analytics	63
Anexo A11 Autenticar	64
Anexo A12 Cerrar sesión	65
Anexo: B Prototipos de Interfaz	66
B1 Prototipo interfaz Gestionar entidad de cada tipo de dato del sistema.....	66
B2 Prototipo interfaz Gestionar entidades relacionadas	68
B3 Prototipo interfaz Imprimir un listado de modelos.....	70
B4 Prototipo interfaz Imprimir los detalles de una instancia de modelo	71
B5 Prototipo interfaz Listar las entidades gestionables.....	72
B6 Prototipo interfaz Exportar a pdf los detalle de una instancia de modelo	73
B7 Prototipo interfaz Exportar a pdf un listado de instancias de modelo	74
B8 Prototipo interfaz Exportar a Excel un listado de instancias de modelo.....	75
B9 Prototipo interfaz Exportar a csv un listado de instancias de modelo	76
B10 Prototipo interfaz Consultar Google Analytics.....	77
B11 Prototipo interfaz Autenticar.....	78

B12 Prototipo interfaz Cerrar sesión	79
--	----

Índice de tablas

Tabla 1 Descripción de los Actores del Sistema.	27
Tabla 2 Descripción de los casos de uso del Sistema.	30
Tabla 3 Factor de peso de los actores del sistema.	38
Tabla 4 Clasificación de los actores atendiendo al factor de peso.	38
Tabla 5 Criterios del factor de peso de los casos de uso sin ajustar.	39
Tabla 6 Clasificación de los casos de uso del sistema.	39
Tabla 7 Factores de complejidad del sistema.	41
Tabla 8 Habilidades del grupo de desarrollo.	42
Tabla 9 Criterios de distribución de esfuerzo.	43

Índice de figuras

Figura 1 El Generador de Modelo.	11
Figura 2 El Generador de CRUD.	12
Figura 3 La página de Gestionar Usuario.....	12
Figura 4 Crear la nueva página de usuario.	13
Figura 5 Modelo Vista Controlador.....	15
Figura 6 La estructura estática de aplicación de Yii.	16
Figura 7 El Modelo del Dominio.	26
Figura 8 Diagrama de Casos de Uso del Sistema.	29
Figura 9 Diagrama de clases del diseño	31
Figura 10 Diagrama de clases de XayModule.....	32
Figura 11 Diagrama de clases de Modelo.....	32
Figura 12 Diagrama de clases de Componente	33
Figura 13 Diagrama de clases de Controlador.....	34
Figura 14 Diagrama de despliegue	36

Introducción

En los nuevos modelos de negocio la gestión de la información, la documentación y el conocimiento se perfilan como un componente estratégico de primera magnitud. La orientación tradicional del carácter táctico de los proyectos de gestión de la información o de la documentación cambia cuando se considera un verdadero componente de la estrategia empresarial. Este cambio no sólo afecta a las empresas que se crean para actuar específicamente en el ámbito de los negocios electrónicos.

La profunda revolución tecnológica que estamos viviendo, ha sido el motor de este cambio. Por esta razón muchas veces los empresarios y directivos simplifican su actuación frente a la nueva realidad, centrándola en la compra e instalación de herramientas informáticas de última generación que deberían dar resultados a corto plazo. Esto ha llevado a fracasos, a directivos desencantados e incluso tecnologías con mala fama que van cambiando su denominación en inteligentes campañas de marketing. Aun así, existen tendencias, que van centrando el compendio de "buenas prácticas" en la gestión de la información en las organizaciones y el cambio de conceptos tradicionales. Es importante que estos cambios se incorporen tanto a las habilidades de los gestores de información como al corpus teórico de nuestra disciplina.

La información para la gestión y la gestión de la información son dos conceptos diferentes; la información para la gestión es un tipo de información (los datos); la gestión de la información es un tipo de gestión (el sistema) [1].

La gestión de la información es el proceso de analizar y utilizar la información que se ha recabado y registrado para permitir a los administradores (de todos los niveles) tomar decisiones documentadas. La información para la gestión es la información necesaria para tomar decisiones de gestión.

La información para la gestión es importante para:

- Tomar las decisiones necesarias para mejorar la gestión de prestaciones y servicios

- Poner en práctica la planificación, implementación, supervisión y evaluación participativas.

Para poder utilizar la información para tomar decisiones de gestión, debe gestionarse la información (recabar, registrar y analizar). Aunque este proceso de recabar y guardar la información y la información necesaria para tomar decisiones bien documentadas son diferentes, se refuerzan entre sí y no pueden separarse en las operaciones cotidianas.

Por lo tanto, la gestión de la información implica:

- Determinar la información que se precisa
- Recoger y analizar la información
- Registrarla y recuperarla cuando sea necesaria
- Utilizarla
- Divulgarla[1].

Vivimos en tiempos en que se presta una atención extraordinaria a una serie de dispositivos que ayudan al intercambio y la gestión de la información. Cada día más habitantes del planeta parecieran necesitar de las nuevas “tecnologías de la información y la comunicación” (TIC). El acceso a estos medios esencial, ya no sólo para permitir la interacción a distancia entre individuos, sino también para facilitar el comercio, la ciencia, el entretenimiento, la educación, y un sin número de actividades relacionadas con la vida moderna del siglo XXI.

Teniendo en cuenta el rol que desempeñan las TIC en la actualidad, resulta estratégico hacer operativo este concepto desde una perspectiva académica, en estrecha relación con la gestión de la información.

Las TIC se desarrollan a partir de los avances científicos producidos en los ámbitos de la informática y las telecomunicaciones. Las TIC son el conjunto de tecnologías que permiten el acceso, producción, tratamiento y comunicación de información presentada en diferentes códigos (texto, imagen, sonido,...).

El elemento más representativo de las nuevas tecnologías es sin duda el ordenador y más específicamente, Internet. Como indican diferentes autores, Internet supone un

salto cualitativo de gran magnitud, cambiando y redefiniendo los modos de conocer y relacionarse del hombre.

Podemos diferenciar los programas y recursos que podemos utilizar con el ordenador en dos grandes categorías: recursos informáticos, que nos permiten realizar el procesamiento y tratamiento de la información y, los recursos telemáticos que nos ofrece Internet, orientados a la comunicación y el acceso a la información.

Los lenguajes de programación, son idiomas artificiales diseñados para expresar instrucciones que pueden ser llevadas a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana [2].

Los lenguajes de programación están formados por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila y se mantiene el código fuente de un programa informático se le llama programación.

La programación web, consiste en escribir programas que, dada una petición web realizada por un cliente (un navegador), procesen el pedido y generen y devuelvan un resultado [3].

Como en todo proceso de desarrollo de software, hay mucho de reutilización. En general, el proceso de recibir el pedido y enviar la respuesta por lo general no cambia, solamente cambia la lógica de negocios de cada pedido y el contenido de la respuesta generada.

La utilización de un framework en el desarrollo de una aplicación implica un cierto coste inicial de aprendizaje, aunque a largo plazo es probable que facilite tanto el desarrollo como el mantenimiento.

Un framework de aplicaciones web es un tipo de framework que permite el desarrollo de sitios web dinámicos, web services (servicios web) y aplicaciones web. El propósito de este tipo de framework es permitir a los desarrolladores construir aplicaciones web y

centrarse en los aspectos interesantes, aliviando la típica tarea repetitiva asociada con patrones comunes de desarrollo web.

El más conocido patrón de diseño de aplicaciones web es la arquitectura Modelo Vista Controlador de sus siglas en inglés Model View Controller (MVC). Un Modelo es un objeto que representa datos o incluso una actividad en forma de tabla en base de datos. Una Vista es algún tipo de visualización del estado del modelo, y un Controlador ofrece funciones (herramientas) para cambiar el estado del modelo [4].

Frameworks conocidos que se pueden mencionar por ejemplo son: CodeIgniter, CakePHP, Zend Framework y Yii PHP Framework.

Yii es un conjunto de código, librerías y componentes de software probados, optimizados para facilitar y agilizar el desarrollo de aplicaciones, basado en las mejores prácticas de desarrollo, que permitirán trabajar de forma uniforme, organizada y con la mayor calidad posible. Normalmente cuentan con documentación, una comunidad organizada que constantemente está trabajando por su constante desarrollo y mantenimiento [5]. Yii viene con un asistente para generar el código de nuestros proyectos de manera automática. Este asistente es una herramienta web llamada Gii la cual proporciona todas las funcionalidades básicas para la gestión de los datos en una aplicación web. GiiX y Awegensson básicamente nuevas plantillas para Gii que le aportan más funcionalidades y YCM es un módulo de Yii que funciona como gestor de contenido. Todas estas herramientas están orientadas a agilizar el desarrollo de las funcionalidades de gestión de la información y a partir del cual se genera la situación problemática de esta investigación.

El proceso de personalización del código generado por Gii CRUD Generator para cada uno de los modelos que se gestionan en un sistema se hace engorroso debido a la gran duplicación del código para idénticas funcionalidades. GiiX y Awegensson solucionan parcialmente este problema pues entienden las relaciones entre las tablas, pero mantiene la duplicidad y YCM evita la duplicidad pero no tiene un diseño correcto por lo que es complicada la modificación de sus funcionalidades. No tiene en cuenta las

relaciones entre las tablas y todo el código se encuentra ubicado en una única clase, lo cual es considerado una mala práctica de ingeniería de software.

El problema de investigación: ¿Cómo facilitar el desarrollo de aplicaciones basadas en Yii PHP Framework?

Como punto de inicio para dar solución a este problema se **defiende la idea** de que: El uso del módulo resultante le facilitará a los desarrolladores la implementación de las funcionalidades de gestión de entidades en cualquier aplicación basada en Yii PHP Framework.

El **objeto de estudio** son los generadores automáticos de código en los frameworks para desarrollo web y el **campo de acción** son las aplicaciones desarrolladas con Yii PHP Framework.

El **objetivo general** desarrollar un módulo de Yii PHP Framework que permita la gestión de las entidades de negocio utilizando un mecanismo común para todos los casos y formularios generados dinámicamente. Para el cumplimiento de este objetivo, el mismo se dividió en los siguientes **objetivos específicos**:

- Analizar el módulo YCM como punto de referencia para la solución
- Diseñar una solución que permita el procesamiento de los datos para la mayoría de los casos.
- Implementar la solución como un módulo de Yii PHP Framework
- Calcular la factibilidad económica del desarrollo de la solución utilizando el método de cálculo de puntos de casos de usos

Para el cumplimiento de estos objetivos se desarrollaron las siguientes **tareas**:

- Estudio del flujo de datos en las aplicaciones web desarrolladas sobre PHP y Yii PHP Framework.
- Análisis del módulo generador de código Gii y del módulo de gestión de contenido YCM.

- Diseño de las interfaces genéricas que permitirán la gestión de la información en los distintos tipos de negocio, así como del nuevo componente generador de formularios.
- Implementación del componente generador de formularios y las nuevas plantillas de código para ser usadas con Gii.
- Cálculo de la factibilidad económica utilizando el método de puntos de casos de uso.

Esta solución tendrá un **significativo aporte práctico** pues la comunidad de desarrolladores de Yii PHP Framework dispondrá de un nuevo módulo que les permitirá agilizar el desarrollo de aplicaciones web ya que no será necesario programar las funcionalidades de gestión de entidades del negocio.

La investigación está **estructurada en los siguientes capítulos**:

Capítulo 1 Generadores automáticos de código. Estudio de Yii PHP Framework:

En este capítulo se abordan los aspectos teóricos de generador de código y los conceptos asociados al dominio del problema. Se explican conceptos asociados de Yii PHP Framework y las diferentes tendencias y metodologías existentes en la actualidad para el desarrollo de aplicaciones informáticas.

Capítulo 2 Descripción y construcción de la solución propuesta: En este capítulo se presenta la descripción actual del dominio y se utiliza la metodología Rational Unified Process (RUP) para el desarrollo del sistema transitando por los flujos de trabajo: dominio, requerimientos, diseño e implementación de módulo Xay.

Capítulo 3 Estudio de factibilidad: En este capítulo se describe el análisis realizado en relación al cálculo de la factibilidad del proyecto, utilizando la estimación del esfuerzo basada en el Análisis de Puntos de Casos de Uso, teniendo en cuenta el análisis de los costos, beneficios y se exponen los resultados alcanzados.

Capítulo 1: Generadores automáticos de código.

Estudio de Yii PHP Framework

1.1. Introducción

En este capítulo se exponen aspectos teóricos de la investigación realizando un análisis sobre los principales conceptos como metaprogramación, reflexión y generador de código vinculados al dominio del problema y estudio de Yii PHP Framework vinculados al campo de acción. Concluye con el estudio de metodologías, tecnologías y herramientas que Yii PHP Framework utiliza para el desarrollo del sistema.

1.2. Conceptos básicos del dominio

1.2.1. Metaprogramación

La metaprogramación consiste en escribir programas que escriben o manipulan otros programas (o a sí mismos) como datos, o que hacen en tiempo de compilación parte del trabajo que, de otra forma, se haría en tiempo de ejecución. Esto permite al programador ahorrar tiempo en la producción de código.

Un ejemplo sencillo de un metaprograma sería este script de Bash:

```
#!/bin/bash
# metaprogram
echo '#!/bin/bash' >program
for ((I=1; I<=992; I++)); do
echo "echo $I" >>program
done
chmod +x program
```

Este script genera un nuevo programa que imprime por pantalla los números 1 a 992. Esto es sólo una muestra de cómo usar código para escribir más código, no la forma más eficiente de imprimir una lista de números

La herramienta de metaprogramación más común es el compilador, el cual permite al programador escribir un programa corto en un lenguaje de alto nivel para, posteriormente, escribir un programa equivalente en lenguaje ensamblador o lenguaje máquina. Esto, por lo general, significa un ahorro de tiempo si se compara con la posibilidad de escribir el programa en lenguaje máquina de forma directa.

A partir del concepto de la metaprogramación surgen los generadores de código para optimizar el trabajo de los desarrolladores y garantizar la consistencia del estilo de programación [6].

1.2.2. Generadores de código

En programación, la generación de código es una de las fases mediante el cual un compilador convierte un programa sintácticamente correcto en una serie de instrucciones a ser interpretadas por una máquina.

En términos generales, la generación de código: es usada para construir programas de manera automática evitando que los programadores tengan que escribir el código a mano. La generación de código puede realizarse en tiempo de ejecución, Tiempo de carga, o Tiempo de compilación. Los compiladores JIT son un ejemplo de generadores de código [7].

Scaffolding es un método de meta-programación para construir aplicaciones de software que soportan bases de datos. Esta es una nueva técnica soportada por algunos frameworks del tipo MVC, donde el programador debe escribir una especificación que escriba como debe ser usada la aplicación de bases de datos. El compilador luego usará esta para generar un código que pueda usar la aplicación para leer, crear, actualizar y borrar entradas de la base de datos (algo conocido como CRUD

o ABM), tratando de poner plantillas como un andamio Scaffold) en la cual construir una aplicación más potente.

Scaffolding es la evolución de códigos generadores de bases de datos desde ambientes más desarrollados, como es CASE Generator de Oracle.

Scaffolding se hizo popular gracias al framework "Ruby onRails", que ha sido adoptado a otros frameworks, incluyendo Django, Monorail, KumbiaPHPframework entre otros[8].

1.2.3. Reflexión

En informática, reflexión (o reflexión computacional) es la capacidad que tiene un programa para observar y opcionalmente modificar su estructura de alto nivel. En un sentido más amplio, la reflexión es una actividad computacional que razona sobre su propia computación.

Normalmente, la reflexión es dinámica o en tiempo de ejecución. Es común en lenguajes de programación de alto nivel ejecutándose sobre una máquina virtual, como Smalltalk o Java, y menos común en lenguajes como C.

Cuando el código fuente de un programa se compila, normalmente se pierde la información sobre la estructura del programa (cuáles son sus métodos y atributos) conforme se genera el código de bajo nivel (normalmente lenguaje ensamblador). Si un sistema permite reflexión, se preserva la estructura como metadatos en el código generado. Dependiendo de la implementación, el código con reflexión tiende a ser más lento que el que no lo tiene [9].

1.2.4. Reflexión en PHP

PHP 5 viene con una API de reflexión completa que añade la capacidad para realizar ingeniería inversa de clases, interfaces, funciones, métodos y extensiones. Además, la API de reflexión ofrece varias formas de obtener comentarios de documentación de funciones, clases y métodos.

Tenga en cuenta que ciertas partes de la API interna no contienen el código necesario para trabajar con la extensión de Reflexión. Por ejemplo, a una clase interna de PHP le

pueden faltar datos de reflexión de sus propiedades. Sin embargo, estos pocos casos son considerados como errores, por lo que deberían ser descubiertos y corregidos [10].

1.3. Descripción del objeto de estudio

1.3.1. Flujo actual de los procesos y análisis crítico de la ejecución de estos

Para implementar la operación CRUD se asume que tenemos una tabla en la base de datos que se llama tbl_user.

En vez de estar lidiando con escribir el código necesario para gestionar esta tabla podemos utilizar la poderosa herramienta Gii (un generador del código poderoso).

Para utilizar Gii, primero necesitamos editar el archivo WebRoot/testdrive/protected/config/main.php, el cual es conocido como el archivo de configuración de aplicación:

```
return array(

    .....

    'import'=>array(

        'application.models.*',

        'application.components.*',

    ),

    'modules'=>array(

        'gii'=>array(

            'class'=>'system.gii.GiiModule',

            'password'=>'pick up a password here',

        ),

    ),
```

);

Luego, visita el URL <http://hostname/testdrive/index.php?r=gii>. A continuación se debe insertar la contraseña que especificamos en la configuración.

Después de login, se debe click en la opción de ModelGenerator. Este va a brindar con la siguiente página de generación modelo:

Model Generator

This generator generates a model class for the specified database table.

Fields with * are required. Click on the highlighted fields to edit them.

Database Connection *
db

Table Prefix
[empty]

Table Name *
tbl_user

Model Class *
User

Base Class *
CActiveRecord

Model Path *
application.models

Build Relations
☒

Code Template *
default (C:_work\jetbrains\yii\framework\gii\generators\model\templates\default)

Preview

Figura 1 El Generador de Modelo.

En el campo de nombre de la tabla, entra en el tbl_user. En el campo de la Clase modelo, entre en el User. Entonces se pulsa en el botón preview. Esto se va a mostrar el nuevo archivo del código a ser generado. Ahora apriete el botón generar. Un nuevo archivo nombrado User.php se generará bajo el protected/models.

Después de crear el archivo de la clase modelo, se debe genera el código que implementar la operación CRUD sobre los datos del User que se muestra en la siguiente figura:

Crud Generator

This generator generates a controller and views that implement CRUD operations for the specified data model.

Fields with * are required. Click on **highlighted fields** to edit them.

Model Class *

Controller ID *

Base Controller Class *

Code Template *

Figura 2 El Generador de CRUD.

En el campo de la clase modelo, se entra el User. En el campo Controller_id, se entra el user. Se pulsa el en botón preview. Se termina la generación de código CRUD[11].

My Web Application

Home About Contact Logout (admin)

Home » Users » Manage

Manage Users

[Advanced Search](#)

Id

Username

Email

Operations

- List User
- Create User

Displaying 1-10 of 21 result(s).

Id	Username	Password	Email
1	test1	pass1	test1@example.com
2	test2	pass2	test2@example.com
3	test3	pass3	test3@example.com
4	test4	pass4	test4@example.com
5	test5	pass5	test5@example.com
6	test6	pass6	test6@example.com
7	test7	pass7	test7@example.com
8	test8	pass8	test8@example.com
9	test9	pass9	test9@example.com
10	test10	pass10	test10@example.com

Go to page: < Previous 1 2 3 Next >

Copyright © 2010 by My Company.
All Rights Reserved.
Powered by [Yii Framework](#).

Figura 3 La página de Gestionar Usuario.

My Web Application

Home About Contact Logout (admin)

Home > Users > Create

Create User

Fields with * are required.

Please fix the following input errors:

- Password cannot be blank.
- Email cannot be blank.

Username *

test

Password *

Password cannot be blank.

Email *

Email cannot be blank.

Create

Operations

- List User
- Manage User

Copyright © 2010 by My Company.
All Rights Reserved.
Powered by [Yii Framework](#).

Figura 4 Crear la nueva página de usuario.

1.3.2. Sistemas automatizados existentes vinculados al campo de acción a nivel nacional e internacional

A nivel internacional existen una serie de extensiones similares a la propuesta para el framework Yii que permiten generar los CRUD como son awegen, giix y ycm.

Extensión: Giix.

Giix es una extensión de Gii, es decir un generador de código para el framework Yii PHP. Dicha extensión está inspirada y basada en la colección de plantillas de Gii.

Extensión: Awegen.

El término de inglés Awegen es **awesomecodegenerator** o un generador de código asombroso para Yii Framework.

El Awegen extiende las bondades de giix y gtc.

Extensión YCM: Sistema de gestión de contenidos para aplicaciones basadas en Yii. A partir del listado de las tablas que debe gestionar proporciona interfaces homogéneas para gestionar los datos de las mismas. En cada clase modelo se configura cómo debe ser editado cada uno de los campos y las relaciones que tiene la clase con otras clases.

1.3.3. Análisis crítico de los sistemas existentes

Todos hacen un buen trabajo pero se quedan cortos a la hora de ayudar a generar los formularios pues hay que detallar manualmente el código necesario para cada campo. El que más se acerca es YCM.

YCM Funciona correctamente está mal documentado y la arquitectura del sistema no es correctamente haciéndose difícil la extensión de las funcionalidades del módulo.

La mayoría de las ideas para XayModule surgen de YCM pero a partir de un rediseño de su arquitectura separando las funcionalidades que realiza en componente y clases base para su posterior extensión y/o generalización.

1.4. Arquitectura de software

En los inicios de la informática, la programación se consideraba un arte y se desarrollaba como tal, debido a la dificultad que entrañaba para la mayoría de las personas, pero con el tiempo se han ido descubriendo y desarrollando formas y guías generales, en base a las cuales se puedan resolver los problemas. A estas, se les ha denominado Arquitectura de Software, porque, a semejanza de los planos de un edificio o construcción, estas indican la estructura, funcionamiento e interacción entre las partes del software.

Generalmente, no es necesario inventar una nueva arquitectura de software para cada sistema de información. Lo habitual es adoptar una arquitectura conocida en función de sus ventajas e inconvenientes para cada caso. Así, las arquitecturas más universales son: Monolítica, Cliente-servidor, Arquitectura de tres niveles, N capas, model-view-controller. En este sistema se utiliza el patrón Modelo Vista Controlador MVC[12].

1.4.1. Modelo Vista Controlador(MVC)

Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos:

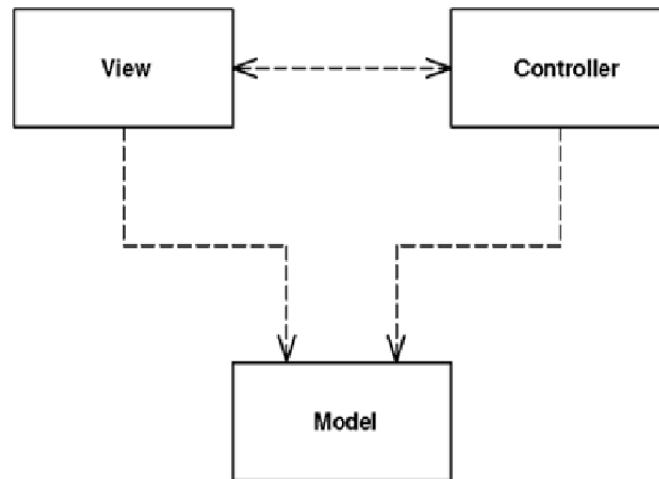


Figura 5Modelo Vista Controlador.

Modelo: Esta es la representación de la información con la cual el sistema opera. En resumen, el modelo se limita a lo relativo de la vista y su controlador facilitando las presentaciones visuales.

Vista: Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.

Controlador: Este responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y, probablemente, a la vista [12].

1.4.2. MVC en Yii

Yii implementa el MVC el cual es adoptado ampliamente en la programación Web. MVC tiene por objeto separar la lógica del negocio de las consideraciones de la interfaz de usuario para que los desarrolladores puedan modificar cada parte más fácilmente sin afectar a la otra. En MVC el modelo representa la información (los datos) y las reglas del negocio; la vista contiene elementos de la interfaz de usuario como textos, formularios de entrada; y el controlador administra la comunicación entre la vista y el modelo.

Más allá del MVC, Yii también introduce un front-controller llamado aplicación el cual representa el contexto de ejecución del procesamiento del pedido. La aplicación resuelve el pedido del usuario y la dispara al controlador apropiado para tratamiento futuro[11].

El diagrama siguiente muestra la estructura estática de una aplicación de Yii:

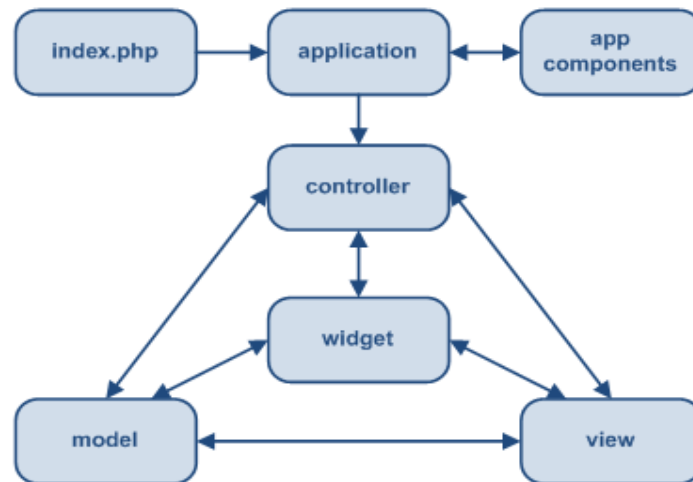


Figura 6 La estructura estática de aplicación de Yii.

1.5. Metodologías de desarrollo de Software

Las metodologías tradicionales de desarrollo de software se originaron en la década de 1960 para desarrollar a gran escala funcional de sistemas de negocio en una época de grandes conglomerados empresariales. La idea principal era continuar el desarrollo de los sistemas de información en una deliberada, estructurada y metódica, reiterando cada una de las etapas del ciclo de vida.

Metodología de desarrollo de software es un conjunto de herramientas, técnicas y procesos que brindan soporte y facilitan el logro u obtención de una meta que hacer, a lo largo de todo el ciclo de vida del software, para construir un producto cumpliendo los requisitos del cliente, dentro del presupuesto y a tiempo [13].

Existen dos tipos principales de metodologías de desarrollo de software: las metodologías tradicionales o pesadas y las metodologías ágiles o livianas. Cada

unatiene prioridades, requerimientos, y tecnologías diferentes. Sin embargo, en todos los proyectos, se debe minimizar el riesgo, garantizar la predictibilidad de los resultados y entregar un software de calidad a tiempo. RationalUnifiedProcess, o RUP, es una plataforma flexible de procesos de desarrollo de software que brinda guías consistentes y personalizadas de procesos para el equipo de proyecto [13].

1.5.1. RationalUnifiedProcess (RUP)

RUP describe cómo utilizar reglas de negocio y procedimientos comerciales en el desarrollo de software para equipos de desarrollo de software, conocidos como “mejores prácticas”. Captura varias de las mejores prácticas en el desarrollo moderno de software en una forma que es aplicable para un amplio rango de proyectos y organizaciones. Es una guía de cómo utilizar de manera efectiva UML. Provee a cada miembro del equipo acceso a una base de conocimiento con guías, plantillas y herramientas para las actividades de desarrollo. Crea y mantiene modelos, en lugar de enfocarse en la producción de papeles de documentación.

Como una plataforma de procesos que abarca las prácticas de la industria, RUP permite seleccionar el conjunto de componentes de proceso que se ajustan a las necesidades del proyecto. Se pueden alcanzar resultados predecibles unificando el equipo con procesos comunes que mejore la comunicación y creen un entendimiento común para realizar las tareas, responsabilidades y artefactos.

Una de las mejores prácticas centrales de RUP es la noción de desarrollar iterativamente. RationalUnifiedProcess organiza los proyectos en términos de disciplinas y fases, consistiendo cada una en una o más iteraciones. Con esta aproximación iterativa, el énfasis de cada artefacto variará a través del ciclo de vida. La aproximación iterativa ayuda a mitigar los riesgos en forma temprana y continua, con un progreso demostrable y frecuente ejecutable. Además provee un entorno de proceso de desarrollo configurable basado en estándares; permite tener definidos los objetivos y accesible el proceso de desarrollo que se sigue y que este sea configurado a las necesidades de la organización y del proyecto [13].

1.6. Lenguajes de modelado/programación utilizados

1.6.1. Lenguaje de Modelación Unificado (UML)

UML (Unified Modeling Language) es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar de facto de la industria, debido a que ha sido concebido por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh. Estos autores fueron contratados por la empresa Rational Software Co. para crear una notación unificada en la que basar la construcción de sus herramientas CASE. En el proceso de creación de UML han participado, no obstante, otras empresas de gran peso en la industria como Microsoft, Hewlett-Packard, Oracle o IBM, así como grupos de analistas y desarrolladores[14].

1.6.2. PHP

Hypertext Pre-processor (inicialmente PHP Tools, o, Personal Home Page Tools) lenguaje de programación interpretado y diseñado originalmente para la creación de páginas web dinámicas. Se usa principalmente para la interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno, destaca su conectividad con MySQL y PostgreSQL. Permite aplicar técnicas de programación orientada a objetos, y no obliga a seguir una metodología a la hora de programar, pero a pesar de todo se ha desarrollado con el patrón de diseño MVC que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes[15].

1.6.3. HTML

HTML, siglas de HyperText Markup Language («lenguaje de marcado de hipertexto»), es el lenguaje de marcado predominante para la elaboración de páginas web[16]. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de

«etiquetas», rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo JavaScript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML [17].

1.6.4. JavaScript

JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, en bases de datos locales al navegador. Aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo [18].

1.6.5. CSS

CSS es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es una forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas.

Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, ya que obliga a crear documentos HTML/XHTML definidos y con significado completo (también llamados "documentos semánticos"). Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en dispositivos diferentes.

Al crear una página web, se utiliza en primer lugar el lenguaje HTML/XHTML para marcar los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla y lista de elementos.

Una vez creados los contenidos, se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, etc[17].

1.7. Herramientas utilizadas para el desarrollo del sistema

1.7.1. Visual Paradigm

Visual Paradigm for UML (Enterprise Edition): Herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientado a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una rápida construcción de aplicaciones de calidad y a un menor costo. Permite dibujarlos tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Algunas de sus características son:

- Software libre.
- Disponibilidad en múltiples plataformas.
- Diseño centrado en casos de uso.
- Permite la realización de ingeniería directa e inversa.
- Posee licencia gratuita y comercial.
- Fácil de instalar y actualizar [19].

1.7.2. Umbrello

Umbrello UML Modeller es una herramienta de diagramas “ que ayuda en el proceso del desarrollo de software. Umbrello UML Modeller le facilitará la creación de un producto de alta rendimiento, especialmente durante fases de análisis y diseño del proyecto. UML también puede usarse para documentar sus diseños de software para ayuda los desarrolladores[20].

1.7.3. PhpMyAdmin

Es una herramienta de software libre escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web. Soporta un amplio rango de

operaciones en MySQL, MariaDB y Drizzle que pueden ser realizadas utilizando la interfaz del usuario o ejecutando directamente las sentencias SQL[21].

1.7.4. NetBeans

NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de Componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas por otros desarrolladores de software [22].

1.7.5. PHP Unit

PHPUnit es un entorno para realizar pruebas unitarias en el lenguaje de programación PHP. PHPUnit es un framework de la familia xUnit originada con SUnit de Kent Beck. PHPUnit se puede encontrar en GitHub y ha sido creado por Sebastian Bergmann[23].

1.7.6. XAMPP

XAMPP es un servidor independiente de plataforma, software libre, que consiste principalmente en la base de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script: PHP y Perl. El nombre proviene del acrónimo de X (para cualquiera de los diferentes sistemas operativos), Apache, MySQL, PHP y Perl.

El programa está liberado bajo la licencia GNU y actúa como un servidor web libre, fácil de usar y capaz de interpretar páginas dinámicas [24].

1.7.7. Yii PHP Framework

Yii es un framework PHP basado en componentes de alta performance para desarrollar aplicaciones Web de gran escala. El mismo permite la máxima reutilización en la programación web y puede acelerar el proceso de desarrollo.

Para correr una aplicación Web Yii, usted necesita tener un servidor Web con soporte PHP 5.1.0 o superior.

Para desarrolladores que deseen utilizar Yii, el entendimiento de Programación Orientada a Objetos (POO) será de gran ayuda ya que Yii es un framework totalmente basado en POO [25].

1.7.8. YiiBooster

YiiBooster es una colección de widgets que aliviará la tarea de desarrollar las aplicaciones de Yii. Básicamente, YiiBooster así los desafíos más comunes que diseñadores de Yii enfrentan mientras intentando mejorar sus aplicaciones [26].

1.8. Conclusiones del capítulo

En este capítulo se realizó un estudio de los conceptos asociados al dominio del problema, así como un análisis de las tecnologías Web, las metodologías, los lenguajes de programación. Se selecciona la metodología RUP como guía para la documentación del software propuesto y UML como lenguaje para modelar el análisis y diseño así como Visual Paradigm como herramienta CASE.

Ya que la solución está enfocada a facilitar el trabajo a desarrolladores de Yii PHP Framework es necesario utilizar esta tecnología basada en PHP. Además, se utilizarán del lado del cliente HTML5, CSS y JavaScript en su conjunto como parte del framework Bootstrap. Como entorno de desarrollo se seleccionaron XAMPP y NetBeans. Toda esta selección, fue realizada sobre la consideración de las potencialidades de dichas herramientas y lenguajes para llevar a cabo la implementación del software propuesto.

Capítulo 2: Descripción y construcción de la solución propuesta

2.1. Introducción

En el presente capítulo, tomando como guía la Metodología RUP, se utiliza uno de los artefactos que brinda dicha metodología: el Modelo de Dominio, el cual ayuda a modelar y describir el comportamiento actual de los procesos objeto de automatización. Se realiza una identificación de los procesos del dominio, así como el enunciado y descripción de sus reglas para garantizar las restricciones existentes en el mismo.

Además, se describe y analiza el modelo de sistema. Se identifican los requerimientos funcionales y no funcionales, se definen los actores del sistema y las funcionalidades que a su disposición se colocan (los casos de uso del sistema).

2.2. Definición de las entidades y los conceptos principales

2.2.1. Aplicación

Programa informático que permite a un usuario utilizar una computadora con un fin específico. Las aplicaciones son parte del software de una computadora, y suelen ejecutarse sobre el sistema operativo.

Una aplicación de software suele tener un único objetivo: navegar en la web, revisar correo, explorar el disco duro, editar textos, jugar (un juego es un tipo de aplicación). Una aplicación que posee múltiples programas se considera un paquete [27].

2.2.2. Modulo

Un módulo es un software que agrupa un conjunto de subprogramas y estructuras de datos. Los módulos son unidades que pueden ser compiladas por separado y los hace reusables y permite que múltiples programadores trabajen en diferentes módulos en forma simultánea, produciendo ahorro en los tiempos de desarrollo.

Los módulos promueven la modularidad y el encapsulamiento, pudiendo generar programas complejos de fácil comprensión [28].

2.2.3. Componente

Componente de Software, un elemento de un sistema software que ofrece un conjunto de servicios, o funcionalidades, a través de interfaces definidas.

Las diferentes posturas acerca de las características de los componentes de software hace difícil la elaboración de una definición única. La utilización de una específica dependerá del contexto en donde se esté empleando el componente y de los acuerdos que se haga dentro del marco de trabajo [29].

2.2.4. Formulario

Un formulario es un documento con espacios (campos) en donde se pueden escribir o seleccionar opciones. Cada campo tiene un objetivo, por ejemplo, el campo "Nombre" se espera que sea llenado con un nombre, el campo "año de nacimiento", se espera que sea llenado con un número válido para un año.

Los formularios presentan una visión ordenada de múltiple información sobre algo, y son útiles para llenar bases de datos.

Los formularios por internet son llamados formularios web, y generalmente son hechos a través de etiquetas HTML, aunque también existen otros medios como Flash, Java, etc[30].

2.2.5. Gii

Apartir de version 1.1.2, Yii trajo una herramienta web de generación de código que se llama Gii.

Gii está implementado como un módulo dentro de una aplicación Yii existente.

Gii es realmente una herramienta muy útil para el desarrollo, ya que nos ayuda a generar una gran cantidad de códigos para nuestros modelos, controladores y vistas. El trabajo que nos queda por hacer es el de acomodar el código o el diseño a nuestras necesidades específicas [31].

2.2.6. Plantilla de Gii

Son definiciones que un desarrollador puede crear para especificar de qué forma se generará el código por la herramienta Gii. Aquí puede incluir detalles de estilo y funcionalidad y puede ser útil para una o varias aplicaciones, en dependencia de las características homogéneas o particulares que se quieran brindar [32].

2.3. Reglas a considerar

- Las llaves primarias de las tablas que se gestionan tienen que ser simples. Se recomiendan campos autoincrementales.
- Para cada entidad que se gestiona se deben definir los métodos de configuración, como mínimo `adminSearch()`.
- Para utilizar el módulo es necesario iniciar sesión con las credenciales de acceso configuradas.

2.4. Modelo del dominio

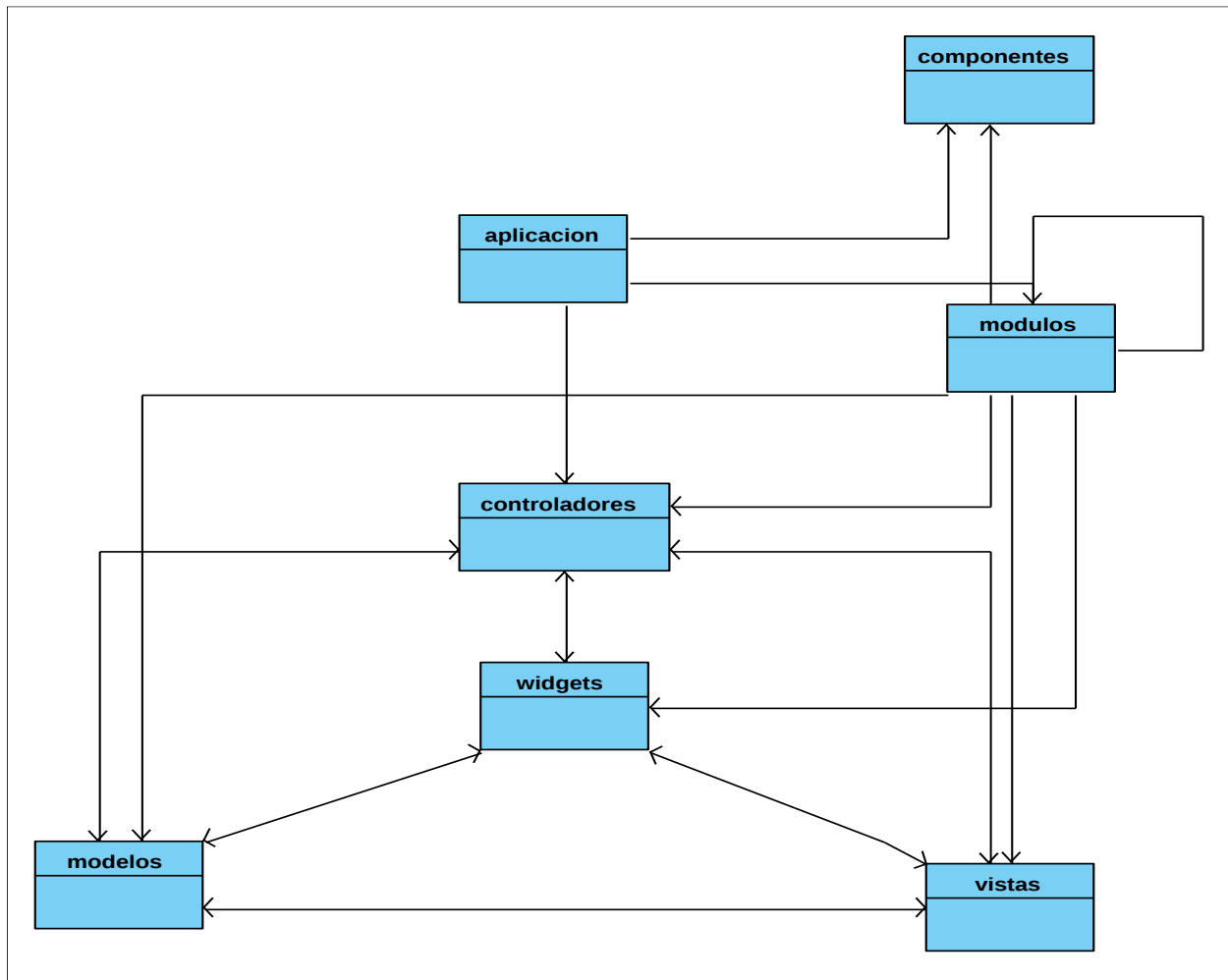


Figura7El Modelo del Dominio.

2.5. Requisitos

2.5.1. Actores del sistema a automatizar

Nombre del actor	Descripción
Administrador	Ejecuta todas las funcionalidades del sistema

Tabla 1 Descripción de los Actores del Sistema.

2.5.2. Los requisitos funcionales

- Insertar entidad de cada tipo de dato del sistema.
- Listar entidades de cada tipo de dato del sistema.
- Buscar entidades de cada tipo de dato del sistema.
- Actualizar entidad de cada tipo de dato del sistema.
- Eliminar entidad de cada tipo de dato del sistema.
- Listar entidades relacionadas de forma uno a mucho desde la vista de la entidad en el lado uno.
- Agregar entidades relacionadas de forma uno a mucho desde la vista de la entidad en el lado uno.
- Eliminar entidades relacionadas de forma uno a mucho desde la vista de la entidad en el lado uno.
- Actualizar entidades relacionadas de forma uno a mucho desde la vista de la entidad en el lado uno.
- Imprimir un listado de modelos
- Imprimir los detalles de una instancia de modelo
- Listar las entidades gestionables
- Exportar a PDF los detalles de una instancia de modelo
- Exportar a PDF un listado de instancias de modelo
- Exportar a Excel un listado de instancias de modelo
- Exportar a CSV un listado de instancias de modelo

2.5.3. Los requisitos no funcionales

2.5.3.1. Apariencia e interfaz externa

El módulo genera vistas basadas en Bootstrap 3 a partir de la extensión YiiBooster 4. Se garantizan interfaces responsivas y de fácil navegación.

2.5.3.2. Usabilidad

A partir del correcto uso de Bootstrap 3 y del estándar HTML 5 se garantiza una visualización y experiencia correctas en dispositivos para personas con dificultades de visión y movilidad.

2.5.3.3. Rendimiento

El uso del módulo no debe afectar el rendimiento clásico de las aplicaciones desarrolladas con Yii PHP Framework.

2.5.3.4. Portabilidad

Se asegura la portabilidad entre sistemas Windows y Unix-like atendiendo a las características específicas de cada uno de estos. Por ejemplo, sensibilidad a mayúsculas para los nombres y rutas de archivos.

2.5.3.5. Hardware

El módulo desarrollado es capaz de desplegarse en entornos de hardware necesarios para alojar aplicaciones web.

2.5.3.6. Software

El módulo desarrollado es capaz de desplegarse en el entorno de software necesario para aplicaciones basadas en Yii PHP Framework.

- Estación de trabajo cliente.
 - Navegador web que supere las pruebas ACID3.
- Servidor.
 - Apache 2, IIS 7, u otros servidores web.
 - PHP 5.3.

2.6. Diagrama de Casos de Uso del Sistema

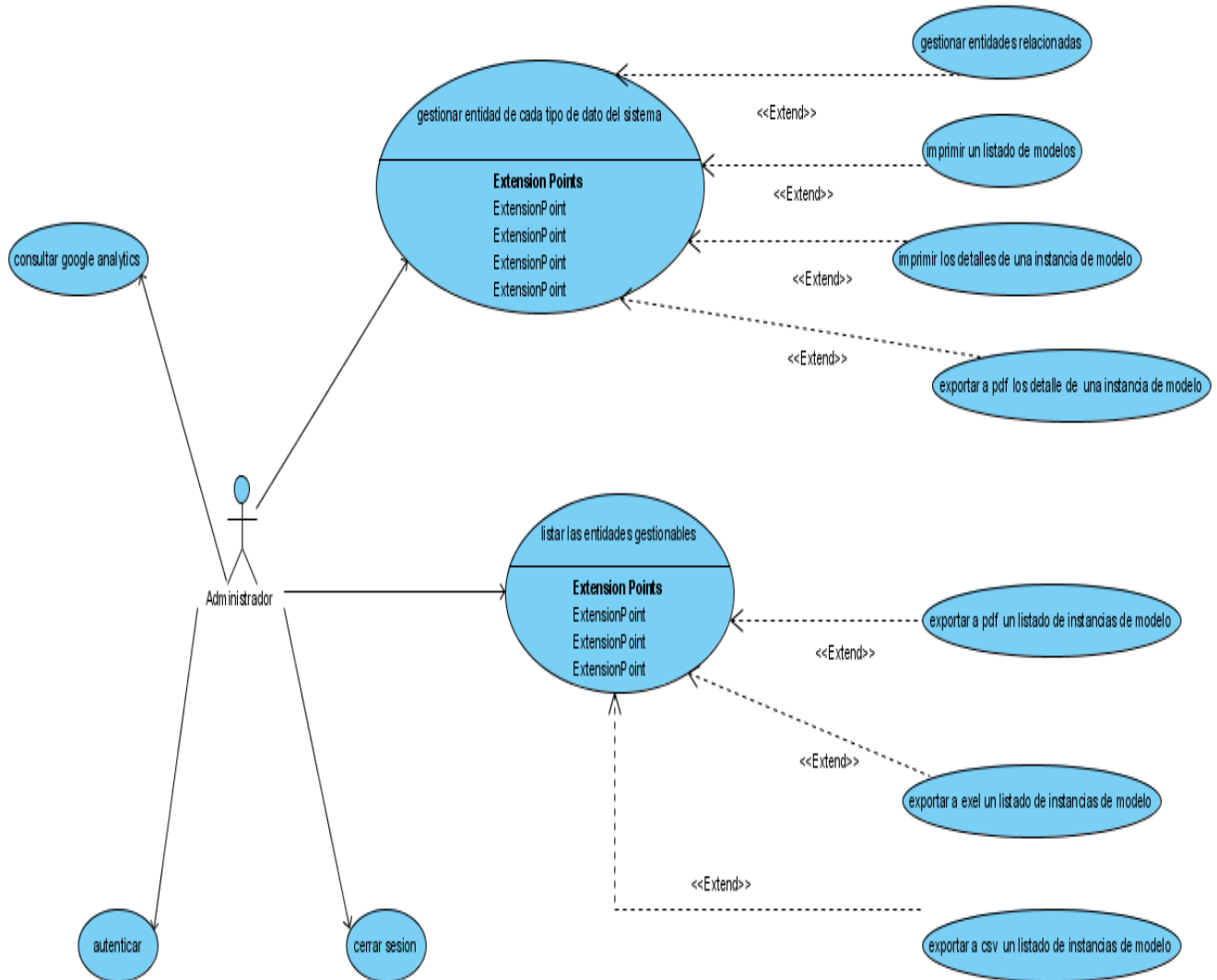


Figura 8 Diagrama de Casos de Uso del Sistema.

2.7. Descripción de los casos de uso

Caso de Uso	Descripción Textual	Prototipo
Gestionar entidad de cada tipo de dato del sistema	Anexo: A1	Anexo: B1
Gestionar entidades relacionadas	Anexo: A2	Anexo: B2
Imprimir un listado de modelos	Anexo: A3	Anexo: B3
Imprimir los detalles de una instancia de modelo	Anexo: A4	Anexo: B4

Listar las entidades gestionables	Anexo: A5	Anexo: B5
Exportar a pdf los detalle de una instancia de modelo	Anexo: A6	Anexo: B6
Exportar a pdf un listado de instancias de modelo	Anexo: A7	Anexo: B7
Exportar a excel un listado de instancias de modelo	Anexo: A8	Anexo: B8
Exportar a csv un listado de instancias de modelo	Anexo: A9	Anexo: B9
Consultar googleanalytics	Anexo: A10	Anexo: B10
Autenticar	Anexo: A11	Anexo: B11
Cerrar sesión	Anexo: A12	Anexo: B12

Tabla 2Descripción de los casos de uso del Sistema.

2.7.1. Diagrama de clases del diseño

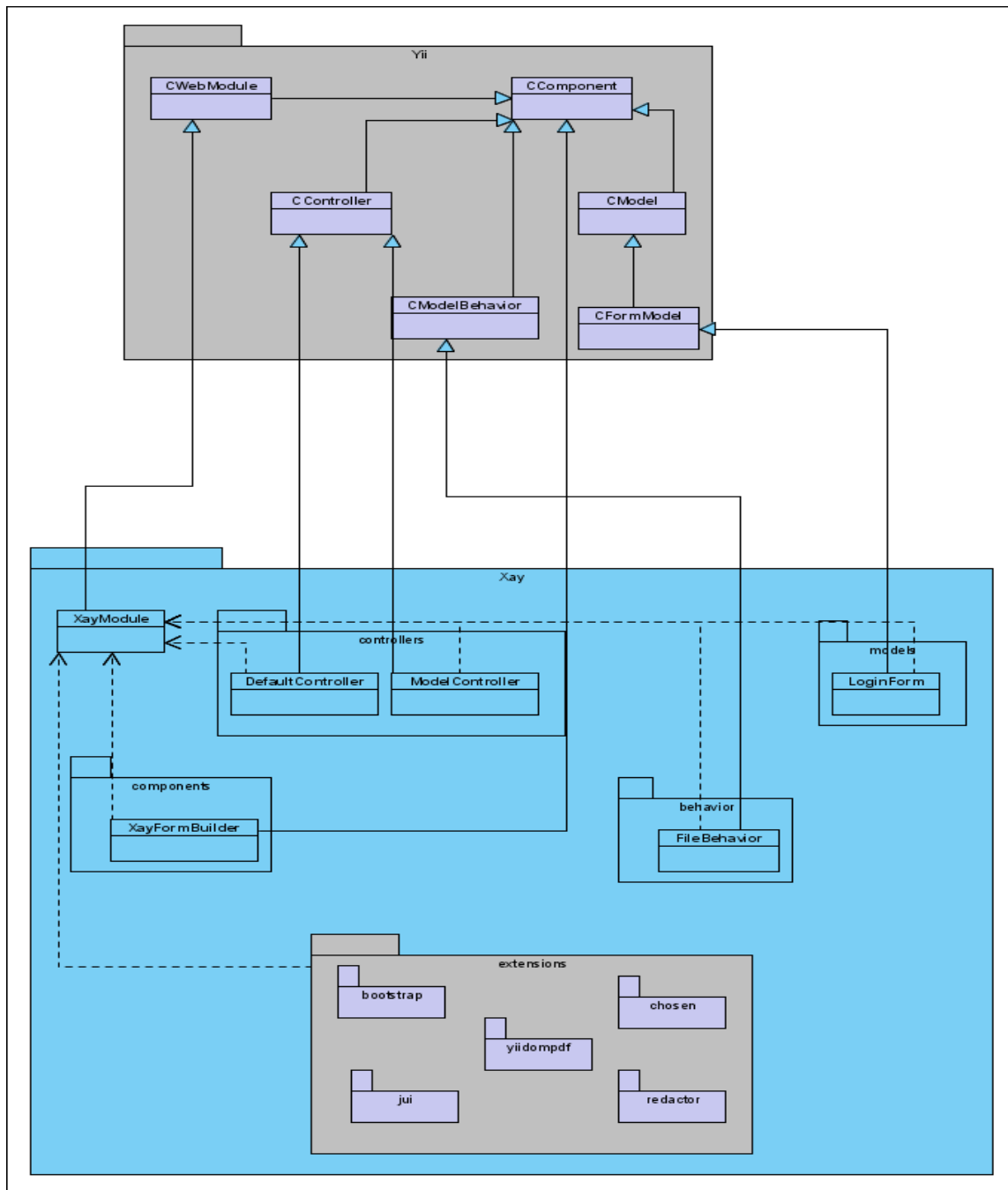


Figura 9 Diagrama de clases del diseño

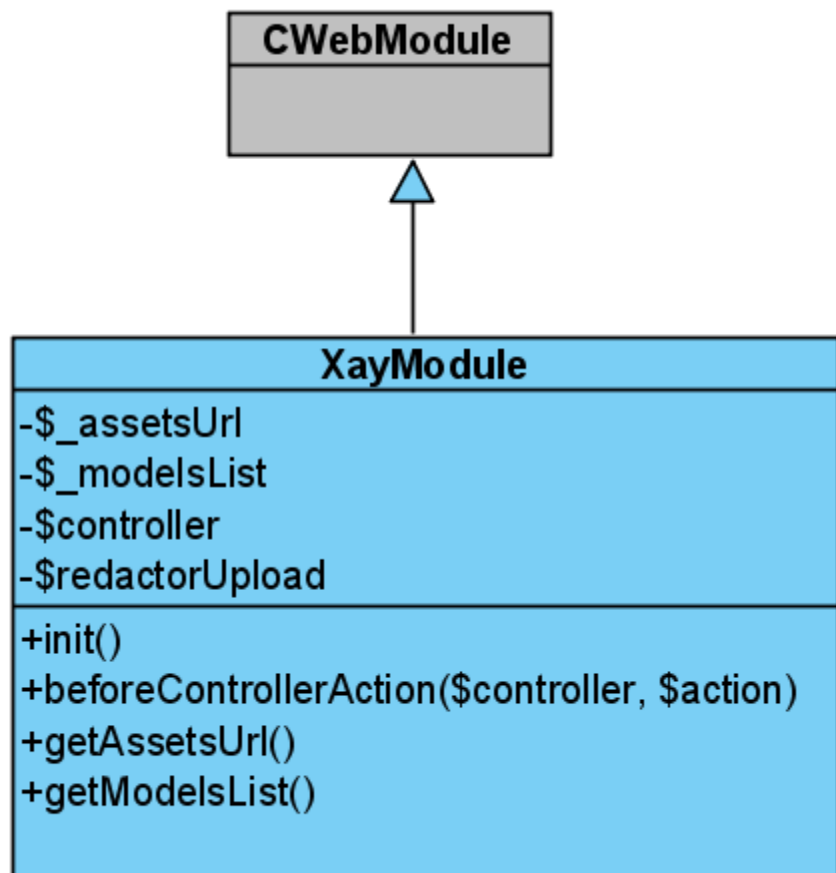


Figura 10 Diagrama de clases de XayModule

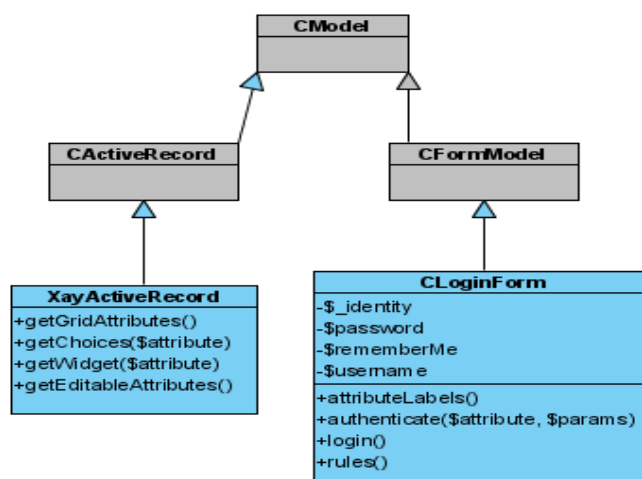


Figura 11 Diagrama de clases de Modelo

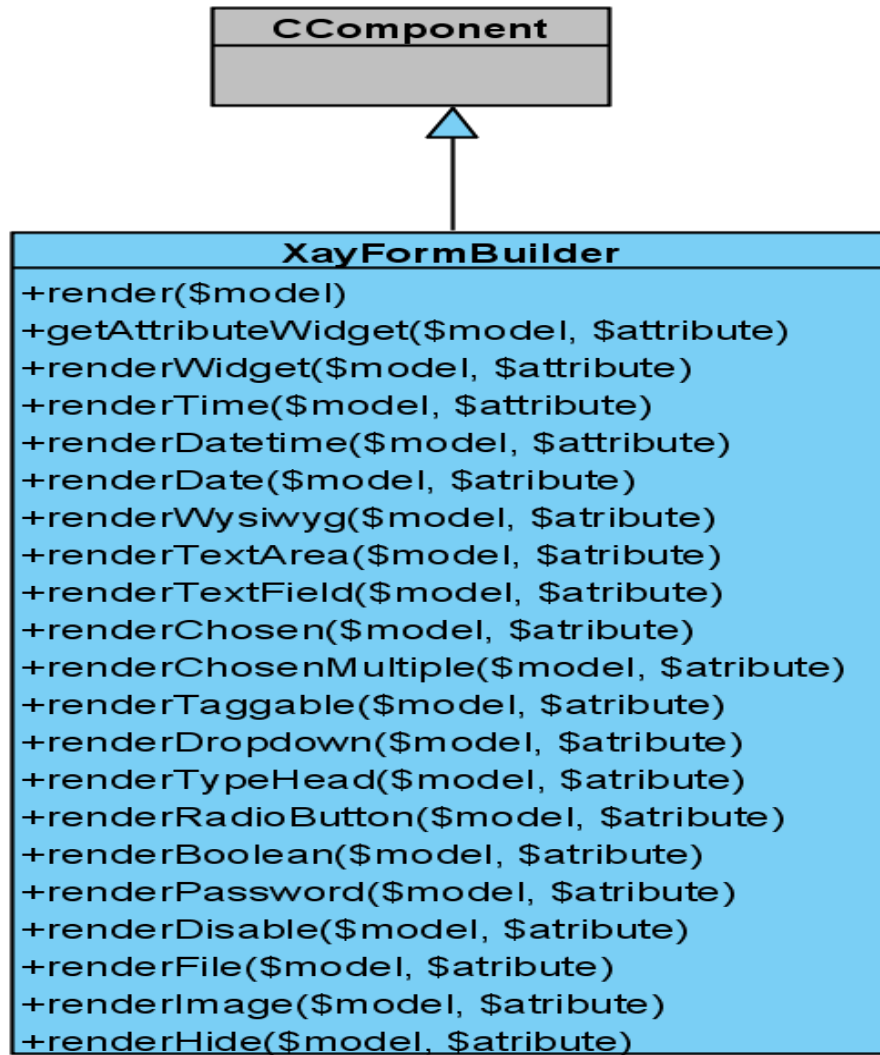


Figura 12 Diagrama de clases de Componente

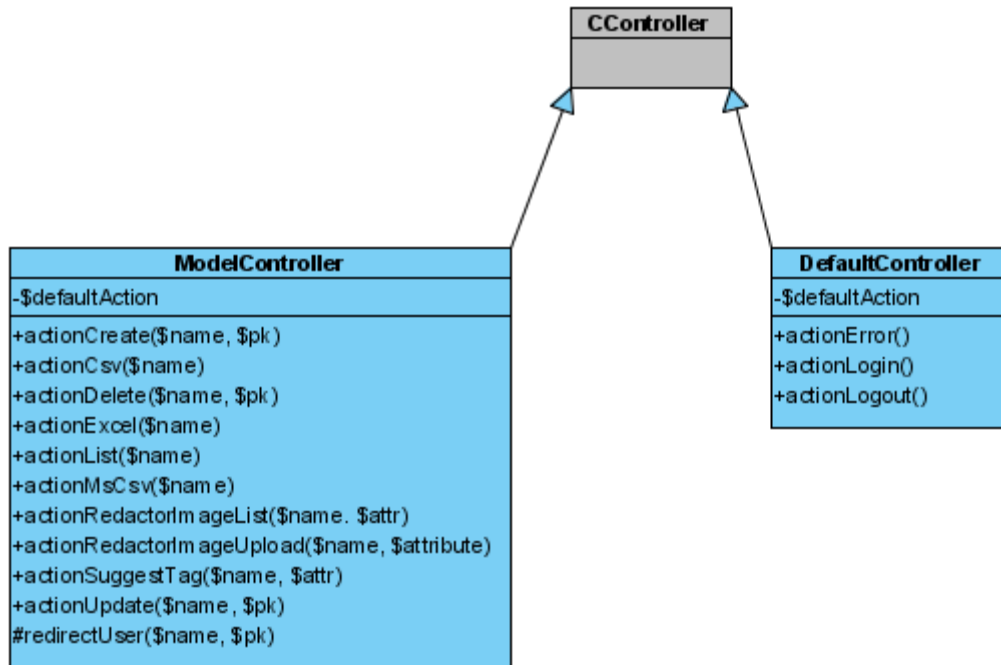


Figura 13 Diagrama de clases de Controlador

2.7.2. Principios de diseño

2.7.2.1. Interfaz de usuario

Utilización de la maquetación con Bootstrap, así como sus elementos de navegación. La interfaz es responsiva de forma tal que se puede utilizar correctamente en todos los tamaños de pantalla.

2.8.2.1. Formato de salida de los reportes

Los reportes se realizarán en los formatos Excel, CSV y PDF así como una vista de impresión.

2.8.2.2. Tratamiento de errores

Las excepciones son generadas utilizando el lenguaje de programación PHP, capturadas por la clase `CHttpException` del framework Yii y finalmente personalizada por el desarrollador, estas se pueden personalizar utilizando el código de estado del protocolo HTTP.

2.8.2.3. Protección del sistema ante vulnerabilidades web

En vista a la protección contra ataques se tomaron en cuenta las recomendaciones brindadas por el grupo OWASP (Open Web Application Security Project) el cual publica manualmente un informe con las vulnerabilidades más comunes encontradas en aplicaciones web y las formas de contrarrestarlas.

La mayoría de las vulnerabilidades mencionadas por los informes de este grupo son cubiertas si se cumplen con las buenas prácticas de desarrollo propuestas para Yii PHP Framework. Entre estas vulnerabilidades se encuentran:

- Inyecciones SQL
 - Utilización de parámetro al realizar consultas a la base de datos
- Inyecciones XSS
 - Utilización del método CHtml::encode para mostrar información entrada por los usuarios
- Cross Site Request Forgery (CSRF)
 - Envío de llaves únicas al navegador de cada usuario para garantizar que quien envía la información es quien dice ser.
- Control de los tipos de datos
 - Especificación de reglas de validación en los modelos para evitar la asignación y posterior uso de tipos de datos incorrectos a los atributos de las clases diseñadas.
- Ocultamiento de información sensible:

La aplicación tiene mensajes de error personalizados evitándose la publicación de información sensible ante cualquier evento anómalo.

2.8.3. Diagrama de despliegue

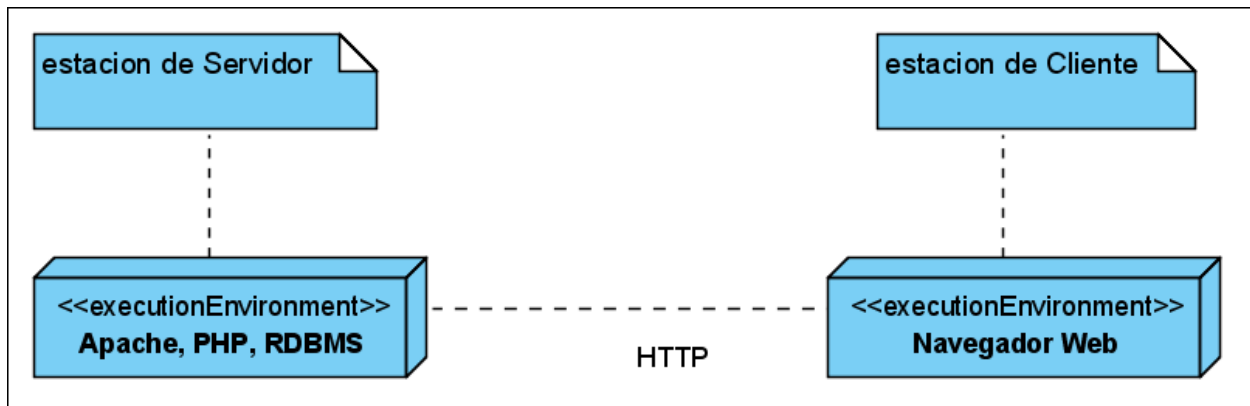


Figura 14 Diagrama de despliegue

2.9. Conclusiones del capítulo

Con el desarrollo de este capítulo se logra un mejor entendimiento de la solución propuesta, se describieron los diagramas de clases Web, diagrama de despliegue, Además fueron definidos, los principios de diseño del sistema que abarcan: los estándares en la interfaz de la aplicación el tratamiento de errores. Todo esto propició un análisis completo y marcó una guía en la implementación del software propuesto.

En este capítulo fueron descritos los procesos del dominio. Se enumeraron las reglas y se identificaron, dando paso al modelado del sistema.

Del modelado del sistema enunciamos sus requerimientos funcionales y no funcionales, identificándose y describiéndose los actores y casos de uso. Se describieron los diagramas de casos de uso del sistema, los diagramas de clases de diseño, y el diagrama de despliegue.

Además fueron definidos, los principios de diseño del sistema que abarcan: la interfaz de usuario, el formato de salida de los reportes y el tratamiento de errores. Todo esto propició un análisis completo y claro del modelo del sistema y marcó una guía en la implementación del software propuesto.

Capítulo 3: Estudio de factibilidad

3.1. Introducción

En este capítulo se presenta el estudio de factibilidad del sistema. Se estiman el esfuerzo humano y el tiempo de desarrollo que se requieren para su elaboración, así como los costos del mismo.

3.2. Factibilidad

Algunas alternativas posibles para la estimación del esfuerzo en proyectos basados en Casos de Uso, son el Análisis de Puntos de Función y COCOMO II, o una variante más reciente denominada Análisis de Puntos de Casos de Uso.

Esta técnica permite cuantificar el tiempo de desarrollo de un proyecto, independientemente del lenguaje de programación, las metodologías, plataformas y/o tecnologías utilizadas, pero si teniendo en cuenta ciertos factores y su influencia en el proyecto.

La estimación mediante el análisis de Puntos de Casos de Uso es un método propuesto originalmente por Gustav Karner, y posteriormente refinado por muchos otros autores. Se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

3.2.1. Cálculos de Puntos de Casos de Uso sin ajustar

El primer paso para la estimación consiste en el cálculo de los Puntos de Casos de Uso sin ajustar. Este valor, se calcula a partir de la siguiente ecuación:

$$UUCP = UAW + UUCW$$

Dónde:

UUCP: Puntos de Casos de Uso sin ajustar

UAW: Factor de Peso de los Actores sin ajustar

UUCW: Factor de Peso de los Casos de Uso sin ajustar

3.2.2. Factor de peso de los actores sin ajustar (UAW)

Este valor se calcula mediante un análisis de la cantidad de Actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Actores se establece teniendo en cuenta en primer lugar si se trata de una persona o de otro sistema, y en segundo lugar, la forma en la que el actor interactúa con el sistema.

Los criterios se muestran en la siguiente tabla:

Tipo de Actor	Descripción	Factor de Peso
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación.	1
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto.	2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica.	3

Tabla 3 Factor de peso de los actores del sistema.

Actor	Tipo de Actor
Administrador	Complejo

Tabla 4 Clasificación de los actores atendiendo al factor de peso.

Como se describe en la tabla anterior existen en el sistema dos actores de tipo complejo (Usuario), (Administrador), ya que interactúa con el sistema mediante una interfaz gráfica.

Multiplicando la cantidad de actores de cada tipo por el peso correspondiente se obtiene que:

$$UAW = 1 \times 3$$

$$UAW = 3$$

3.2.3. Factor de peso de los casos de uso sin ajustar (UUCW)

Este valor se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómica, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia. Los criterios se muestran en la siguiente tabla:

Tipo de CU	Transacciones	Peso
Simple	menos de 4	5
Medio	de 4 a 7	10
Complejo	más de 7	15

Tabla 5 Criterios del factor de peso de los casos de uso sin ajustar.

Caso de Uso	Tipo
Gestionar entidad de cada tipo de dato del sistema	Medio
Gestionar entidades relacionadas	Medio
Imprimir un listado de modelos	Simple
Imprimir los detalles de una instancia de modelo	Simple
Listar las entidades gestionables	Simple
Exportar a pdf los detalle de una instancia de modelo	Simple
Exportar a pdf un listado de instancias de modelo	Simple
Exportar a exel un listado de instancias de modelo	Simple
Exportar a csv un listado de instancias de modelo	Simple
Consultar googleanalytics	Simple
Autenticar	Simple
Cerrar sesión	Simple

Tabla 6 Clasificación de los casos de uso del sistema.

De acuerdo a la tabla anterior se cuenta con 2 casos de uso de tipo medio (peso 10) y 10 de tipo simple (peso 5) con lo cual el factor de peso de los casos de uso sin ajustar resulta:

$$UUCW = 2 \times 10 + 10 \times 5$$

$$UUCW = 70$$

Finalmente, los Puntos de Casos de Uso sin ajustar resultan

$$UUCP = UAW + UUCW$$

$$UUCP = 3 + 70$$

$$UUCP = 73$$

3.2.4. Cálculo de puntos de casos de uso ajustados

Una vez que se tienen los Puntos de Casos de Uso sin ajustar, se debe ajustar este valor mediante la siguiente ecuación:

$$UCP = UUCP \times TCF \times EF$$

Dónde:

UCP: Puntos de Casos de Uso ajustados

UUCP: Puntos de Casos de Uso sin ajustar

TCF: Factor de complejidad técnica

EF: Factor de ambiente

Es necesario calcular los valores de TCF y EF.

3.2.5. Factor de complejidad técnica (TCF)

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante. En la siguiente tabla se muestra el significado, el peso de cada uno, el valor asignado y el peso total:

Factor	Descripción	Peso	Valor Asignado	Peso Total
T1	Sistema distribuido	2	0	0
T2	Objetivos de performance o tiempo de respuesta	1	5	5
T3	Eficiencia de usuario final	1	5	5
T4	Procesamiento interno complejo	1	2	2

T5	El código debe ser reutilizable	1	5	5
T6	Facilidad de instalación	0.5	5	2.5
T7	Facilidad de uso	0.5	5	2.5
T8	Portabilidad	2	5	10
T9	Facilidad de cambio	1	4	4
T10	Concurrencia	1	5	5
T11	Incluye objetos especiales de seguridad	1	2	2
T12	Provee acceso directo a terceras partes	1	0	0
T13	Se requieren facilidades especiales de entrenamiento a usuarios	1	1	1

Tabla 7 Factores de complejidad del sistema.

El Factor de complejidad técnica se calcula mediante la siguiente ecuación:

$$TCF = 0.6 + 0.01 * \sum \text{Peso } i * \text{Valor Asignado } i$$

$$TCF = 0.6 + 0.01 * 44$$

$$TCF = 1.04$$

3.2.6. Factor de ambiente (EF)

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del Factor de ambiente. El cálculo del mismo es similar al cálculo del Factor de complejidad técnica, es decir, se trata de un conjunto de factores que se cuantifican con valores de 0 a 5.

En la siguiente tabla se muestra el significado y el peso de cada uno de éstos factores.

Factor	Descripción	Peso	Valor Asignado	Peso Total
E1	Familiaridad con el modelo utilizado.	1.5	3	4.5
E2	Experiencia con la aplicación.	0.5	1	0.5
E3	Experiencia en orientación a objeto.	1	2	2

E4	Capacidad del analista líder.	0.5	4	2
E5	Motivación.	1	5	5
E6	Estabilidad de los requerimientos.	2	5	10
E7	Personal Part-time.	-1	0	0
E8	Dificultad del lenguaje de programación.	-1	2	-2

Tabla 8 Habilidades del grupo de desarrollo.

El Factor de ambiente se calcula mediante la siguiente ecuación:

$$EF = 1.4 - 0.03 * \sum P_{esoi} * ValorAsignado_i$$

$$EF = 1.4 - 0.03 * 22$$

$$EF = 0.74$$

Con el cálculo de estos valores, es posible sustituir en la ecuación inicial y obtener el valor de los puntos de caso de uso ajustado.

De esta forma:

$$UCP = UUCP \times TCF \times EF$$

$$UCP = 73 \times 1.04 \times 0.74$$

$$UCP = 56.1808$$

Estimación de esfuerzo a través de los puntos de casos de uso

CF: Factor de Conversión

CF= 20 Horas-Hombre

El esfuerzo en horas /hombre está dado por:

$$E = UCP \times CF$$

$$E = 56.1808 * 20$$

$$E = 1123.616 \text{ horas-hombre.}$$

Duración:

Trabajando 25 días al mes y 8 horas diarias como promedio, se tiene que:

Duración (días)= Total de Horas /Hombre entre 8 horas al día

$$= 1123.616/8$$

Duración (días)=140.452días.

Duración (meses) =Total de días / 25 días por mes

$$= 140.452/ 25$$

Duración (meses) = 5.61808≈ 6 meses.

Actividad	Porcentaje	Valor (Horas-Hombre)
Análisis	20%	224.7232
Diseño	40%	449.4464
Programación	30%	337.0848
Prueba	10%	112.3616
Sobrecarga	0	0
Total de Horas	100%	1123.616

Tabla 9 Criterios de distribución de esfuerzo.

3.2.7. Cálculo de costos

Tomando como salario promedio mensual \$555.00.

Costo = 6 meses x \$555.00

Costo = \$3330.00 MN

Costo = \$133.2 CUC

3.3. Beneficios tangibles e intangibles

Los beneficios tangibles no son más que las ventajas económicas cuantificables que obtiene la organización a través del uso del sistema de información.

Los beneficios obtenidos con el desarrollo del presente software permiten agilizar el proceso de desarrollo de aplicaciones con el frameworkYii, contribuyendo a disminuir la

posibilidad de errores de duplicidad de código e inconsistencias en la gestión de la información.

Como beneficios intangibles se puede citar la contribución en forma de código para la comunidad de software libre, específicamente para la comunidad desarrolladora de Yii PHP Framework.

3.4. Análisis de costos y beneficios

Este sistema, como resultado del presente trabajo de diploma, no implica costo alguno para el lugar donde se va a implantar, sin embargo, al desarrollo de todo producto informático va asociado un costo y su justificación económica viene dado por los beneficios tangibles e intangibles que este proporciona.

Anteriormente el proceso de personalización de los formularios generados por Gii para cada uno de los modelos que se vayan a gestionar en el sistema y su posterior procesamiento se hace engorroso debido a su carácter estático y la duplicación del código para idénticas funcionalidades. Con la ayuda del sistema informático propuesto se logra eliminar los problemas antes descritos. Teniendo en cuenta el análisis del costo del sistema y los beneficios que brinda el mismo al cumplir con los objetivos planteados, se concluye que la aplicación es factible.

3.5. Conclusiones del capítulo

En el presente capítulo se determinó la factibilidad del sistema utilizando el método de cálculo de puntos de casos de usos. La realización del estudio de factibilidad del producto informático proyectó una cantidad significativa de beneficios tangibles e intangibles. Una vez concluido el estudio de factibilidad del sistema, se estima un tiempo de 6 meses para su construcción por un hombre y su costo asciende a \$3330.00.

Conclusiones

A partir de los objetivos planteados con anterioridad se arriba a las siguientes conclusiones:

- Se diseñó una solución capaz de gestionar la mayoría de los casos de gestión de información.
- Se implementó la solución como un módulo de Yii PHP Framework.
- La utilización de XAY reduce la cantidad de archivos que se deben editar en el desarrollo de una aplicación basada en Yii PHP Framework.
- Se calculó que es económicamente factible el desarrollo de la solución.

Recomendaciones:

A pesar de que los objetivos trazados para la realización del trabajo de diploma fueron cumplidos, se recomienda tomar esta propuesta como la primera etapa de un proyecto más amplio.

Se recomienda como pasos que den continuidad:

- Extender las funcionalidades para permitir la personalización de las operaciones en los casos que se requiera.
- Ejecutar métodos de refactorización para mejorar la arquitectura del módulo.

Referencias Bibliográficas

- [1] “Gestión de la información — Biblioteca Virtual de la UCF.” [Online]. Available: <http://biblioteca.ucf.edu.cu/biblioteca/articulos-descargados/sociocultural/Gestion%20de%20la%20informacion.pdf/view>. [Accessed: 28-Mar-2015].
- [2] J. C. Cobo Romaní, “El concepto de tecnologías de la información. Benchmarking sobre las definiciones de las TIC en la sociedad del conocimiento.” 22-Sep-2009.
- [3] “La programación web,” in *PROGRAMACIÓN WEB JAVA*, lectores@mpediciones.com, p. 26.
- [4] “¿Qué es un ‘framework’? | jordisan.net.” [Online]. Available: http://www.addtoany.com/share_save?url=http%3A%2F%2Fjordisan.net%2Fblog%2F2006%2Fque-es-un-framework%2F&title=%C2%BFQu%C3%A9%20es%20un%20%E2%80%98framework%E2%80%99%3F&description=El%20t%C3%A9rmino%20framework%20se%20utiliza%20constantemente%20en%20el%20desarrollo%20de%20software%2C%20pero...%20%C2%BFsabr%C3%ADamos%20definir%20qu%C3%A9%20es%20un%20framework%20%3F%20He%20aqu%C3%AD%20una%20descripci%C3%B3n. [Accessed: 28-Mar-2015].
- [5] Luzcila, “Los 11 Mejores Frameworks gratuitos para Aplicaciones Web,” *El Baúl del Programador*. [Online]. Available: <http://elbauldelprogramador.com/los-10-mejores-frameworks-gratis-de-aplicaciones-web/>. [Accessed: 27-Mar-2015].
- [6] A. Chlipala, “Ur: statically-typed metaprogramming with type-level record computation,” *ACM SIGPLAN Notices*, vol. 45, no. 6, pp. 122–133, Jun. 2010.
- [7] A. V. Aho, *Compilers: Principles, Techniques, and Tools*. Addison-Wesley, 1987.
- [8] “Scaffolding - KumbiaPHP Framework.” [Online]. Available: <http://wiki.kumbiaphp.com/Discusión:Scaffolding>. [Accessed: 26-Mar-2015].
- [9] K. Czarnecki and U. W. Eisenecker, *Generative Programming*. 2000.
- [10] “Getting Started.” [Online]. Available: <http://php.net/manual/en/getting-started.php>.
- [11] Q. Xue and X. Wei Zhuo, “yii-guide-1.1.13-RC.” 2008.

- [12] M. Fowler, *Patterns of Enterprise Application Architecture*. Addison Wesley, 2002.
- [13] “Por qué utilizar RUP para desarrollar aplicaciones web.” [Online]. Available: <http://www.eumed.net/libros-gratis/2009c/584/584.zip>. [Accessed: 28-Mar-2015].
- [14] J. de D. Bátis Paredes, *Desarrollo Orientado a Objetos con UML*. .
- [15] W. Jason Gilmore, *Beginning PHP and PostgreSQL 8: From Novice to Professional*. 2006.
- [16] “Tags used in HTML.” [Online]. Available: www.w3.org/History/19921103-hypertext/hypertext/WWW/MarkUp/Tags.html. [Accessed: 29-Mar-2015].
- [17] “HTML & CSS - W3C.” [Online]. Available: <file:///I:/HTML%20&%20CSS%20-%20W3C.htm#whathtml>. [Accessed: 26-Mar-2015].
- [18] T. Powell and F. Schneider, “JavaScript 2.0 The Complete Reference, 2nd Edition.” .
- [19] “Visual Paradigm for UML (ME) (Paradigma Visual para UML (ME)) (Visual Paradigm for UML (ME)) por Visual Paradigm International Ltd. - reporte y descarga.” May-2013.
- [20] P. Hensgen, “Manual de Umbrello UML Modeller.” 2001.
- [21] phpMyAdmin contributors, “phpMyAdmin,” *phpMyAdmin*. [Online]. Available: www.phpmyadmin.net. [Accessed: 29-Mar-2015].
- [22] J. M. Gimeno and J. L. González, “Introducción a Netbeans.” José Luis González.
- [23] S. Bergmann, *PHPUnit Manual*. 2015.
- [24] “XAMPP Installers and Downloads for Apache Friends.” [Online]. Available: <https://www.apachefriends.org/es/index.html>. [Accessed: 29-Mar-2015].
- [25] “Introducción: Qué es Yii | The Definitive Guide to Yii | Yii PHP Framework.” [Online]. Available: <http://www.yiiframework.com/doc/guide/1.1/es/index>. [Accessed: 29-Mar-2015].
- [26] “YiiBooster - For an easier Yii application development.” [Online]. Available: <http://yiibooster.cleverttech.biz/site/index>. [Accessed: 29-Mar-2015].
- [27] “¿Cuál es la definición de Aplicación (informática)?” [Online]. Available: <http://www.alegsa.com.ar/Dic/aplicacion.php>. [Accessed: 29-Mar-2015].

- [28] “¿Cuál es la definición de Módulo (programación)?” [Online]. Available: <http://www.alegsa.com.ar/Dic/modulo.php>. [Accessed: 29-Mar-2015].
- [29] “Componentes - Monografias.com.” [Online]. Available: <http://www.monografias.com/Computacion/index.shtml>. [Accessed: 29-Mar-2015].
- [30] “¿Cuál es la definicion de Formulario?” [Online]. Available: <http://www.alegsa.com.ar/Dic/formulario.php>. [Accessed: 29-Mar-2015].
- [31] P. por D. Valenzuela, “Yii Framework en Español: Gii - Generación automática de código.” .
- [32] P. por D. Valenzuela, “Yii Framework en Español: Personalizando Gii.” .

Bibliografía

- [1] Jason Gilmore, W. *Beginning PHP and PostgreSQL 8: From Novice to Professional*. 2006.
- [2] A. V. Aho, *Compilers: Principles, Techniques, and Tools*. Addison-Wesley, 1987.
- [3] “Componentes - Monografias.com.” [Online]. Available: <http://www.monografias.com/Computacion/index.shtml>. [Accessed: 29-Mar-2015].
- [4] “¿Cuál es la definición de Aplicación (informática)?” [Online]. Available: <http://www.alegsa.com.ar/Dic/aplicacion.php>. [Accessed: 29-Mar-2015].
- [5] “¿Cuál es la definicion de Formulario?” [Online]. Available: <http://www.alegsa.com.ar/Dic/formulario.php>. [Accessed: 29-Mar-2015].
- [6] “¿Cuál es la definición de Módulo (programación)?” [Online]. Available: <http://www.alegsa.com.ar/Dic/modulo.php>. [Accessed: 29-Mar-2015].
- [7] *Curse on Program Analysis and Transformation*. .
- [8] J. de D. Bátis Paredes, *Desarrollo Orientado a Objetos con UML*. .
- [9] Cobo Romaní, J. C. “El concepto de tecnologías de la información. Benchmarking sobre las definiciones de las TIC en la sociedad del conocimiento.” 22-Sep-2009.
- [10] “Features.” [Online]. Available: <http://php.net/manual/en/features.php>.
- [11] “Function Reference.” [Online]. Available: <http://php.net/manual/en/funcref.php>.
- [12] Czarnecki, K. and U. W. Eisenecker, *Generative Programming*. 2000.
- [13] “Gestión de la información — Biblioteca Virtual de la UCF.” [Online]. Available: <http://biblioteca.ucf.edu.cu/biblioteca/articulos-descargados/sociocultural/Gestion%20de%20la%20informacion.pdf/view>. [Accessed: 28-Mar-2015].
- [14] “Getting Started.” [Online]. Available: <http://php.net/manual/en/getting-started.php>.
- [15] “Guia para escribir documentos HTML.pdf.” .
- [16] “HTML & CSS - W3C.” [Online]. Available: <file:///I:/HTML%20&%20CSS%20-%20W3C.htm#whathtml>. [Accessed: 26-Mar-2015].
- [17] Eguíluz Pérez., J. *Introduccion a CSS - Javier Eguiluz-Introduccion a CSS.pdf*. .
- [18] Gimeno, J. M. and J. L. González, “Introducción a Netbeans.” José Luis González.
- [19] “Introducción: Qué es Yii | The Definitive Guide to Yii | Yii PHP Framework.” [Online]. Available: <http://www.yiiframework.com/doc/guide/1.1/es/index>.

- [Accessed: 29-Mar-2015].
- [20] T. Powell and F. Schneider, "JavaScript 2.0 The Complete Reference, 2nd Edition."
- [21] "Language Reference." [Online]. Available: <http://www.php.net/manual/en/langref.php>.
- [22] "La programación web," in *PROGRAMACIÓN WEB JAVA*, lectores@mpediciones.com, p. 26.
- [23] Luzcila, "Los 11 Mejores Frameworks gratuitos para Aplicaciones Web," *El Baúl del Programador*. [Online]. Available: <http://elbauldelprogramador.com/los-10-mejores-frameworks-gratis-de-aplicaciones-web/>. [Accessed: 27-Mar-2015].
- [24] Hensgen, P. "Manual de Umbrello UML Modeller." 2001.
- [25] Fowler, M. *Patterns of Enterprise Application Architecture*. Addison Wesley, 2002.
- [26] phpMyAdmin contributors, "phpMyAdmin," *phpMyAdmin*. [Online]. Available: www.phpmyadmin.net. [Accessed: 29-Mar-2015].
- [27] Bergmann, S. *PHPUnit Manual*. 2015.
- [28] "Por qué utilizar RUP para desarrollar aplicaciones web." [Online]. Available: <http://www.eumed.net/libros-gratis/2009c/584/584.zip>. [Accessed: 28-Mar-2015].
- [29] "¿Qué es un 'framework'? | jordisan.net." [Online]. Available: http://www.addtoany.com/share_save?url=http%3A%2F%2Fjordisan.net%2Fblog%2F2006%2Fque-es-un-framework%2F&title=%C2%BFQu%C3%A9%20es%20un%20%E2%80%98framework%E2%80%99%3F&description=El%20t%C3%A9rmino%20framework%20se%20utiliza%20constantemente%20en%20el%20desarrollo%20de%20software%20C%20pero...%20%C2%BFsabr%C3%ADamos%20definir%20qu%C3%A9%20es%20un%20framework%20%3F%20He%20aqu%C3%AD%20una%20descripci%C3%B3n. [Accessed: 28-Mar-2015].
- [30] *Rdoc for Class: BasicObject (Ruby 1.9.3) - instance_eval*. .
- [31] "Scaffolding - KumbiaPHP Framework." [Online]. Available: <http://wiki.kumbiaphp.com/Discusión:Scaffolding>. [Accessed: 26-Mar-2015].
- [32] "Security." [Online]. Available: <http://www.php.net/manual/en/security.php>.
- [33] "Tags used in HTML." [Online]. Available: www.w3.org/History/19921103-

- hypertext/hypertext/WWW/MarkUp/Tags.html. [Accessed: 29-Mar-2015].
- [34] Q. Xue and X. Wei Zhuo, "The Definitive Guide to Yii 1.1." 2008.
- [35] A. Chlipala, "Ur: statically-typed metaprogramming with type-level record computation," *ACM SIGPLAN Notices*, vol. 45, no. 6, pp. 122–133, Jun. 2010.
- [36] "Visual Paradigm for UML (ME) (Paradigma Visual para UML (ME)) (Visual Paradigm for UML (ME)) por Visual Paradigm International Ltd. - reporte y descarga." May-2013.
- [37] "XAMPP Installers and Downloads for Apache Friends." [Online]. Available: <https://www.apachefriends.org/es/index.html>. [Accessed: 29-Mar-2015].
- [38] "YiiBooster - For an easier Yii application development." [Online]. Available: <http://yiibooster.cleverttech.biz/site/index>. [Accessed: 29-Mar-2015].
- [39] P. por D. Valenzuela, "Yii Framework en Español: Gii - Generación automática de código." .
- [40] P. por D. Valenzuela, "Yii Framework en Español: Personalizando Gii." .
- [41] Q. Xue and X. Wei Zhuo, "yii-guide-1.1.13-RC." 2008.

Anexos

Anexo: A Descripción de los casos de uso

Anexo A1 Gestionar entidad de cada tipo de dato del sistema

Nombre del caso de uso Gestionar entidad de cada tipo de dato del sistema	
Actores	Administrador
Propósito	Permitir al administrador Insertar, Listar, Buscar, Actualizar, Eliminar entidad de cada tipo de dato del sistema
Resumen	El caso de uso se inicia cuando el usuario desea insertar, actualizar, eliminar, listar y buscar las entidades de cada tipo de dato del sistema. Si desea eliminar o actualizar selecciona una de estas opciones del menú. En caso de actualizar el sistema muestra un formulario con los datos del tipo seleccionado permitiendo al usuario modificar los mismos, el sistema valida los datos insertados y de no ocurrir ningún error los datos son modificados mostrando un mensaje de confirmación, en caso de una situación contraria se muestra un mensaje de error. De ser eliminar se muestra un mensaje de confirmación en caso de aceptar se eliminan los datos de los tipos seleccionados y se confirma la acción realizada. En el caso de insertar se muestra un formulario solicitando los datos del nuevo tipo de entidad. Una vez especificado los datos se verifica que los mismos sean válidos y que no se esté duplicando el mismo, en caso de que no cumpla alguna de estas condiciones se muestra un mensaje de error en caso contrario se insertan los datos y se muestra un mensaje de confirmación. El caso de uso finaliza con la actualización de los datos.
Prototipos	Anexo B1
Precondiciones	El Administrador debe estar registrado en el sistema. En el caso de insertar deben existir los datos de entidad realizada.

	En el caso de eliminar o actualizar deben existir los datos de la entidad que deseada.
Poscondiciones	Se han insertado/eliminado/actualizado/listado/buscado entidad de cada tipo de dato del sistema

Anexo A2 Gestionar entidades relacionadas

Nombre del caso de uso Gestionar entidades relacionadas	
Actores	Administrador
Propósito	Permitir al administrador listar, agregar, eliminar o actualizar entidades relacionadas de forma uno a mucho desde la vista de la entidad en el lado uno.
Resumen	El caso de uso se inicia cuando el administrador desea listar, agregar, eliminar o actualizar entidades relacionadas de forma uno a mucho desde la vista de la entidad en el lado uno. Si desea eliminar o actualizar escoge la entidad de la lista y selecciona una de estas opciones en el menú, si es modificar se le muestra un formulario con los datos del seleccionado para que pueda modificarlos y de ser eliminar se le muestra un mensaje de confirmación y este será eliminado. En el caso de insertar se le muestra un formulario solicitando los datos de la entidad. Después de llenado se verifica que este no exista en la BD, de existir se muestra un mensaje de que ya existe, de lo contrario se inserta la nueva entidad.
Prototipos	Anexo B2
Precondiciones	El Administrador debe estar registrado en el sistema. En el caso de insertar deben existir los datos de entidad realizada. En el caso de eliminar o actualizar deben existir los datos de la entidad que deseada.
Poscondiciones	Se han insertado/eliminado/actualizado/listado/entidades relacionadas de forma uno a mucho desde la vista de la entidad en el lado uno.

Anexo A3Imprimir un listado de modelos

Nombre del caso de uso	Imprimir un listado de modelos
Actores	Administrador
Propósito	Permitir al administrador imprimir un listado de modelos
Resumen	El caso de uso se inicia cuando el administrador desea Imprimir un listado de modelos. Se selecciona el botónImprimir de la barra de menú y el sistema muestra la página para imprimir.
Prototipos	Anexo B3
Precondiciones	El Administrador debe estar registrado en el sistema.
Poscondiciones	-

Anexo A4Imprimir los detalles de una instancia de modelo

Nombre del caso de uso Imprimir los detalles de una instancia de modelo	
Actores	Administrador
Propósito	Permitir al administrador imprimir los detalles de una instancia de modelo
Resumen	El caso de uso se inicia cuando el usuario desea Imprimir los detalles de una distancia de modelo. Se selecciona el modelo que deseada y después selecciona el botónImprimir de la barra de menú y el sistema muestra la página para imprimir.
Prototipos	Anexo B4
Precondiciones	El Administrador debe estar registrado en el sistema.
Poscondiciones	-

Anexo A5Listar las entidades gestionables

Nombre del caso de uso	Listar las entidades gestionables
Actores	Administrador
Propósito	Permitir al administrador listar las entidades gestionables
Resumen	El caso de uso se inicia cuando usuario termina de autenticarse y el sistema muestra una lista de las entidades del sistema.
Prototipos	Anexo B5
Precondiciones	El Administrador debe estar registrado en el sistema Estén correctamente insertados los datos de las entidades
Poscondiciones	-

Anexo A6Exportar a PDF los detalle de una instancia de modelo

Nombre del caso de uso Exportar a PDF los detalle de una instancia de modelo	
Actores	Administrador
Propósito	Permitir al administrador exportar a pdf los detalle de una instancia de modelo
Resumen	El caso de uso se inicia cuando el administrador desea exportar a pdf los detalles de una instancia de modelo seleccionado. Se selecciona el modelo que deseada y después selecciona el botón exportar pdf de la barra de menú y el sistema se muestra como queda conformado el pdf.
Prototipos	Anexo B6
Precondiciones	Deben estar registrados los datos del modelo seleccionado.
Poscondiciones	-

Anexo A7Exportar a PDF un listado de instancias de modelo

Nombre del caso de uso	Exportar a PDF un listado de instancias de modelo
Actores	Administrador.
Propósito	Permitir al administrador exportar a pdf un listado de instancias de modelo.
Resumen	El caso de uso se inicia cuando el administrador desea exportar a pdf un listado de instancias de modelo. Se da clic izquierdo en la flecha abajo de botón download, luego se selecciona a pdf, el sistema se muestra como queda conformado el pdf. Terminando así el caso de uso.
Prototipos	Anexo B7
Precondiciones	El Administrador debe estar registrado en el sistema. Deben estar registrados los datos del modelo seleccionado.
Poscondiciones	-

Anexo A8Exportar a Excel un listado de instancias de modelo

Nombre del caso de uso	Exportar a Excel un listado de instancias de modelo
Actores	Administrador.
Propósito	Permitir al administrador exportar a Excel un listado de instancias de modelo.
Resumen	El caso de uso se inicia cuando el administrador desea exportar a Excel un listado de instancias de modelo. Se da clic izquierdo en la flecha abajo de botón download, luego se selecciona a Excel, el sistema se muestra como queda conformado el Excel. Terminando así el caso de uso.
Prototipos	Anexo B8
Precondiciones	El Administrador debe estar registrado en el sistema. Deben estar registrados los datos del modelo seleccionado.
Poscondiciones	-

Anexo A9Exportar a CSV un listado de instancias de modelo

Nombre del caso de uso	Exportar a CSV un listado de instancias de modelo
Actores	Administrador
Propósito	Permitir al administrador exportar a csv un listado de instancias de modelo
Resumen	El caso de uso se inicia cuando el administrador desea exportar a csvun listado de instancias de modelo. Se da clic izquierdo en la flecha abajo de botón download, luego se selecciona a Excel, el sistema se muestra como queda conformado el Excel. Terminando así el caso de uso.
Prototipos	Anexo B9
Precondiciones	El Administrador debe estar registrado en el sistema. Deben estar registrados los datos del modelo seleccionado.
Poscondiciones	

Anexo A10Consultar GoogleAnalytics

Nombre del caso de uso Consultar GoogleAnalytics	
Actores	Administrador
Propósito	Permitir al administrador consultar googleanalytics
Resumen	El caso de uso se inicia cuando el administrador desea consultar GoogleAnalytics. Se da clic izquierdo en el botónStatistics, el sistema se muestra la página de consultar googleanalytics. Terminando así el caso de uso.
Prototipos	Anexo B10
Precondiciones	El Administrador debe estar registrado en el sistema.
Poscondiciones	-

Anexo A11Autenticar

Nombre del caso de uso	Autenticar
Actores	Administrador
Propósito	Permitir al administradorautenticar
Resumen	El caso de uso se inicia cuando el usuario desea ingresar al sistema. Para ello debe introducir su usuario y contraseña, a continuación se chequea. Si los datos son correctos podrá acceder a las opciones del sistema que le corresponden, en el caso de que no lo sean se mostrará un mensaje de error, denegando el acceso.
Prototipos	Anexo B11
Precondiciones	El Administrador debe estar registrado en el sistema, teniendo un identificador y contraseña.
Poscondiciones	El Administrador accede a la información dentro de su sesión.

Anexo A12 Cerrar sesión

Nombre del caso de uso	Cerrar sesión
Actores	Administrador
Propósito	Permitir al administrador cerrar sesión
Resumen	El caso de uso se inicia cuando el Administrador desea salir del sistema. Se brinda la opción de cerrar la sesión del usuario que esté registrado.
Prototipos	Anexo B12
Precondiciones	El Administrador debe estar registrado en el sistema.
Poscondiciones	-

Anexo: B Prototipos de Interfaz

B1 Prototipo interfaz Gestionar entidad de cada tipo de dato del sistema

Listar entidad de cada tipo de dato del sistema

Displaying 1-3 of 3 results.

ID	Nombre	
1	Cuba	 
2	Laos	 
3	Cambodia	 

Insertar entidad de cada tipo de dato del sistema

[Home](#) / [Pais](#) / Create pais

Nombre *

Modificar entidad de cada tipo de dato del sistema

[Home](#) / [Pais](#) / Edit pais

Nombre *

Eliminar entidad de cada tipo de dato del sistema

Are you sure you want to delete this item?

Buscar entidad de cada tipo de dato del sistema

ID	Nombre

B2 Prototipo interfaz Gestionar entidades relacionadas

Listar entidad de cada tipo de dato del sistema

Pais 1

[Download as PDF](#) [Print](#)

ID	1
Nombre	Cuba

Estudiantes

ID	Nombre	Sexo	Biografia	Fecha Nacimiento	Grupo	Pais	Imagen	
4	Chanly	0	kjeseg	2015-05-11	1	1	43ed6687eff941994233add4cdaa74ca.jpg	  

Agregar entidad de cada tipo de dato del sistema

Nombre *

Sexo ☐ Hombre
☐ Mujer

Biografia



Fecha Nacimiento *



Imagen

No file selected.

Grupo *

Choose Grupo

Pais *

Choose Pais

Modificar entidad de cada tipo de dato del sistema

Nombre *

Sexo ☒ Hombre
☐ Mujer

Biografia

B *I* T  **</>**

kjsesg

Fecha Nacimiento *

2015-05-11

Imagen

Browse...

 No file selected.

Preview

Grupo * ▼

Pais * ▼

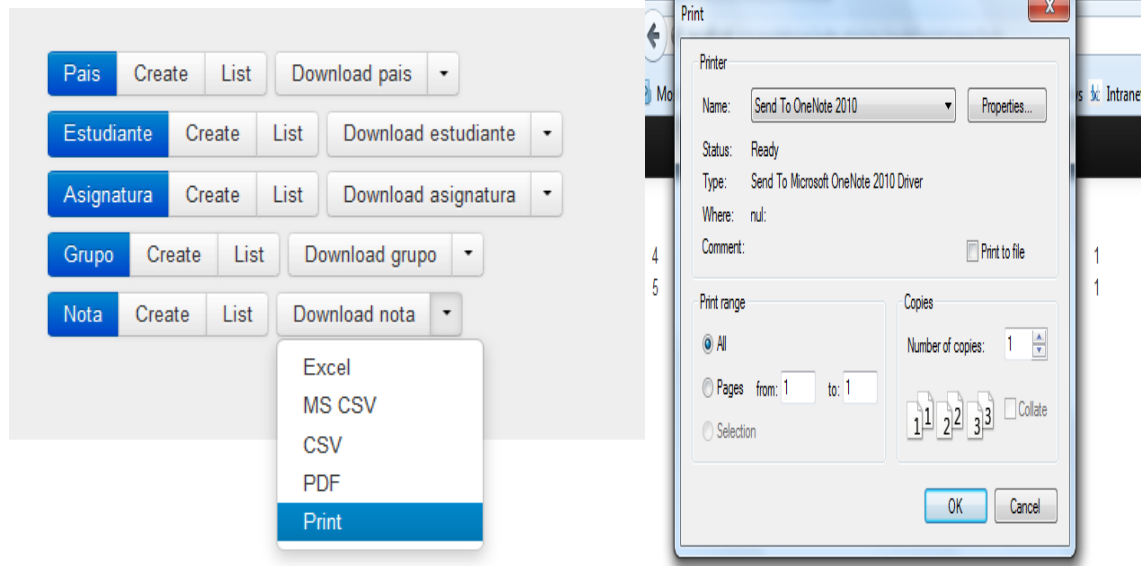
Save Save and add another Save and continue editing Delete

Eliminar entidad de cada tipo de dato del sistema

Are you sure you want to delete this item?

OK Cancel

B3 Prototipo interfaz Imprimir un listado de modelos



B4 Prototipo interfaz Imprimir los detalles de una instancia de modelo

Pais 1

Download as PDF

Print

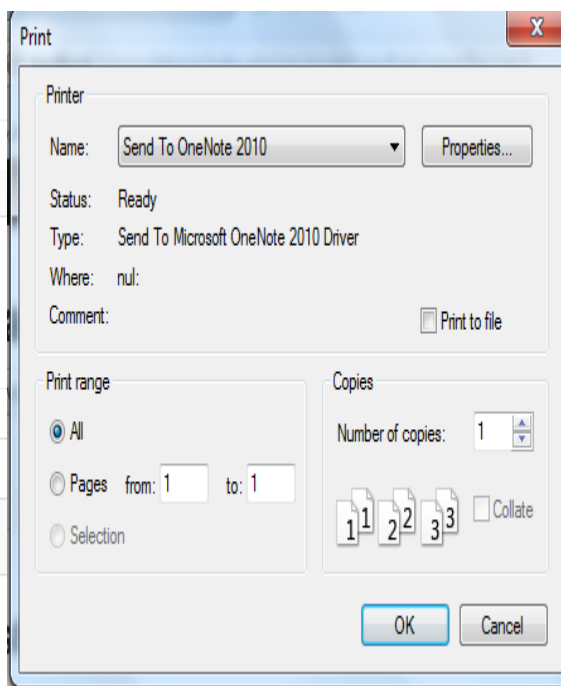
ID 1

Nombre Cuba

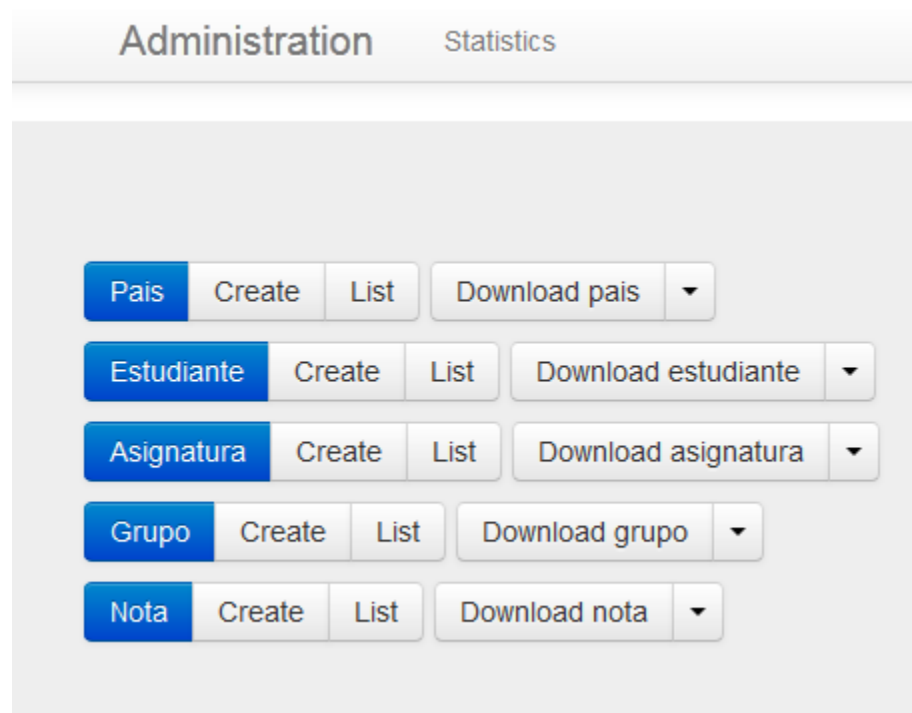
Estudiantes

ID	Nombre	Sexo
4	Chanly	0

Create



B5Prototipo interfaz Listar las entidades gestionables



B6 Prototipo interfaz Exportar a pdf los detalle de una instancia de modelo

Pais 1

Download as PDF

Print

ID 1

Nombre Cuba

ID 1

Nombre Cuba

Estudiantes

Estudiantes

ID Nombre Sexo Biografia Fecha Nacimiento Grupo Pais

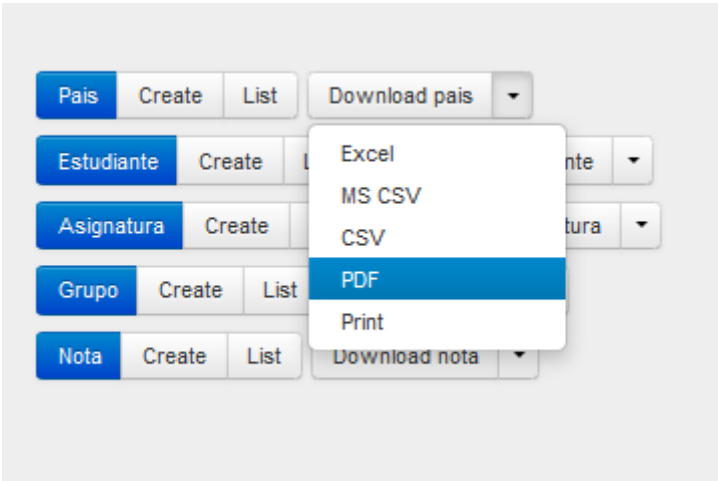
Imagen

ID	Nombre
4	Chanly

4 Chanly 0 kjsesg 2015-05-11 1 1 43ed6687eff941994233add4cdaa74ca.jpg

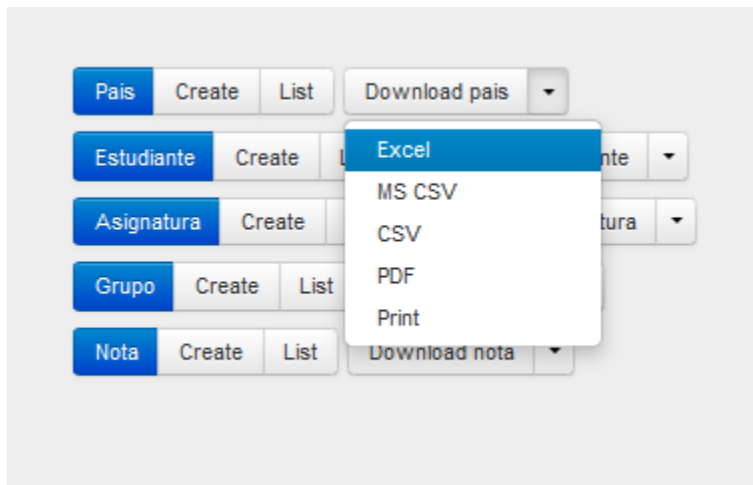
Create

B7 Prototipo interfaz Exportar a pdf un listado de instancias de modelo



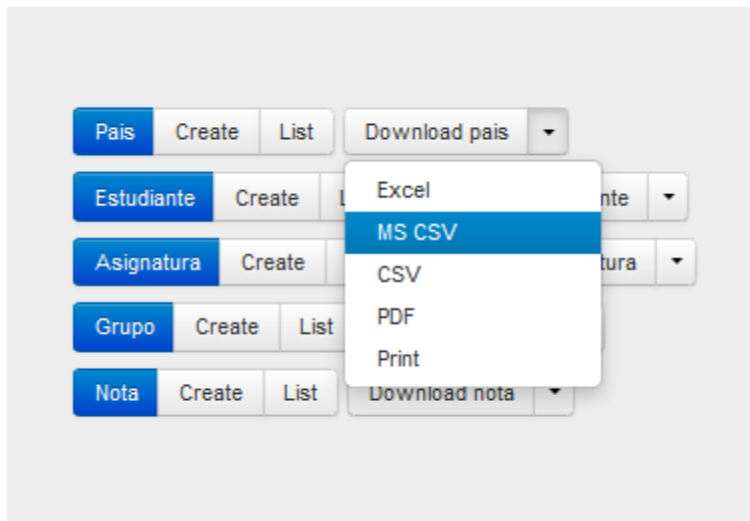
	ID	Nombre
1		Cuba
2		Laos
3		Cambodia

B8 Prototipo interfaz Exportar a Excel un listado de instancias de modelo



	1	2
1	ID	Nombre
2		1 Cuba
3		2 Laos
4		3 Cambodia
5		
6		
7		

B9 Prototipo interfaz Exportar a csv un listado de instancias de modelo



	1	2	3
1	ID	Nombre	
2		1 Cuba	
3		2 Laos	
4		3 Cambodia	
5			
6			

Google Analytics setup

Step 1

Connect with your Google Analytics account.

Connect

Step 2

Paste authorization code here:

Next

B11 Prototipo interfaz Autenticar

Please enter your username and password.

	xay		...	Login
---	----------------------------------	---	--------------------------------------	--------------------------------------

☐ Remember me next time

B12 Prototipo interfaz Cerrar sesión

