

Instalación y configuración de YII – Ejercicio

Participante:
Jesus Cardozo

Caracas, Venezuela Noviembre del 2015

Índice de contenido

Propósito.....	3
Instalación.....	3
Requisitos de tecnológico.....	3
Pasos para realizar la instalación	3
Crear la Base de Datos.....	4
Crear la aplicación.....	6
Configuración.....	6
Generar CRUD.....	7
Configurar Menu.....	10

Propósito

Realizar la instalación y configuración , seguido de un ejemplo de un programa en el cual se crea un CRUD, de una base de datos.

Instalación

Requisitos de tecnológico

Requisitos de Hardware:

- Se recomienda que la maquina
 - Procesador de 4.00 GHz
 - Memoria RAM – 1 Gb recomendado.
 - Disco duro al menos 20 Gb Mínimo.

Requisitos de Software:

- Ultima versión estable del Sistema operativo Debían o cualquier sistema linux.
- Servidor web Apache2, PHP5.4 (apache2,libapache2-mod-php5,php5,php5-pgsql).
- Gestor de base de datos Postgres en su ultima version estable.
- Aplicación pgadmin3.
- Aplicación Yii en su version 1.16.

Pasos para realizar la instalación

Descargamos, los archivos de yii en su version estable 1.16. de la siguiente manera.

```
jesus@jc:~$ wget https://github.com/yiisoft/yii/releases/download/1.1.16/yii-1.1.16.bca042.tar.gz
```

Después de realizar la descarga se sigue descomprimir.

```
jesus@jc:~$ tar -vxf yii-1.1.16.bca042.tar.gz
```

Se necesita ser super usuario para poder mover la carpeta de yii a /usr/share/

```
jesus@jc:~$ su
```

Contraseña:

Luego se mueve la carpeta de framework a la carpeta /usr/share/

```
root@jc:/home/jesus# mv yii-1.1.16.bca042/framework/ /usr/share/yii
```

Listo para crear aplicaciones utilizando el framework Yii.

Crear la Base de Datos

Para crear la base de datos es necesario que este instalado el gestor de bases de datos relacional postgresql.

Hacer login con el usuario postgres:

```
jesus@jc:/usr/share/yii$ su postgres
```

Contraseña:

despues de entrar con el usuario postgres se procede a entrar en la consola de comandos SQL de postgres,

```
postgres@jc:/usr/share/yii$ psql
```

Digite «help» para obtener ayuda.

```
postgres=# CREATE DATABASE ip WITH ENCODING='UTF8' OWNER=postgres;
```

Luego salimos de la base de datos postgres y entramos a la base de datos creada

```
postgres=# \q
```

```
postgres@jc:/usr/share/yii$ psql ip
```

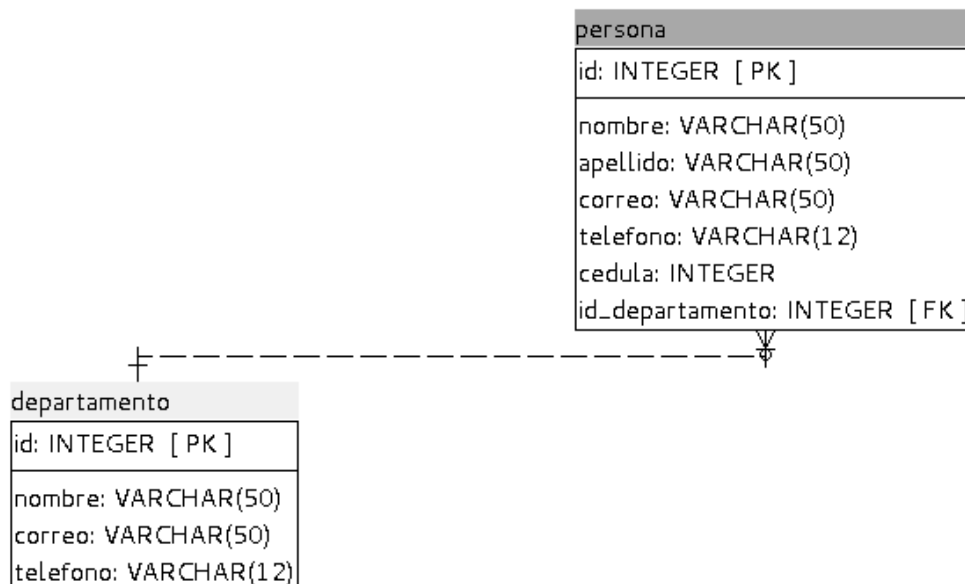
Crear las tablas con relaciones escribimos los comandos SQL:

- Crear tabla departamento

```
ip=# CREATE TABLE departamento(id serial NOT NULL,nombre character  
varying(50),correo character varying(50), telefono character  
varying(12),CONSTRAINT departamento_pkey PRIMARY KEY (id ));
```

- Crear tabla persona

```
ip=# CREATE TABLE persona(id serial NOT NULL,nombre character  
varying(50),apellido character varying(50),correo character  
varying(50),telefono character varying(12),cedula integer,id_departamento  
integer,direccion_ip character varying(15), CONSTRAINT persona_pkey PRIMARY  
KEY (id ), CONSTRAINT persona_id_departamento_fkey FOREIGN KEY  
(id_departamento) REFERENCES departamento (id) MATCH SIMPLE ON UPDATE  
NO ACTION ON DELETE NO ACTION);
```



Crear la aplicación

Para crear la aplicación, necesitamos estar ubicados en la carpeta /usr/share/yii y buscar el archivo yiic ejemplo:

```
jesus@jc:/usr/share/yii$ php yiic webapp /var/www/sistema_yii
```

Después de crear la aplicación se puede abrir una navegador web y se coloca en la URL: http://localhost/sistema_yii

Asignar la aplicación al usuario www-data para poder realizar las operaciones CRUD

```
root@jc:/usr/share/yii# chown -R www-data:www-data /var/www/sistema_yii
```

Configuración

Para configurar la administración del framework, hay que editar el siguiente archivo

```
jesus@jc:/usr/share/yii$ nano  
/var/www/sistemajesus2/protected/config/main.php
```

Se descomenta la siguiente linea y se cambia el password ejemplo:

```
'modules'=>array(  
    // uncomment the following to enable the Gii tool  
    'gii'=>array(  
        'class'=>'system.gii.GiiModule',  
        'password'=>'admin',  
        // If removed, Gii defaults to localhost only. Edit car$  
        'ipFilters'=>array('127.0.0.1','::1'),  
    ),  
),
```

),

para configurar la base de datos es necesario editar el siguiente archivo

```
jesus@jc:/usr/share/yii$ nano  
/var/www/sistemajesus2/protected/config/database.php
```

Se edita el siguiente bloque de código que estamos a continuación

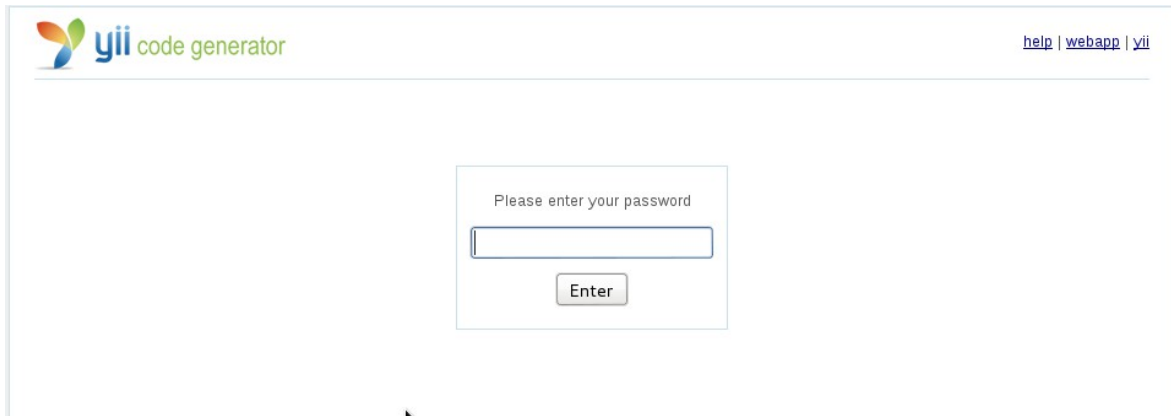
```
return array(  
    'connectionString' => 'sqlite:'.dirname(__FILE__).'../data/testdrive.d$  
    // uncomment the following lines to use a MySQL database  
  
    'connectionString' => 'pgsql:host=localhost;dbname=ip',  
    'emulatePrepare' => true,  
    'username' => 'postgres',  
    'password' => 'postgresqldb',  
    'charset' => 'utf8',  
);
```

Generar CRUD

Para generar los crud en hay que ingresar en la aplicación tipo administrador framework.

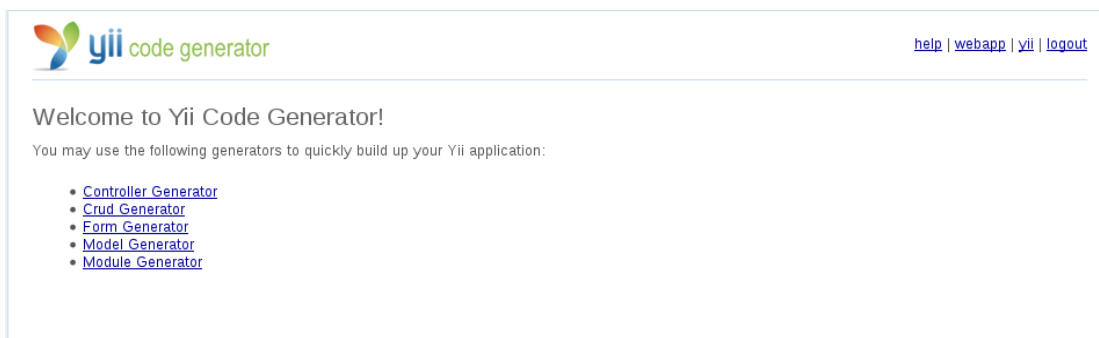
Abrir navegador web y colocar el la URL: http://localhost/sistema_yii/index.php?r=gii

se coloca la contraseña configurada que es: admin.



The screenshot shows the login interface of the Yii Code Generator. At the top left is the logo with the text "yii code generator". At the top right are links for "help", "webapp", and "yii". In the center, there is a box with the text "Please enter your password" above a text input field. Below the input field is a button labeled "Enter".

Selecciona la opción Model Generator para crear el modelo de la tabla que queremos realizar el CRUD.




The screenshot shows the welcome page of the Yii Code Generator. At the top left is the logo with the text "yii code generator". At the top right are links for "help", "webapp", "yii", and "logout". Below the header, it says "Welcome to Yii Code Generator!" followed by the text "You may use the following generators to quickly build up your Yii application:". Below this text is a bulleted list of links: "Controller Generator", "Crud Generator", "Form Generator", "Model Generator", and "Module Generator".

en la opción table Name, escribes el nombre de la tabla que quieres modelar.

Luego seleccionas la opción Preview.

Después interfaz muestra los archivos y pregunta si deseas generar.

[help](#) | [webapp](#) | [yii](#) | [logout](#)

Generators
[Controller Generator](#)
[Crud Generator](#)
[Form Generator](#)
[Model Generator](#)
[Module Generator](#)

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 *
persona

Model Class *
Persona

Base Class *
 CActiveRecord

Model Path *
application.models

Build Relations
☒

Use Column Comments as Attribute Labels
☐

Code Template *
default (/var/www/yii/framework/gii/generators/model/templates/default)

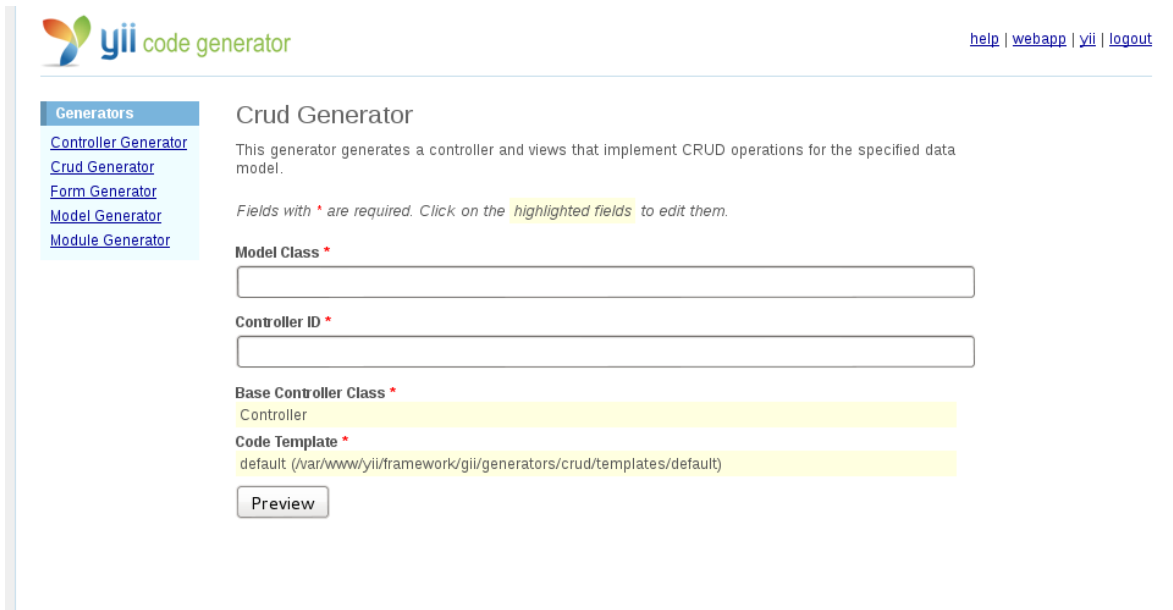
Preview

The code has been generated successfully.

```
Generating code using template "/var/www/yii/framework/gii/generators/model/templates/default"
skipped models/Persona.php
done!
```

Potenciado por [Yii Framework](#).
A product of [Yii Software LLC](#).

Para Generar los CRUD, seleccionar la Opción *CRUD Generator*



The screenshot shows the 'yii code generator' interface. On the left, a sidebar lists 'Generators' with links for 'Controller Generator', 'Crud Generator', 'Form Generator', 'Model Generator', and 'Module Generator'. The 'Crud Generator' is selected. The main area is titled 'Crud Generator' and contains the following fields and options:

- Model Class ***: An empty text input field.
- Controller ID ***: An empty text input field.
- Base Controller Class ***: A dropdown menu with 'Controller' selected.
- Code Template ***: A dropdown menu with 'default (/var/www/yii/framework/gii/generators/crud/templates/default)' selected.
- Preview**: A button at the bottom.

Below the fields, there is a note: 'Fields with * are required. Click on the highlighted fields to edit them.'

Escribir en el campo Model Class, el modelo generado por el modulo Model Generator, ejemplo: Persona.

Dar click a la opcion Preview , y generar los archivos, y listo.

Configurar Menu

Hay que configurar el menú para que salga en la interfaz principal.

Editamos el archivo

```
jesus@jc:/usr/share/yii$ nano  
/var/www/sistemajesus2/protected/views/layouts/main.php
```

Agregamos la link del modulo creado:

```
<div id="mainmenu">  
    <?php $this->widget('zii.widgets.CMenu',array(  
        'items'=>array(  
            array('label'=>'Home', 'url'=>array('/site/index')),  
            array('label'=>'About', 'url'=>array('/site/page', 'view'=>'about')),  
            array('label'=>'Contact', 'url'=>array('/site/contact')),
```

```
        array('label'=>'Persona', 'url'=>array('/persona')),  
        array('label'=>'Login', 'url'=>array('/site/login'), 'visible'=>Yii::app()->user->isGuest),  
        array('label'=>'Logout ('.Yii::app()->user->name.)', 'url'=>array('/site/logout'),  
'visible'=>!Yii::app()->user->isGuest)  
    ),  
    ); ?>  
</div><!-- mainmenu -->
```