

YII 1 for PHP Development



JUNNEL T. TORATE
Software Developer

Objectives

- Basic understanding of the structure, concepts, and help to promote rapid application development (RAD), which saves time, helps build more stable applications, and reduces the amount of repetitive coding for developers.
- Take away performing tedious coding tasks using pre-built modules and spend time on developing the actual application rather than re-building the foundation.
- Awareness of using modern PHP frameworks to improve knowledge and skills to easily adapt other frameworks more flexible.



Course Outline

- Introduction to PHP: The PHP Environment, Software and Frameworks
- PHP Programming with Yii 1
- Web UI using Bootstrap
- Database Connection
- Generating Code with Gii
- Git: Clone, Add, Commit and Push
- Exercise: Develop Web Application



Introduction

PHP (Hypertext Preprocessor)

- Server-side scripting language
- Used to manage dynamic content, databases, session tracking, even build entire e-commerce sites
- Performs system functions like create, open, read and write files
- Handle pages to view, add, edit and delete data from the database

```
<html>
  <head>
    <title>Hello World</title>
  </head>
  <body>

    <?php echo "Hello, World!"; ?>

  </body>
</html>
```

Result:

Hello, World!

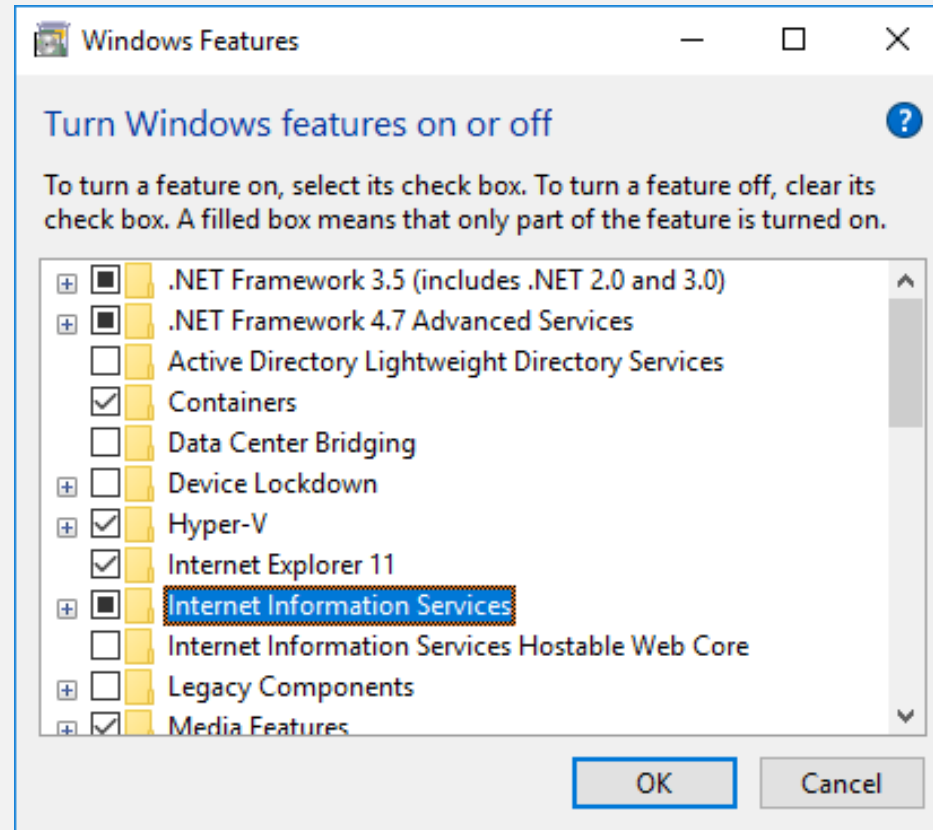
Web Server

- The basic objective of the web server is to store, process and deliver web pages to the users.
- Intercommunication is done through **Hypertext Transfer Protocol** (HTTP)

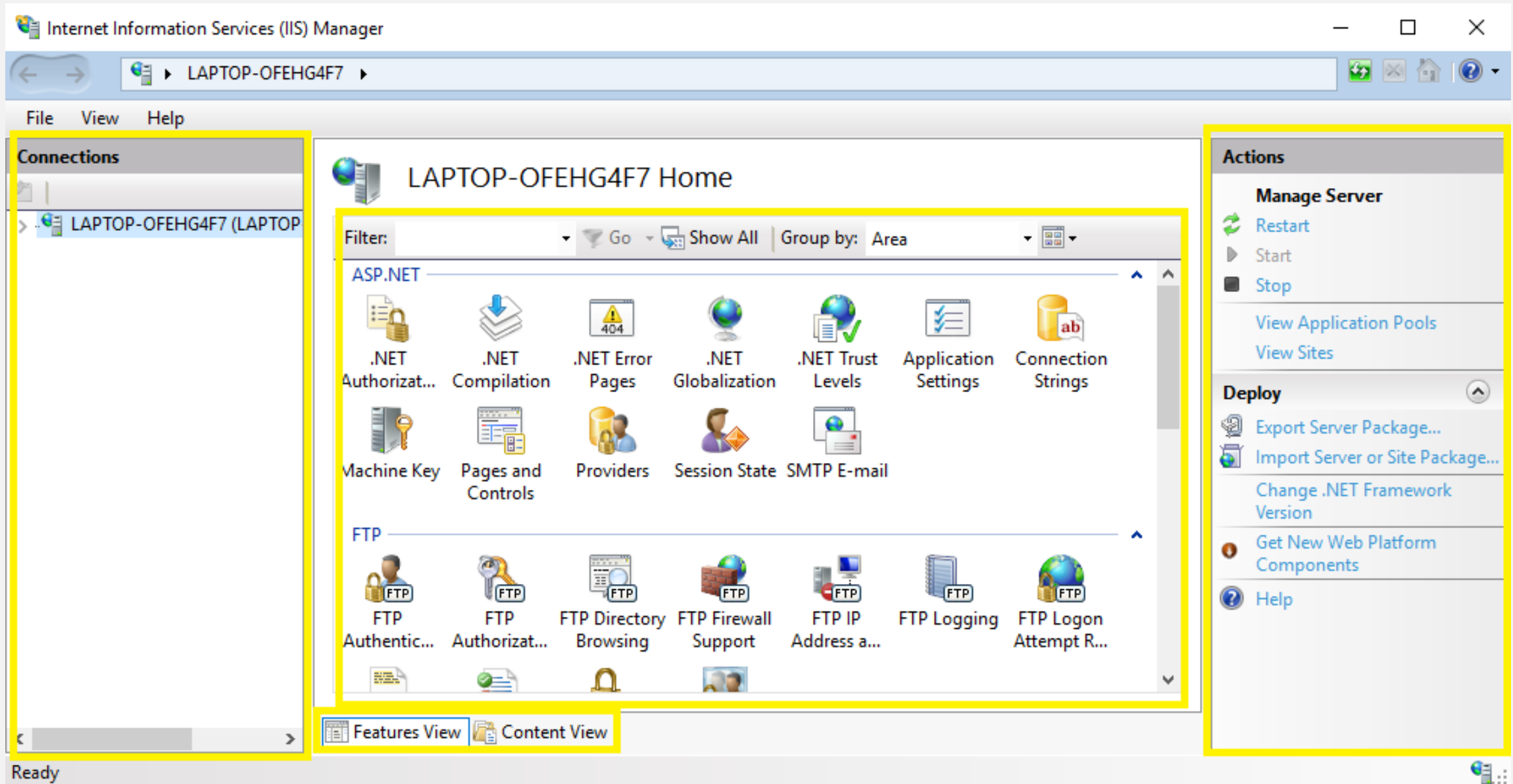


Microsoft IIS

- **Internet Information Services**
- IIS is a web server software package designed for Windows.
- It is used for hosting websites and other content on the Web.

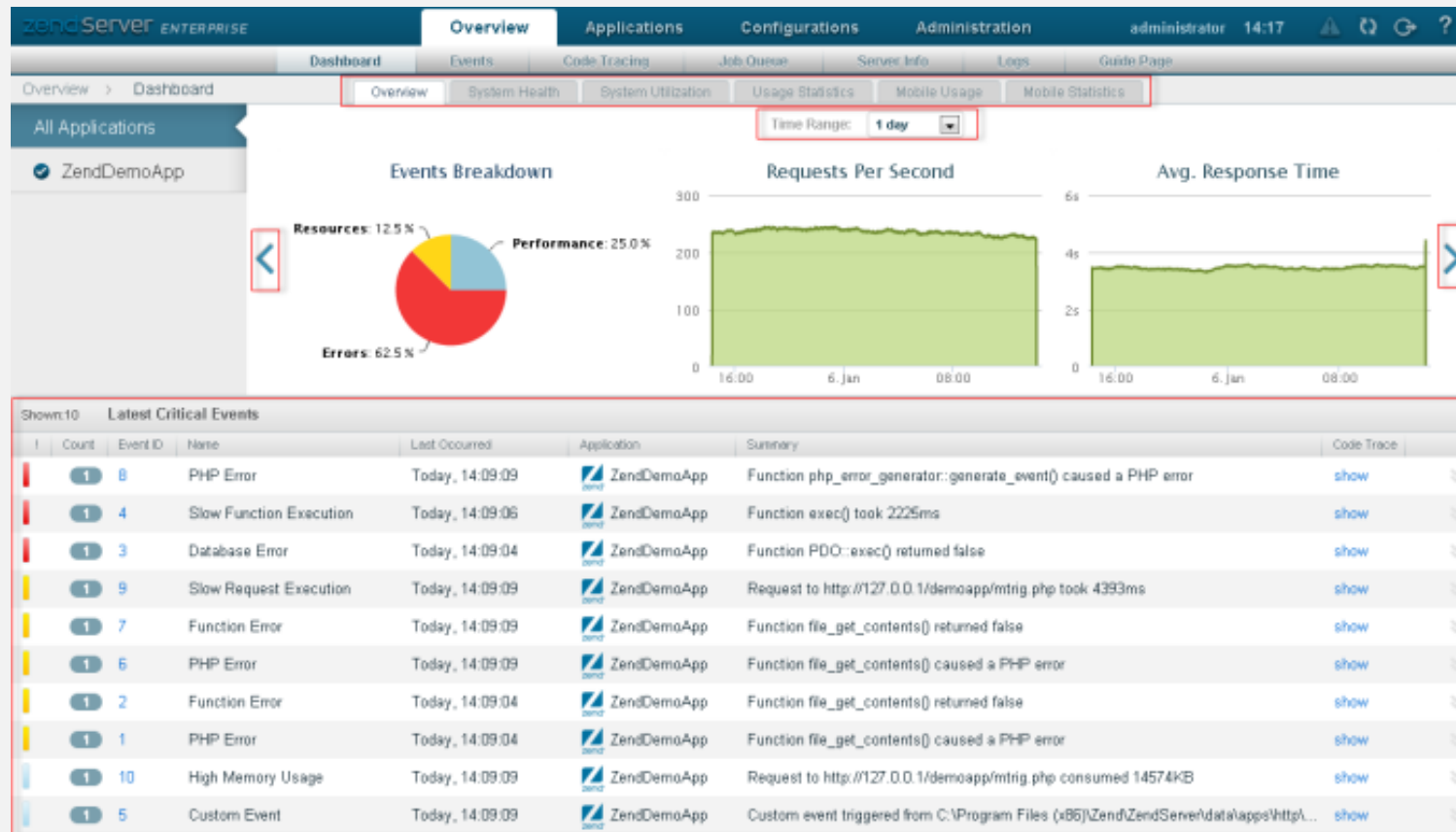


IIS Web Server: Internet Information Services



Zend Server

- **Zend Server** is a complete and certified PHP distribution stack fully maintained and supported by Zend Technologies.
- It includes over 80 PHP extensions and supports Apache, NGINX and IIS Web servers.



Code Editors and IDE

Code Editor

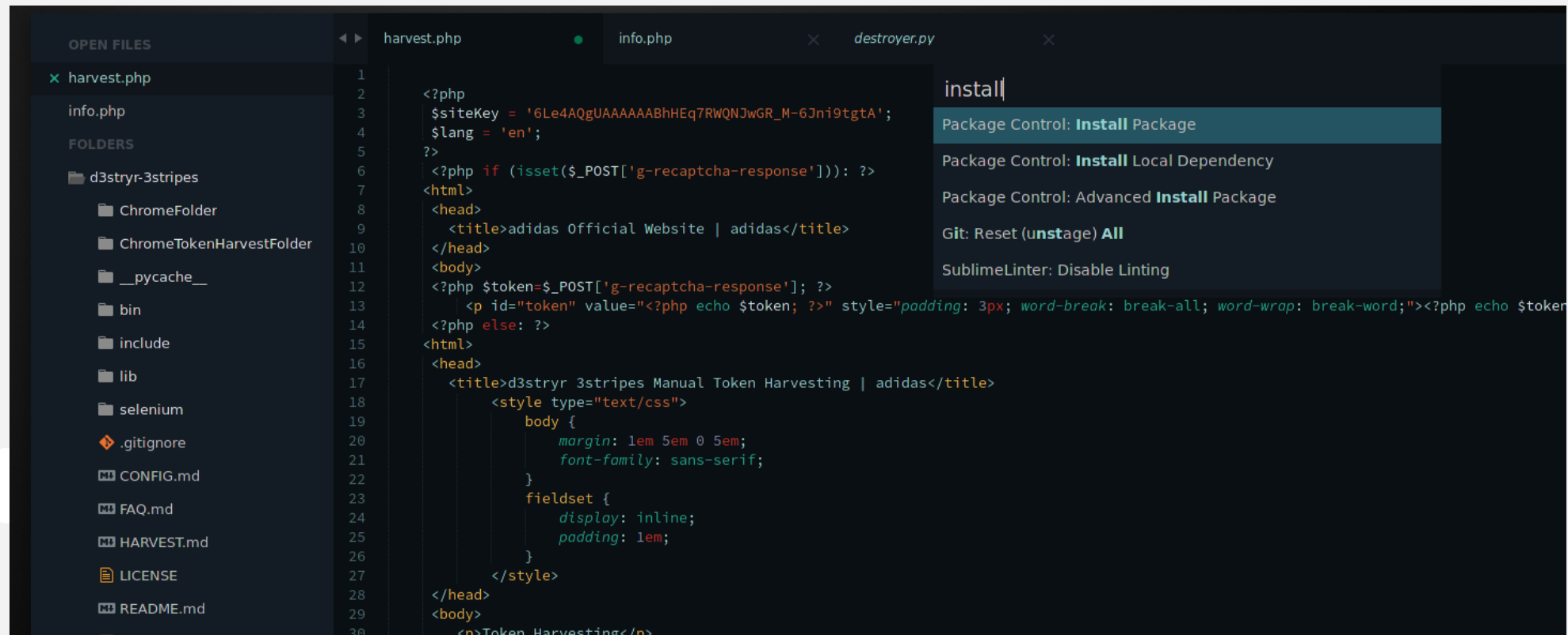
- Small piece of software like a text editor.
- It highlights special keywords, give suggestions, automatic indentation features and other has a integrated terminal.

IDE (Integrated Development Environment)

- Has more sophisticated complex functionalities to make the developer life more easy.
- It supports debugging, build automation tools, Git plug-ins, code generators, managing project history and etc.

Sublime Text

- **Sublime Text** is a fast and lightweight cross-platform code editor packed with features and plug-ins.
- With its **“Goto Anything”** keyboard shortcut, you can easily jump to any symbol or line of code within your program.



Visual Studio Code

- **Visual Studio Code** is a cross-platform (Linux, Mac OS, Windows) editor that can be extended with plug-ins.
 - IntelliSense
 - Debugging
 - Built-in Git

```
gpf_booking_calendar.php — gpf_booking_calendar — Visual Studio Code
File Edit Selection View Go Debug Tasks Help

EXPLORER
- public.php
  OPEN EDITORS
  gpf_booking_calendar-p...
  gpf_booking_calendar-a...
  gpf_booking_calendar.php
  gpf_booking_calendar-a...
  GPF-BOOKING-CALENDAR
  js
  # admin-styles.css
  gpf_booking_calendar-asset...
  gpf_booking_calendar-asset...
  gpf_booking_calendar-asset...
  gpf_booking_calendar-asset...
  gpf_booking_calendar-book...
  gpf_booking_calendar-book...
  gpf_booking_calendar-publi...
  gpf_booking_calendar.php
  # styles.css

254 'Asset types',
255 'Asset types',
256 'manage_options',
257 'gpfbc-asset-types', // slug
258 'gpf_booking_calendar_asset_types'
259 );
260 }
261
262 add_action('admin_menu', 'gpf_booking_calendar_top_menu');
263 add_action('admin_menu', 'gpf_booking_calendar_assets_menu');
264
265 function gpf_booking_calendar_activate()
266 {
267     global $wpdb;
268
269     $table_name = $wpdb->prefix . '_gpfbc_assets';
270     $charset_collate = $wpdb->get_charset_collate();
271
272     $sql = "CREATE TABLE $table_name (
273         id mediumint(9) NOT NULL AUTO_INCREMENT,
274         name varchar(30) NOT NULL,
275         released varchar(25) NOT NULL,
276         asset_type_id int(11) NOT NULL,
277         PRIMARY KEY (id)
278     ) $charset_collate;";
279
280     require_once( ABSPATH . 'wp-admin/includes/upgrade.php' );
281     dbDelta( $sql );
282
283     $table_name = $wpdb->prefix . '_gpfbc_bookings';
284     $sql = "CREATE TABLE $table_name (
285         id mediumint(9) NOT NULL AUTO_INCREMENT,
286         customer varchar(100) NOT NULL,
```

PHP Frameworks

Advantages

- Speed up custom web application development
- Simplify web application maintenance
- Work with databases more efficiently
- Automate common web development tasks
- Protect websites from targeted security attacks
- Perform unit testing efficiently
- No need to increase web development cost



PHP Frameworks

Disadvantages

- Programmers need to learn PHP frameworks instead of PHP Basics
- Quality of PHP frameworks differs
- Lack of option to modify core behavior
- Affect speed and performance of websites



Yii Framework

- **Yii** is an open source, object-oriented, component-based MVC PHP web application framework.
- Pronounced as “**Yee**” and in Chinese it means “**simple and evolutionary**” and it can be an acronym for “**Yes It Is!**”.
- Light-weight and equipped with caching mechanisms, suited for high-traffic applications, such as portals, forums, content management systems (CMS), e-commerce systems, etc.

Yii 1.1 Installation

- Download Yii Framework from this [link](#).
- Unpack the Yii release file to a Web-accessible directory.

Requirements:

After installing Yii, to verify that your server satisfies Yii's requirements, you can do so by accessing the **requirement checker** script via the following URL in a Web browser: <http://hostname/path/to/yii/requirements/index.php>

e.g. <http://localhost/yii-app/yii/requirements/index.php>



Creating Your First Yii Application

- Using **yiic** (command line tool) to create a new Yii application
- Assume that **YiiRoot** is the directory where Yii is installed, and **WebRoot** is the document root of our Web server

Run **yiic** on the command line as follows:

% **YiiRoot/framework/yiic webapp WebRoot/yii-app**

```
C:\WINDOWS\system32\cmd.exe  
  
C:\>C:\inetpub\wwwroot\yii-app\yii\framework\yiic webapp C:\inetpub\wwwroot\yii-app
```


Creating Your First Yii Application

- This will create a skeleton Yii application under the directory WebRoot/yii-app

```
Your application has been created successfully under C:\inetpub\wwwroot\yii-app.
```

By accessing the following URL in a Web browser:

<http://localhost/yii-app/index.php>

The application has four default pages: the homepage, the about page, the contact page and the login page.

The contact page displays a contact form that users can fill in to submit their inquiries to the webmaster, and the login page allows users to be authenticated before accessing privileged contents.



Database Connection

- WebRoot/yii-app/protected/config/database.php

```
// This is the database connection configuration.
return array(
    'connectionString' => 'sqlite:'.dirname(__FILE__).'/../data/testdrive.db',
    // uncomment the following lines to use a MySQL database
    /*
    'connectionString' => 'mysql:host=localhost;dbname=testdrive',
    'emulatePrepare' => true,
    'username' => 'root',
    'password' => '',
    'charset' => 'utf8',
    */
);
```

- The above code is the default database connection
- Instructs Yii that the application should connect to the SQLite database:

<WebRoot/yii-app/protected/data/testdrive.db>

Connecting to MySQL Database

- In MySQL GUI, create a database named: **testdrive**
- Select that database and execute the SQL file found in this directory:

[WebRoot/yii-app/protected/data/schema.mysql.sql](#)

The SQL schema contains only a single table named **tbl_user** with the initial data to be inserted.

```
CREATE TABLE tbl_user (  
    id INTEGER NOT NULL PRIMARY KEY AUTO_INCREMENT,  
    username VARCHAR(128) NOT NULL,  
    password VARCHAR(128) NOT NULL,  
    email VARCHAR(128) NOT NULL  
);  
  
INSERT INTO tbl_user (username, password, email) VALUES ('test1', 'pass1', 'test1@example.com');  
INSERT INTO tbl_user (username, password, email) VALUES ('test2', 'pass2', 'test2@example.com');  
INSERT INTO tbl_user (username, password, email) VALUES ('test3', 'pass3', 'test3@example.com');  
.....
```

Connecting to MySQL Database

- Edit the file: [WebRoot/yii-app/protected/config/database.php](#)
- In editor, comment the following line of code:

```
return array(  
    'connectionString' => 'sqlite:'.dirname(__FILE__).'/../data/testdrive.db',  
    .....  
);
```

- Uncomment the following lines of code to connect in MySQL Database:

```
return array(  
    .....  
    'connectionString' => 'mysql:host=localhost;dbname=testdrive',  
    'emulatePrepare' => true,  
    'username' => 'root',  
    'password' => '',  
    'charset' => 'utf8',  
    .....  
);
```

Gii

- A powerful Web-based code generator

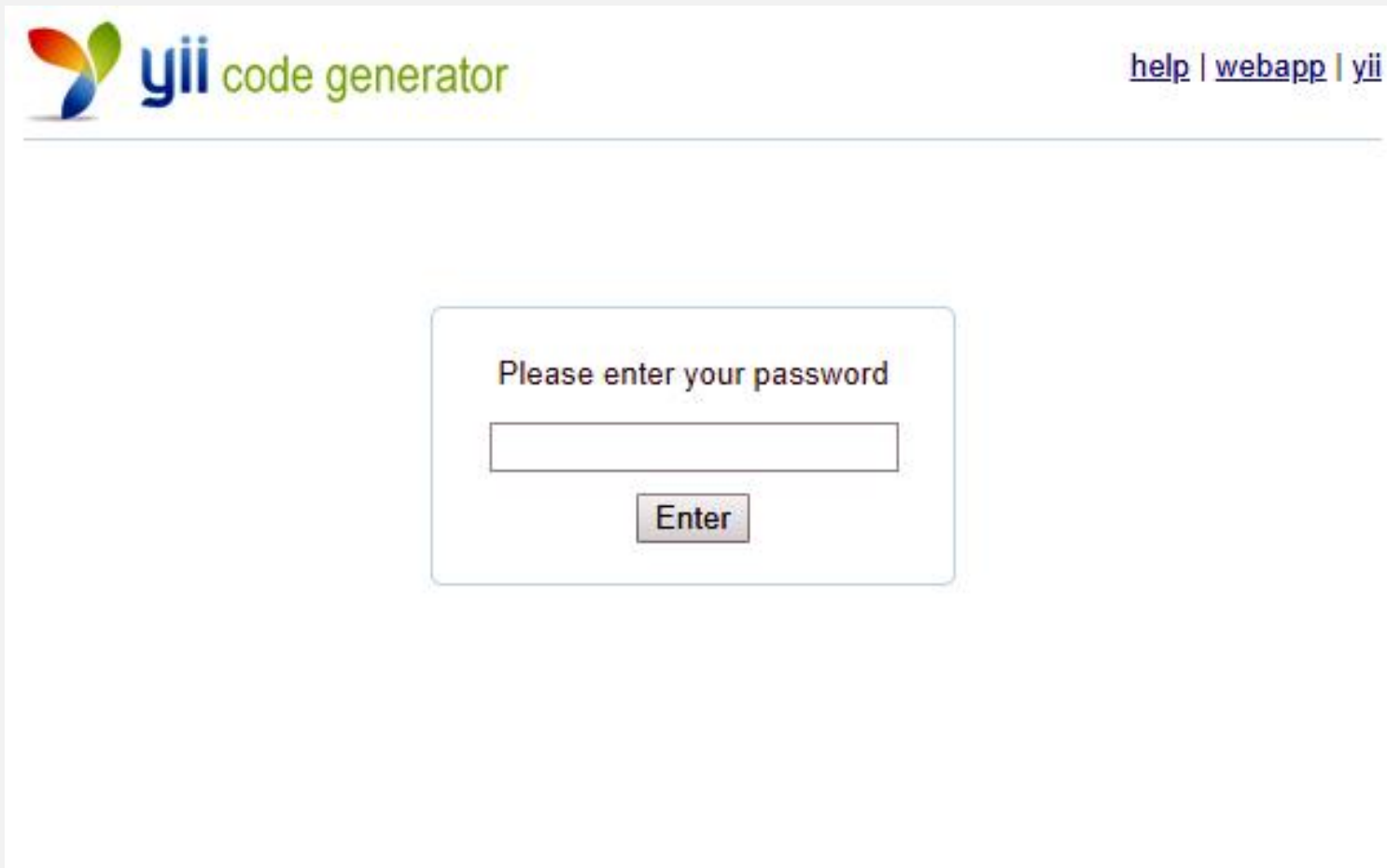
Configuring Gii:

- Edit the file: [WebRoot/yii-app/protected/config/main.php](#)
- Uncomment the Gii module and update the password

```
.....  
'import'=>array(  
    'application.models.*',  
    'application.components.*',  
)  
,  
'modules'=>array(  
    // uncomment the following to enable the Gii tool  
    'gii'=>array(  
        'class'=>'system.gii.GiiModule',  
        'password'=>'Enter Your Password Here',  
        // If removed, Gii defaults to localhost only. Edit carefully to taste.  
        'ipFilters'=>array('127.0.0.1','::1'),  
    ),  
)  
,  
.....
```

Gii Page

- Visit the URL <http://localhost/yii-app/index.php?r=gii>
- Login page will be prompted for a password



The screenshot displays the 'yii code generator' interface. At the top left is the Yii logo (a stylized 'Y' with red, green, and blue segments) followed by the text 'yii code generator'. At the top right are links for 'help', 'webapp', and 'yii'. A horizontal line separates the header from the main content area. In the center, a rounded rectangular box contains the text 'Please enter your password', an empty password input field, and an 'Enter' button below it.

Generating the User Model

- After login, click on the link [Model Generator](#).
- This will bring us to the following model generation page.

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 *

Model Class *

Base Class *
CActiveRecord

Model Path *
application.models

Build Relations
☒

Use Column Comments as Attribute Labels
☐

Code Template *
default (C:\inetpub\wwwroot\yii-app\yii\framework\gii\generators\model\templates\default)

Generating the User Model

Enter the following values to generate model file:

- Table Name: `tbl_user`
- Model Class: `User`
- Press the `Preview` button

This will show us the new code file to be generated

- Press the `Generate` button

A new file named `User.php` will be generated under `protected/models`

This User model class allows us to talk to the underlying database `tbl_user` table in an object-oriented fashion.



Generating User CRUD Code

- In Gii page, click on the link [Crud Generator](#).
- Generate the code that implements the CRUD operations

Crud Generator

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

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

Model Class *

Controller ID *

Base Controller Class *

Controller

Code Template *

default (C:\inetpub\wwwroot\yii-app\yii\framework\gii\generators\crud\templates\default)

Preview

Generating User CRUD Code

Enter the following values to generate CRUD files:

- Model Class: **User**
- Controller ID: **user** (in lower case)
- Press the **Preview** button

This will show us the new code files to be generated

- Press the **Generate** button

A new file named **UserController.php** will be generated under **protected/controllers**

And, view files under **protected/views/user** used for the front-end



Accessing User CRUD Pages

- Visit the URL <http://localhost/yii-app/index.php?r=user>
- This will display a list of user entries in the `tbl_user` table.

Pages Access Rules

- [Allow all users \(*\)](#) - List User, View User
- [Allow authenticated user \(@\)](#) - Create User, Update User
- [Allow admin user \(admin\)](#) - Manage User, Delete User



Controller Access Control Rules

- WebRoot/yii-app/protected/controllers/UserController.php

```
/**
 * Specifies the access control rules.
 * This method is used by the 'accessControl' filter.
 * @return array access control rules
 */
public function accessRules()
{
    return array(
        array('allow', // allow all users to perform 'index' and 'view' actions
            'actions'=>array('index','view'),
            'users'=>array('*'),
        ),
        array('allow', // allow authenticated user to perform 'create' and 'update' actions
            'actions'=>array('create','update'),
            'users'=>array('@'),
        ),
        array('allow', // allow admin user to perform 'admin' and 'delete' actions
            'actions'=>array('admin','delete'),
            'users'=>array('admin'),
        ),
        array('deny', // deny all users
            'users'=>array('*'),
        ),
    );
}
```

Bootstrap

Bootstrap is the most popular **CSS Framework** for developing responsive and mobile-first websites.

Advantages

- Ensure cross-browser compatibility
- Allow website to adapt in various screen sizes – mobile and desktop
- Add consistency to design and code between projects and between developers

Disadvantage

- Lot of requirements to style overrides or rewriting the files tends for customization of design used in Bootstrap.

Install Bootstrap Extension in Yii

- Download bootstrap extension from this [link](#).
- Unpack the file archive to [WebRoot/yii-app/protected/extensions](#).

Configuring Extension:

- Edit the file: [WebRoot/yii-app/protected/config/main.php](#)

In editor, add the following line of code before starting of array:

```
Yii::setPathOfAlias('bootstrap', dirname(__FILE__).'../extensions/bootstrap');
```

```
.....  
Yii::setPathOfAlias('bootstrap', dirname(__FILE__).'../extensions/bootstrap');  
  
return array(  
.....
```

Install Bootstrap Extension in Yii

- Edit the file: [WebRoot/yii-app/protected/config/main.php](#)

In editor, add the following line of code:

```
return array(  
    'basePath' => dirname(__FILE__).DIRECTORY_SEPARATOR.'..',  
    'name' => 'My Web Application',  
    'theme' => 'bootstrap',  
    // preloading 'log' component  
    'preload' => array('log'),  
    // ...  
);
```

In editor, add the following line of codes inside Gii module:

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

Install Bootstrap Extension in Yii

- Edit the file: [WebRoot/yii-app/protected/config/main.php](#)

In editor, add the following line of codes inside the components:

```
// application components
'components'=>array(
    .....
    'bootstrap'=>array(
        'class'=>'bootstrap.components.Bootstrap',
    ),
    .....
),
```


Install Bootstrap Extension in Yii

- Edit the file: [WebRoot/yii-app/protected/extensions/bootstrap/components/Bootstrap.php](#)

In editor, add the following line of codes inside the class:

```
.....  
public function init() {  
    $this->registerAllCss();  
    $this->registerJs();  
    parent::init();  
}  
.....
```

Install Bootstrap Extension in Yii

- Edit the file: [WebRoot/yii-app/protected/views/layout/main.php](#)

In editor, add the following line of code before the <html> tag:

```
<?php echo Yii::app()->bootstrap->init();?>
```

```
<?php /* @var $this Controller */ ?>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
    <meta name="language" content="en">
```

Install Bootstrap Extension in Yii

- Default classic template:

My Web Application

[Home](#) [About](#) [Contact](#) [User](#) [Logout \(admin\)](#)

Welcome to *My Web Application*

Congratulations! You have successfully created your Yii application.

You may change the content of this page by modifying the following two files:

- View file: `C:\inetpub\wwwroot\yii-app\protected\views\site\index.php`
- Layout file: `C:\inetpub\wwwroot\yii-app\protected\views\layouts\main.php`

For more details on how to further develop this application, please read the [documentation](#).

Copyright © 2019 by My C
All Rights Reserve
Powered by [Yii Framework](#)

Contact Us

If you have business inquiries or other questions, please fill out the following form.

Fields with * are required.

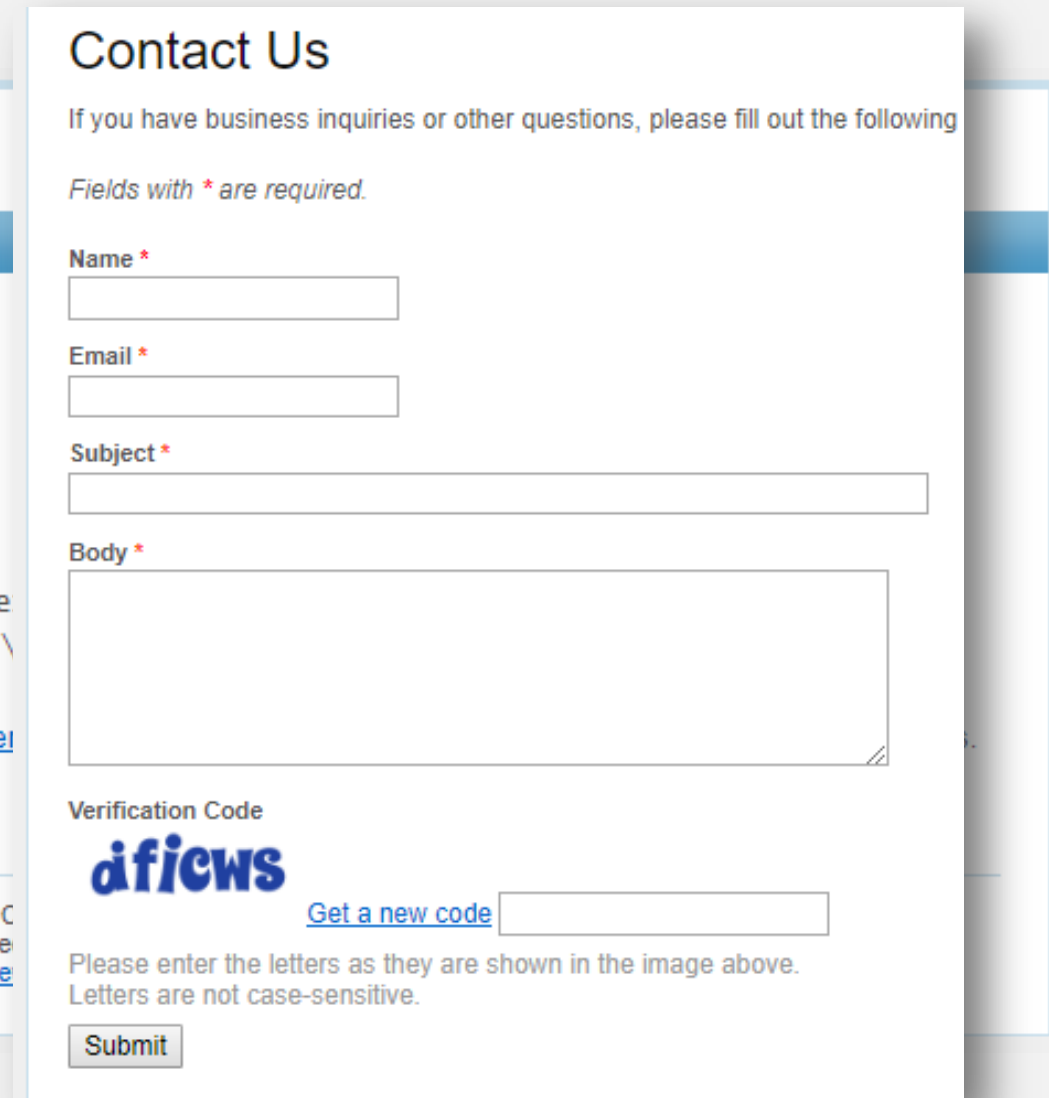
Name *

Email *

Subject *

Body *

Verification Code

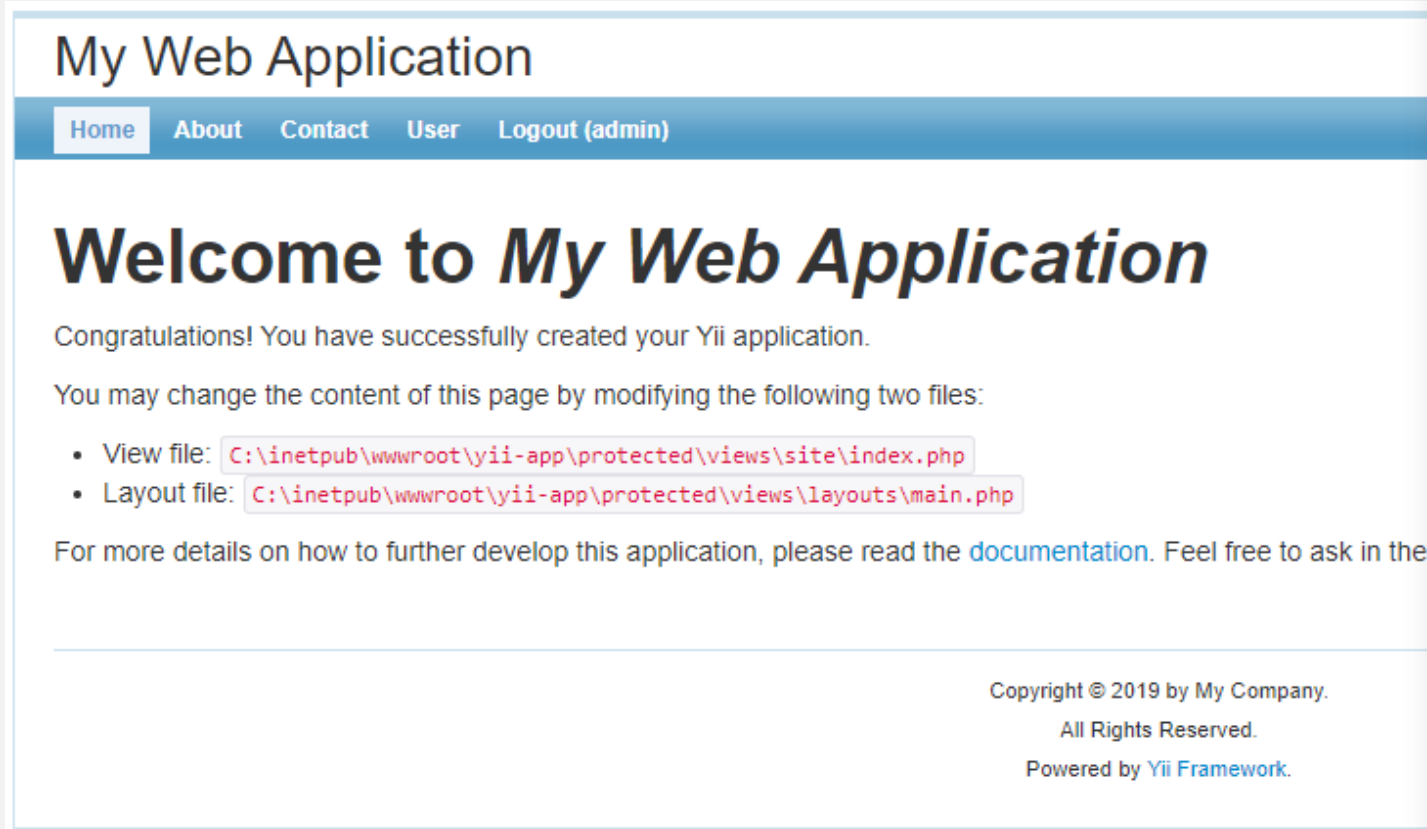
 [Get a new code](#)

Please enter the letters as they are shown in the image above.
Letters are not case-sensitive.

Submit

Install Bootstrap Extension in Yii

- Bootstrap template:



The screenshot shows a web application titled "My Web Application". It features a blue navigation bar with links: Home, About, Contact, User, and Logout (admin). The main content area has a large heading "Welcome to *My Web Application*" and a message: "Congratulations! You have successfully created your Yii application. You may change the content of this page by modifying the following two files:" followed by a list of files with their paths highlighted in a light blue box. At the bottom, there is a footer with copyright information and a note about the Yii Framework.

My Web Application

Home About Contact User Logout (admin)

Welcome to *My Web Application*

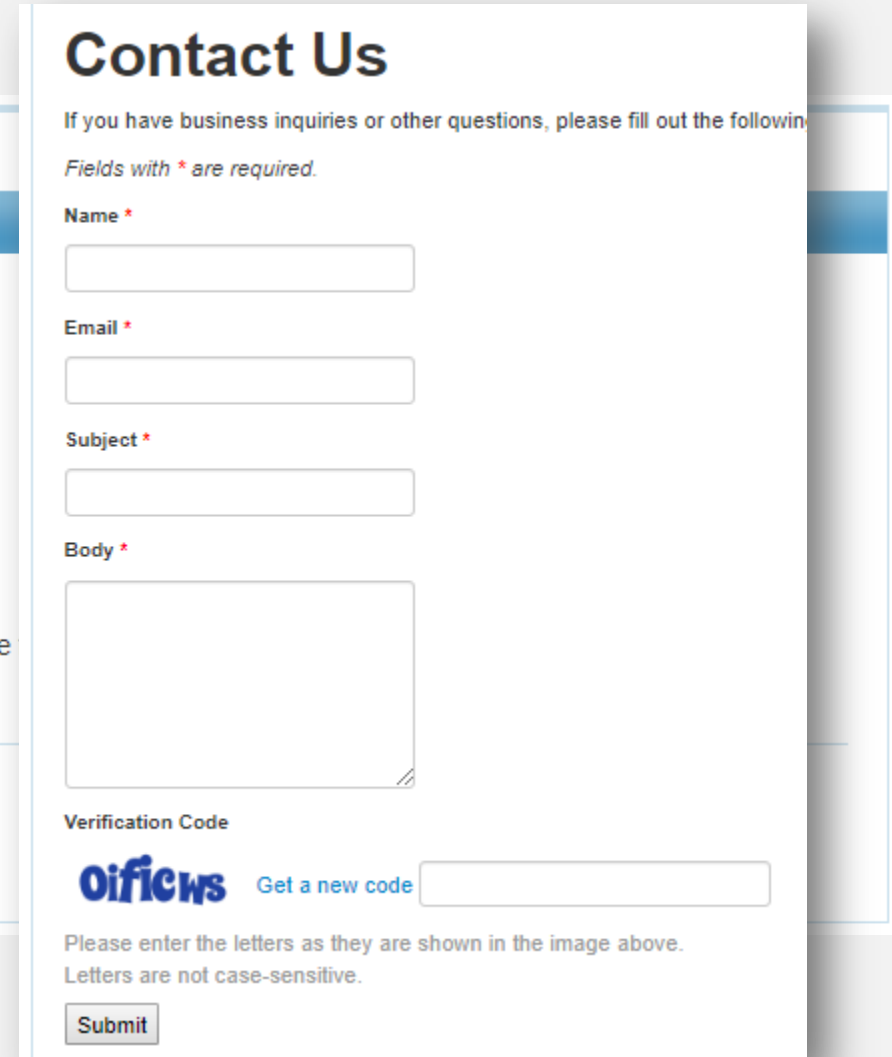
Congratulations! You have successfully created your Yii application.

You may change the content of this page by modifying the following two files:

- View file: `C:\inetpub\wwwroot\yii-app\protected\views\site\index.php`
- Layout file: `C:\inetpub\wwwroot\yii-app\protected\views\layouts\main.php`

For more details on how to further develop this application, please read the [documentation](#). Feel free to ask in the

Copyright © 2019 by My Company.
All Rights Reserved.
Powered by [Yii Framework](#).



The screenshot shows a "Contact Us" form. It includes a title, a brief instruction, and a note about required fields. The form has input fields for Name, Email, Subject, and a large text area for the Body. Below these is a "Verification Code" section featuring a CAPTCHA image and a "Get a new code" link. A "Submit" button is at the bottom.

Contact Us

If you have business inquiries or other questions, please fill out the following form.

*Fields with * are required.*

Name *

Email *

Subject *

Body *

Verification Code

oifcws [Get a new code](#)

Please enter the letters as they are shown in the image above.
Letters are not case-sensitive.

Bootstrap Generator

- Bootstrap Generator creates CRUD using bootstrap widgets



Welcome to Yii Code Generator!

You may use the following generators to quickly build up your Yii application:

- [Controller Generator](#)
- [Crud Generator](#)
- [Form Generator](#)
- [Model Generator](#)
- [Module Generator](#)
- [Bootstrap Generator](#)

GridView Widget





















Default

Manage Users

You may optionally enter a comparison operator (<, <=, >, >=, <> or =) at the beginning of each of your search values to specify how the comparison should be done.

[Advanced Search](#)

Displaying 1-10 of 22 results.

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 >





















Bootstrap

Manage Users

You may optionally enter a comparison operator (<, <=, >, >=, <> or =) at the beginning of each of your search values to specify how the comparison should be done.

[Advanced Search](#)

Displaying 1-10 of 22 results.

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	 

← 1 2 3 →

Widget

- Difference between default widget and bootstrap widget

Default Widget

```
<?php $this->widget('zii.widgets.grid.CGridView', array(
    'id' => 'user-grid',
    'dataProvider' => $model->search(),
    'filter' => $model,
    'columns' => array(
        'id',
        'username',
        'password',
        'email',
        array(
            'class' => 'CButtonColumn',
        ),
    ),
)); ?>
```

Bootstrap Widget

```
<?php $this->widget('bootstrap.widgets.TbGridView', array(
    'id' => 'user-grid',
    'dataProvider' => $model->search(),
    'filter' => $model,
    'columns' => array(
        'id',
        'username',
        'password',
        'email',
        array(
            'class' => 'bootstrap.widgets.TbButtonColumn',
        ),
    ),
)); ?>
```

Connect it to GitHub

- **GitHub** is a Git repository hosting service.
- **Git** is a command line tool while **GitHub** provides a Web-based graphical interface.

Start a new git repository:

- Go to github.com.
- Log in to your account.
- Click the [new repository](#) button in the top-right. You'll have an option there to initialize the repository with a README file.
- Click the [“Create repository”](#) button.

Git Push

- Open a [Git Bash](#) or a [Command-Line](#) in this directory [WebRoot/yii-app](#)
- Follow this set of instructions to “[Push an existing repository](#)” to GitHub

```
git init
```

```
git remote add origin https://github.com/<username>/yii-app.git
```

```
git add .
```

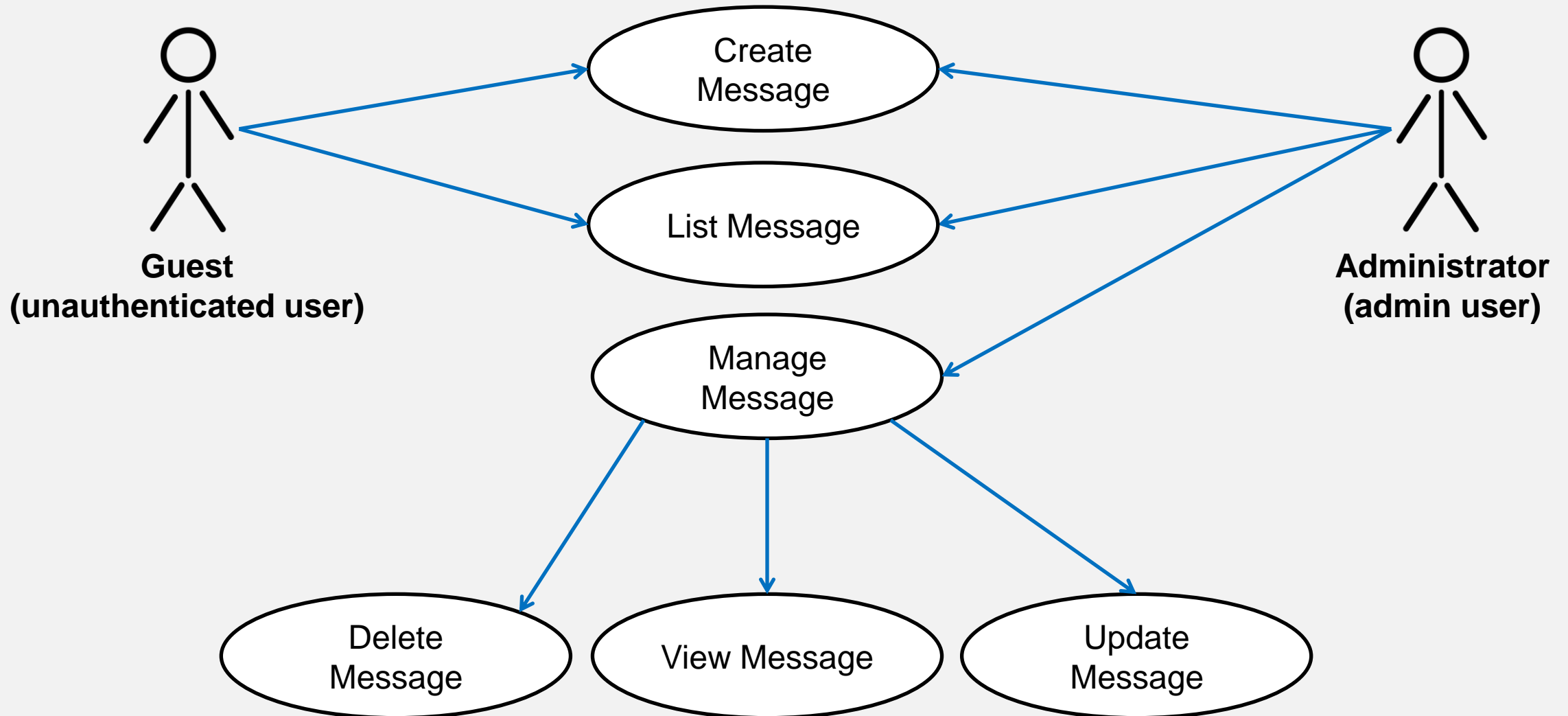
```
git commit -m "first commit"
```

```
git push -u origin master
```

- Visit the URL <https://github.com/<username>/yii-app> to view the code repository

Exercise

- Create “**Message**” System



Exercise

Guest

- Can **create** a message
- Can **view** list of message

Administrator

- Can **create** a message
- Can **view** list of message
- Can **manage** list of message
- Can **update** message
- Can **delete** message