

Programmer Thailand



# Yii2 Web Application Basic

เรียนรู้ได้ไว เข้าใจได้ง่าย ใช้งานได้จริง

*Yii Framework 2.0.2 & 2.0.3 Dev*

nanop กองอุ่น

WWW.PROGRAMMERTHAILAND.COM

# สารบัญ

สารบัญ .....	1
บทนำ .....	6
บทที่ 1 เริ่มต้นกับ Yii 2 .....	7
Yii2 คืออะไร .....	7
ความหมายและที่มา .....	7
ความต้องการระบบของ Yii2 .....	8
เตรียมเครื่องมือ .....	9
ติดตั้ง WAMP Server .....	9
ติดตั้ง Composer .....	19
การ Update Composer .....	22
แม่แบบ(Template) ใน Yii2 .....	23
บทที่ 2 การติดตั้ง Yii2 .....	24
การติดตั้ง Yii2 Basic Application Template .....	24
การสร้างฐานข้อมูลสำหรับ yii2-basic .....	26
แก้ไขการเขียนต่อฐานข้อมูล .....	27
การใช้ Migration ใน Yii Basic .....	28
การใช้ Gii Generator สำหรับ Basic Template .....	30
การสร้าง Virtual Host สำหรับ Yii2 Basic Application Template .....	38
การจัดการ URL แบบสวยงาม (Pretty URL) .....	41
การติดตั้ง Yii2 Advanced Application Template .....	43
สร้างฐานข้อมูลสำหรับ yii2-advanced .....	45
แก้ไขการเขียนต่อฐานข้อมูล .....	46
การใช้ Migration ใน Yii Advanced .....	49

Gii Generator สำหรับ Advanced Template .....	51
การสร้าง Virtual Host สำหรับ Yii2 Advanced Application Template .....	57
การจัดการ URL แบบสวยงาม (Pretty URL) .....	61
การ Update Yii2 .....	67
การ Upgrade Version .....	67
บทที่ 3 เข้าใจ Yii2 .....	69
หลักการ MVC .....	69
เข้าใจ Model .....	69
เข้าใจ View .....	69
เข้าใจ Controller .....	69
การทำงานของ Yii2 .....	70
โครงสร้าง Yii2 .....	71
Hello World กับ Yii2 .....	72
คุณสมบัติพื้นฐานที่ควรเข้าใจ .....	73
เรียนรู้ Routing ใน Yii2 .....	73
รู้จัก Namespace และ Use .....	76
ชื่อ Alias ใน Yii2 .....	76
ทดลองสร้าง Model .....	77
ทดลองสร้าง Controller .....	78
ทดลองสร้าง View .....	79
การเลือกข้อมูล .....	81
การสร้าง Relation ใน Model .....	90
การแสดง เพิ่ม แก้ไข และลบข้อมูล .....	90
การควบคุมการเข้าถึงโดยใช้ Access Control Filter (ACF) .....	101
การควบคุมการเข้าถึงโดยใช้ Role Base Access Control (RBAC) .....	103

บทที่ 4 โมดูล (Module) .....	111
การสร้างโมดูลสำหรับ Yii2 Basic Application Template.....	111
การสร้าง Model ใน Module .....	113
การสร้าง CRUD สำหรับ Module .....	115
การสร้างโมดูลสำหรับ Yii2 Advanced Application Template .....	119
การสร้างโมดูลสำหรับ Frontend .....	119
การสร้างโมดูลสำหรับ Backend .....	127
บทที่ 5 ธีม (Theme).....	136
ธีมสำหรับ Yii2 Basic Application Template.....	136
ติดตั้งแบบไม่ใช้ Composer .....	136
ติดตั้งแบบใช้ Composer .....	141
ธีมสำหรับ Yii2 Advanced Application Template.....	165
ธีมสำหรับ Frontend .....	166
ธีมสำหรับ Backend.....	167
บทที่ 6 การติดตั้ง Extension เสริม.....	170
Yii2 User.....	170
วิธีการติดตั้ง.....	170
วิธีการตั้งค่า .....	171
Yii2 Admin .....	174
วิธีติดตั้ง .....	174
วิธีตั้งค่า.....	175
การใช้งาน .....	178
บทที่ 7 ระบบจัดการงาน (Task Manager).....	188
วิเคราะห์ระบบ.....	188
ผู้ดูแลระบบ.....	188

เจ้าหน้าที่ .....	188
ออกแบบฐานข้อมูล .....	188
สร้าง ER Diagram .....	188
การ Forward Engineer ไปยังฐานข้อมูล .....	189
พจนานุกรมข้อมูล (Data Dictionary) .....	195
สร้าง Project Skeleton .....	196
ติดตั้ง Extension .....	199
ติดตั้ง Date Picker .....	199
เขียนโปรแกรม .....	200
สร้าง Virtual host .....	200
แก้ไขการตั้งค่า Yii2 .....	203
เพิ่ม user ในตาราง user .....	204
แก้ไขไฟล์ User Model .....	206
เพิ่มข้อมูลเข้าในตาราง project และ task .....	207
เพิ่ม urlManager .....	207
แก้ไขหน้า index .....	212
สร้าง Model Task และ Project .....	213
แก้ไข Relation ใน Model Task .....	216
สร้าง CRUD ด้วย Gii Generator .....	216
แก้ไขหน้าแสดงผล Task .....	220
แก้ไขฟอร์มเพิ่มข้อมูล .....	223
เพิ่ม Access Control สำหรับ Controller .....	226
เขียนโปรแกรมหน้า Dashboard .....	227
สร้าง RBAC .....	232
บทสรุป .....	241

บทที่ 8 ระบบสารสนเทศทางภูมิศาสตร์ (GIS) ด้วย Google Map .....	242
วิเคราะห์ระบบ.....	242
เจ้าหน้าที่ .....	242
บุคคลทั่วไป.....	242
ออกแบบฐานข้อมูล.....	243
ER Diagram.....	243
พจนานุกรมข้อมูล (Data Dictionary).....	243
สร้าง Project Skeleton .....	246
ติดตั้งส่วนเสริม.....	248
Yii2 User.....	248
Yii2 Google Map Library .....	250
Yii2 Export Table .....	251
เขียนโปรแกรม .....	252
แก้ไขการตั้งค่า Yii2.....	252
สร้าง Model ด้วย Gii Generator .....	256
สร้าง CRUD ด้วย Gii Generator.....	259
เขียนโปรแกรมแสดงแผนที่ .....	263
แก้ไข controllers/SiteController.php.....	263
แก้ไข views/site/index.php .....	265
กำหนดสิทธิ์ให้เฉพาะเจ้าหน้าที่ สามารถเพิ่ม แก้ไข และลบข้อมูล .....	269
บทสรุป .....	271
คำอปคุณ .....	273
บรรชณ .....	274

## บทนำ

หนังสือเล่มนี้เกิดจากความตั้งใจของผู้เขียนที่ต้องการสื่อให้ผู้อ่านเข้าใจ Yii Framework 2 อย่างง่ายดาย โดยผู้เขียนได้ศึกษาค้นคว้าจากแหล่งต่างๆ เพื่อกำหนดเป็นแนวทางในการเรียนรู้ Yii Framework 2 และพยายามจัดเรียงบทต่างๆ ให้เป็นขั้นตอนเพื่อให้ผู้อ่านทำความเข้าใจได้ง่ายดายมากยิ่งขึ้น โดยเนื้อหาจะมุ่งเน้นเรื่องพื้นฐานที่ควรรู้และเข้าใจก่อนการเขียนโปรแกรม ซึ่งเป็นการบูรณาการให้เข้าใจ Yii Framework 2 หรือ Yii2 โดยเน้นการเรียนรู้ลักษณะ Workshop โดยทำไปด้วยเรียนรู้ไปด้วยทดลอง Run ไปด้วย ซึ่งจะทำให้เข้าใจมากขึ้นทีละเรื่องทีละ Step จนเกิดความเข้าใจและเรียนรู้ได้อย่างรวดเร็ว หนังสือเล่มนี้จึงได้เน้นไปที่จุดหลักๆ ของการเขียนโปรแกรมโดยใช้ Yii2 ที่จะสามารถพัฒนา Web Application ได้อย่างรวดเร็วมากยิ่งขึ้น และสามารถต่อยอดในอนาคตได้

ทั้งนี้หนังสือเล่มนี้เป็นเพียงแนวทางหนึ่งในการเรียนรู้การพัฒนา Web Application ด้วย Yii Framework 2 เท่านั้น หากหนังสือเล่มนี้มีข้อขาดตกบกพร่องส่วนไหนทางผู้เขียนต้องกราบขออภัยมา ณ โอกาสนี้ และหวังเป็นอย่างยิ่งว่าผู้อ่านทุกๆ ท่านจะให้คำแนะนำต่างๆ เพื่อนำไปปรับปรุงแก้ไขต่อไป โดยสามารถส่งคำแนะนำมาได้ที่ contact@programmerthailand.com จักขอบพระคุณยิ่ง

มานพ กองอุ่น

Programmer Thailand Team

## บทที่ 1 เริ่มต้นกับ Yii 2

Yii2 หรือเรียกเต็มๆ ว่า Yii PHP Framework Version 2 ซึ่งทั้งหมดในหนังสือเล่มนี้จะเรียก Yii2 เพื่อให้เข้าใจง่าย โดยใน Version นี้ถือว่ามีการพัฒนาใหม่ทั้งหมดเลยก็ว่าได้

### Yii2 คืออะไร



รูปที่ 1 โลโก้ Yii Framework

### ความหมายและที่มา

Yii หรือ ยี ย่อมาจาก Yes It Is เป็นเหมือนกับคำตอบหรือคำอุทานก็ว่าได้ที่ว่า มันเร็วจริงหรือเปล่า? มันมีความปลอดภัยไหม? มันมีความเป็นมืออาชีพหรือเปล่า? มันควรนำมาสร้าง Project ต่อไปหรือเปล่า? ใช่แล้วมันใช่จริงๆ (Yes It Is)

สำหรับ Yii นั้นเป็น Open Source ซึ่งเป็น PHP Framework สำหรับสร้าง Web Application ที่ถูกเขียนขึ้นให้รองรับ PHP5 ช่วยให้สร้าง Web Application ได้อย่างรวดเร็ว

Yii นั้นเกิดขึ้นจากคุณ Qiang Xue เป็นผู้ริเริ่ม Yii เมื่อ 1 มกราคม 2008 ซึ่งก่อนหน้าเขาเคยพัฒนา Prado Framework มา ก่อนซึ่งก็มีประสบการณ์ต่างๆ มากหมาย และเห็นจุดอ่อนจุดแข็งของ Framework และได้นำประสบการณ์เหล่านั้นมาพัฒนาเป็น Yii Framework ในเวอร์ชันแรก

หลังจากนั้น Yii Framework ได้รับความนิยมเป็นอย่างมากในวงกว้างมากขึ้นเรื่อยๆ จนถึงปัจจุบัน ได้ออก Yii Framework ในเวอร์ชัน 2 เรียบร้อยแล้วเมื่อวันที่ 12 ตุลาคม 2557 ที่ผ่านมา

## ความต้องการระบบของ Yii2

ความต้องการระบบของ Yii2 นั้น จะใช้ PHP 5.4 ขึ้นไปซึ่งรองรับการเขียนโปรแกรม OOP และมีการปรับปรุงให้สามารถประมวลผลได้ดีมากขึ้น และการเขียนโปรแกรมที่สั้นลง เช่น array() ก็เปลี่ยนเป็น [] เป็นต้น โดยการตรวจสอบความต้องการนั้นสามารถเรียก URL ไปที่ไฟล์ requirement.php เพื่อตรวจสอบความต้องการของระบบสำหรับ Yii2 โดยจะมีหน้า Web ดังนี้

### Yii Application Requirement Checker

#### Description

This script checks if your server configuration meets the requirements for running Yii applications. It checks if the server is running the right version of PHP. If appropriate PHP extensions have been loaded, and if `php.ini` file settings are correct.

There are two kinds of requirements being checked. Mandatory requirements are those that have to be met to allow Yii to work as expected. There are also some optional requirements being checked which will show you a warning when they do not meet. You can use Yii framework without them but some specific functionality may be not available in this case.

#### Conclusion

Your server configuration satisfies the minimum requirements by this application.  
Please pay attention to the warnings listed below and check if your application will use the corresponding features.

#### Details

Name	Result	Required By	Memo
PHP version	Passed	Yii Framework	PHP 5.4.0 or higher is required.
Reflection extension	Passed	Yii Framework	
PCRE extension	Passed	Yii Framework	
SPL extension	Passed	Yii Framework	
gettext extension	Passed	Multiple string processing	Required for multiple encoding string processing.
Mcrypt extension	Passed	Security Component	Required by encrypt and decrypt methods.
Intl extension	Passed	Internationalization support	PHP int extension 1.0.2 or higher is required when you want to use advanced parameters formating in <code>Intl::format()</code> , international languages with <code>Intl::translate()</code> , ION-feature of <code>IntlValidator</code> or <code>IntlValidator::getIntlFormatter</code> .
Finfo extension	Passed	File Information	Required for file upload to detect correct file mime-type.
Dom extension	Passed	Disconnected Object Model	Required for REST API to send XML responses via <code>yii\web\XmlResponseFormatter</code> .
PDO extension	Passed	All DB-related classes	
PDO SQLite extension	Passed	All DB-related classes	Required for SQLite database.
PDO MySQL extension	Passed	All DB-related classes	Required for MySQL database.
PDO PostgreSQL extension	Passed	All DB-related classes	Required for PostgreSQL database.
Memcached extension	Passed	MemCache	
APC extension	Warning	AppCache	
PHP safe mode	Passed	Pre-uploading and console command execution	"safe_mode" should be disabled at <code>php.ini</code>
Expose PHP	Passed	Security reasons	"expose_php" should be disabled at <code>php.ini</code>
PHP allow url include	Passed	Security reasons	"allow_url_include" should be disabled at <code>php.ini</code>
PHP mail SMTP	Passed	Email sending	PHP mail (SMTP) server required.

Server: Apache/2.4.9 (Win64) 2014-10-02 13:27

Powered By: Yii Framework

รูปที่ 2 ความต้องการ (Requirement) ของ Yii2



www.programmerthailand.com

สำหรับ APC Extension นั้นยังไม่มี Version รองรับ PHP5.5.12 จึงยังไม่สามารถติดตั้งได้ครบหมด  
ซึ่งผู้เขียนได้ใช้ WAMP Server 2.5 64bit เป็น PHP5.5.12

## เตรียมเครื่องมือ

สำหรับการเรียนรู้ Yii2 ในหนังสือเล่มนี้จะใช้ WAMP Server ในการทดสอบ โดยสามารถดูรายละเอียดได้ที่ [www.wampserver.com](http://www.wampserver.com) ซึ่งมีวิธีการดาวน์โหลดและติดตั้งดังนี้

## ติดตั้ง WAMP Server

WAMP Server เป็น Web Server ที่สามารถจำลองเครื่องของเป็น Web Server สำหรับ Run Script PHP และมีฐานข้อมูล MySQL ด้วย นอกจากนั้น Version ล่าสุดของ WAMP Server คือ 2.5 ซึ่งมาพร้อมกับ Apache 2.4.9 PHP 5.5.12 และ MySQL 5.6.17 ซึ่งเหมาะสมสำหรับการพัฒนา Web Application ด้วย Yii2

## การดาวน์โหลด WAMP Server

เข้าเว็บไซต์ [www.wampserver.com](http://www.wampserver.com)



รูปที่ 3 หน้าเว็บไซต์ [www.wampserver.com](http://www.wampserver.com)

จากนั้นเปลี่ยนภาษาเป็น ENGLISH ที่มุมขวาบน เพื่อให้ง่ายต่อการทำความเข้าใจ



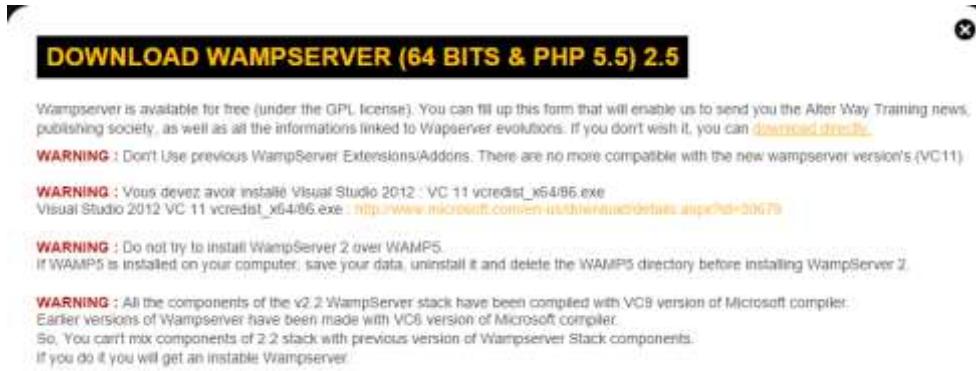
รูปที่ 4 หน้า www.wampserver.com เป็นภาษาอังกฤษ

จากนั้นเลือกเมนู Download หน้าเว็บจะเลื่อนลงมาที่ส่วนของดาวน์โหลด



รูปที่ 5 หน้าดาวน์โหลด WAMP Server

แล้วเลือก Version ตาม Bit ของเครื่องที่ต้องการ ในที่นี้จะเลือก WAMP SERVER (64 BITS & PHP 5.5) 2.5

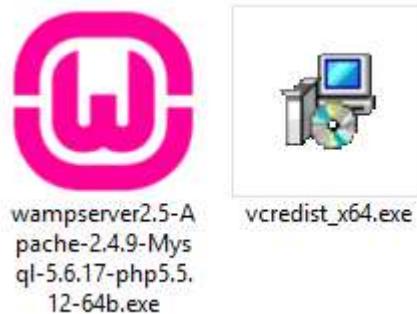


รูปที่ 6 หน้าสำหรับดาวน์โหลด WAMP Server

ให้ดาวน์โหลด 2 ไฟล์ นั่นคือลิ้งค์ download directly และ <http://www.microsoft.com/en-us/download/details.aspx?id=30679>

Download directly คือ WAMP Server และ Link ที่ 2 คือ VC 11 ซึ่งดาวน์โหลดจากเว็บไซต์ของ Microsoft

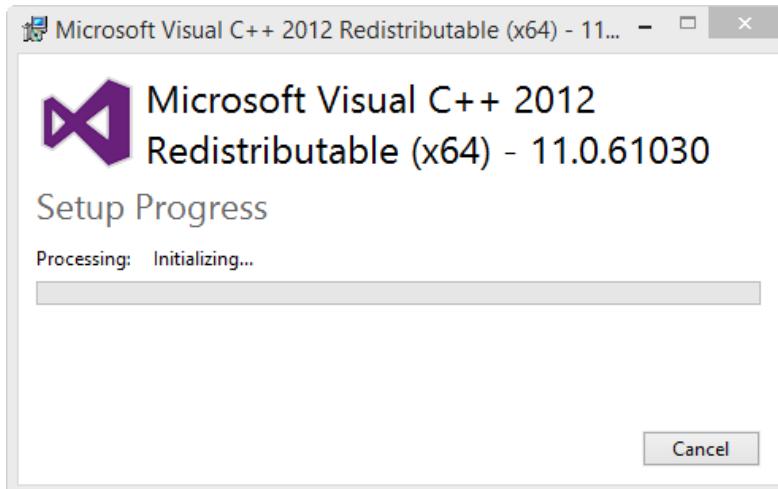
หลังจากดาวน์โหลดแล้วจะได้ 2 ไฟล์ ดังนี้



รูปที่ 7 ไฟล์ต่างๆ หลังจากดาวน์โหลดแล้ว

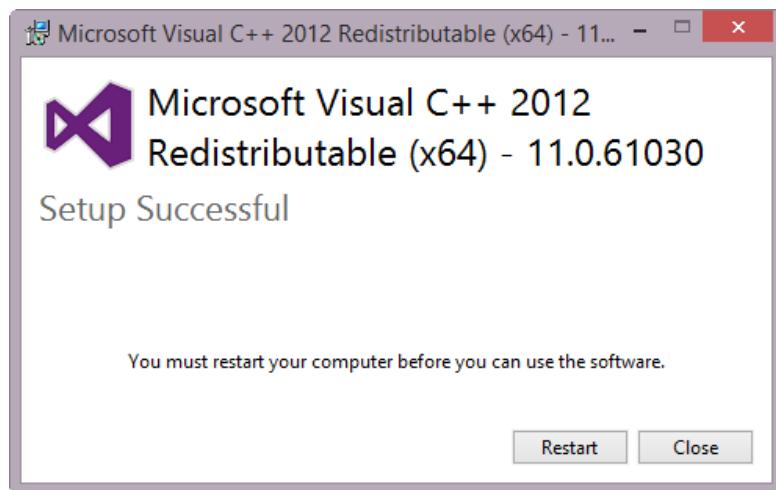
## ติดตั้ง WAMP Server

ให้ติดตั้ง Vcredist ก่อน โดยเปิดไฟล์ vcredist\_x64.exe จะเห็นการ Initializing ดังนี้



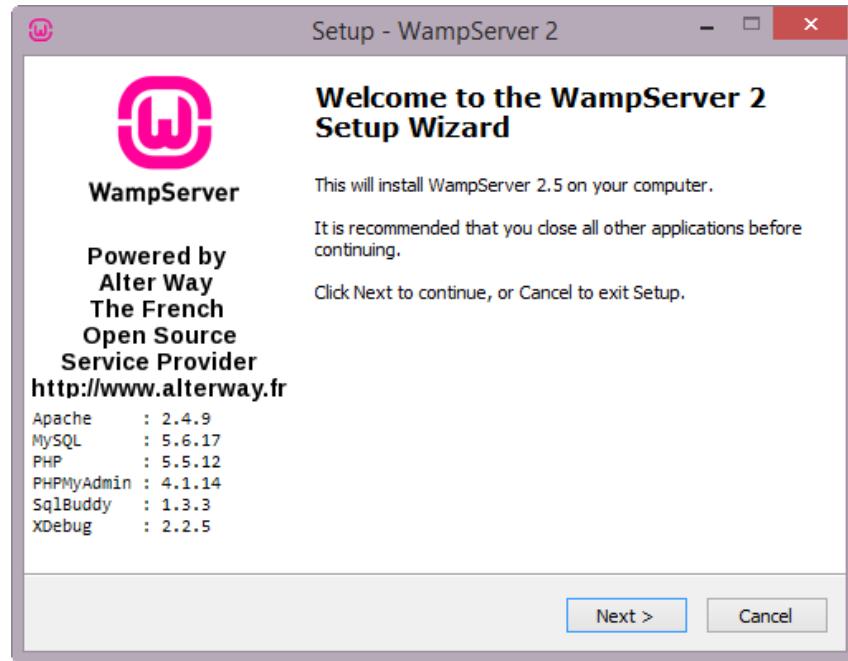
รูปที่ 8 การติดตั้ง Visual C++2012

เมื่อติดตั้งเสร็จโปรแกรมจะให้ Restart เครื่อง



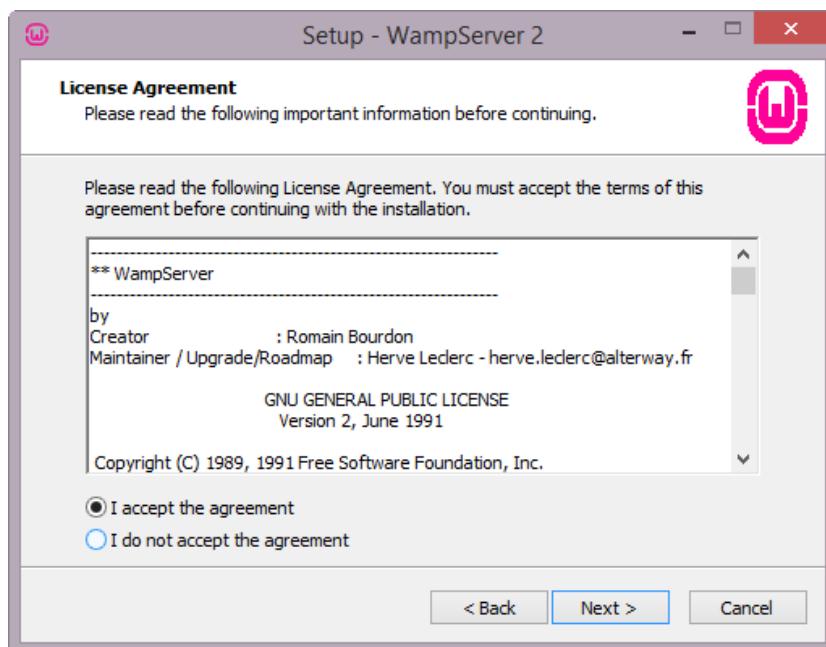
รูปที่ 9 หลังจากติดตั้ง Visual C++2012 เสร็จแล้ว

จากนั้นติดตั้ง WAMP Server โดยเปิดไฟล์ wampserver2.5-Apache-2.4.9-Mysql-5.6.17-PHP-5.5.12-64b.exe



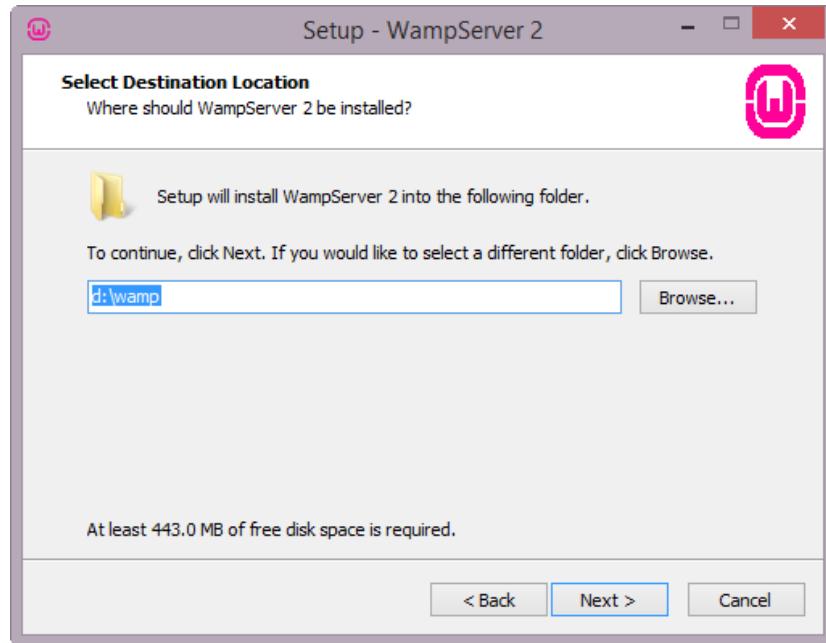
รูปที่ 10 การติดตั้ง WAMP Server

จากนั้นกดปุ่ม Next >



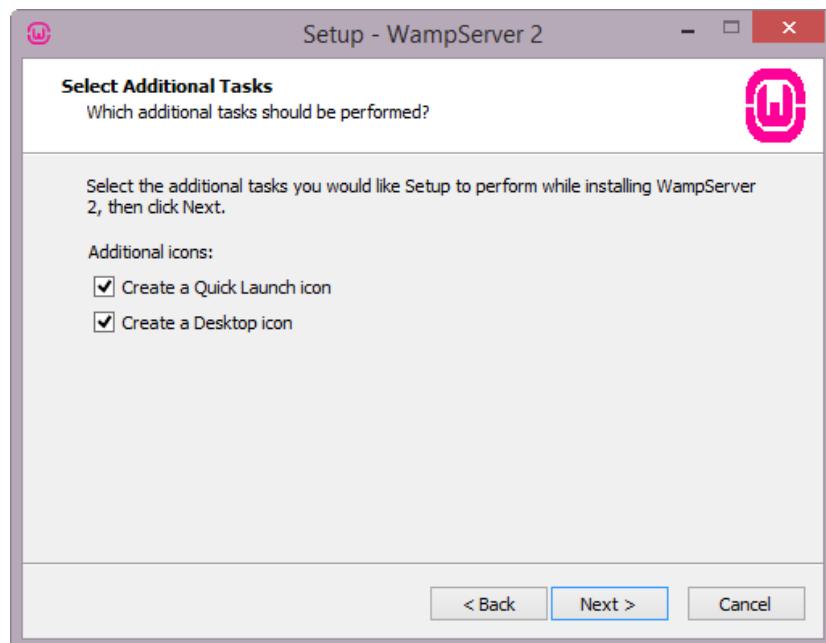
รูปที่ 11 การยอมรับเงื่อนไขข้อตกลงการใช้งาน

กดยอมรับเงื่อนไขการใช้งานแล้วกดปุ่ม Next >



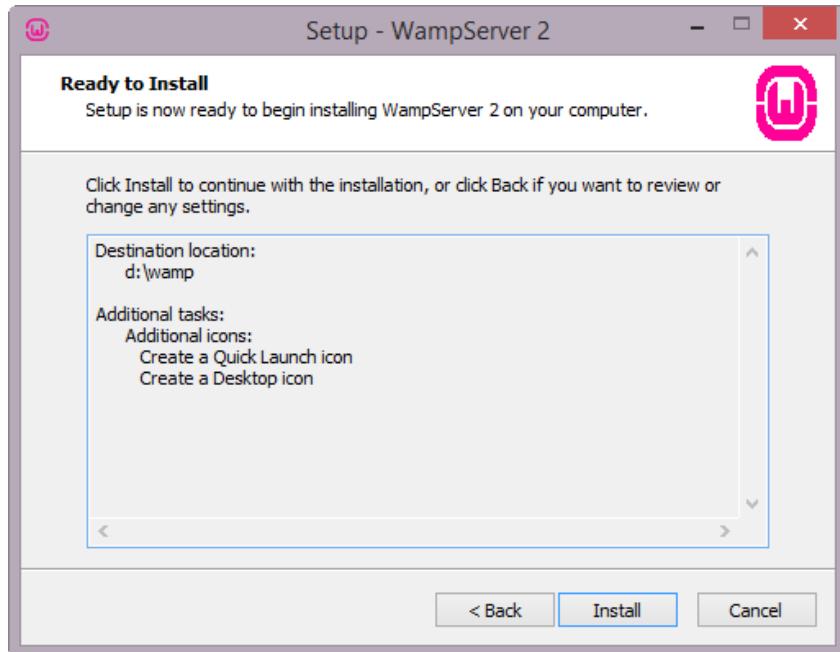
รูปที่ 12 การเลือกที่เก็บ WAMP Server

เปลี่ยนที่เก็บ Project จาก C:\ เป็น d:\ หรือ Drive อื่นๆ ตามต้องการ และกดปุ่ม Next >



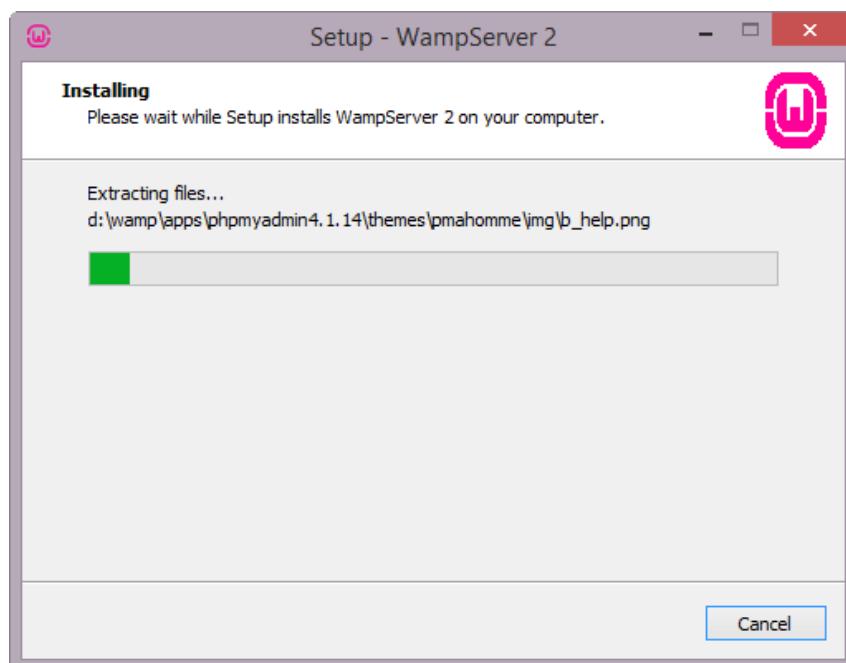
รูปที่ 13 การสร้าง Shortcut สำหรับ WAMP Server

สร้าง Short cut ตามต้องการ และกดปุ่ม Next >



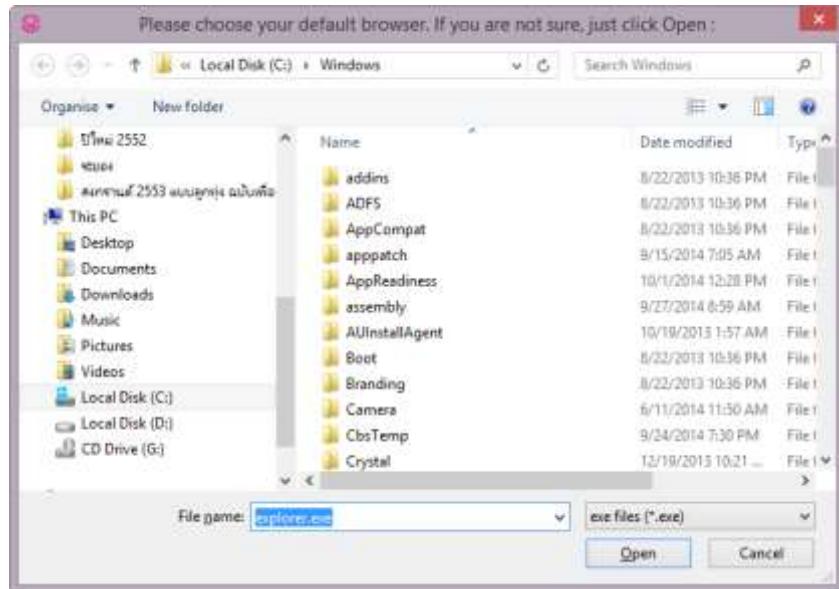
รูปที่ 14 ข้อความสรุปก่อนการติดตั้ง WAMP Server

จะพบกับหน้า Ready to Install จากนั้นกดปุ่ม Install



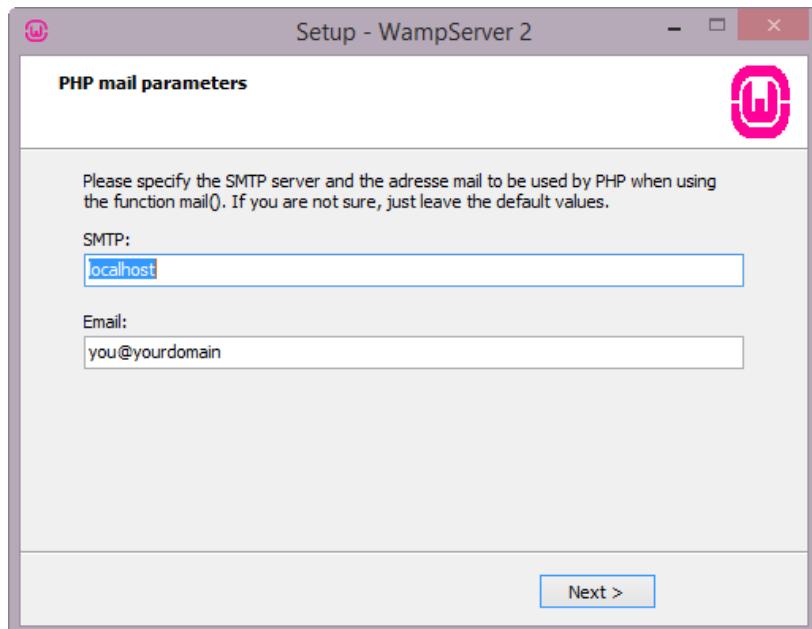
รูปที่ 15 กำลังติดตั้ง WAMP Server

ระบบจะติดตั้ง WAMP Server ตามขั้นตอน รอสักพัก



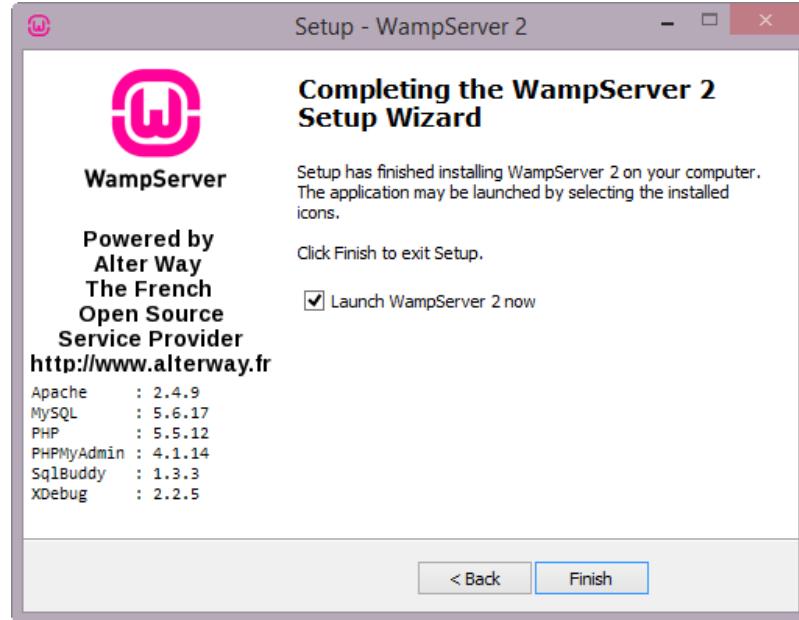
รูปที่ 16 การเลือกไฟล์ explorer.exe ให้กด Open

โปรแกรมจะเลือกไฟล์ explorer.exe ให้กด Open



รูปที่ 17 การกำหนด SMTP สำหรับ WAMP Server

โปรแกรมจะให้เลือกการตั้งค่า SMTP และ Email ในขั้นตอนนี้สามารถกดปุ่ม Next > ได้เลย



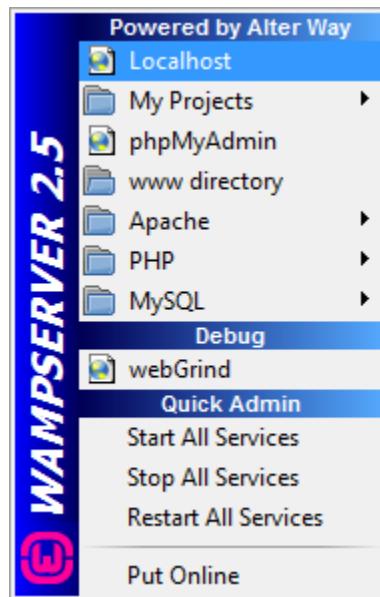
รูปที่ 18 ขั้นตอนสุดท้ายของการติดตั้ง WAMP Server

จะเข้าสู่ขั้นตอนสุดท้ายสามารถกดปุ่ม Finish เพื่อ Run WAMP Server หลังการติดตั้งทันที



รูปที่ 19 Icon ของ WAMP Server ที่ทำงานเป็นปกติ (สีเขียว)

จะพบ W สีเขียวแสดงว่าระบบทำงาน 100% นั่นคือ Service ของ Apache และ MySQL ทดสอบโดยกด W สีเขียวแล้วเลือก Localhost



รูปที่ 20 เมนู WAMP Server หลังจากคลิก W สีเขียว



www.programmerthailand.com

จะปรากฏหน้า localhost ดังนี้

The screenshot shows the WampServer configuration interface. At the top, it displays the WampServer logo and version information: Version 2.5 Version Française. Below this, the "Server Configuration" section provides details about the installed software:

- Apache Version : 2.4.9 - Documentation
- PHP Version : 5.5.12 - Documentation
- Server Software: Apache/2.4.9 (Win64) PHP/5.5.12

The "Loaded Extensions" section lists numerous PHP extensions, each with a small icon:

- apache2handler
- bcmath
- bz2
- calendar
- com\_dotnet
- Core
- ctype
- curl
- date
- dom
- ereg
- exif
- fileinfo
- filter
- ftp
- gd
- gettext
- gmp
- hash
- iconv
- imap
- intl
- json
- idap
- mbstring
- mcrypt
- mhash
- mysql
- mysqlnd
- odbc
- openssl
- pcre
- PDO
- pdo\_mysql
- pdo\_sqlite
- Phar
- Reflection
- session
- shmop
- SimpleXML
- soap
- SPL
- sqlite3
- standard
- sockets
- SPL
- xml
- xmireader
- xmlrpc
- wddx
- xmlwriter
- xmlwriter
- zip
- zlib

MySQL Version : 5.6.17 - Documentation

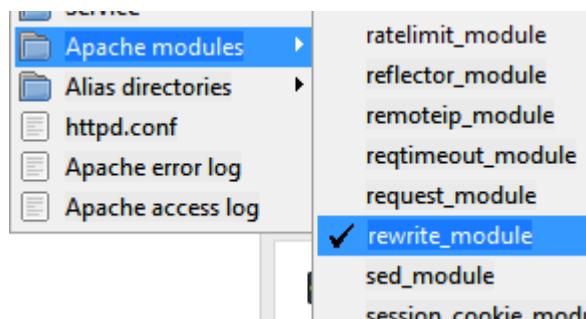
Below the configuration section are three toolbars:

- Tools**: phpinfo(), phpmyadmin
- Your Projects**: .metadata, 100-Flamingo88
- Your Aliases**: phpmyadmin, phpsysinfo

รูปที่ 21 หน้า localhost หน้าแรก

แสดงว่าติดตั้ง WAMP Server เสร็จเรียบร้อยแล้ว

เนื่องจาก Yii Framework 2 นั้นมีความต้องการ rewrite\_module เพื่อการทำ Pretty URL ดังนั้นควรเปิดการใช้งานส่วนนี้โดยให้กด W->Apache->Apache modules ทำให้แนใจว่ามีเครื่องหมายถูกหน้า rewrite\_module



รูปที่ 22 การกำหนด rewrite\_module สำหรับ Apache

## ติดตั้ง Composer

Composer เป็นโปรแกรมสำหรับการเรียกใช้งานและติดตั้งโปรแกรมต่างๆ ผ่านเครือข่าย อินเทอร์เน็ต โดยสามารถดาวน์โหลดได้ที่เว็บไซต์ <https://getcomposer.org/>



Dependency Manager for PHP

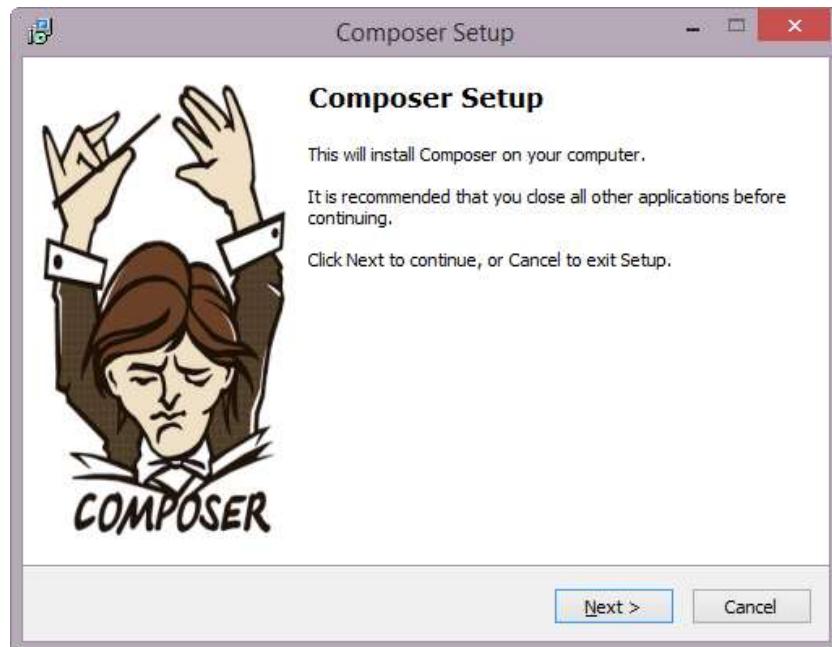
รูปที่ 23 โลโก้ของ Composer

จากนั้นคลิกที่ปุ่ม Download และในที่นี่จะเลือกการติดตั้งสำหรับ Windows โดยคลิกที่ Composer-Setup.exe

Download and run Composer-Setup.exe .

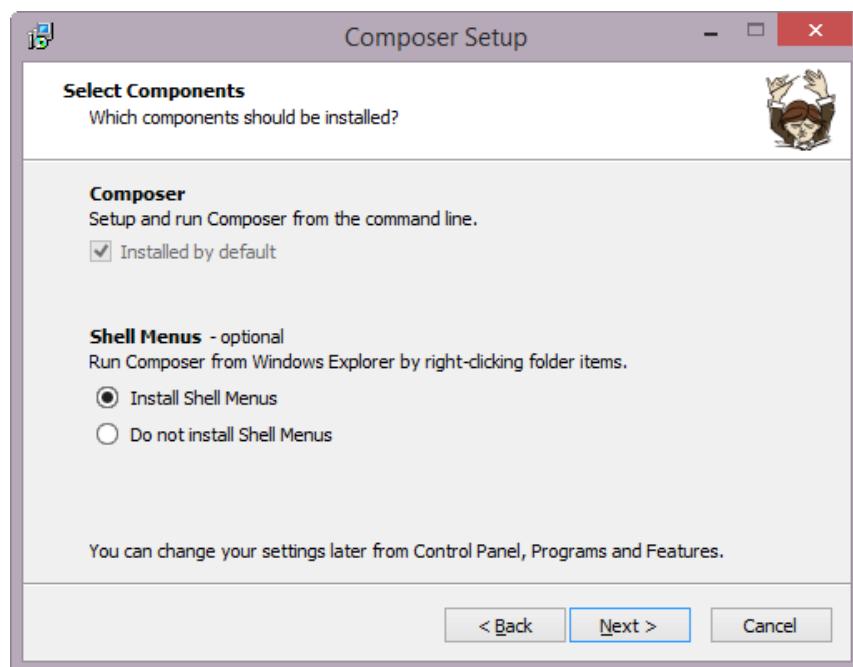
รูปที่ 24 ลิ้งค์สำหรับดาวน์โหลด Composer

และทำการติดตั้ง โดยเปิดไฟล์ Composer-Setup.exe



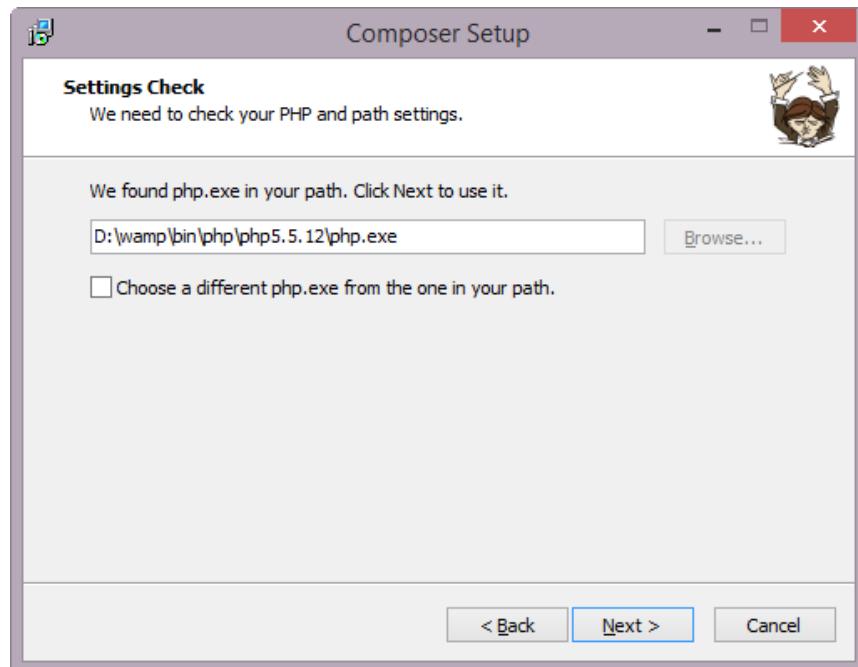
รูปที่ 25 หน้าแรกของการติดตั้ง Composer

จากนั้นกดปุ่ม Next>



รูปที่ 26 การเลือก Component สำหรับการติดตั้ง Composer

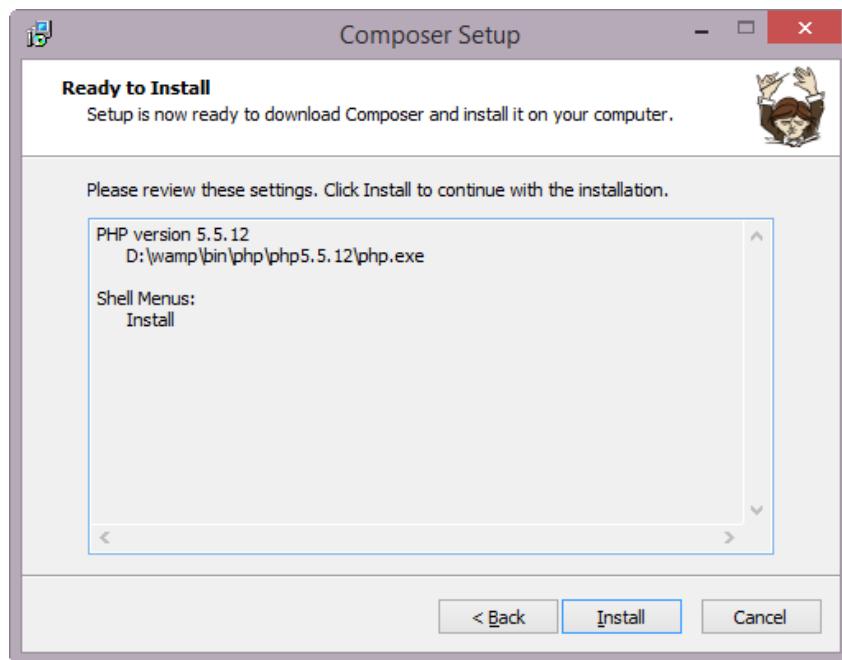
ใช้การตั้งค่าแบบเริ่มต้น และกดปุ่ม Next>



รูปที่ 27 การเลือกที่เก็บ PHP สำหรับ Composer

ทำการเลือกที่เก็บ Path ของ PHP ในที่นี่จะอยู่ที่ D:\wamp\bin\php\php5.5.12\php.exe จากนั้นกด

Next>



รูปที่ 28 สรุปข้อมูลก่อนการติดตั้ง

จะปรากฏ Ready to Install ให้กดปุ่ม Install



รูปที่ 29 หน้าเมื่อทำการติดตั้ง Composer เรียบร้อยแล้ว  
จะปรากฏการติดตั้งเสร็จสมบูรณ์ และกดปุ่ม Finish

## การ Update Composer

สำหรับการ Update Composer นั้นมีความจำเป็นเนื่องจากจะมีคำเตือนตอนการเรียกใช้งาน composer หากไม่มีการ update หลังจาก 30 วันไปแล้ว สำหรับวิธีการ update ทำได้่ายดายมาก เพียงพิมพ์คำสั่งดังนี้

```
composer self-update
```

หลังจากพิมพ์คำสั่งแล้วกดปุ่ม Enter จะปรากฏหน้า Command Prompt ดังนี้

```
D:\wamp\www\yii2-basic>composer self-update
Updating to version b23a3cd36870ff0eefc161a4638d9fcf49d998ba.
  Downloading: 100%
use composer self-update --rollback to return to version 533eeddc7575f11ddf1d98
93237ab1fa0ec1fc9
D:\wamp\www\yii2-basic>
```

รูปที่ 30 การใช้คำสั่ง Update Composer

## แม่แบบ(Template) ใน Yii2

สำหรับ Yii2 จะมี Web Application Template อิฐ 2 แบบ นั่นคือ Yii2 Basic ใช้สำหรับสร้าง Web Application ทั่วไป และ Yii2 Advanced ใช้สำหรับสร้าง Web Application ที่มีความซับซ้อนมากกว่า โดยข้อแตกต่างระหว่าง Yii2 Basic และ Yii2 Advanced นั่นคือการแบ่งระบบของ Yii2 Advanced นั้นจะแบ่งส่วน backend และ frontend ออกจากกัน แต่ในขณะที่ Yii2 Basic ไม่ได้แบ่งแยก นั่นเอง



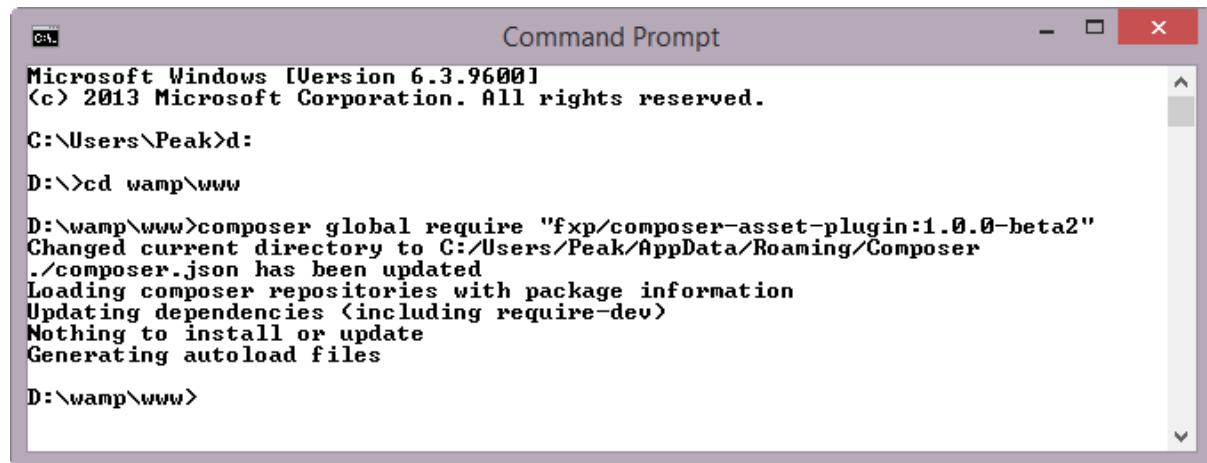
## บทที่ 2 การติดตั้ง Yii2

ในการติดตั้ง Yii2 นั้นสามารถทำได้หลายแบบในที่นี้จะแนะนำการติดตั้งผ่าน Composer ซึ่งเป็นมาตรฐานในการติดตั้ง โดยสามารถนำลักษณะนี้ไปประยุกต์ใช้ในการติดตั้ง Framework ตัวอื่นๆ ได้ โดยมีขั้นตอนในการติดตั้งดังนี้

ติดตั้ง Plugin เสริมสำหรับ Composer โดยเปิด Command Prompt จากนั้นไปที่ D:\wamp\www และพิมพ์คำสั่งดังนี้

```
composer global require "fxp/composer-asset-plugin:1.0.0-beta4"
```

แล้วกดปุ่ม Enter โปรแกรมจะทำการ Update Plugin ที่จำเป็นในการสร้าง Web Application ด้วย Yii2



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Peak>d:
D:\>cd wamp\www
D:\wamp\www>composer global require "fxp/composer-asset-plugin:1.0.0-beta4"
Changed current directory to C:/Users/Peak/AppData/Roaming/Composer
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Nothing to install or update
Generating autoload files
D:\wamp\www>
```

รูปที่ 31 การเพิ่ม Plugin สำหรับ Composer

## การติดตั้ง Yii2 Basic Application Template

สำหรับ Yii2 Basic Application Template นั้นเป็นการใช้ Yii2 ในการพัฒนา Web Application แบบทั่วไป สำหรับ Web Application ที่ไม่ซับซ้อนมากนัก โดยมีวิธีการติดตั้งดังนี้

เปิด Command Prompt และไปที่ D:\wamp\www

รูปที่ 32 การใช้ Command Prompt เข้า Root Directory ของ Web Server

พิมพ์คำสั่งในการเรียกใช้ composer plugin

```
composer global require "fxp/composer-asset-plugin:1.0.0"
```

พิมพ์คำสั่งในการสร้าง Web Application จาก Basic Template ดังนี้

```
composer create-project --prefer-dist --stability=dev  
yiisoft/yii2-app-basic yii2-basic
```

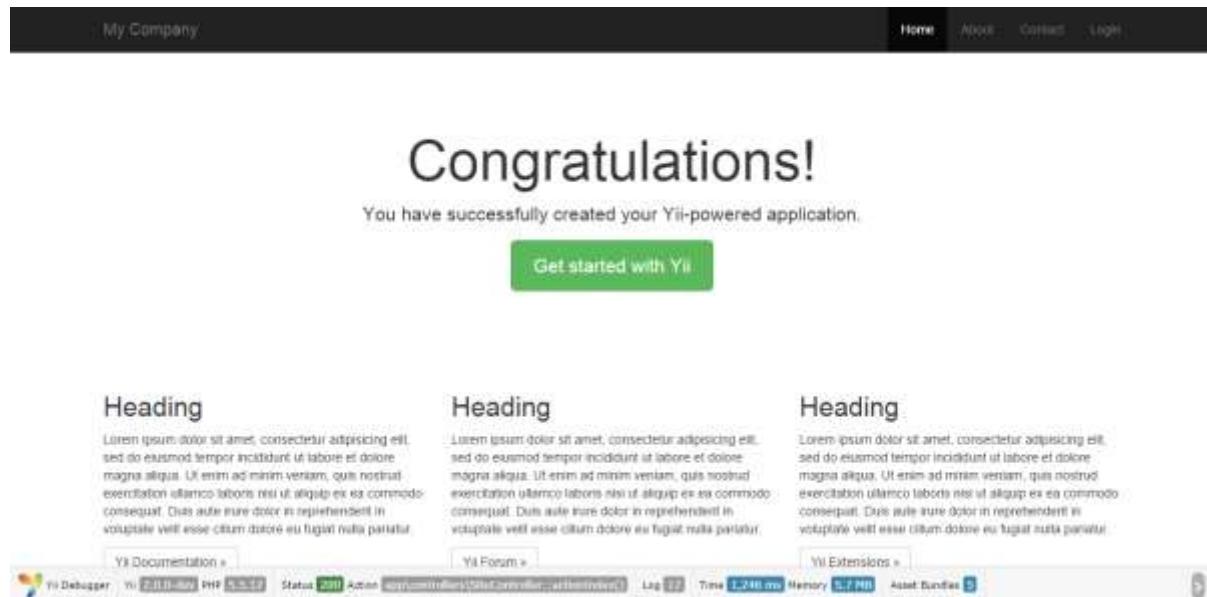
หมายเหตุ หากต้องการ Yii2 Version ที่เสถียรแล้วไม่ต้องใส่ --stability=dev

แล้วกดปุ่ม Enter จะแสดงรายการไฟล์ต่างๆ ที่สร้างจาก Composer

รูปที่ 33 การติดตั้ง Yii2 Basic Application Template



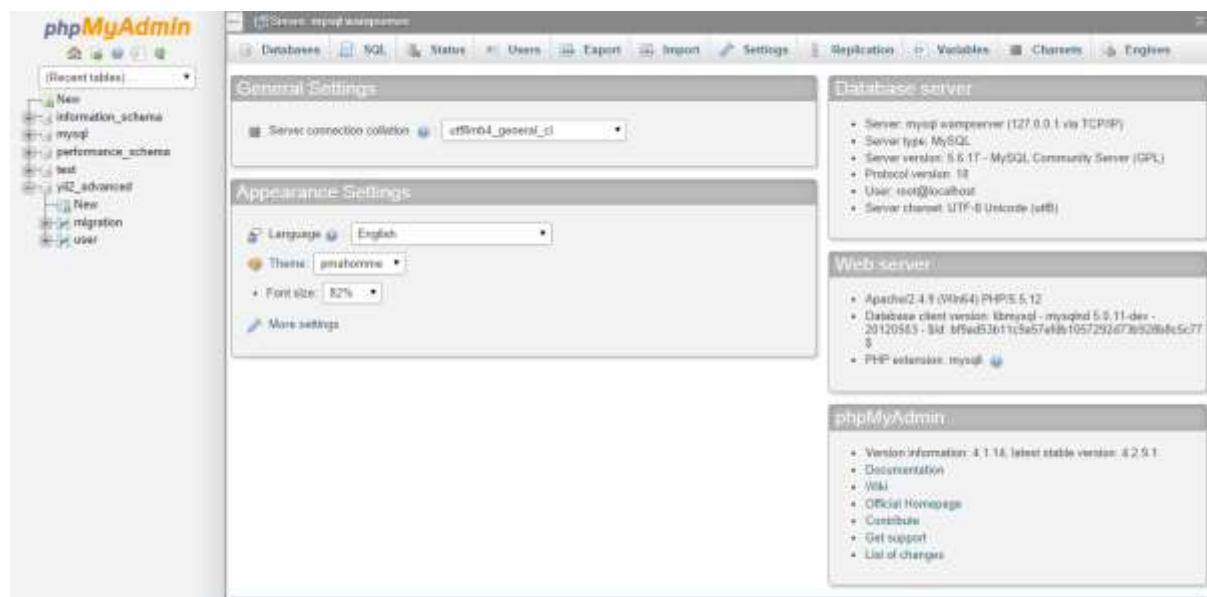
ทดสอบ Run หน้าเว็บโดยเข้า URL <http://localhost/yii2-basic/web> จะปรากฏหน้า Web Application ดังนี้



รูปที่ 34 หน้าหลักของ Web Application

## การสร้างฐานข้อมูลสำหรับ yii2-basic

สำหรับการสร้างฐานข้อมูลนั้นสามารถเข้าบริหารจัดการโดยใช้ phpMyAdmin โดยเข้า URL <http://localhost/phpmyadmin> หรือ <http://127.0.0.1/phpmyadmin> ซึ่งจะได้หน้า Web ดังนี้

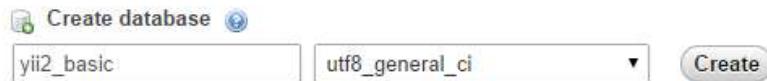


รูปที่ 35 หน้า PHPMyAdmin



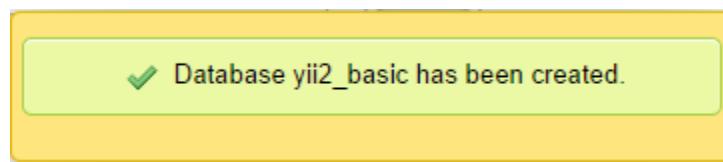
คลิกที่ Database จากนั้น ใส่ชื่อฐานข้อมูลในที่นี่จะใช้ yii2\_basic และใช้การเข้ารหัสอักขระเป็น utf8\_general\_ci ดังนี้ จากนั้นกดปุ่ม Create

## Databases



รูปที่ 36 การสร้างฐานข้อมูล

จะปรากฏ Popup สีเขียวว่าสร้างฐานข้อมูลชื่อ yii2\_basic เรียบร้อยแล้ว



รูปที่ 37 POPUP หลังจากสร้างฐานข้อมูลเสร็จแล้ว

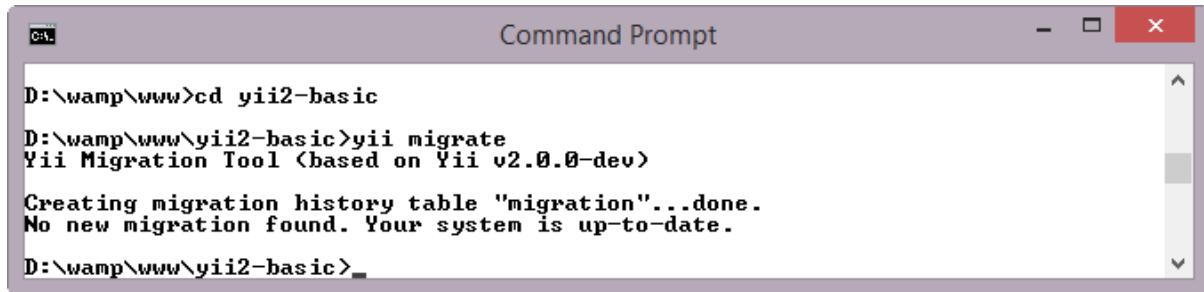
## แก้ไขการเชื่อมต่อฐานข้อมูล

แก้ไขการเชื่อมต่อฐานข้อมูลโดยเปิดไฟล์ config/db.php จากนั้นใส่ชื่อฐานข้อมูลที่ต้องการเชื่อมต่อพร้อม Username และ Password สำหรับเชื่อมต่อฐานข้อมูล MySQL ในที่นี่คือฐานข้อมูล: yii2\_basic  
Username: root และ Password ไม่มี ดังนี้

```
return [
    'class' => 'yii\db\Connection',
    'dsn' => 'mysql:host=localhost;dbname=yii2_basic',
    'username' => 'root',
    'password' => '',
    'charset' => 'utf8',
];
```

รูปที่ 38 การแก้ไขการติดต่อฐานข้อมูล

## การใช้ Migration ใน Yii Basic



```
D:\wamp\www>cd yii2-basic
D:\wamp\www\yii2-basic>yii migrate
Yii Migration Tool (based on Yii v2.0.0-dev)

Creating migration history table "migration"...done.
No new migration found. Your system is up-to-date.

D:\wamp\www\yii2-basic>
```

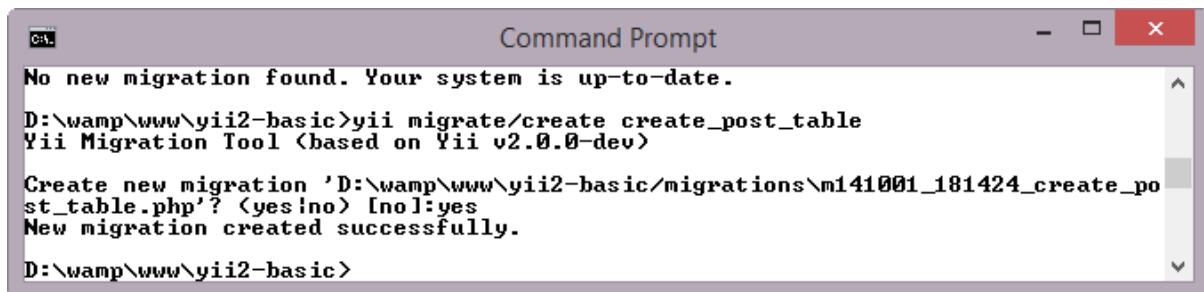
รูปที่ 39 การใช้ Migration ใน Yii Basic

## การสร้าง Migration

สร้าง Migration ชื่อ create\_post\_table โดยพิมพ์คำสั่งดังนี้

```
yii migrate/create create_post_table
```

กดปุ่ม Enter จากนั้นพิมพ์ yes เพื่อยืนยันแล้วกดปุ่ม Enter



```
No new migration found. Your system is up-to-date.

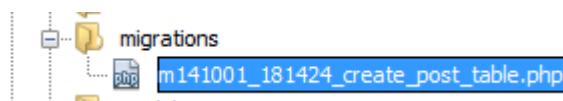
D:\wamp\www\yii2-basic>yii migrate/create create_post_table
Yii Migration Tool (based on Yii v2.0.0-dev)

Create new migration 'D:\wamp\www\yii2-basic/migrations/m14001_181424_create_po
st_table.php'? (yes|no) [no]:yes
New migration created successfully.

D:\wamp\www\yii2-basic>
```

รูปที่ 40 การเรียกใช้งานเพื่อสร้าง Migration

จะปรากฏไฟล์ m14001\_181424\_create\_post\_table.php ใน Folder migrations เรียบร้อยแล้ว



รูปที่ 41 ไฟล์ Migration ที่ถูกสร้างขึ้น

จากนั้นทำการปรับแต่งไฟล์เพื่อใช้ในการสร้างตารางต่างๆ ดังนี้

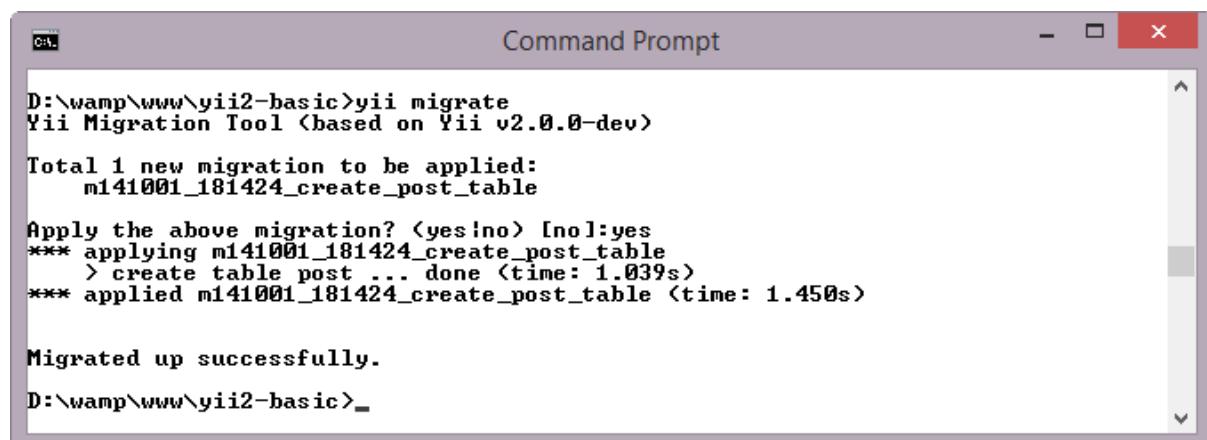
```
1 <?php
2
3     use yii\db\Schema;
4     use yii\db\Migration;
5
6     class m141001_181424_create_post_table extends Migration
7     {
8         public function up()
9         {
10             $this->createTable('post', [
11                 'id' => 'pk',
12                 'title' => Schema::TYPE_STRING.' NOT NULL',
13                 'content' => Schema::TYPE_TEXT,
14             ]);
15         }
16
17         public function down()
18         {
19             $this->dropTable('post');
20             //echo "m141001_181424_create_post_table cannot be reverted.\n";
21
22             //return false;
23         }
24     }
```

รูปที่ 42 การแก้ไขไฟล์ Migration ที่ได้สร้างขึ้น

ทดสอบโดยการเรียก

```
yii migrate
```

จะได้การประมวลผลดังนี้



```
D:\wamp\www\yii2-basic>yii migrate
Yii Migration Tool (based on Yii v2.0.0-dev)

Total 1 new migration to be applied:
  m141001_181424_create_post_table

Apply the above migration? <yes|no> [no]:yes
*** applying m141001_181424_create_post_table
  > create table post ... done (time: 1.039s)
*** applied m141001_181424_create_post_table (time: 1.450s)

Migrated up successfully.

D:\wamp\www\yii2-basic>
```

รูปที่ 43 การสั่งให้ Migration ทำงาน

จะเห็นว่ามีตาราง post เกิดขึ้นที่ฐานข้อมูล yii2\_basic ดังนี้



<a href="#">migration</a>	<a href="#">Browse</a>	<a href="#">Structure</a>	<a href="#">Search</a>	<a href="#">Insert</a>	<a href="#">Empty</a>	<a href="#">Drop</a>	~2 InnoDB utf8_general_ci 16 KiB
<a href="#">post</a>	<a href="#">Browse</a>	<a href="#">Structure</a>	<a href="#">Search</a>	<a href="#">Insert</a>	<a href="#">Empty</a>	<a href="#">Drop</a>	~0 InnoDB utf8_general_ci 16 KiB

รูปที่ 44 รายการตารางหลังจากสั่ง Migration ให้ทำงาน

เข้าไปปรับแก้ตาราง โดยเพิ่ม Comment ในแต่ละ Field ดังนี้

Name	Type	Length/Values	Default	Collation	Attributes	Null	A_I	Comments
title	VARCHAR	255	None	utf8_general_ci		✓		หัวข้อ
content	TEXT		NULL	utf8_general_ci		✓		รายละเอียด

รูปที่ 45 การปรับแต่ง Comment ในตาราง post

จากนั้นกดปุ่ม Save เพื่อบันทึกการเปลี่ยนแปลง

## การใช้ Gii Generator สำหรับ Basic Template

Gii Generator เป็นเครื่องมือที่มีความสำคัญสำหรับ Yii Framework ซึ่งเป็นเครื่องมือที่ช่วยในการสร้าง Code ให้สามารถพัฒนา Web Application ได้อย่างรวดเร็วมากยิ่งขึ้นภายใต้ธรรมเนียมของ Yii Framework

การเข้าใช้งาน Gii Generator นั้นสามารถเรียกจาก URL web/index.php?r=gii สำหรับการติดตั้งแบบ Basic โดยจะเห็นหน้าตาของ Web Application ดังนี้

Welcome to Gii a magical tool that can write code for you

Start the fun with the following code generators:

**Model Generator**  
This generator generates an ActiveRecord class for the specified database table.  
[Start >](#)

**CRUD Generator**  
This generator generates a controller and views that implement CRUD (Create, Read, Update, Delete) operations for the specified data model.  
[Start >](#)

**Controller Generator**  
This generator helps you to quickly generate a new controller class, one or several controller actions and their corresponding views.  
[Start >](#)

**Form Generator**  
This generator generates a view script file that displays a form to collect input for the specified model class.  
[Start >](#)

**Module Generator**  
This generator helps you to generate the skeleton code needed by a Yii module.  
[Start >](#)

**Extension Generator**  
This generator helps you to generate the files needed by a Yii extension.  
[Start >](#)

Yii Debugger | Yii 2.0.16 PHP 7.0.32 Status 200 Actions [View] [Edit] [Delete] [Revert] Log [+] Time 357 ms Memory 6.2 MB Asset Bundles [+]

รูปที่ 46 หน้าแรกของ Gii Generator

## การสร้าง Model ด้วย Gii

หลังจากที่กดปุ่ม Start > ในส่วนของ Model Generator และ جانนั้นใส่ชื่อตารางที่ต้องการสร้าง Model ในที่นี่คือ post และเลือกใช้ Generate Labels from DB Comments

# Model Generator

This generator generates an ActiveRecord class for the specified database table.

**Table Name**

post

**Model Class**

Post

**Namespace**

app\models

**Base Class**

yii\db\ActiveRecord

**Use Table Prefix**

**Generate Relations**

**Generate Labels from DB Comments**

**Enable I18N**

**Code Template**

default (D:\wamp\www\yii2-basic\vendor\yiisoft\yii2-gii\generators\model/default)

**Preview**

รูปที่ 47 การตั้งค่าสำหรับ Generate Model

จากนั้นกดปุ่ม Preview



www.programmerthailand.com

Preview    Generate

Click on the above **Generate** button to generate the files selected below:

Code File	Action	
models\Post.php	create	<input checked="" type="checkbox"/>

รูปที่ 48 การแสดงผลเมื่อกด Preview

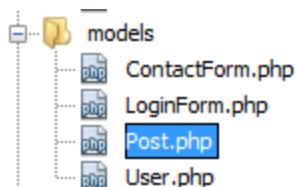
แลกดปุ่ม Generate เพื่อสร้าง Model ก็จะปรากฏข้อความดังนี้ แสดงว่าสร้าง Model เสร็จเรียบร้อยแล้ว

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-basic\vendor\yiisoft\yii2-gii\generators\model/default"...
generated models\Post.php
done!
```

รูปที่ 49 การแสดงผลเมื่อกด Generate

เปิดดูใน Folder models จะเห็นว่ามีไฟล์ Post.php เกิดขึ้น



รูปที่ 50 โครงสร้างไฟล์ Model หลังจาก Generate แล้ว

เปิดไฟล์ models/Post.php ขึ้นมาจะเห็นข้อมูลลักษณะนี้

```
<?php

namespace app\models;

use Yii;

/**
 * This is the model class for table "post".
 *
 * @property integer $id
 *
 * @property string $title
 *
 * @property string $content

```

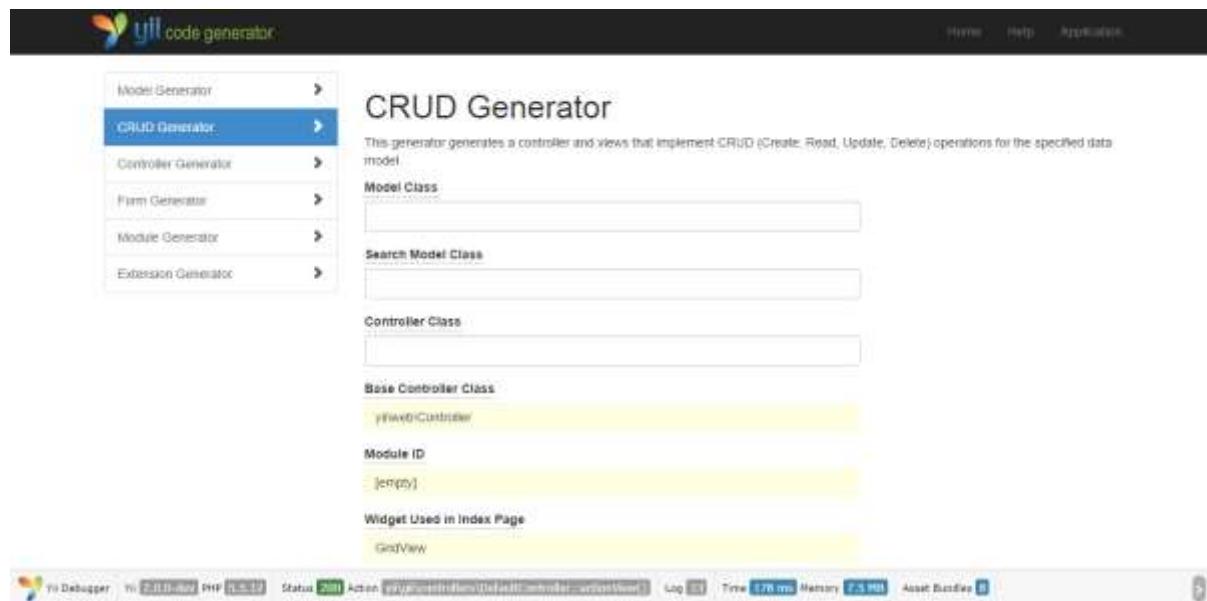
```
*/  
  
class Post extends \yii\db\ActiveRecord  
{  
  
    /**  
  
     * @inheritdoc  
  
     */  
  
    public static function tableName()  
    {  
  
        return 'post';  
  
    }  
  
    /**  
  
     * @inheritdoc  
  
     */  
  
    public function rules()  
    {  
  
        return [  
  
            ['title', 'required'],  
  
            ['content', 'string'],  
  
            ['title', 'string', 'max' => 255]  
        ];  
  
    }  
  
    /**  
  
     * @inheritdoc  
  
     */  
  
    public function attributeLabels()
```

```
{  
    return [  
        'id' => 'ID',  
        'title' => 'หัวข้อ',  
        'content' => 'รายละเอียด',  
    ];  
}  
}
```

จะเห็นว่าในส่วนของ attributeLabels() นั้นจะมีภาษาไทยเกิดขึ้น

## การสร้าง CRUD ด้วย Gii

CRUD เป็นการสร้าง Controller และ View พร้อมกัน โดยเป็นการสร้างคำสั่งพื้นฐานนั้นคือ Create Read Update และ Delete ในเวลาเดียวกัน โดยระบุชื่อ Model ที่ได้สร้างไว้แล้ว โดยคลิกที่เมนู CRUD Genterator



รูปที่ 51 หน้าแรกของ CRUD Generator

จากนั้นระบุชื่อ Class ของ Model ที่ต้องการ ในที่นี่คือ Post และกำหนดรายละเอียดดังนี้

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update model).

## Model Class

app\models\Post

## Search Model Class

app\models\PostSearch

## Controller Class

app\controllers\PostController

## Base Controller Class

## Widget Used in Index Page

GridView

## Enable I18N

## Code Template

default (D:\wamp\www\yii2-basic\vendor\yiisoft\yii2-gii\generators\crud/default)

Preview

รูปที่ 52 การตั้งค่าต่างๆ ของ CRUD Generator

จากนั้นกดปุ่ม Preview เพื่อดูตัวอย่าง จะปรากฏรายละเอียดดังนี้



www.programmerthailand.com

Preview    Generate

Click on the above **Generate** button to generate the files selected below:

Code File	Action	
controllers\PostController.php	create	<input checked="" type="checkbox"/>
models\PostSearch.php	create	<input checked="" type="checkbox"/>
views\post\_form.php	create	<input checked="" type="checkbox"/>
views\post\_search.php	create	<input checked="" type="checkbox"/>
views\post\create.php	create	<input checked="" type="checkbox"/>
views\post\index.php	create	<input checked="" type="checkbox"/>
views\post\update.php	create	<input checked="" type="checkbox"/>
views\post\view.php	create	<input checked="" type="checkbox"/>

**Create** **Unchanged** **Overwrite**

รูปที่ 53 หน้าเว็บหลังกดปุ่ม Preview

จากนั้นกดปุ่ม Generate จะปรากฏข้อความแสดงการ Generate CRUD เรียบร้อยแล้ว

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-basic\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated controllers\PostController.php
generated models\PostSearch.php
generated views\post\_form.php
generated views\post\_search.php
generated views\post\create.php
generated views\post\index.php
generated views\post\update.php
generated views\post\view.php
done!
```

รูปที่ 54 หน้าเว็บหลังกดปุ่ม Generate

ทดลองเรียก URL เพื่อตุ้นหน้า Web Application สำหรับ Post โดยไปที่ URL <http://localhost/yii2-basic/web/index.php?r=post> จะได้หน้า Web Application ดังนี้

My Company

Home About Contact Log in

Home Posts

## Posts

Create Post

#	ID	Title	Description

No results found.

รูปที่ 55 หน้าเว็บหลังเรียก post

ทดลองสร้าง Post ใหม่ โดยกดปุ่ม Create Post แล้วใส่รายละเอียด

My Company

Home About Contact Log in

Home Posts Create Post

## Create Post

ชื่อกระทู้

บทความ Web Application ด้วย Yii Framework 2 คุณจะได้

รูปภาพ

Yii Framework 2 สร้าง Web Application ที่ดีและมีประสิทธิภาพโดยใช้ภาษา PHP 5.6 หรือใหม่กว่า คุณสามารถเรียนรู้การใช้งาน Yii Framework ในการสร้าง

**Create**

### รูปที่ 56 การเพิ่มข้อมูลจากฟอร์ม

จากนั้นกดปุ่ม Create จะปรากฏหน้าสำหรับแสดงรายละเอียด Post ที่ได้ทำการบันทึก และสามารถกดปุ่ม Update เพื่อแก้ไขข้อมูล และปุ่ม Delete เพื่อลบข้อมูล

The screenshot shows a Yii Framework 2 application interface. At the top, there's a navigation bar with links for Home, Posts, and Yii2Basic Web Application. Below that is a title bar with 'My Company' and other navigation links. The main content area has a heading 'มาพัฒนา Web Application ด้วย Yii Framework 2 กันเถอะ'. Underneath is a table with one row, containing fields for ID, หัวข้อ (Topic), and รายละเอียด (Details). There are 'Update' and 'Delete' buttons at the top of the table row.

Yii Debugger    CPU Usage    PHP    SQLs    Status 200 Action userController@PostController@index Log 13 Time 180 ms Memory 8.2 MB    C0 2 Asset Bundles 3

รูปที่ 57 การเปิดดูข้อมูลหลังจากเพิ่มข้อมูลแล้ว

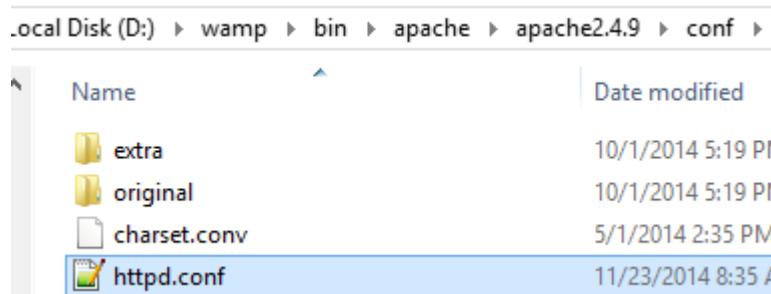
## การสร้าง Virtual Host สำหรับ Yii2 Basic Application

### Template

สำหรับ Virtual Host นั้นเป็นการจำลองการเรียก Web Application ผ่าน Domain URL เช่น `http://yii2-basic.dev` เป็นต้น สำหรับวิธีการนั้นทำได้ไม่ยาก มี 2 ขั้นตอนดังนี้

### แก้ไขไฟล์ `httpd-vhosts.conf`

สำหรับไฟล์นี้จะบรรจุอยู่ใน `d:/wamp/bin/apache/apachex.x.x/conf/extra/httpd-vhosts.conf` แต่ก่อนที่จะไปแก้ไขไฟล์นี้ ต้องทำให้แน่ใจก่อนว่าได้เปิดการใช้งานแล้วในการตั้งค่าในไฟล์ `httpd.conf` ที่อยู่ใน `d:/wamp/bin/apache/apachex.x.x/conf/httpd.conf`



รูปที่ 58 ที่เก็บไฟล์ `httpd.conf`

จากนั้นแก้ไขโดยยกเลิก Comment ในบรรทัด 513 เพื่อให้เรียกไฟล์ `httpd-vhosts.conf` ได้

```
512 # Virtual hosts  
513 Include conf/extr/httpd-vhosts.conf
```

รูปที่ 59 การเปิดการใช้งาน Virtual Host

ทำการบันทึกไฟล์ httpd.conf

จากนั้นเปิดไฟล์ httpd-vhosts.conf และทำให้แน่ใจว่ามีคำสั่งต่อไปนี้อยู่หากไม่มีให้เพิ่ม

```
<VirtualHost *:80>  
  
    ServerName localhost  
  
    DocumentRoot "d:/wamp/www/"  
  
</VirtualHost>
```

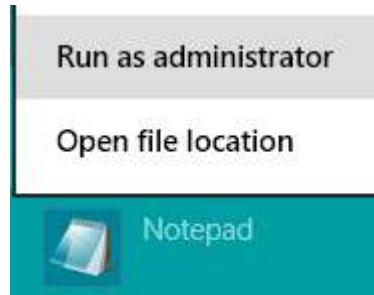
จากนั้นเพิ่มคำสั่งในการสร้าง Virtual Host ดังนี้

```
<VirtualHost *:80>  
  
    ServerAdmin webmaster@yii2-basic.dev  
  
    DocumentRoot "d:/wamp/www/yii2-basic/web"  
  
    ServerName yii2-basic.dev  
  
    ErrorLog "logs/yii2-basic.log"  
  
    CustomLog "logs/yii2-basic.log" common  
  
</VirtualHost>
```

ส่วนที่สำคัญคือ DocumentRoot คือ Root ของ Web Application และ ServerName คือ ชื่อ Domain ที่เราต้องการ แล้วทำการบันทึกไฟล์ httpd-vhosts.conf

**แก้ไขไฟล์ c:/windows/system32/driver/etc/hosts**

เปิด Notepad แบบ Administrator คลิกขวา Run as administrator



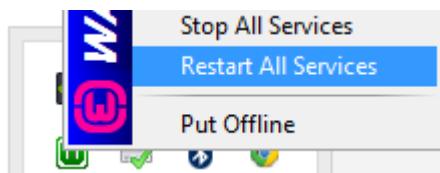
รูปที่ 60 การเปิด Notepad แบบ Administrator

จากนั้นไปที่ File->Open แล้วไปเปิดไฟล์ hosts ใน c:/windows/system32/driver/etc/ และเพิ่มข้อมูลต่อไปนี้

127.0.0.1                yii2-basic.dev

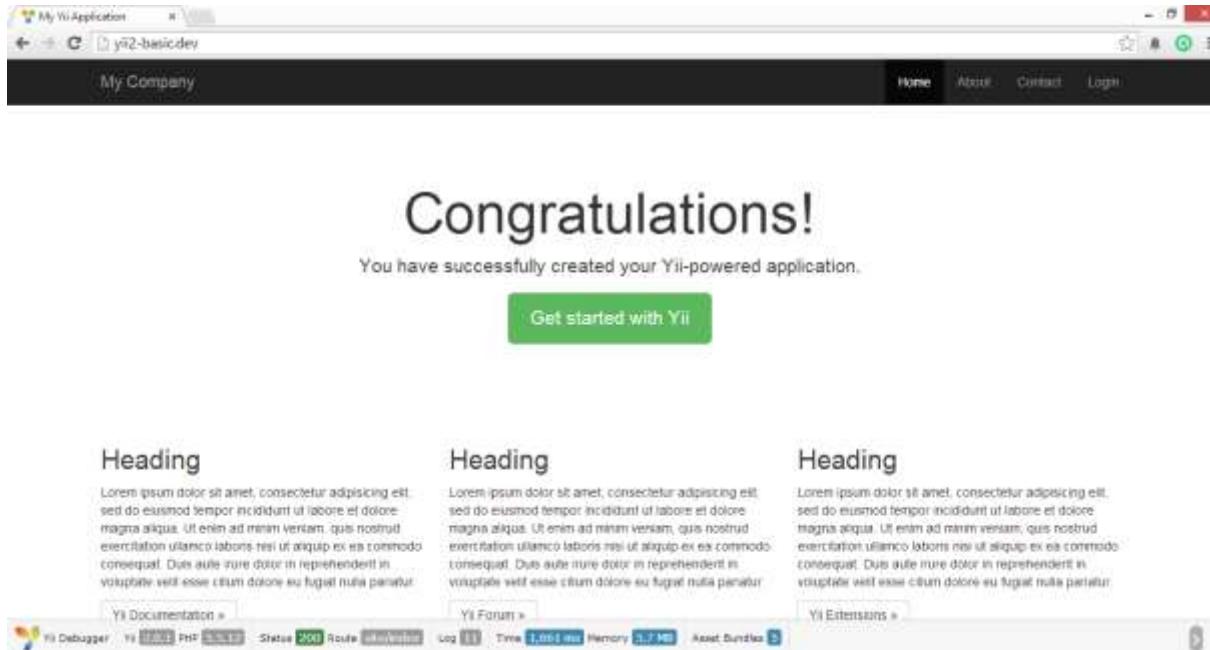
รูปที่ 61 การเพิ่มการเรียก Domain ใน Localhost

จากนั้นบันทึก และทำการ Restart Apache โดยคลิก W สีเขียวแล้วเลือก Restart All Services



รูปที่ 62 การ Restart All Services

หมายเหตุ ชื่อ Domain ใน httpd-vhosts.conf กับ ชื่อ Domain ใน hosts นั้นต้องเป็นชื่อเดียวกัน  
จากนั้นทดลองเข้า URL ตามชื่อ Domain http://yii2-basic.dev จะได้หน้า Web Application ดังนี้



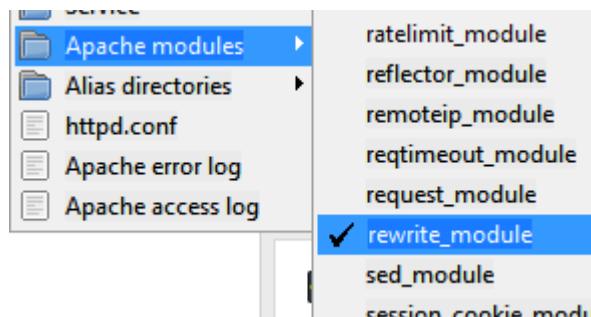
รูปที่ 63 การเรียก URL หลังการกำหนด Virtual Host แล้ว

## การจัดการ URL แบบสวยงาม (Pretty URL)

สำหรับการจัดการ URL ให้สวยงามนั้นเรียกว่า Pretty URL โดยมีตัวอย่างการเรียกใช้งานเช่น

```
http://yii2-basic.dev/site/about
```

จะเห็นว่า index.php ไม่มี และตัวแปร ?r= ก็ไม่มีเช่นกัน ทำให้ URL ดูสวยงามมากยิ่งขึ้น โดยเป็นการใช้งาน rewrite\_module ของ Apache ดังนั้นก่อนทำขั้นตอนนี้ควรเป็น rewrite\_module ของ Apache ก่อน ถ้าเป็น WAMP Server ให้กด W->Apache->Apache modules ทำให้แนใจว่ามีเครื่องหมายถูกหน้า rewrite\_module



รูปที่ 64 การเปิดใช้งาน rewrite\_module

สำหรับ Pretty URL นั้นมีประโยชน์กับ Search Engine ในการรวบรวมข้อมูลต่างๆ ใน Web Application และทำให้ URL มีความสวยงามมากยิ่งขึ้น สำหรับวิธีการจัดการนั้นสามารถทำได้ดังนี้

## เพิ่ม UrlManager

ทำการเพิ่ม UrlManager ในส่วนของ Component ในไฟล์ config/web.php ดังนี้

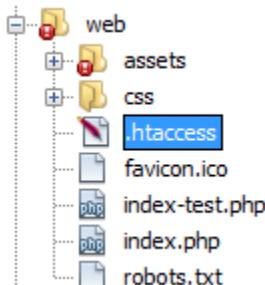
```
15     'components' => [
16         'urlManager' => [
17             'class' => 'yii\web\UrlManager',
18             'enablePrettyUrl' => true,
19             'showScriptName' => false,
20         ],

```

รูปที่ 65 การเพิ่มคำสั่ง urlManager ใน components

## สร้างไฟล์ .htaccess

ทำการสร้างไฟล์ .htaccess ในโฟลเดอร์ web ดังนี้



รูปที่ 66 ที่เก็บไฟล์ .htaccess

โดยการใส่ข้อมูลดังนี้

```
1 RewriteEngine on
2 # if a directory or file exists, use it directly
3 RewriteCond %{REQUEST_FILENAME} !-f
4 RewriteCond %{REQUEST_FILENAME} !-d
5
6 # otherwise forward it to index.php
7 RewriteRule . index.php
```

รูปที่ 67 คำสั่งในไฟล์ .htaccess

หมายเหตุ ดูเพิ่มเติมเรื่องการสร้าง Virtual Host สำหรับ Yii Basic Application Template ที่จะได้ URL  
<http://yii2-basic.dev>

## การติดตั้ง Yii2 Advanced Application Template

เปิด Command Prompt และไปที่ D:\wamp\www

รูปที่ 68 การ Point ไปยัง Root ของ Web Server

พิมพ์คำสั่งในการเรียกใช้ composer asset plugin

```
composer global require "fxp/composer-asset-plugin:1.0.0"
```

พิมพ์คำสั่งในการสร้าง Web Application จาก Basic Template ดังนี้

```
composer create-project --prefer-dist --stability=dev  
yiisoft/yii2-app-advanced yii2-advanced
```

หมายเหตุ หากต้องการ Version ที่เสถียรไม่ต้องใส่ --stability=dev

แล้วกดปุ่ม Enter จะแสดงรายการไฟล์ต่างๆ ที่สร้างจาก Composer

รูปที่ 69 หน้า Command Prompt หลังการติดตั้ง Yii2 Advanced Template

จากนั้นเข้าไปที่ Folder yii2-advanced ที่สร้างขึ้น

```
D:\wamp\www>cd yii2-advanced  
D:\wamp\www\yii2-advanced>
```

รูปที่ 70 การเข้าไปยัง Project ที่เพิ่งสร้างขึ้น

แล้วเรียกคำสั่ง

```
init
```

แล้วกดปุ่ม Enter จะได้ดังนี้

```
D:\wamp\www\yii2-advanced>init  
Yii Application Initialization Tool v1.0  
Which environment do you want the application to be initialized in?  
[0] Development  
[1] Production  
Your choice [0-1, or "q" to quit]
```

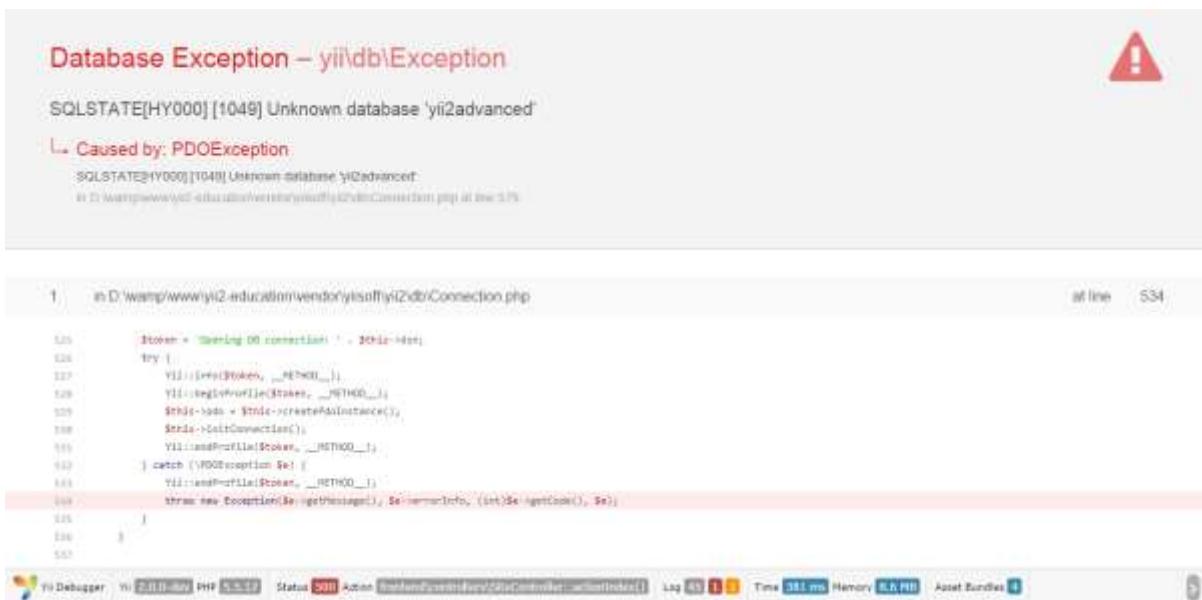
รูปที่ 71 การเรียกใช้งานคำสั่ง init

เลือกรูปแบบของ Web Application ในที่นี่จะเลือก 0 คือ Development แล้วกดปุ่ม Enter และกด yes เพื่อ Confirm อีกครั้ง แล้วกด Enter โปรแกรมจะ Generate Project Skeleton ให้ดังนี้

```
Command Prompt  
generate backend/config/main-local.php  
generate backend/config/params-local.php  
generate backend/web/index-test.php  
generate backend/web/index.php  
generate common/config/main-local.php  
generate common/config/params-local.php  
generate console/config/main-local.php  
generate console/config/params-local.php  
generate frontend/config/main-local.php  
generate frontend/config/params-local.php  
generate frontend/web/index-test.php  
generate frontend/web/index.php  
generate yii  
generate cookie validation key in backend/config/main-local.php  
generate cookie validation key in frontend/config/main-local.php  
    chmod 0777 backend/runtime  
    chmod 0777 backend/web/assets  
    chmod 0777 frontend/runtime  
    chmod 0777 frontend/web/assets  
    chmod 0755 yii  
... initialization completed.  
D:\wamp\www\yii2-advanced>_
```

รูปที่ 72 หน้า Command Prompt หลังจาก init เสร็จแล้ว

จากนั้นเปิด URL <http://localhost/yii2-advanced/frontend/web> จะได้หน้า Web Application ดังนี้

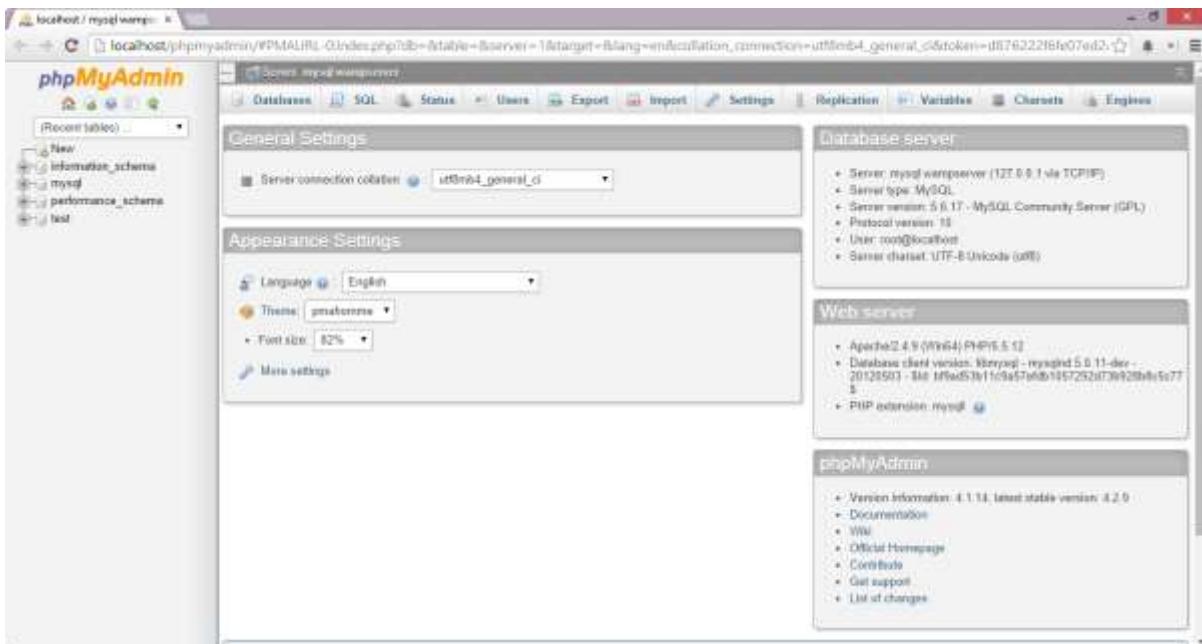


รูปที่ 73 การเปิดหน้า Web Application

จะแสดง Error เนื่องจากยังไม่ได้เขียนต่อฐานข้อมูล

## สร้างฐานข้อมูลสำหรับ yii2-advanced

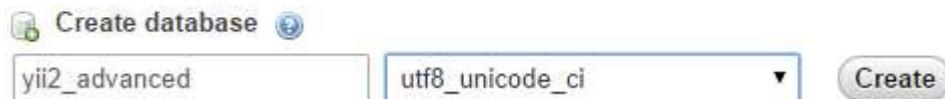
เปิด <http://localhost/phpmyadmin> หรือ <http://127.0.0.1/phpmyadmin>



รูปที่ 74 หน้าแรกของ PHPMyAdmin

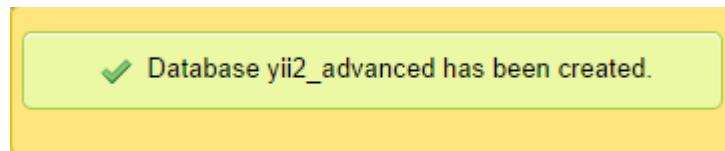
ไปที่เมนู Database และกรอกชื่อฐานข้อมูลในที่นี่จะใช้ชื่อว่า yii2\_advanced และ Collation เลือกเป็น utf8\_unicode\_ci และกดปุ่ม Create

## Databases



รูปที่ 75 การฐานข้อมูล

จะแสดง POPUP ขึ้นว่าสร้างฐานข้อมูลเสร็จเรียบร้อยแล้ว



รูปที่ 76 POPUP หลังการสร้างฐานข้อมูลเสร็จเรียบร้อยแล้ว

### แก้ไขการเชื่อมต่อฐานข้อมูล

แก้ไขไฟล์ติดต่อฐานข้อมูล common/config/main-local.php

```
1 <?php
2 return [
3     'components' => [
4         'db' => [
5             'class' => 'yii\db\Connection',
6             'dsn' => 'mysql:host=localhost;dbname=yii2_advanced',
7             'username' => 'root',
8             'password' => '',
9             'charset' => 'utf8',
10        ],
11    ],
12 ];
```

รูปที่ 77 การกำหนดการติดต่อฐานข้อมูล

dbname คือ ชื่อฐานข้อมูล

username ชื่อผู้ใช้ฐานข้อมูล

password คือรหัสผ่านเชื่อมต่อฐานข้อมูล

เรียกใช้คำสั่ง migrate เพื่อสร้างตารางในฐานข้อมูลที่ได้เชื่อมต่อไว้โดยพิมพ์คำสั่ง

```
yii migrate
```

แล้วกดปุ่ม Enter จากนั้นโปรแกรมจะให้ Confirm โดยพิมพ์ yes และกดปุ่ม Enter จะได้หน้าระบบดังนี้

```
ca.                                         Command Prompt
D:\wamp\www\yii2-advanced>yii migrate
Yii Migration Tool (based on Yii v2.0.0-dev)

Creating migration history table "migration"...done.
Total 1 new migration to be applied:
  m130524_201442_init

Apply the above migration? <yes|no> [no]:yes
*** applying m130524_201442_init
  > create table <{{user}} ... done (time: 1.463s)
*** applied m130524_201442_init (time: 1.627s)

Migrated up successfully.
D:\wamp\www\yii2-advanced>
```

รูปที่ 78 กด Migration

เพิ่ม User เข้าในระบบโดยคลิกที่เมนู Signup หรือ URL <http://localhost/yii2-advanced/frontend/web/index.php?r=site/signup>

## Signup

Please fill out the following fields to signup:

**Username**

admin

**Email**

contact@programmerthailand.com

**Password**

.....

**Signup**

รูปที่ 79 หน้าลงทะเบียนผู้ใช้งาน

ใน Username Email และ Password แล้วกด Signup จะเห็นว่า Yii Login เข้าระบบให้เรียบร้อย ในที่นี้จะใช้ admin และรหัสผ่าน 44121165



[www.programmerthailand.com](http://www.programmerthailand.com)

The screenshot shows the main page of a Yii-powered application. At the top, there's a navigation bar with 'My Company' on the left and links for 'Home', 'About', 'Contact', and 'Logout (admin)' on the right. The main content area features a large green header 'Congratulations!' with the subtext 'You have successfully created your Yii-powered application.' Below this is a green button labeled 'Get started with Yii'. The page is styled with a light gray background and some placeholder text ('Lorem ipsum...') in the sidebar.

รูปที่ 80 หน้าแรกของ Web Application

เปิดดูในฐานข้อมูลในตาราง user จะเห็นว่ามีข้อมูล 1 แล้วปรากฏอยู่

<b>id</b>	<b>username</b>	<b>auth_key</b>	<b>password_hash</b>	<b>password_reset_token</b>	<b>email</b>
1	admin	gt2CJnD9PDAJD-DXRhhF5Ylt0KSxwSi	\$2y\$13\$BaEbB12axoAjZB17.IGNuryi9sqng.1als6rfQ6n5x...	NULL	contact@programmerthailand.com

รูปที่ 81 Username หลังลงทะเบียน

และ URL <http://localhost/yii2-advanced/backend/web> จะได้หน้า Web Application ดังนี้

The screenshot shows the login page of the Yii application. It has a dark header with 'My Company' and a 'Login' link. The main content is titled 'Login' with the subtext 'Please fill out the following fields to login.' It contains two input fields for 'Username' and 'Password', a 'Remember Me' checkbox, and a blue 'Login' button. Below the form is a Yii debugger toolbar with various status indicators like Status 200, Action /backend/controllers/SiteController::actionLogin(), Log 13, Time 500 ms, Memory 6.8 MB, and Asset Bundles.

รูปที่ 82 หน้าเข้าสู่ระบบ

## การใช้ Migration ใน Yii Advanced

พิมพ์คำสั่งสำหรับสร้างตารางดังนี้

```
yii migrate/create create_post_table
```

แล้วกด Enter จะปรากฏให้ Confirm โดยพิมพ์ yes แล้วกด Enter อีกครั้ง



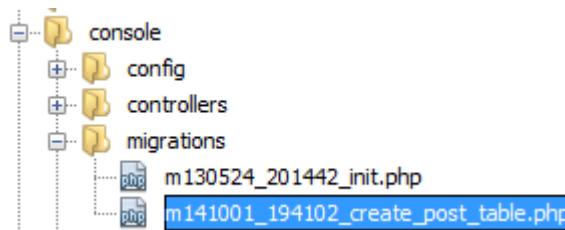
```
D:\wamp\www>cd yii2-advanced
D:\wamp\www\yii2-advanced>yii migrate/create create_post_table
Yii Migration Tool (based on Yii v2.0.0-dev)

Create new migration 'D:\wamp\www\yii2-advanced\console\migrations\m141001_194102_create_post_table.php'? <yes|no> [no]:yes
New migration created successfully.

D:\wamp\www\yii2-advanced>_
```

รูปที่ 83 การสร้างไฟล์ Migration

โดยไฟล์จะถูกเก็บไว้ที่ console/migrations



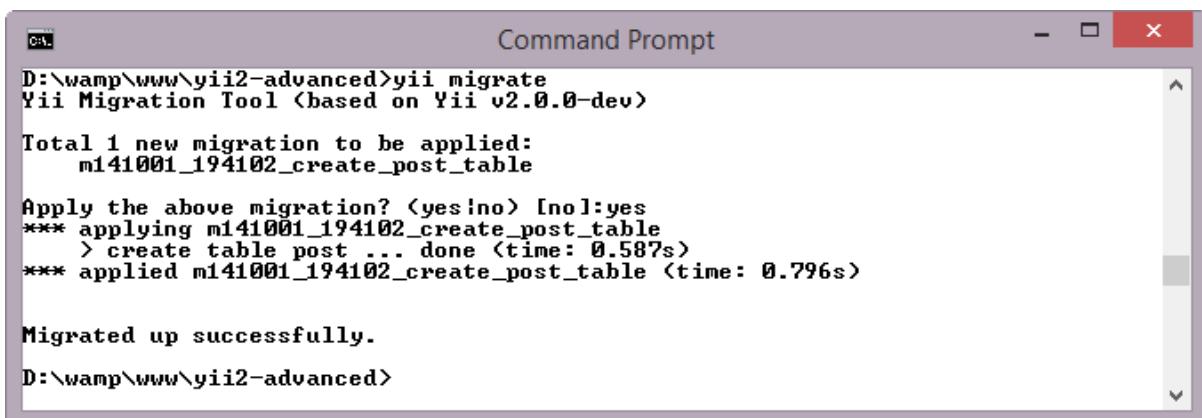
รูปที่ 84 ที่เก็บไฟล์ Migration

เปิดไฟล์เพื่อแก้ไข โดยเพิ่มคำสั่งให้สร้างตาราง post ดังนี้

```
<?php  
2  
3     use yii\db\Schema;  
4     use yii\db\Migration;  
5  
6     class m141001_194102_create_post_table extends Migration  
7     {  
8         public function up()  
9         {  
10             $this->createTable('post', [  
11                 'id' => 'pk',  
12                 'title' => Schema::TYPE_STRING . ' NOT NULL',  
13                 'content' => Schema::TYPE_TEXT,  
14             ]);  
15         }  
16  
17         public function down()  
18         {  
19             $this->dropTable('post');  
20             //echo "m141001_194102_create_post_table cannot be r  
21  
22             //return false;  
23         }  
24     }
```

รูปที่ 85 การแก้ไขไฟล์ Migration

เรียกคำสั่ง migration อีกครั้ง ระบบจะสร้างตาราง post ในฐานข้อมูลให้ทันที โดยพิมพ์ yii migrate แล้วกดปุ่ม Enter จากนั้นกด y แล้วกดปุ่ม Enter เพื่อยืนยันอีกครั้ง



```
D:\wamp\www\yii2-advanced>yii migrate  
Yii Migration Tool (based on Yii v2.0.0-dev)  
Total 1 new migration to be applied:  
m141001_194102_create_post_table  
  
Apply the above migration? <yes|no> [no]:yes  
*** applying m141001_194102_create_post_table  
    > create table post ... done (time: 0.587s)  
*** applied m141001_194102_create_post_table (time: 0.796s)  
  
Migrated up successfully.  
D:\wamp\www\yii2-advanced>
```

รูปที่ 86 การเรียกใช้งาน Migration

เปิดดูในฐานข้อมูลจะเห็นว่ามีตาราง post เกิดขึ้น

migration	Browse	Structure	Search	Insert	Empty	Drop	~3 InnoDB utf8_unicode_ci 16 KiB
post	Browse	Structure	Search	Insert	Empty	Drop	~0 InnoDB utf8_unicode_ci 16 KiB
user	Browse	Structure	Search	Insert	Empty	Drop	~1 InnoDB utf8_unicode_ci 16 KiB

รูปที่ 87 ตารางที่ได้ขึ้นกลังจากการเรียกใช้งาน Migration

จากนั้นให้แก้ไขโดยเพิ่ม Comment ให้เป็นภาษาไทยในแต่ละ Field

Name	Type	Length/Values	Default	Collation	Attributes	Null	A_I	Comments
file	VARCHAR	255	None	utf8_unicode_ci				ไฟล์
content	TEXT		NULL	utf8_unicode_ci				รายละเอียด

รูปที่ 88 การเพิ่ม Comment ใน Field ต่างๆ

แล้วกดปุ่ม Save เพื่อบันทึกการเปลี่ยนแปลง

## Gii Generator สำหรับ Advanced Template

การเข้าใช้งาน Gii Generator นั้นเข้าผ่าน URL <http://localhost/yii2-advanced/frontend/web/index.php?r=gii> จะปรากฏหน้า Web Application ดังนี้

Welcome to Gii a magical tool that can write code for you

Start the fun with the following code generators:

**Model Generator**  
This generator generates an ActiveRecord class for the specified database table.  
[Start ▾](#)

**CRUD Generator**  
This generator generates a controller and views that implement CRUD (Create, Read, Update, Delete) operations for the specified data model.  
[Start ▾](#)

**Controller Generator**  
This generator helps you to quickly generate a new controller class, one or several controller actions and their corresponding views.  
[Start ▾](#)

**Form Generator**  
This generator generates a view script file that displays a form to collect input for the specified model class.  
[Start ▾](#)

**Module Generator**  
This generator helps you to generate the skeleton code needed by a Yii module.  
[Start ▾](#)

**Extension Generator**  
This generator helps you to generate the files needed by a Yii extension.  
[Start ▾](#)

Yii Debugger | Yii 2.0.0-beta PHP 5.5.17 Status 200 Action: /gii/generator/test/default/index [Time: 352 ms] Log: 11 | Time: 352 ms Memory: 6.2 MB Asset Bundles: 0

รูปที่ 89 หน้าแรกของ Gii Generator

## การสร้าง Model ด้วย Gii

การสร้าง Model ด้วย Gii นั้นมีขั้นตอนที่ง่ายดายเพียงเข้าเมนู Model Generator โดยคราวที่จะสร้าง Model ก่อนสร้างอย่างอื่น เพราะต้องใช้ Model ในการสร้างอย่างอื่น เช่น CRUD Generator ดังนั้นจะต้องสร้าง Model ขึ้นมาก่อน หลังจากคลิกที่ปุ่ม Start ในส่วนของ Model Generator จะปรากฏหน้า Web Application ดังนี้



รูปที่ 90 หน้า Model Generator

จากนั้นใส่รายละเอียดของ Model ที่ต้องการสร้าง ในที่นี้ต้องการสร้าง Model post จากตารางในฐานข้อมูล ใส่รายละเอียด ชื่อตาราง และชื่อ Model จากนั้นกด Preview

# Model Generator

This generator generates an ActiveRecord class for the specified database table.

## Table Name

post

## Model Class

Post

## Namespace

app\models

## Base Class

yii\db\ActiveRecord

## Use Table Prefix

## Generate Relations

## Generate Labels from DB Comments

## Enable I18N

## Code Template

default (D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\model/default)

Preview

Generate

รูปที่ 91 การกำหนดค่าต่างๆ ในการ Generate Model

เมื่อกำหนดตามต้องการแล้วกดปุ่ม Generate จะปรากฏข้อความว่าสร้าง Model เสร็จเรียบร้อยแล้ว

The code has been generated successfully.

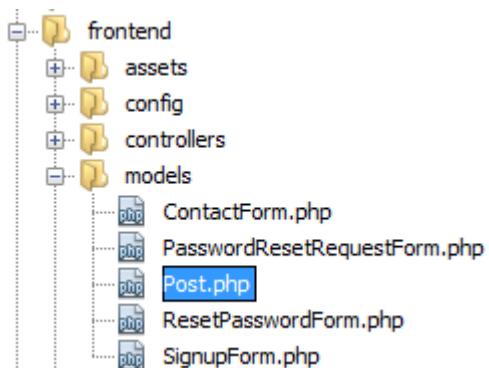
```
Generating code using template "D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\model/default"...
generated models\Post.php
done!
```

รูปที่ 92 หน้าหลักการ Generate Model



www.programmerthailand.com

ช่อง Model จะไปสร้างที่ frontend/models



รูปที่ 93 ตำแหน่งไฟล์ Model ที่ได้สร้างขึ้น

## การสร้าง CRUD ด้วย Gii

CRUD เป็นการ Generate คำสั่งพื้นฐานที่ใน 1 ตารางจำเป็นต้องมีนั่นคือ Create Read Update และ Delete โดยใน Yii Framework มีตัวช่วยในการสร้างเรียกว่า CRUD Generator โดยระบุ Model ที่ต้องการสร้างดังนี้

เข้าเมนู CRUD Generator จากนั้นสร้าง CRUD สำหรับ Model Post ที่ได้สร้างไว้ก่อนหน้านี้ โดยตั้งค่าดังนี้

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update model).

## Model Class

frontend\models\Post

## Search Model Class

frontend\models\PostSearch

## Controller Class

frontend\controllers\PostController

## Base Controller Class

## Widget Used in Index Page

GridView

## Enable I18N

## Code Template

default (D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\crud/default)

Preview

รูปที่ 94 การกำหนดค่าในการสร้าง CRUD

จากนั้นกดปุ่ม Preview เพื่อแสดงตัวอย่าง จะแสดง Error ขึ้นลักษณะนี้

## Model Class

frontend\models\Post

Class 'frontend\models\Post' does not exist or has syntax error.

รูปที่ 95 การแสดงข้อผิดพลาด

ให้เปลี่ยนชื่อในส่วน namespace จาก app\models เป็น namespace frontend\models ในไฟล์ frontend\models\Post.php



www.programmerthailand.com

```
1 <?php  
2  
3 namespace frontend\models;  
4
```

รูปที่ 96 การแก้ไข namespace

จากนั้นกดปุ่ม Preview อีกครั้ง

Click on the above **Generate** button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	
controllers\PostController.php	create	<input checked="" type="checkbox"/>
models\PostSearch.php	create	<input checked="" type="checkbox"/>
views\post\_form.php	create	<input checked="" type="checkbox"/>
views\post\_search.php	create	<input checked="" type="checkbox"/>
views\post\create.php	create	<input checked="" type="checkbox"/>
views\post\index.php	create	<input checked="" type="checkbox"/>
views\post\update.php	create	<input checked="" type="checkbox"/>
views\post\view.php	create	<input checked="" type="checkbox"/>

รูปที่ 97 หน้าหลังจากกดปุ่ม Preview

และกดปุ่ม Generate จะปรากฏข้อความว่าสร้าง Code สำเร็จแล้ว

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated controllers\PostController.php
generated models\PostSearch.php
generated views\post\_form.php
generated views\post\_search.php
generated views\post\create.php
generated views\post\index.php
generated views\post\update.php
generated views\post\view.php
done!
```

รูปที่ 98 หน้าหลังจากกดปุ่ม Generate

ทดสอบเปิด URL สำหรับเรียก Post ดังนี้ <http://localhost/yii2-advanced/frontend/web/index.php?r=post> จากนั้นทดสอบสร้าง Post โดยกดปุ่ม Create Post

My Company

Home About Contact Signup Login

Home Posts Create Post

## Create Post

หัวข้อ

สร้างหัวข้อ Web Application ด้วย Yi Framework ของคุณ

รายละเอียด

รายละเอียดของหัวข้อ Web Application ด้วย Yi Framework ของคุณ ที่คุณ ต้องการจะนำเสนอ

**Create**

จากนั้นกดปุ่ม Create Yii Framework จะเปิดหน้า View สำหรับดูรายการที่ Post ให้โดยอัตโนมัติ

ID	1
ชื่อ	มาเรียมกันพัฒนา Web Application ด้วย Yii Framework นะครับ
รายละเอียด	ตอนนี้เรามาเรียนรู้การพัฒนา Web Application ด้วย Yii Framework บนเบื้องต้น ก่อนจะไปเรียนรู้รายลึกต่อไป

ซึ่งสามารถแก้ไข หรือ Update ข้อมูล และสามารถ ลบ หรือ Delete ข้อมูลได้ ตามปุ่มที่แสดงนั้น

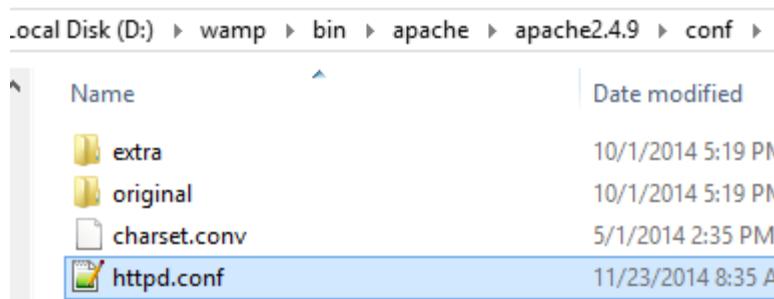
# การสร้าง Virtual Host สำหรับ Yii2 Advanced Application Template

สำหรับ Yii2 Advanced Application Template นั้นมี 2 ส่วนด้วยกันนั่นคือ Frontend และ Backend โดยทำ Virtual Host ให้กับ 2 ส่วนนี้ โดย Frontend นั้นจะเป็นค่าเริ่มต้นสำหรับ Domain เช่น

http://yii2-advanced.dev ส่วน Backend นั้นจะเป็น Subdomain เช่น http://backend.yii2-advanced.dev โดยมีการตั้งค่าดังนี้

## แก้ไขไฟล์ *httpd-vhosts.conf*

สำหรับไฟล์นี้จะบรรจุอยู่ใน d:/wamp/bin/apache/apachex.x.x/conf/extra/httpd-vhosts.conf แต่ก่อนที่จะไปแก้ไขไฟล์นี้ ต้องทำให้แน่ใจก่อนว่าได้เปิดการใช้งานแล้วในการตั้งค่าในไฟล์ httpd.conf ที่อยู่ใน d:/wamp/bin/apache/apachex.x.x/conf/httpd.conf



รูปที่ 101 ที่เก็บไฟล์ *httpd.conf*

จากนั้นแก้ไขโดยยกเลิก Comment ในบรรทัด 513 เพื่อให้เรียกไฟล์ *httpd-vhosts.conf* ได้

```
512 # Virtual hosts
513 Include conf/extra/httpd-vhosts.conf
```

รูปที่ 102 การเปิดใช้งาน Virtual Host

ทำการบันทึกไฟล์ *httpd.conf*

จากนั้นเปิดไฟล์ *httpd-vhosts.conf* และทำให้แน่ใจว่ามีคำสั่งต่อไปนี้อยู่หากไม่มีให้เพิ่ม

```
<VirtualHost *:80>
    ServerName localhost
    DocumentRoot "d:/wamp/www/"
</VirtualHost>
```

แล้วเพิ่มคำสั่งในการสร้าง Virtual Host ดังนี้

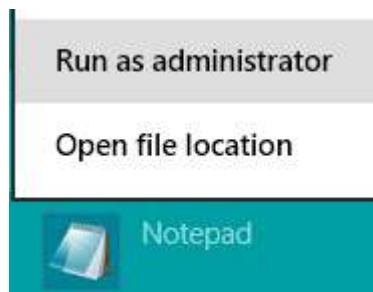
```
<VirtualHost *:80>
    ServerAdmin webmaster@yii2-advanced.dev
    DocumentRoot "d:/wamp/www/yii2-advanced/frontend/web"
```

```
ServerName yii2-advanced.dev
ErrorLog "logs/yii2-advanced.log"
CustomLog "logs/yii2-advanced.log" common
</VirtualHost>
<VirtualHost *:80>
    ServerAdmin webmaster@yii2-advanced.dev
    DocumentRoot "d:/wamp/www/yii2-advanced/backend/web"
    ServerName backend.yii2-advanced.dev
    ErrorLog "logs/yii2-advanced.log"
    CustomLog "logs/yii2-advanced.log" common
</VirtualHost>
```

ส่วนที่สำคัญคือ DocumentRoot คือ Root ของ Web Application และ ServerName คือ ชื่อ Domain ที่เราต้องการ แล้วทำการบันทึกไฟล์ httpd-vhosts.conf

## แก้ไขไฟล์ *c:/windows/system32/driver/etc/hosts*

เปิด Notepad แบบ Administrator คลิกขวา Run as administrator



รูปที่ 103 การเปิด Notepad แบบ Administrator

จากนั้นไปที่ File->Open แล้วไปเปิดไฟล์ hosts ใน c:/windows/system32/driver/etc/ และเพิ่มข้อมูลต่อไปนี้

127.0.0.1	yii2-advanced.dev
127.0.0.1	backend.yii2-advanced.dev

รูปที่ 104 การเพิ่ม Domain ใน localhost

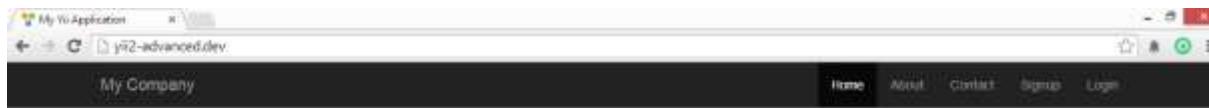
จากนั้นบันทึก และทำการ Restart Apache โดยคลิก W สีเขียวแล้วเลือก Restart All Services



รูปที่ 105 การ Restart All Services

หมายเหตุ ชื่อ Domain ใน httpd-vhosts.conf กับ ชื่อ Domain ใน hosts นั้นต้องเป็นชื่อเดียวกัน

จากนั้นทดลองเข้า URL ตามชื่อ Domain <http://yii2-advanced.dev> จะได้หน้า Web Application ดังนี้



## Congratulations!

You have successfully created your Yii-powered application.

[Get started with Yii](#)

### Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

### Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

### Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

[Yii Documentation](#)

[Yii Forum](#)

[Yii Extensions](#)

[Status](#)

[Log](#)

[Time](#)

[Memory](#)

[Asset Bundles](#)

[Yiisoft](#)

[Yii2](#)

[Yii2 Advanced](#)

[Yii2 Basic](#)

[Yii2 Gii](#)

[Yii2 Test](#)

[Yii2 Gridview](#)

[Yii2 Kartik](#)

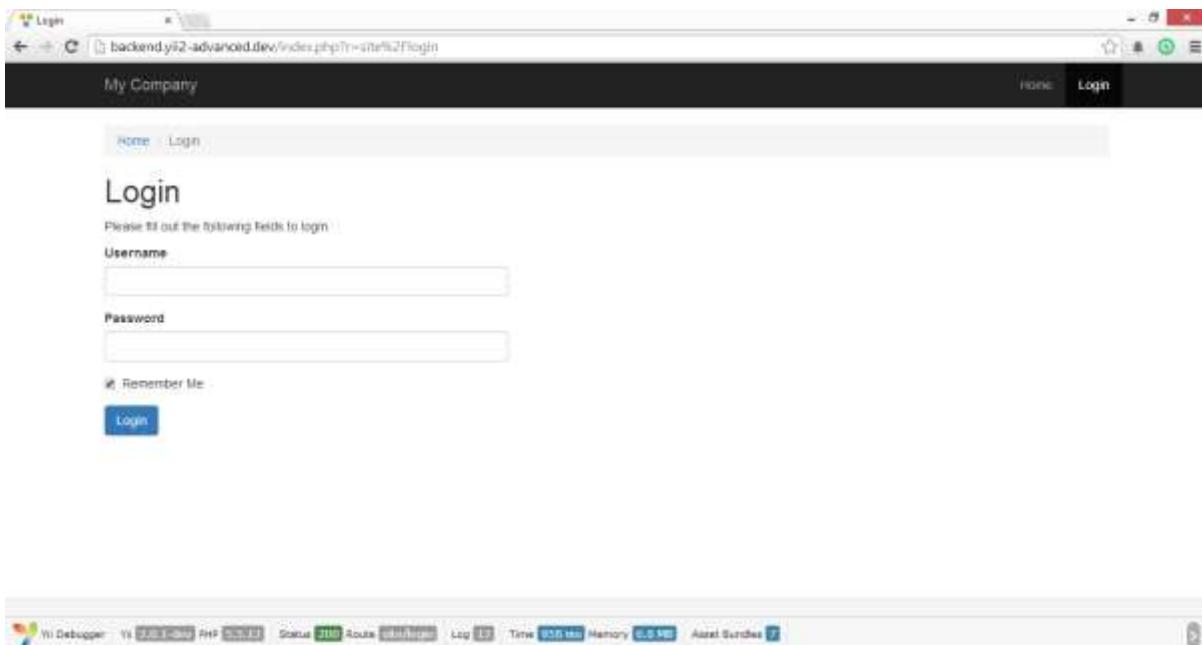
[Yii2 Pdf](#)

[Yii2 Gii](#)

[Yii2 Gridview](#)

[Yii2 Kartik](#)

[Yii2 Pdf](#)

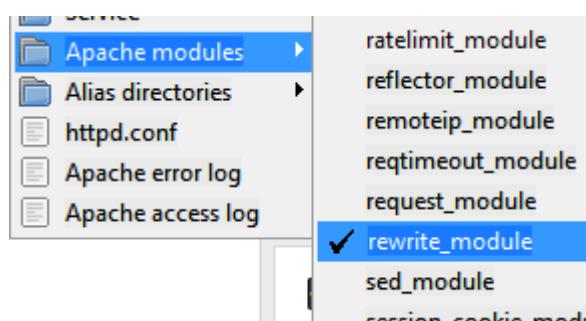


รูปที่ 107 หน้าแรกของ Web Application ส่วน Backend

## การจัดการ URL แบบสวยงาม (Pretty URL)

สำหรับ Pretty URL สำหรับ Yii Advanced Application Template นั้นเราต้องจัดการ 2 ส่วน คือ ส่วนของ Frontend และส่วนของ Backend

สำหรับการทำ Pretty URL นั้นจำเป็นต้องเปิดใช้งาน rewrite\_module ของ Apache ก่อน เพื่อให้การทำงานของขั้นตอนนี้ถูกต้อง ถ้าเป็น WAMP Server ให้กด W->Apache->Apache modules ทำให้แน่ใจว่า มีเครื่องหมายถูกหน้า rewrite\_module



รูปที่ 108 การเปิดใช้งาน rewrite\_module ใน Apache

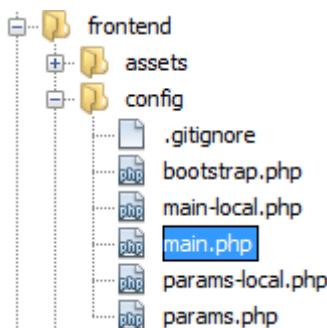
โดยมีการจัดการและตั้งค่าต่างๆ ดังนี้

## การจัดการ URL สำหรับ Frontend

สำหรับ Frontend นั้นเป็นการจัดการส่วนการแสดงผลด้านหน้า Web Application ซึ่งมีการจัดการและตั้งค่าต่างๆ ดังนี้

### เพิ่ม UrlManager

เปิดไฟล์ การตั้งค่า frontend/config/main.php



รูปที่ 109 ตำแหน่งไฟล์ main.php

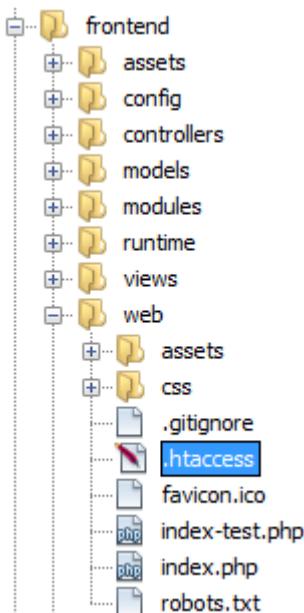
จากนั้นเพิ่ม Array urlManager ใน components ดังนี้

```
19 |     'components' => [
20 |         ...
21 |         'urlManager' => [
22 |             'class' => 'yii\web\UrlManager',
23 |             'enablePrettyUrl' => true,
24 |             'showScriptName' => false,
|         ],
|     ],
| 
```

รูปที่ 110 การกำหนด components โดยใช้ UrlManager

## สร้างไฟล์ .htaccess

ทำการสร้างไฟล์ .htaccess ใน frontend/web



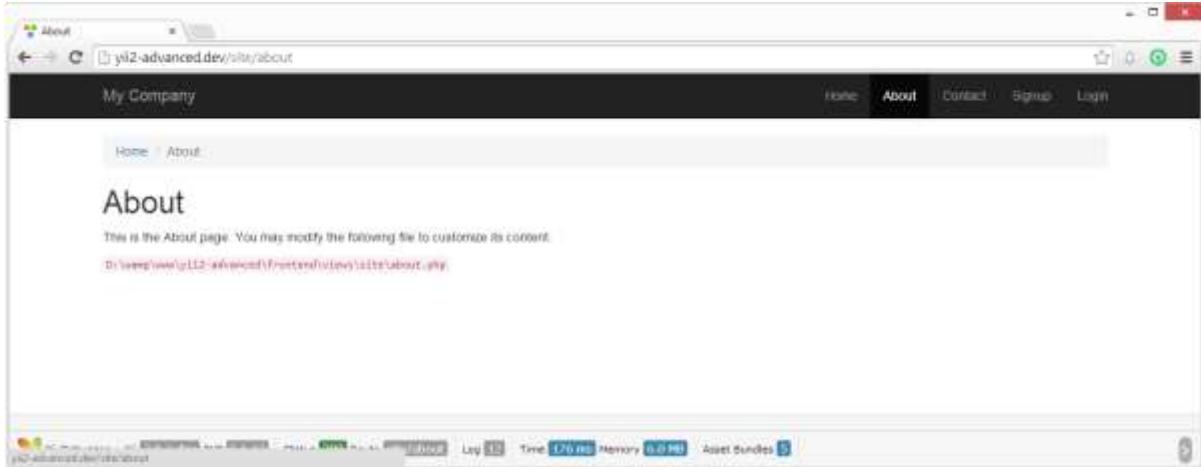
รูปที่ 111 ตำแหน่งไฟล์ .htaccess

จากนั้นเพิ่มคำสั่งดังนี้

```
1 RewriteEngine on
2 # if a directory or file exists, use it directly
3 RewriteCond %{REQUEST_FILENAME} !-f
4 RewriteCond %{REQUEST_FILENAME} !-d
5
6 # otherwise forward it to index.php
7 RewriteRule . index.php
```

รูปที่ 112 คำสั่งสำหรับไฟล์ .htaccess

ทดลองคลิกที่เมนูต่างๆ



รูปที่ 113 การทดสอบเข้า Link ต่างๆ

จากรูปภาพด้านบนจะเห็นว่า URL กล้ายเป็น

`http://yii2-advanced.dev/site/about`

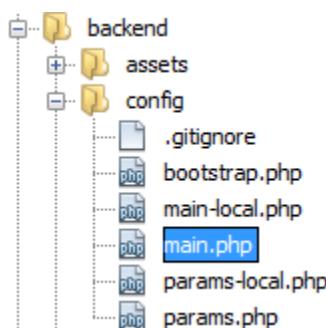
จะเห็นว่า index.php ไม่มี และ ตัวแปร ?r= ก็ไม่มีเช่นเดียวกัน

## การจัดการ URL สำหรับ Backend

สำหรับ Backend นั้นเป็นการจัดการส่วนแสดงผลหลังบ้าน โดยมีขั้นตอนวิธีการจัดการและตั้งค่าต่างๆ ดังนี้

### เพิ่ม UrlManager

เปิดไฟล์ การตั้งค่า backend/config/main.php



รูปที่ 114 ตำแหน่งไฟล์ main.php

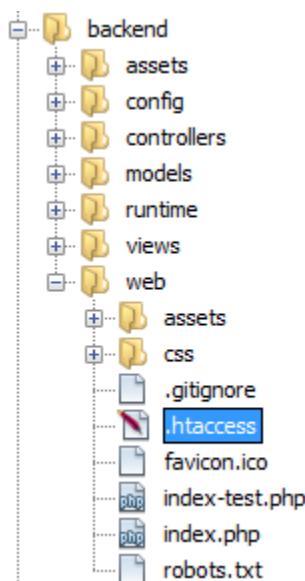
แล้วเพิ่ม Array UrlManager ในส่วน components ดังนี้

```
15     'components' => [
16         'urlManager' => [
17             'class' => 'yii\web\UrlManager',
18             'enablePrettyUrl' => true,
19             'showScriptName' => false,
20         ],
21     ],
22 
```

รูปที่ 115 การกำหนด Component โดยใช้ UrlManager

## สร้างไฟล์ .htaccess

สร้างไฟล์ .htaccess ใน backend/web



รูปที่ 116 ตำแหน่งไฟล์ .htaccess

จากนั้นเพิ่มคำสั่งดังนี้

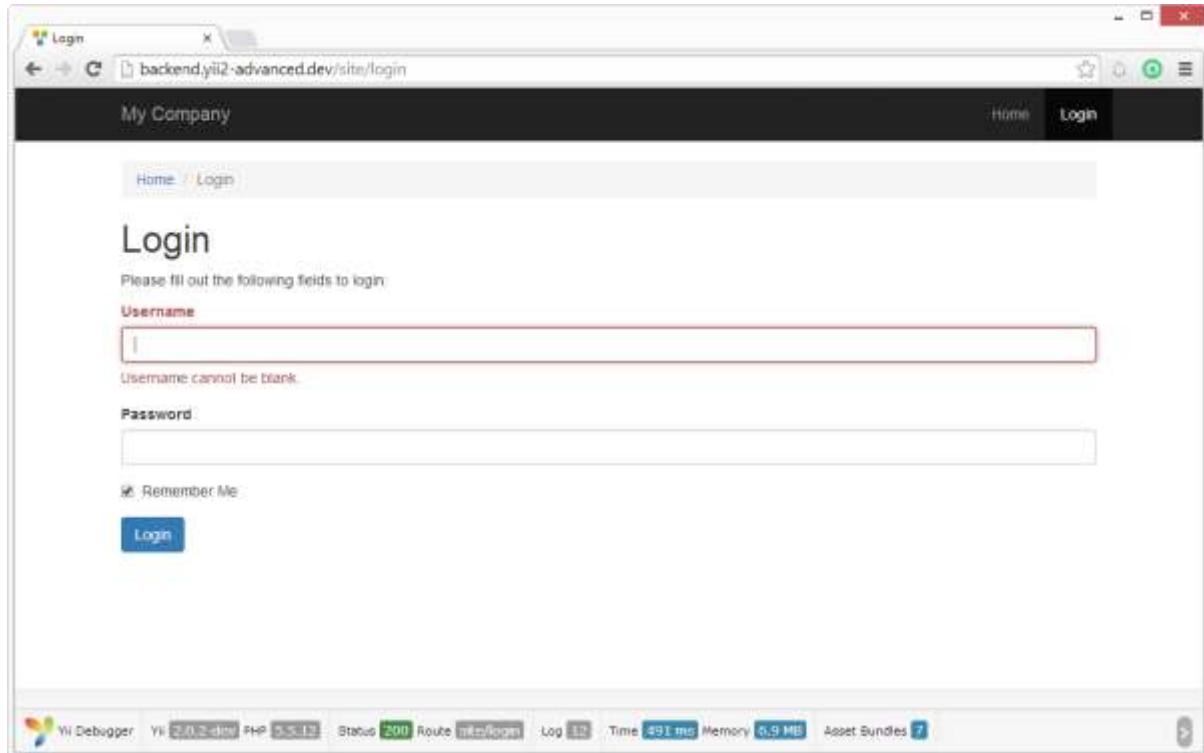
```
1 RewriteEngine on
2 # if a directory or file exists, use it directly
3 RewriteCond %{REQUEST_FILENAME} !-f
4 RewriteCond %{REQUEST_FILENAME} !-d
5
6 # otherwise forward it to index.php
7 RewriteRule . index.php
```

รูปที่ 117 คำสั่งในไฟล์ .htaccess

ทดลองเปิด URL

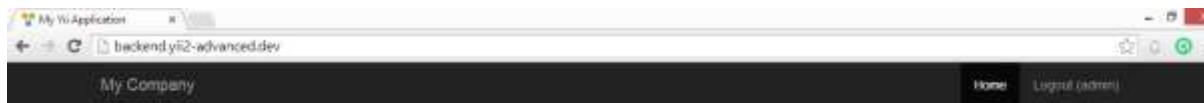
<http://backend.yii2-advanced.dev/>

จะเห็นว่าระบบจะ Redirect ไปยังหน้า Login



รูปที่ 118 การเปิด Web Application ส่วน Backend

เมื่อเข้าสู่ระบบเรียบร้อยแล้วจะได้หน้า Web Application ดังนี้



## Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

[Yii Documentation](#)

## Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

[Yii Forum](#)

## Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

[Yii Extensions](#)

รูปที่ 119 หน้า Web Application หลังจาก Login เสร็จเรียบร้อยแล้ว



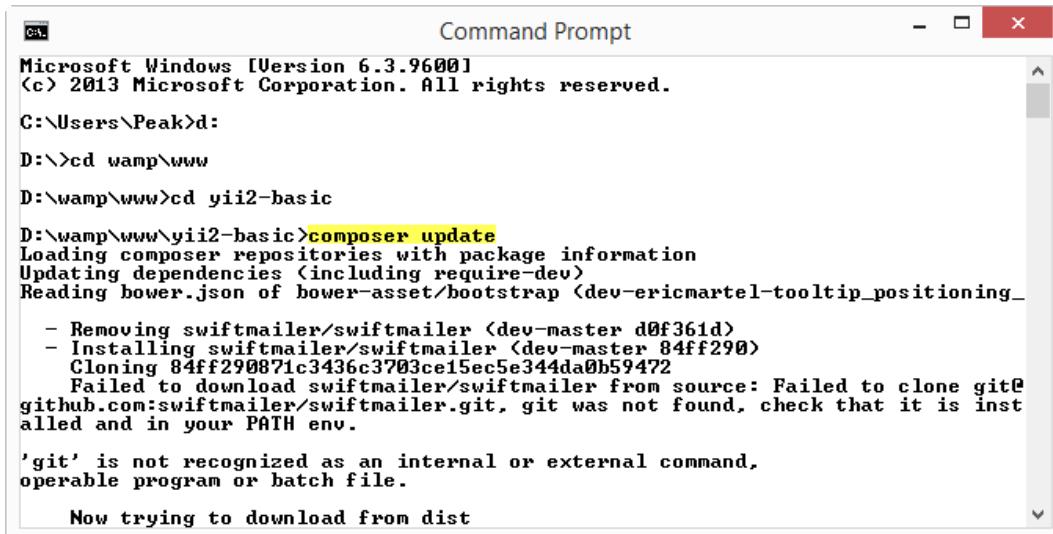
[www.programmerthailand.com](http://www.programmerthailand.com)

## การ Update Yii2

สำหรับ Yii2 เมื่อเวลาผ่านไปทาง Yii2 Official ได้มีการอัพเดทหรือปรับปรุงคุณสมบัติต่างๆ เพิ่มเติม ในระบบ และส่วนเสริมต่างๆ เพื่อให้ระบบ Web Application ที่เราสร้างมีความเป็นปัจจุบันเรاجึงต้องมีการ อัพเดทด้วยเซ่นกัน สำหรับคำสั่งในการอัพเดทนั้นให้เข้าไปที่ Folder Project และใช้คำสั่งดังนี้

```
composer update
```

หลังจากที่พิมพ์คำสั่งในการอัพเดท Yii2 และกดปุ่ม Enter จะปรากฏหน้า Command Prompt ดังนี้



```
Command Prompt
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Peak>d:
D:>cd wamp\www
D:\wamp\www>cd yii2-basic
D:\wamp\www\yii2-basic>composer update
Loading composer repositories with package information
Updating dependencies (including require-dev)
Reading bower.json of bower-asset/bootstrap <dev-ericmartel-tooltip_positioning_
- Removing swiftmailer/swiftmailer (dev-master d0f361d)
- Installing swiftmailer/swiftmailer (dev-master 84ff290)
  Cloning 84ff290871c3436c3703ce15ec5e344da0b59472
Failed to download swiftmailer/swiftmailer from source: Failed to clone git@github.com:swiftmailer/swiftmailer.git, git was not found, check that it is installed and in your PATH env.

'git' is not recognized as an internal or external command,
operable program or batch file.

Now trying to download from dist
```

รูปที่ 120 คำลั่งในการ Update Composer

ก็เป็นอันเสร็จเรียบร้อยสำหรับการอัพเดท Yii2 และส่วนเสริมต่างๆ

## การ Upgrade Version

สำหรับการ Upgrade Version ใหม่ของ Yii2 นั้นสามารถทำได้โดยพิมพ์คำสั่ง require เข้าไปใน composer เพื่อให้สามารถสั่ง Update ผ่าน composer ได้ ดังนี้

```
composer require yiisoft/yii2 2.0.1
```

เช่นตัวอย่างนี้เป็นการ Upgrade เป็น Yii 2.0.1 จะพิมพ์คำสั่งดังนี้

```
Command Prompt  
D:\wamp\www>cd yii2-basic  
D:\wamp\www\yii2-basic>composer require yiisoft/yii2 2.0.1  
./composer.json has been updated  
Loading composer repositories with package information  
Updating dependencies (including require-dev)  
- Removing yiisoft/yii2 <dev-master 3b09730>  
- Installing yiisoft/yii2 <2.0.1>  
  Downloading: 100%  
  
Writing lock file  
Generating autoload files  
D:\wamp\www\yii2-basic>
```

รูปที่ 121 คำสั่งในการ Update Yii2



www.programmerthailand.com

## บทที่ 3 เข้าใจ Yii2

Yii2 หรือ PHP Framework โดยทั่วไปจะถูกออกแบบสถาปัตยกรรมแบบ MVC ซึ่งจะแบ่งส่วน Model View และ Controller ในบทนี้ จะนำไปสู่การเรียนรู้ Yii2 เพื่อให้เข้าใจหลักการและโครงสร้างต่างๆ ที่ควรรู้ก่อนการพัฒนา Web Application

### หลักการ MVC

ก่อนที่จะเริ่มการพัฒนา Web Application ด้วย Yii2 นั้นความมาทำความเข้าใจในหลักการของ Model View และ Controller หรือเรียกย่อๆ ว่า MVC ก่อน เพื่อให้เข้าใจในแนวคิดของการออกแบบ

### เข้าใจ Model

Model เป็นส่วนที่ทำงานกับข้อมูลซึ่งมี 2 แบบนั่นคือ Model แบบใช้ตาราง และ Model แบบไม่ใช้ตาราง

Model แบบใช้ตาราง 1 ตารางคือ 1 Model ซึ่งรายใน Model จะประกอบไปด้วย Attribute ที่มาจาก Field ต่างๆ ของตารางเก็บข้อมูลในคำศัพท์ด้านฐานข้อมูลเรียกว่า Attribute และมีส่วนของการตรวจสอบความถูกต้องของข้อมูลอยู่ด้วย ดังนั้นการทำงานที่เกี่ยวข้องกับ Model จะเขียนโปรแกรมในส่วนของ Controller ให้สามารถ Create Read Update และ Delete ข้อมูลได้

Model แบบไม่ใช้ตาราง เช่น Model ในการสร้างฟอร์ม โดยจะมีส่วนของการตรวจสอบความถูกต้องของข้อมูลซึ่งหลังจากที่มีการตรวจสอบแล้วจะต้องเขียนโปรแกรมใน Controller ให้กระทำการบางอย่างกับข้อมูลใน Model นี้ เช่น ฟอร์ม Contact หลังจากที่มีการส่งข้อมูลมาให้ส่ง Email ไปยัง Email ที่ต้องการ เป็นต้น

### เข้าใจ View

View คือส่วนการแสดงผล หรือ GUI (Graphic User Interface) สำหรับนำข้อมูลที่ได้จากการประมวลผลมาสร้างการแสดงผลใน View เช่น เขียนโปรแกรมใน Controller เพื่อดึงข้อมูลจาก Model Post จากนั้นนำข้อมูลทั้งหมดมาแสดงเป็นตารางในส่วน View เป็นต้น

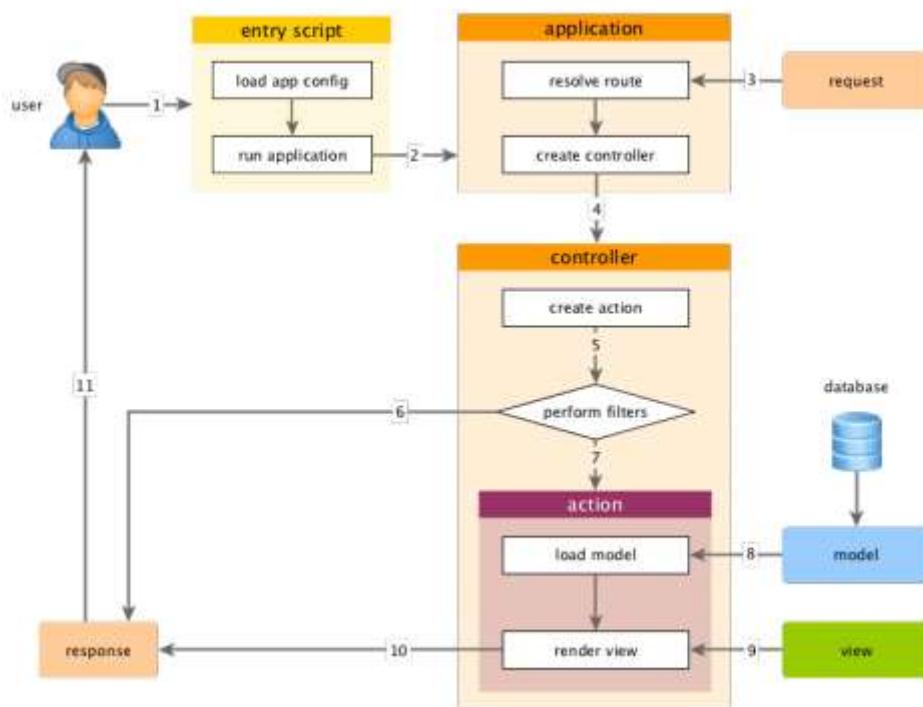
### เข้าใจ Controller

Controller คือส่วนการประมวลผลบางอย่างกับข้อมูล เช่น การรับข้อมูลจากฟอร์มใน View และ เขียนโปรแกรมให้บันทึกข้อมูลผ่าน Model เป็นต้น

ใน Controller จะมี Program ต่างๆ ที่ถูกเขียนขึ้นเพื่อให้ Web Application ทำงานได้ตรงตามความต้องการ เช่น การเลือกรายการสินค้าทั้งหมดมาพร้อมกับหมวดหมู่สินค้าจาก Model และส่งชุดข้อมูลนี้ไปแสดงผลที่ View เป็นต้น

## การทำงานของ Yii2

ทำความเข้าใจหลักการทำงานของ Yii2 เพื่อให้เข้าใจการประมวลผลในส่วนต่างๆ ว่ามีขั้นตอนอย่างไรบ้าง



รูปที่ 122 หลักการทำงานของ Yii2

- เมื่อผู้ใช้งานทำการ Request หน้า web/index.php
- ไฟล์สคริป index.php จะโหลดการตั้งค่าของ Application และสร้าง Application Instance ขึ้นมาเพื่อทำการใช้งานการ Request
- จากนั้น Application จะทำการสร้างเส้นทางจากตัวช่วยสร้างเส้นทางใน Application Component
- Application จะสร้าง Controller Instance และทำการเรียกใช้งาน
- Controller สร้าง Instance และ ดำเนินการเรียกใช้ Filter สำหรับ Action
- ถ้า Filter มีข้อผิดพลาด Action จะถูกยกเลิก
- ถ้า Filter ผ่าน Action จะถูกประมวลผล
- Action มีการโหลด Data Model ซึ่งอาจมาจากฐานข้อมูล

9. Action ทำการ Render ส่วน View ซึ่งมีข้อมูลจาก Model
10. การ Render ส่งค่าตอบกลับไปยัง Application Component
11. ส่วนของ Response Component จะส่งผลลัพธ์จากการ Render ไปยัง Browser ของ User

ที่มา: *The Definition Guide to Yii 2.0* หน้า 23

สำหรับ Yii2 นั้น Method หลัก เช่น create, index, update, และ view จะเรียกว่า Action โดยส่วนใหญ่จะเป็น Method ที่ Render View และเป็น Method ที่สามารถเรียกผ่าน URL ได้ โดยการตั้งชื่อ Method จะมีคำว่า action นำหน้า เช่น

```
public function actionShop () {  
}
```

เวลาเรียกผ่าน URL จะเรียก index.php?r=controller/shop สามารถเพิ่มเติมในหัวข้อ เรียนรู้ Routing ใน Yii2

## โครงสร้าง Yii2

ใน Yii2 นั้นมีการเปลี่ยนแปลงในส่วนของโครงสร้างไฟล์ โดยตัด Folder protected ใน yii1.1 ออก แต่เปลี่ยน Web Root เป็น basic/web แทน ซึ่งต้องมาตั้งค่าในส่วน Web Server ให้ชี้ Web Root มา Folder นี้อีกรอบ สำหรับรายละเอียดโครงสร้างไฟล์มีดังนี้

basic/	ที่อยู่ของ Web Application
composer.json	ไฟล์สำหรับ composer ที่รวม package ต่างๆ
config/	ที่เก็บการตั้งค่าสำหรับ Web Application และการตั้งค่าต่างๆ
console.php	การตั้งค่าสำหรับ console
web.php	การตั้งค่าสำหรับ Web application
commands/	ที่เก็บ Class ต่างๆ สำหรับ console
controllers/	ที่เก็บ Controller ของ Web Application
models/	ที่เก็บ Model ของ Web Application
runtime/	ที่เก็บไฟล์ที่สร้างโดย Yii ในขณะที่ทำงาน เช่นไฟล์ logs และ cache
vendor/	ที่เก็บ Package ต่างๆ ที่ติดตั้ง รวมไปถึง Yii Framework



views/	ที่เก็บ View ของ Web Application
web/	เป็น Web Root ของ Web Application
assets/	ที่เก็บ assets ไฟล์ เช่น javascript และ css)
index.php	ไฟล์เริ่มต้นสำหรับการเรียก Web Application
yii	Yii console สำหรับ execute คำสั่งต่างๆ

## Hello World กับ Yii2

มาทดลอง Hello World กับ Yii2 และทดสอบการส่งค่าเข้า action ผ่าน URL โดยเปิดไฟล์ controllers/SiteController.php แล้วเพิ่ม action Hello() ด้านล่าง action Index() ดังนี้

```
48     public function actionIndex() {
49         return $this->render('index');
50     }
51     public function actionHello($var=null) {
52         return $this->render('hello', ['myvar'=>$var]);
53     }
```

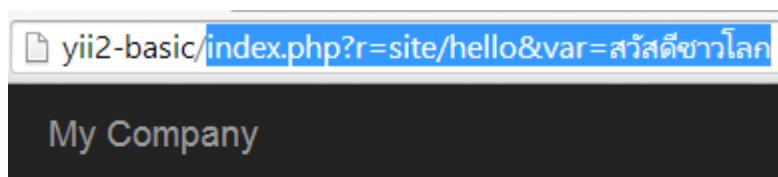
รูปที่ 123 การเพิ่ม actionHello()

จากนั้นเพิ่ม View ใน views/site/hello.php และเขียนโปรแกรมแสดงค่าตัวแปรดังนี้

```
1 <?php
2 echo $myvar;
```

จากนั้นทดลองเรียก URL index.php?r=site/hello&var=สวัสดีชาวโลก

จะเห็น Web Application แสดงผลดังนี้



สวัสดีชาวโลก

รูปที่ 124 การเรียก URL



จะเห็นว่าเราสามารถส่งตัวแปรเข้าไปใน Action ได้เลย โดยกำหนดชื่อตัวแปรให้ตรงกัน ในที่นี่คือ var จากนั้นนำ var ไป set ค่า ให้กับ myvar เพื่อส่งต่อให้ views/site/hello.php นำไปแสดงผลนั้นเอง

## คุณสมบัติพื้นฐานที่ควรเข้าใจ

### เรียนรู้ Routing ใน Yii2

Routing แบล็คแบบตรงตัวคือ การกำหนดเส้นทาง ในที่คือการกำหนดเส้นทางเพื่อเรียกใช้งาน Controller และ Action ใน URL เช่น index.php?r=site/index เป็นต้น

site คือชื่อของ Controller

index คือชื่อของ Action

### สร้าง Controller และ Action ใหม่

สร้างไฟล์ TestController.php ใน controllers/



รูปที่ 125 ตำแหน่งไฟล์ TestController.php

กำหนด namespace และ เรียกใช้คำสั่ง use

```
2     namespace app\controllers;
3
4     use Yii;
5     use yii\web\Controller;
```

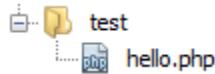
รูปที่ 126 การใช้งาน namespace และ use

สร้าง Class TestController

```
7     class TestController extends Controller{
8         public function actionHello(){
9             return $this->render('hello');
10        }
11    }
```

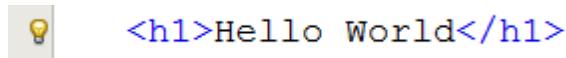
รูปที่ 127 Class TestController

สร้าง Folder สำหรับเก็บ view ใน views/test/ และสร้างไฟล์แสดงผล hello.php



รูปที่ 128 ตำแหน่งไฟล์ hello.php

เขียนการแสดงผลใน hello.php ตามต้องการในที่นี่คือ



รูปที่ 129 ข้อมูลในไฟล์ hello.php

ทดสอบเรียกการแสดงผลใน URL index.php?r=test/hello



## Hello World

รูปที่ 130 หน้า Web Application หลังเรียก actionHello()

### การส่งค่าให้ view

การส่งค่าให้ view นั้นสามารถทำได้โดยส่งข้อมูลออกไปในลักษณะ array() ดังนี้

```
7 class TestController extends Controller{
8     public function actionHello() {
9         $var = 'Manop Kongoon';
10        return $this->render('hello', array('myvar'=>$var));
11    }
12 }
```

รูปที่ 131 การส่งค่าตัวแปรให้ View

หรือสามารถกำหนด Array เป็นลักษณะนี้ได้

```
7 | class TestController extends Controller{  
8 | | public function actionHello(){  
9 | | | $var = 'Manop Kongoon';  
10| | | return $this->render('hello', ['myvar'=>$var]);  
11| }  
12| }
```

รูปที่ 132 การกำหนด Array สำหรับ PHP 5.4 ขึ้นไป

ความหมายคือนำค่าในตัวแปร \$var มากำหนดให้ myvar และส่งให้ view

เรียกใช้งานใน view ดังนี้

```
1 | <h1>Hello World</h1>  
2 | <?php echo $myvar;?>  
3 | 
```

รูปที่ 133 การนำค่าตัวแปรมาแสดงผล

เมื่อทดสอบเรียก URL index.php?r=test/hello จะได้การแสดงผลดังนี้

My Company

# Hello World

Manop Kongoon

รูปที่ 134 หน้า Web Application หลังการเรียก actionHello()

## การเรียก URL สำหรับ Module

สำหรับการเรียก URL ใน Module นั้นจะเรียกผ่าน URL เมื่อong กับ Yii1 โดยเรียกคลักษณะดังนี้

index.php?r=module/controller/action

โดยสามารถดูรายละเอียดในหัวข้อการสร้างโมดูล

## รู้จัก Namespace และ Use

### Namespace

สำหรับ Namespace เปรียบเสมือนการตั้งชื่อเพื่อแบ่งกลุ่มของ Class ต่างๆ ให้สามารถใช้ชื่อซ้ำกันได้แต่อยู่คนละ Namespace นั่นเองเพื่อแก้ปัญหาการตั้งชื่อคลาสที่ยาวๆ

ยกตัวอย่างเช่น

```
<?php  
namespace myspace;
```

### Use

เป็นการระบุกับ Web Application ว่าเราต้องการจะใช้ Class ไหนบ้าง เป็นลักษณะของการ Import Class ต่างๆ เข้ามาใช้งานใน Web Application

ตัวอย่างเช่น

```
<?php  
namespace myspace;  
use ArrayObject;
```

## ชื่อ Alias ใน Yii2

### Basic Template

@app: root directory ของ application

@vendor: การอ้างถึง Folder vendor ที่อยู่ใน @app

@runtime: การอ้างถึง Folder runtime

@web: URL path ไปยัง Application

@webroot: web root ของ Application

@tests: ที่เก็บการ Tests

## Advanced Template

@app: root directory ของ application

@vendor: การอ้างถึง Folder vendor ที่อยู่ใน @app

@runtime: การอ้างถึง Folder runtime

@web: URL path ไปยัง Application

@webroot: web root ของ Application

@tests: ที่เก็บการ Tests

@common: การอ้างถึง Folder common ที่อยู่ใน @app

@frontend: การอ้างถึง Folder frontend ที่อยู่ใน @app

@backend: การอ้างถึง Folder backend ที่อยู่ใน @app

@console: การอ้างถึง Folder console ที่อยู่ใน @app

## ทดลองสร้าง Model

สำหรับ Model นั้นมี 2 แบบคือ Model แบบไม่เชื่อมตารางได้แก่ Model ที่นำไปสร้างฟอร์มรับข้อมูลเพื่อมาประมวลผล และ Model แบบเชื่อมกับตารางในฐานข้อมูล โดย Model จะถูกเก็บไว้ใน Folder models เช่นหากต้องการสร้าง Model แบบไม่เชื่อมตารางชื่อ TestForm จะตั้งชื่อไฟล์ TestForm.php เก็บไว้ใน models จะได้ models/TestForm.php จะเห็นได้ว่าชื่อ File และชื่อ Class เป็นชื่อเดียวกัน และมีการระบุ namespace app\models; และใช้ use yii\base\Model; และ TestForm extends มาจาก Model สำหรับ Yii2 นั้นเพิ่มส่วนที่เรียกว่า namespace และ use เข้ามาเพื่อให้สอดคล้องกับการเขียนโปรแกรมด้วย PHP ในยุคใหม่นี้

```
1 <?php
2 namespace app\models;
3 use yii\base\Model;
4
5 class TestForm extends Model{
6     public $firstname;
7     public $lastname;
8
9
10    public function rules(){
11        return [
12            [['firstname','lastname'],'required'],
13        ];
14    }
15
16    public function attributeLabels() {
17        return [
18            'firstname'=>'ชื่อ',
19            'lastname'=>'นามสกุล',
20        ];
21    }
22}
```

รูปที่ 135 รูปแบบของ Model

## ทดลองสร้าง Controller

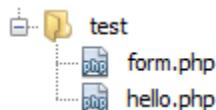
สำหรับ Controller นั้น จะเก็บไว้ใน Folder controllers/ โดยตัวอย่างนี้จะสร้างไฟล์ controller/TestController.php โดยเขียนโปรแกรมดังนี้

```
1 <?php
2 namespace app\controllers;
3 use Yii;
4 use yii\web\Controller;
5 use app\models\TestForm;
6 class TestController extends Controller{
7     public function actionHello(){
8         $var = 'Manop Kongoon';
9         return $this->render('hello', ['myvar'=>$var]);
10    }
11    public function actionForm(){
12        $model = new TestForm;
13        if($model->load(Yii::$app->request->post()) && $model->validate()){
14            //หากมีการ post ข้อมูลเข้ามาจะทำการบันทึก
15            //เขียนเนื้อหาข้อมูลลงฐานข้อมูล ให้เวลาส่งให้ Flash
16            Yii::$app->session->setFlash('success', 'ສາສດຖານ');
17            . $model->firstname.' '
18            . $model->lastname);
19        }
20        return $this->render('form', ['model'=>$model]);
21    }
22 }
```

รูปที่ 136 รูปแบบของ Controller

## ทดลองสร้าง View

ทดลองสร้าง view ใน views/test/โดยสร้างไฟล์ form.php



รูปที่ 137 ตำแหน่งของ View

จากนั้นเขียนโปรแกรมเพื่อแสดงฟอร์มกรอกข้อมูลดังนี้

```
1 <?php
2     use yii\helpers\Html;
3     use yii\widgets\ActiveForm;
4 ?>
5 <?php
6     if(Yii::$app->session->hasFlash('success')) {
7         echo Yii::$app->session->getFlash('success');
8     }
9 ?>
10    <?php $form=ActiveForm::begin();?>
11    <?= $form->field($model, 'firstname');?>
12    <?= $form->field($model, 'lastname');?>
13    <?= Html::submitButton('ส่งข้อมูล', ['class'=>'btn btn-success']);?>
```

รูปที่ 138 การเขียนโปรแกรมใน form.php

ทดลองเรียก URL index.php?r=test/form

The screenshot shows a web page titled "My Company". Below the title, there is a form with two input fields and a green "submit" button. The first input field is labeled "ชื่อ" (Name) and the second is labeled "นามสกุล" (Last Name). The "submit" button is green and labeled "ส่งข้อมูล" (Send Data).

รูปที่ 139 หน้า Web Application ใน test/form

ทดลองกรอกข้อมูลแล้วกดปุ่ม ส่งข้อมูล

My Company

สวัสดีคุณมานพ กองอุ่น  
ชื่อ

มานพ

นามสกุล

กองอุ่น

ส่งข้อมูล

รูปที่ 140 การกรอกข้อมูลลงในฟอร์ม

จะเห็นว่ามีคำว่า สวัสดีคุณมานพ กองอุ่น

จะเห็นว่ามีการทำงานที่แยกอิสระต่อ กันนั้นคือ Model เป็นส่วนของการกำหนดการเก็บข้อมูล การตรวจสอบความถูกต้องของข้อมูล Controller เป็นส่วนโปรแกรมที่จะกระทำการกับข้อมูล และ View เป็นส่วนของการแสดงผลซึ่งทำให้การพัฒนาระบบที่มีขนาดใหญ่มีความคล่องตัวมากยิ่งขึ้น

## การเลือกข้อมูล

สำหรับการเลือกข้อมูลมาแสดงนั้นสามารถทำได้หลายแบบ

### การเลือกข้อมูลโดยใช้ DataProvider

สำหรับการเลือกข้อมูลวิธีนี้โดยส่วนใหญ่แล้วจะส่งต่อให้ View แสดงผลแบบต่างๆ โดยใช้ Widget เช่น GridView

```
55 public function actionIndex() { //เรียกข้อมูลทั้งหมดไปแสดงผล
56     $query = Content::find();
57     $dataProvider = new ActiveDataProvider([
58         'query'=>$query,
59     ]);
60     return $this->render('index', [
61         'dataProvider'=>$dataProvider,
62     ]);
63 }
```

รูปที่ 141 การเรียกข้อมูลโดยใช้ ActiveDataProvider

สำหรับการเขียนโปรแกรมแสดงผลใน View โดยใช้ GridView ดังนี้

```
9    <?= GridView::widget([
10        'dataProvider'=>$dataProvider,
11        'columns'=>[
12            ['class'=>'yii\grid\SerialColumn'], //ลำดับคอลัมน์
13            'title', //หัวข้อ
14            'description', //รายละเอียด
15            'created', //วันที่สร้าง
16            ['class'=>'yii\grid\ActionColumn'], //action button
17        ]
18    ]);?>
```

รูปที่ 142 การแสดงผลข้อมูลแบบ GridView

เมื่อเรียก URL จะแสดงผลดังนี้

ผลลัพธ์ทั้งหมด				
#	ชื่อ	รายละเอียด	วันที่	
1	ทดสอบการเรียกข้อมูลจากไฟล์	ทดสอบการเรียกข้อมูลจากไฟล์	2014-11-20 12:19:14	
2	การเขียนไฟล์ทดสอบเรื่อง Yii2	เรียนรู้การเขียนไฟล์ทดสอบเรื่อง Yii2	2014-11-20 14:45:17	

รูปที่ 143 ข้อมูลที่แสดงผลแบบ GridView

จะเห็นว่าเป็นการเรียกใช้ Widget GridView ที่สามารถแสดงผลออกมาให้อยู่ในรูปของตาราง โดยที่เราไม่ต้องมาออกแบบตารางเองเลย ทำให้มีความสะดวกมากและเขียนโค้ดได้สั้นมากๆ

## การเลือกข้อมูลโดยใช้ DB Query Building Method

การสร้างคำสั่ง SQL ที่มีความซับซ้อน อาจไม่เหมาะสมสำหรับคนที่ใช้งาน SQL ไม่คล่องมากนัก การเรียกใช้งาน Method สำหรับ สร้าง Query นั้นช่วยลดข้อจำกัดดังกล่าวได้

```
$posts = (new \yii\db\Query())
```



```
->select('id, name')  
->from('post')  
->limit(10)  
->all();
```

หรือ

```
$query = (new \yii\db\Query())  
->select('id, name')  
->from('post')  
->limit(10);
```

## Method ในการ Query ข้อมูล

### SELECT

เป็น Method ที่ใช้ในการระบุ Filed ในตารางว่าต้องการเลือกข้อมูลมาจาก Field ใดบ้าง เช่น

```
$query->select('id', 'title')  
->from('post');
```

หรือสามารถกำหนดในรูปแบบของ Array() ได้ เช่น

```
$query->select(['id', 'title'])  
->from('post');
```

นอกจากนี้ยังสามารถใส่คำสั่งพิเศษได้อีกด้วย เช่น

```
CONCAT(first_name, ' ', last_name) AS full_name
```

และนอกจากนี้ยังสามารถสร้าง sub-queries ให้เป็น Column ได้อีกด้วย เช่น

```
$q1 = (new Query)->select('COUNT(*)')->from('post');  
$q2 = (new Query)->select(['id', 'count'=>$q1])->from('user');
```

เมื่อทำงานจะได้คำสั่ง SQL ดังนี้

```
SELECT id, (SELECT COUNT(*) FROM post) AS count FROM user
```

นอกจากนี้ยังสามารถเลือกเฉพาะที่ไม่ซ้ำโดยใช้ Method distinct() เช่น



```
$query->select('user_id')->distinct()->from('post');
```

## FROM

ใช้สำหรับเลือกตารางที่ต้องการสำหรับเลือกข้อมูล เช่น

```
$query->select('*')->from('post');
```

ทั้งนี้ยังสามารถใช้ชื่อ Alias ได้อีกด้วย เช่น

```
$query->select(u.username,p.*)->from(['user u', 'post p']);
```

## WHERE

สำหรับ Method where() นั้นเป็น Method สำหรับการระบุเงื่อนไขต่างๆ เช่น

```
$query->where('approved=:approved', [':approved'=>$approved]);
```

ข้อสำคัญไม่ควรกำหนดค่าโดยตรง เนื่องจากอาจมีการ SQL Injection เข้ามาได้ ควรทำการกำหนดค่าตามตัวอย่างข้างต้น ซึ่งตัวอย่างการกำหนดค่าโดยตรงเช่น

```
$query->where("approved=$approved"); //ไม่ปลอดภัย
```

ทั้งนี้ยังสามารถใช้การ Bind ค่าให้กับตัวแปรที่กำหนดขึ้นได้เช่น

```
$query->where('approved=:approved');  
$query->addParams([':approved'=>$approved]);
```

นอกจากนี้ยังสามารถใช้งานในรูปแบบ hash ได้เช่น

```
$query->where([  
    'approved'=> 1,  
    'id'=>[2,3,4],  
]);
```

เมื่อแปลงเป็นคำสั่ง SQL จะได้ลักษณะนี้

```
WHERE (approved=1) AND (id IN (2,3,4))
```

และยังสามารถใช้ NULL ซึ่งเป็นค่าพิเศษในฐานข้อมูลได้อีกด้วย เช่น



```
$query->where(['approved'=>null]);
```

จะได้ผลลัพท์ดังนี้

```
WHERE (approved IS NULL)
```

นอกจากนั้นยังสามารถใช้ sub-query จาก Query Object ได้อีกด้วย เช่น

```
$q1 = (new \Query)->select('id')->form('user');  
$q2->where(['user_id'=>$q1]);
```

จะได้ผลลัพท์ดังนี้

```
WHERE user_id IN (SELECT id FROM user)
```

และยังสามารถใช้ตัวดำเนินการ (Operator) ได้อีกด้วยในรูปแบบ [ตัวดำเนินการ (Operator, ค่าที่ 1(Operand1), ค่าที่2(Operand2,...)]

ตัวดำเนินการ	รูปแบบ	ผลลัพท์
and	['and', 'id=1', 'id=2']	id=1 AND id=2
or	['or', 'id=1', 'id=2']	id=1 OR id=2
between	['between', 'id',1,5]	id BETWEEN 1 AND 5
not between	['not between', 'id',1,5]	id NOT BETWEEN 1 AND 5
in	['in', 'id',[1,2,3]]	id IN (1,2,3)
not in	['not in', 'id',[1,2,3]]	id NOT IN (1,2,3)
like	['like', 'title', 'hello']	title LIKE '%hello%'
	['like', 'title', ['hello', 'world']]	title LIKE '%hello%' AND title LIKE '%world%'
or like	['or like', 'title',['hello', 'world']]	title LIKE '%hello%' OR title LIKE '%world%'
not like	['not like', 'title', 'hello']	title NOT LIKE '%hello%'
or not like	['or not like', 'title',['hello', 'world']]	title NOT LIKE '%hello%' OR title NOT LIKE '%world%'

และยังสามารถใช้คำสั่งการเปรียบเทียบต่างๆ ได้เช่น

```
$query->select('id')  
      ->form('post')
```



```
->where(['>=' , 'id' , 10]);
```

จะได้ผลลัพท์ดังนี้

```
SELECT id FROM post WHERE id >=10;
```

และยังสามารถใช้ Method andWhere() และ orWhere() เช่น

```
$query->where('approved'=>1);  
$query->andWhere(['like', 'title', 'hello']);
```

จะได้ผลลัพท์ดังนี้

```
WHERE (approved=1) AND (title LIKE '%hello%')
```

นอกจากนั้นยังสามารถใช้ Method พิเศษ นั่นคือ filterWhere() ซึ่งจะช่วยในการตรวจสอบค่าว่าง เช่น

```
$query->filterWhere([  
    'username'=>$username,  
    'email'=>$email,  
]);
```

ในคำสั่งนี้จะเป็นการตรวจสอบค่าว่าง หากค่าใดค่าหนึ่งว่าระบบจะสร้าง WHERE เฉพาะค่านั้นๆ แต่ถ้าว่างทั้ง 2 ระบบจะไม่สร้าง WHERE นั้นเอง นอกจากนั้นยังสามารถใช้ Method andFilterWhere() และ orFilterWhere() ในการกำหนดเงื่อนไขเพิ่มเติมได้อีกด้วย

## ORDER BY

Order by หรือการเรียงลำดับ สามารถใช้ตัวแปรพิเศษในการกำหนดการเรียงลำดับได้ คือ SORT\_ASC คือเรียงจากน้อยไปมาก และ SORT\_DESC คือเรียงจากมากไปน้อย เช่น

```
$query->orderBy([  
    'id' => SORT_ASC,  
    'title' => SORT_DESC,  
]);
```

ผลลัพท์จะได้ id เรียงจากน้อยไปมาก และ title เรียงจากมากไปน้อย

## GROUP BY และ HAVING

ในการใช้งาน GROUP BY สามารถเรียกใช้งานได้ดังนี้

```
$query->groupBy('id, status');
```

หากต้องการเพิ่ม Field อื่นๆ หลังจากใช้ groupBy แล้ว ให้ใช้ Method addGroupBy เช่น

```
$query->addGroupBy(['created_at', 'updated_at']);
```

สำหรับการใช้ HAVING นั้น สามารถเรียกใช้งานได้เช่นกัน

```
$query->having(['status'=>$status);
```

และยังสามารถใช้ Method andHaving และ orHaving ได้อีกด้วย

## LIMIT และ OFFSET

หากต้องการเลือกแค่มา 20 แถวจะใช้ Method limit ดังนี้

```
$query->limit(20);
```

หากต้องการเว้นการเลือก 50 แถวแรกจะใช้ Method offset ดังนี้

```
$query->offset(50);
```

## JOIN

Method join เป็น Method ในการเข้ามาร่วม โดยมี Method 3 รูปแบบดังนี้

innerJoin() leftJoin() และ rightJoin()

ตัวอย่างเช่น

```
$query->select(['user.firstname as firstname', 'user.lastname as lastname', 'post.title as title'])  
->from('user')  
->leftJoin('post', 'post.user_id = user.id');
```

และยังสามารถใช้ Method join ในการกำหนดการ join ในรูปแบบอื่นๆ เช่น

```
$query->join('FULL OUTER JOIN', 'post', 'post.user_id = user.id');
```

## UNION

สำหรับการใช้ Union นั้นเป็นการเปรียบเทียบการ Query จาก 2 Query โดยจะเลือกเฉพาะค่าที่ตรงกันทั้ง 2 Query

```
$q1 = new Query();
$q1->select('id, category_id as type, name')->from('post')
->limit(10);

$q2 = new Query();
$q2->select('id, type, name')->from('user')->limit(10);
$query->union($q2);
```

ที่มา <http://www.yiiframework.com/doc-2.0/guide-db-query-builder.html>

## การเลือกข้อมูลแบบกำหนด SQL Query เอง

### การเลือกข้อมูลแบบที่ 1

เป็นการเลือกข้อมูลทั้งหมดในฐานข้อมูล โดยมีรูปแบบการเลือกข้อมูลดังนี้

```
public function actionSelect1() {
    $con = \Yii::$app->db;
    $cmd = $con->createCommand('SELECT * FROM post');
    $posts = $cmd->queryAll(); // เลือกข้อมูลทั้งหมด
    return $this->render('select1', ['posts'=>$posts]);
}
```

### การเลือกข้อมูลแบบที่ 2

เป็นการเลือกข้อมูลเฉพาะเจาะที่ต้องการ โดยมีรูปแบบการเลือกข้อมูลดังนี้

```
public function actionSelect2() {
    $con = \Yii::$app->db;
    $cmd = $con->createCommand('SELECT * FROM post WHERE id=2');
    $post = $cmd->queryOne(); // เลือกข้อมูลเฉพาะเดียว
    return $this->render('select2', ['post'=>$post]);
}
```

รูปที่ 144 การเลือกข้อมูลโดยใช้ queryOne()



## การเลือกข้อมูลแบบที่ 3

เป็นการเลือกข้อมูลเฉพาะ Field ที่ต้องการ โดยมีรูปแบบการเลือกข้อมูลดังนี้

```
public function actionSelect3() {
    $con = \Yii::$app->db;
    $cmd = $con->createCommand('SELECT title FROM post');
    $posts = $cmd->queryColumn(); //เลือกข้อมูลมา N Field
    return $this->render('select3', ['posts'=>$posts]);
}
```

รูปที่ 145 การเลือกข้อมูลโดยใช้ queryColumn()

## การเลือกข้อมูลแบบที่ 4

เป็นการเลือกข้อมูลแบบนับจำนวนหรือมีการคำนวณ โดยมีรูปแบบการเลือกข้อมูลดังนี้

```
public function actionSelect4() {
    $con = \Yii::$app->db;
    $cmd = $con->createCommand('SELECT COUNT(*) FROM post');
    $post = $cmd->queryScalar(); //คำนวน
    return $this->render('select4', ['post'=>$post]);
}
```

รูปที่ 146 การเลือกข้อมูลด้วย queryScalar()

## การเลือกข้อมูลแบบที่ 5

เป็นการเลือกข้อมูลแบบ PDO โดยมีการ bindValue ร่วมด้วยซึ่งจะมีความปลอดภัยจาก SQL

Injection มากยิ่งขึ้น โดยเลือกมา 1 แถวตามเงื่อนไขที่ต้องการ ซึ่งมีรูปแบบการเลือกข้อมูลดังนี้

```
public function actionSelect5() {
    $con = \Yii::$app->db;
    $cmd = $con->createCommand('SELECT * FROM post WHERE id=:id');
    $cmd->bindValue(':id', 2);
    $post = $cmd->queryOne(); //เลือกมา1แถว
    return $this->render('select5', ['post'=>$post]);
}
```

รูปที่ 147 การเลือกข้อมูลแบบ bindValue()

## การสร้าง Relation ใน Model

Relation เป็นตัวช่วยให้สามารถอ้างอิงข้อมูลจากตารางอื่นๆ ได้โดยการสร้าง Relation ใน Model เพื่อเป็นการระบุว่า Field Foreign Key นี้ เชื่อมไปยัง Primary Key ของตารางไหน เป็นต้น ตัวอย่างเช่นการสร้าง Relation ไปยัง Model Comment จาก Model Post

```
class Post extends yii\db\ActiveRecord
{
    public function getComments()
    {
        return $this->hasMany('Comment', array('post_id' =>
'id'));
    }
}
```

ส่วนการเรียกข้อมูลก็สามารถเรียกผ่าน Object ได้ตามตัวอย่าง

```
$comments = $post->getComments()->andWhere('status=1')->all();
```

## การแสดง เพิ่ม แก้ไข และลบข้อมูล

สำหรับ Yii Framework หรือแม้แต่การเขียนโปรแกรมโดยทั่วไปจะต้องมาเรียนรู้คำสั่งในการสร้าง CRUD นั่นก็คือ Create Read Update และ Delete ข้อมูลโดยเป็นพื้นฐานสำหรับการพัฒนา Web Application โดยในส่วนนี้จะเป็นการแสดงรายละเอียดของการเขียนโปรแกรมสำหรับการทำ CRUD ให้มองเห็นภาพทั้งหมดก่อน แต่โดยทั่วไปสามารถใช้ Gii ในการ Generate Code ได้ เพื่อความเข้าใจมากขึ้น เราสามารถเขียนรู้คำสั่งต่างๆ เหล่านั้นกันก่อน

## ส่วนของตารางในฐานข้อมูล

สำหรับตารางในฐานข้อมูลจะสร้างโดยใช้คำสั่ง SQL ดังต่อไปนี้

```
CREATE TABLE IF NOT EXISTS `content` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `title` varchar(255) NOT NULL COMMENT 'เรื่อง',
```

```

`description` text NOT NULL COMMENT 'เนื้อหา',
`created` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP
COMMENT 'โพสเมื่อ',
PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;

```

## ส่วนของ Model

สำหรับส่วนของ Model นั้น การเขียนจะแตกต่างจาก Yii1.1 พอสมควร แต่ Concept หลักยังคงเหมือนกัน คือส่วนการกำหนดชื่อตารางหรือ method tableName() กำหนดกฎเกณฑ์หรือ method rule() และส่วนที่เป็นคำอธิบายหรือ method attributeLabel() แต่ที่เพิ่มเติมคือการใช้งาน namespace โดยจะตั้ง namespace เป็น app\models; ดังมีรายละเอียดต่อไปนี้

```

<?php
1    namespace app\models;
2
3    use Yii;
4    use yii\db\ActiveRecord;
5
6    class Content extends ActiveRecord{
7        public static function tableName() {
8            return 'content';//ระบุชื่อตาราง
9        }
10       public function rules() {
11           return [
12               [['title','description'],'required',//ระบุเงื่อนไขต้อง
13               [['title']], 'string', 'max'=>255]
14           ];
15       }
16       public function attributeLabels() {//ระบุ Label
17           return [
18               'id'=>'ID',
19               'title'=>'เรื่อง',
20               'description'=>'รายละเอียด',
21               'created'=>'โพสเมื่อ',
22           ];
23       }
}

```

รูปที่ 148 การกำหนด Model

## ส่วนของ Controller

ในส่วนของ Controller นั้นโดยปกติจะสร้างชื่อ Controller ให้ตรงกับชื่อตารางในที่นี่คือตาราง content ดังนั้นจะสร้าง Controller ชื่อ ContentController ใน controllers/ ดังนี้



รูปที่ 149 ตำแหน่ง Controller

```
1 <?php
2 namespace app\controllers;
3 use yii\web\Controller;//เรียกใช้ Controller เพื่อ extends
4 use app\models\Content;//เรียกใช้ Content model
5 use yii\web\NotFoundHttpException;
6 use yii\data\ActiveDataProvider;
7 use Yii;
8 class ContentController extends Controller{
9     public function actionCreate(){
10         $model = new Content(); //สร้าง Content model ชื่อ $model
11         //ถ้ามีการ post คำมาจากฟอร์มและถ้ามันเท่ากับ 1 ให้ไปหน้า view ส่ง id ไปแสดง
12         if($model->load(Yii::$app->request->post()) && $model->save()){
13             return $this->redirect(['view','id'=>$model->id]);
14         }else{//ถ้าไม่ได้ post ให้ render create
15             return $this->render('create', [
16                 'model'=>$model,
17             ]);
18         }
19     }
}
```

รูปที่ 150 การเขียนโปรแกรมใน Controller

### การแก้ไขข้อมูล

```
18     public function actionUpdate($id) {//แก้ไขข้อมูลโดยล็อก ID มาแก้
19         $model = $this->findModel($id); //โหลดข้อมูลเก่า
20         //ถ้ามีการ post คำมาจากฟอร์มและถ้ามันเท่ากับ 1 ให้ไปหน้า view ส่ง id ไปแสดง
21         if($model->load(Yii::$app->request->post()) && $model->save()){
22             return $this->redirect(['view','id'=>$model->id]);
23         }else{//ถ้าไม่ได้ post ให้ render update
24             return $this->render('update', [
25                 'model'=>$model,
26             ]);
27         }
28     }
}
```

รูปที่ 151 การเขียนโปรแกรมส่วน actionUpdate()



### การเปิดดูข้อมูล

```
31     public function actionView($id) { //แสดงผลโดยเลือกข้อมูลจาก id  
32         return $this->render('view', [  
33             'model'=>$this->findModel($id),  
34         ]);  
35     }
```

รูปที่ 152 การเขียนโปรแกรม actionView()

### การลบข้อมูล

```
36     public function actionDelete($id) { //ลบข้อมูลจากการส่ง id มา  
37         $this->findModel($id)->delete();  
38         return $this->redirect(['index']);  
39     }
```

รูปที่ 153 การเขียนโปรแกรม actionDelete()

### การเรียกข้อมูลทั้งหมดไปแสดงผล

```
40     public function actionIndex() { //เรียกข้อมูลทั้งหมดไปแสดงผล  
41         $query = Content::find();  
42         $dataProvider = new ActiveDataProvider([  
43             'query'=>$query,  
44         ]);  
45         return $this->render('index', [  
46             'dataProvider'=>$dataProvider,  
47         ]);  
48     }
```

รูปที่ 154 การเขียนโปรแกรม actionIndex()

### เขียน Method เพิ่มเติมเพื่อใช้ใน Controller

```
49     protected function findModel($id) { // method ในการค้นหา  
50         if(($model = Content::findOne($id)) !==null){  
51             return $model;  
52         }else{  
53             throw new NotFoundHttpException('หน้าที่ต้องการไม่มี');  
54         }  
55     }  
56 }
```

รูปที่ 155 การเขียนโปรแกรม findModel()



## ส่วนของ View

ในส่วนของ View หรือส่วนของการแสดงผลนั้นมีอยู่ 3 ส่วนหลัก นั่นคือ ส่วนแสดงรายการทั้งหมด (index) ส่วนเพิ่มข้อมูล(create) และส่วนแก้ไขข้อมูล(update)

การสร้าง view นั้นจะต้องมี folder เก็บไฟล์ต่างๆ ตามชื่อ Controller ซึ่งจะเก็บไว้ใน views ในที่นี้คือ views/content และการตั้งชื่อไฟล์ส่วนใหญ่จะตั้งชื่อตาม action เช่น actionIndex() ชื่อไฟล์คือ index เป็นต้น

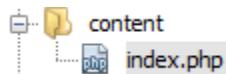


รูปที่ 156 ตำแหน่งที่เก็บ View

สำหรับการเขียนโปรแกรมเราจะเริ่มเขียนในส่วนการแสดงรายการทั้งหมดก่อนโดยมีรายละเอียดดังนี้

### การแสดงข้อมูลทั้งหมด (index)

สำหรับการแสดงข้อมูลทั้งหมดนั้นเราจะใช้ actionIndex() ที่ได้เขียนไว้แล้วใน Controller โดยส่งตัวแปรdataProvider อกมาให้ index.php เพื่อนำไปแสดงผล ในที่นี้จะแสดงออกเป็น GridView หรือตาราง โดยสร้างไฟล์ index.php ใน views/content



รูปที่ 157 ตำแหน่งที่เก็บ index.php

จากนั้นกำหนดรายละเอียดต่างๆ ดังนี้

```

1 <?php
2     use yii\helpers\Html;
3     use yii\grid\GridView;
4     ?>
5     <h1>รายการทั้งหมด</h1>
6     <p>
7         <?= Html::a('เพิ่มข้อมูล', ['create'], ['class'=>'btn btn-success']);?>
8     </p>
9     <?= GridView::widget([
10         'dataProvider'=>$dataProvider,
11         'columns'=>[
12             ['class'=>'yii\grid\SerialColumn'], //ลำดับคอลัมน์
13             'title',//หัวข้อ
14             'description',//รายละเอียด
15             'created',//วันที่สร้าง
16             ['class'=>'yii\grid\ActionColumn'], //action button
17         ]
18     ]);?>

```

รูปที่ 158 เขียนโปรแกรมแสดงรายการข้อมูล

เมื่อเรียก URL สำหรับแสดงผลคือ web/index.php?r=content จะแสดงผลดังนี้

#	หัวข้อ	รายละเอียด	วันที่
1	ทดสอบหัวข้อ	ทดสอบรายละเอียด	2014-11-20 12:19:14

รูปที่ 159 หน้า Web Application รายการข้อมูล

ทดลองแทรกข้อมูลเข้าในตาราง content โดยใช้ phpMyAdmin

<b>id</b>	<b>title</b> เรื่อง	<b>description</b> เนื้อหา	<b>created</b> วันที่
1	ทดสอบหัวข้อ	ทดสอบรายละเอียด	2014-11-20 12:19:14

รูปที่ 160 การแทรกข้อมูลในตาราง

จากนั้น Refresh หน้าเว็บจะพบรายการดังนี้

## รายการทั้งหมด

ผลลัพธ์			
#	ชื่อผู้ใช้งาน	รายละเอียด	เวลาเมื่อ
1	ทดสอบที่วิจัย	ทดสอบรายละเอียด	2014-11-20 12:19:14

รูปที่ 161 รายการข้อมูล

จะเห็นว่ามีปุ่มสำหรับคลิก เปิดดู แก้ไข และลบข้อมูลให้เรียบร้อยพร้อมใช้งาน

### การเปิดดูข้อมูล (view)

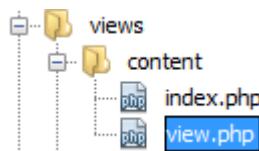
เมื่อคลิกไปที่รูปตาของรายการจะเห็นว่ามีข้อความ Error ดังนี้

**Invalid Parameter – yii\base\InvalidParamException**

The view file does not exist: D:\wamp\www\yii2-basic\views\content\view.php

รูปที่ 162 ข้อความแสดง Error เมื่อไม่มี view.php

แสดงว่าเรายังไม่ได้สร้าง views/content/view.php ให้ทำการสร้าง view และกำหนดการแสดงผลแบบ DetailView ดังนี้



รูปที่ 163 ตำแหน่งที่เก็บ view.php

ซึ่งมีรายละเอียดของโปรแกรมดังนี้

```
1 <?php
2 use yii\widgets\DetailView;
3 ?>
4 <h1><?php echo $model->title;?></h1>
5 <?=DetailView::widget([
6     'model'=>$model,
7     'attributes'=>[
8         'title',
9         'description:ntext',
10        'created',
11    ]
12 ]);?>
```

รูปที่ 164 การเขียนโปรแกรมใน view.php

จากนั้น Refresh หน้า web/index.php?r=content/view&id=1 จะได้ดังนี้

## ทดสอบหัวข้อ

ชื่อ	ผลลัพธ์ที่ได้
นามสกุลเดิม	ทดสอบรายละเอียด
โพสต์	2014-11-20 12:19:14

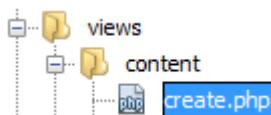
รูปที่ 165 รายการข้อมูลจาก id=1

## การเพิ่มข้อมูล (create)

สำหรับการเพิ่มข้อมูลนั้นมี 2 ส่วนคือ ส่วนของฟอร์มซึ่งจะใช้งานทั้งการเพิ่มและการแก้ไขข้อมูล โดยส่วนใหญ่จะตั้งชื่อว่า \_form.php และไฟล์สำหรับแสดงผลการเพิ่มข้อมูล คือ create.php

### ไฟล์ create.php

สร้างไฟล์ create.php ใน views/content/ ดังนี้



รูปที่ 166 ตำแหน่งที่เก็บไฟล์ create.php

จากนั้นเพิ่มคำสั่งในการดึงฟอร์มมาแสดงผลดังนี้

```
1 <h1>เพิ่มข้อมูล</h1>
2 <?= $this->render('_form', [
3     'model'=>$model,
4 ]) ;?>
5
```

รูปที่ 167 โปรแกรมใน create.php

## ไฟล์ \_form.php

ไฟล์ \_form.php เป็นไฟล์ สำหรับแสดงฟอร์มในการกรอกข้อมูลโดยจะใช้ทั้ง Create และ Update โดยมีรายละเอียดดังนี้

```
1 <?php
2     use yii\helpers\Html;
3     use yii\widgets\ActiveForm;
4 >
5     <?php $form = ActiveForm::begin(); ?>
6     <?= $form->field($model, 'title')->textInput(['maxlength'=>255]); ?>
7     <?= $form->field($model, 'description')->textarea(['row'=>6]); ?>
8     <?= Html::submitButton('บันทึกข้อมูล', ['class'=>'btn btn-success']) ;?>
9     <?php ActiveForm::end(); ?>
```

รูปที่ 168 โปรแกรมใน \_form.php

เมื่อแสดงผลโดยเรียก URL web/index.php?r=content/create จะได้ดังนี้

เพิ่มข้อมูล

ชื่อ

รายละเอียด

บันทึกข้อมูล

รูปที่ 169 หน้า Web Application

ทดสอบโดยการกดปุ่มบันทึกโดยที่ไม่ต้องกรอกข้อมูลใดๆ จะเห็นว่าระบบ validation ทำงานโดยที่เราไม่ต้องมาเขียนโปรแกรมในการตรวจสอบเลย สะดวกมากเลยครับ

# เพิ่มข้อมูล

เรื่อง

เรื่อง ต้องไม่ว่างเปล่า

รายละเอียด

รายละเอียด ต้องไม่ว่างเปล่า

บันทึกข้อมูล

รูปที่ 170 การแสดงฟอร์มสำหรับเพิ่มข้อมูล

เมื่อเพิ่มข้อมูลและกดปุ่มบันทึกข้อมูลระบบจะ redirect ไปที่หน้า view อัตโนมัติพร้อมแสดงรายละเอียดของรายการที่เราพึ่งเพิ่มเข้าไป

## การเขียนโปรแกรมด้วย Yii2

เรื่อง	การเขียนโปรแกรมด้วย Yii2
รายละเอียด	เรียนรู้การเขียนโปรแกรมด้วย Yii2
โพสเมื่อ	2014-11-20 14:45:17

รูปที่ 171 หน้า view หลังการเพิ่มข้อมูล

## การแก้ไขข้อมูล

เมื่อกดปุ่มรูปดินสอในหน้า index และระบบจะลิงค์ไปที่หน้าสำหรับการแก้ไขข้อมูลคือ web/index.php?r=content/update&id=1 และจะปรากฏข้อความแสดง Error ดังนี้

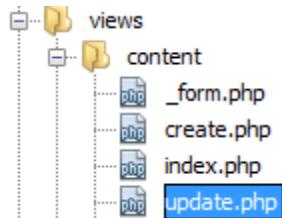


## Invalid Parameter – yii\\base\\InvalidParamException

The view file does not exist: D:\\wamp\\www\\yii2-basic\\views\\content\\update.php

รูปที่ 172 ข้อความ Error เมื่อไม่มีไฟล์ update.php

แสดงว่าเรายังไม่ได้สร้างไฟล์ views/content/update.php ตามคำแนะนำ ดังนั้นไปสร้างไฟล์นี้ได้เลย



รูปที่ 173 ตำแหน่งที่ตั้งไฟล์ update.php

โดยให้มีข้อมูลักษณะนี้

```
1 <h1><?php echo $model->title;?></h1>
2 <?= $this->render('_form', [
3   'model'=>$model,
4 ]);?>
```

รูปที่ 174 โปรแกรมใน update.php

จากนั้น Refresh หน้าเว็บ web/index.php?r=content/update&id=1 จะได้หน้า Web Application ดังนี้

## ทดสอบหัวข้อ

เรื่อง

ทดสอบหัวข้อ

รายละเอียด

ทดสอบรายละเอียด

บันทึกข้อมูล

รูปที่ 175 หน้า Web Application เมื่อกด update

จะเห็นว่ามีรายการที่เราเคยบันทึกไว้ในตาราง Content มาแสดงให้เราเห็นซึ่งเราเก็บสามารถแก้ไขได้ โดยการพิมพ์ข้อมูลเข้าไปใหม่แล้วกดปุ่ม บันทึกข้อมูล และระบบจะ Redirect ไปหน้า view ให้อัตโนมัติ ตามเงื่อนไขที่ได้เขียนไว้ใน actionUpdate() ที่อยู่ใน Controller

## ทดสอบหัวข้อทดสอบแก้ไข

เรื่อง	ทดสอบหัวข้อทดสอบแก้ไข
รายละเอียด	ทดสอบรายละเอียดทดสอบการแก้ไข
โพสเมื่อ	2014-11-20 12:19:14

รูปที่ 176 หน้า view หลังกดปุ่ม update

## การควบคุมการเข้าถึงโดยใช้ Access Control Filter (ACF)

การควบคุมการเข้าถึงหรือเรียกว่า Access Control โดยสามารถกำหนดการเข้าถึงได้ในระดับ Action ได้เลยที่เดียว南北 ได้ว่าสามารถกำหนด Web Application ให้มีความปลอดภัยมากยิ่งขึ้น ตัวอย่างของการกำหนดการควบคุมการเข้าถึงด้านล่าง จะเป็นการกำหนดให้กับ action create และ update โดยใช้งานได้เฉพาะคนที่เข้าสู่ระบบเท่านั้น

การกำหนดการควบคุมการเข้าถึงนั้นสามารถกำหนดได้ใน method behaviors() โดยกำหนดในส่วน return array ที่เป็น ‘access’ โดยมีรายละเอียดดังนี้



```
8     use yii\filters\AccessControl;
9     class ContentController extends Controller{
10         public function behaviors() {
11             return [
12                 'access'=>[
13                     'class'=>AccessControl::className(),
14                     'only'=>['create','update'], //เฉพาะ action create, update
15                     'rules'=>[
16                         [
17                             'allow'=>true, //อนุญาตเข้าถึง
18                             'roles'=>['@'] //คนที่เข้าสู่ระบบ
19                         ]
20                     ]
21                 ],
22             ];
23         }
24     }
```

รูปที่ 177 การกำหนด access control filter

จะสังเกตว่าพอใส่ Access Control แล้วเมื่อเรียก URL ใน action create และ update ระบบจะให้เข้าสู่ระบบก่อน

The screenshot shows a Yii2 application's login interface. At the top, there is a dark header bar with the text "My Company". Below it, a navigation bar contains the links "หน้าหลัก / Login". The main content area has a title "Login" and a sub-instruction "Please fill out the following fields to login:". There are two input fields: "Username" and "Password", each with its own label. Below these fields is a checked checkbox labeled "Remember Me". At the bottom of the form is a blue "Login" button. To the right of the form, there is explanatory text: "You may login with admin/admin or demo/demo." and "To modify the username/password, please check out the code app\models\User::\$users".

รูปที่ 178 การเข้าสู่หน้า login

เมื่อ Login Username และ Password admin/admin ก็จะสามารถเข้าระบบและสามารถใช้งาน action create ได้แล้ว

My Company

## เพิ่มข้อมูล

เรื่อง

รายละเอียด

บันทึกข้อมูล

รูปที่ 179 ฟอร์มสำหรับเพิ่มข้อมูล

สัญลักษณ์ที่ใช้มี 3 แบบคือ

? สำหรับบุคคลทั่วไปหรือ Guest

@ สำหรับผู้ที่เข้าสู่ระบบแล้วหรือ Authenticated Users

## การควบคุมการเข้าถึงโดยใช้ Role Base Access Control (RBAC)

Role Base Access Control เป็นการควบคุมระดับการเข้าถึง Action ใน Yii2 ซึ่งช่วยให้สามารถสร้างสิทธิ์การเข้าถึงได้หลายสิทธิ์ เช่น สิทธิ์ที่เป็น Admin สามารถจัดการข้อมูลได้ทั้งหมด สิทธิ์ที่เป็น Author สามารถจัดการเฉพาะที่ตัวเอง Post เท่านั้น เป็นต้น ทำให้มีความสะดวกต่อการกำหนดสิทธิ์ และการเขียนโปรแกรมการจัดการระดับการเข้าถึงมากยิ่งขึ้น

ทั้งนี้ขอใช้ Yii Advanced Template เป็นตัวทดสอบและเรียนรู้นะครับ โดยเริ่มจากการติดตั้ง Yii Advanced Template (ดูเพิ่มเติมได้ที่หัวข้อ การติดตั้ง Yii2 Advanced Application Template) ในที่นี้จะตั้งชื่อ Project ว่า yii2-rbac และทำการ Migration ให้เรียบร้อย จะได้ตาราง user พร้อมกับ Signup เพิ่ม admin user และทำการ Migrate ตาราง post และ สร้าง CRUD ด้วย Gii Generator ให้เรียบร้อย



## สร้างตารางโครงสร้าง RBAC

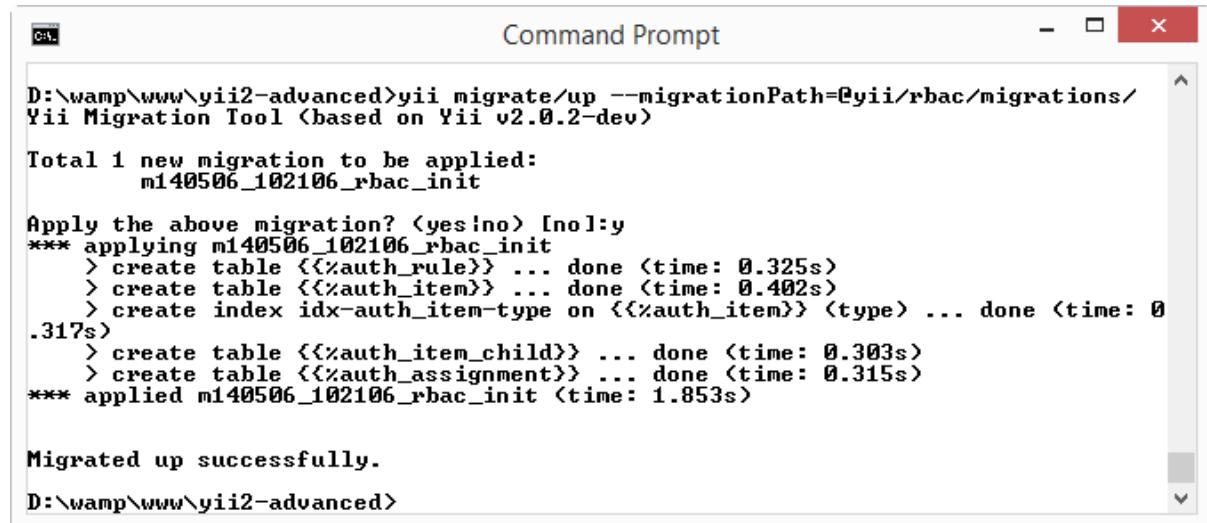
ก่อนสร้างตารางโครงสร้าง RBAC นั้นต้องทำการเพิ่ม authManager ในส่วน Component ของระบบก่อนโดยเปิดไฟล์ common/config/main.php และเพิ่มคำสั่งในส่วน Component ดังนี้

```
'authManager' => [
    'class' => 'yii\rbac\DbManager',
    'defaultRoles' => ['author'], //กำหนด roles
],
```

จากนั้นพิมพ์คำสั่งใน Command Prompt เพื่อสร้างโครงสร้าง RBAC โดยเปิด Command Prompt และเข้าไปใน Project yii2-advanced จากนั้นพิมพ์คำสั่ง

```
yii migrate --migrationPath=@yii/rbac/migrations/
```

แล้วกดปุ่ม Enter จากนั้นกด y และกดปุ่ม Enter อีกครั้งเพื่อดำเนินการ



```
D:\wamp\www\yii2-advanced>yii migrate/up --migrationPath=@yii/rbac/migrations/
Yii Migration Tool (based on Yii v2.0.2-dev)

Total 1 new migration to be applied:
m140506_102106_rbac_init

Apply the above migration? <yes|no> [no]:y
*** applying m140506_102106_rbac_init
    > create table `{{%auth_rule}}` ... done <time: 0.325s>
    > create table `{{%auth_item}}` ... done <time: 0.402s>
    > create index idx-auth_item-type on `{{%auth_item}}` <type> ... done <time: 0.317s>
    > create table `{{%auth_item_child}}` ... done <time: 0.303s>
    > create table `{{%auth_assignment}}` ... done <time: 0.315s>
*** applied m140506_102106_rbac_init <time: 1.853s>

Migrated up successfully.

D:\wamp\www\yii2-advanced>
```

รูปที่ 180 การ Migration RBAC

เมื่อเสร็จเรียบร้อยจะได้ตารางต่างๆ ดังนี้

Table
auth_assignment
auth_item
auth_item_child
auth_rule

รูปที่ 181 ตารางที่เกิดจากการ Migration



## สร้าง AuthorRule Class

ทำการสร้างไฟล์ AuthorRule.php ใน common/rbac/AuthorRule.php จากนั้นเขียนโปรแกรมดังนี้

```
<?php

namespace common\rbac;

use yii\rbac\Rule;

class AuthorRule extends Rule{

    public $name = 'isAuthor';

    public function execute($user, $item, $params) {

        return isset($params['post']) ? $params['post']->user_id == $user : false;
    }

}
```

## สร้าง RbacController

สร้าง Class RbacController ใน console/controllers/RbacController.php

```
<?php

namespace console\controllers;

use Yii;
use yii\console\Controller;
use common\rbac\AuthorRule;

class RbacController extends Controller{

    public function actionInit()      {

        $auth = Yii::$app->authManager;

        // add "createPost" permission
        $createPost = $auth->createPermission('createPost');
    }
}
```

```
$createPost->description = 'Create a post';
$auth->add($createPost);

// add "updatePost" permission
$updatePost = $auth->createPermission('updatePost');
$updatePost->description = 'Update post';
$auth->add($updatePost);

// add "author" role and give this role the
"createPost" permission
$author = $auth->createRole('author');
$auth->add($author);
$auth->addChild($author, $createPost);

// add "admin" role and give this role the
"updatePost" permission
// as well as the permissions of the "author" role
$admin = $auth->createRole('admin');
$auth->add($admin);
$auth->addChild($admin, $updatePost);
$auth->addChild($admin, $author);

// add the rule
$rule = new AuthorRule;
$auth->add($rule);

// add the "updateOwnPost" permission and associate
the rule with it.

$updateOwnPost = $auth-
>createPermission('updateOwnPost');

$updateOwnPost->description = 'Update own post';
$updateOwnPost->ruleName = $rule->name;
```



```
$auth->add($updateOwnPost);  
  
// "updateOwnPost" will be used from "updatePost"  
$auth->addChild($updateOwnPost, $updatePost);  
  
// allow "author" to update their own posts  
$auth->addChild($author, $updateOwnPost);  
  
// Assign roles to users. 1 and 2 are IDs returned by  
IdentityInterface::getId()  
// usually implemented in your User model.  
$auth->assign($author, 2);  
$auth->assign($admin, 1);  
}  
}
```

จากนั้นเรียก Action init โดยพิมพ์คำสั่งใน Command Prompt ดังนี้

```
yii rbac/init
```

จะเห็นว่ามีแค่ต่างๆ เกิดขึ้นในตารางในฐานข้อมูล

Table	Action	Rows	Type
auth_assignment	Browse Structure Search Insert Empty Drop	~8	InnoDB
auth_item	Browse Structure Search Insert Empty Drop	~5	InnoDB
auth_item_child	Browse Structure Search Insert Empty Drop	~4	InnoDB
auth_rule	Browse Structure Search Insert Empty Drop	~1	InnoDB

รูปที่ 182 และที่เกิดขึ้นหลังการใช้คำสั่ง init

## แก้ไขการ Update

เปิดไฟล์ backend/controllers/PostController.php จากนั้นเพิ่มโปรแกรมตรวจสอบการเข้าถึงใน actionUpdate() ดังนี้

```
public function actionUpdate($id) {
```

```
$model = $this->findModel($id);

if (Yii::$app->user->can('updatePost', ['post' =>
$model])) {

    if ($model->load(Yii::$app->request->post()) &&
$model->save()) {

        return $this->redirect(['view', 'id' =>
$model->id]);

    } else {

        return $this->render('update', [
            'model' => $model,
        ]);

    }
} else{

    throw new \yii\web\ForbiddenHttpException ("ไม่สามารถ
แก้ไขโพสของคนอื่นได้");
}

}
```

## ทดสอบ RBAC

ทดสอบการเข้าถึงโดยทดสอบ Permission ในการแก้ไข Post ของตัวเองและแก้ไข Post ของคนอื่น  
โดยใช้ User 2 ระดับคือ

Username: admin

Password: 44121165

Username: kongoon

Password: 44121165

ทำการเข้า URL สำหรับ Web Application โดยใช้ Username: kongoon

`http://localhost/yii2-rbac/backend/web/index.php?r=post`

## จะได้หน้า Web Application ดังนี้



The screenshot shows a Yii2 Rbac Admin interface for managing posts. The URL is `localhost/yii2-rbac/backend/web/index.php?r=post`. The page title is "Posts". There is a "Create Post" button. A table lists two posts:

#	ID	หัวข้อ	รายละเอียด	ผู้อ่าน	วันที่	操作	
1	1	ทดสอบการเขียนโค้ด kongoon	ทดสอบเมื่อการไฟล์สำหรับ kongoon	2	2015-01-03 10:03:35		
2	2	ทดสอบการเขียนโค้ด admin	ทดสอบเมื่อการไฟล์สำหรับ admin	1	2015-01-03 10:06:11		

At the bottom, there is a Yii Debugger panel with various metrics.

รูปที่ 183 หน้ารายการ Post

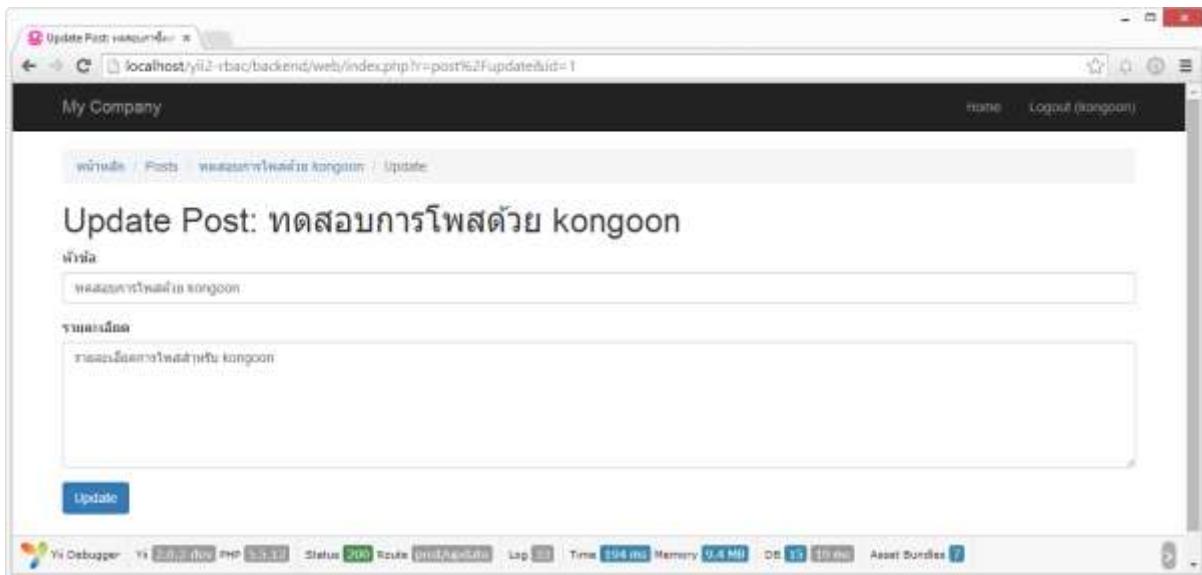
จากนั้นทดสอบการแก้ไขโดยการคลิกที่ Link แก้ไข Post ที่ Admin เป็นคน Post จะเห็นว่ามีการแสดงหน้า Error ดังนี้



The screenshot shows a Yii2 Rbac Admin interface displaying a "Forbidden (#403)" error. The URL is `localhost/yii2-rbac/backend/web/index.php?r=post%2fupdate&id=2`. The page title is "My Company". The error message is: "ไม่สามารถเข้าถึงค่าของหน้าที่นี้ได้". Below the message, it says: "The above error occurred while the Web server was processing your request. Please contact us if you think this is a server error. Thank you." At the bottom, there is a Yii Debugger panel.

รูปที่ 184 หน้าแสดง Error เมื่อไม่ได้แก้ไข Post ของตัวเอง

หากแก้ไข Post ของตัวเองจะสามารถแก้ไขได้



รูปที่ 185 หน้าที่สามารถแก้ไข Post ของตัวเองได้

นี่คือแนวทางหนึ่งในการทำ RBAC ในการเขียนโปรแกรมสำหรับบริการจัดการการเข้าถึงตามหน้าที่ ทั้งนี้ สามารถใช้ Module ในการจัดการ RBAC เช่น

```
https://github.com/mmdmsoft/yii2-admin
https://github.com/robregonm/yii2-auth
https://github.com/zelenin/yii2-rbac-module
```

หมายเหตุ สามารถดูเพิ่มเติมได้ที่ <https://github.com/yiisoft/yii2/blob/master/docs/guide/security-authorization.md>

## บทที่ 4 โมดูล (Module)

โมดูลเป็นส่วนสำคัญอีกส่วนหนึ่งซึ่งหากเราพัฒนา Web Application ที่มีความซับซ้อนหรือมีระบบย่อยๆ หลายๆ ระบบการแยกระบบต่างๆ นั้นออกเป็นโมดูลจะช่วยให้ลดความซับซ้อนของระบบที่จะพัฒนาด้วย และนอกจากนั้นยังช่วยจัดกลุ่มของระบบย่อยๆ ให้เป็นระบบระเบียบมากยิ่งขึ้น

### การสร้างโมดูลสำหรับ Yii2 Basic Application Template

โมดูลสำหรับ Yii2 Basic นั้นให้เข้า Gii Generator และไปที่เมนู Module Generator จากนั้นพิมพ์ชื่อโมดูลที่ต้องการ เช่น Module Class ชื่อ Content

app\modules\content\Content

และกำหนด Module ID ชื่อ content

content

จากนั้นกดปุ่ม Preview จะปรากฏหน้าเว็บดังนี้

### Module Generator

This generator helps you to generate the skeleton code needed by a Yii module.

#### Module Class

app\modules\content\Content

#### Module ID

content

#### Code Template

default (D:\wamp\www\yii2-basic\vendor\yiisoft\yii2-gii\generators\module/default)

Preview Generate

Click on the above **Generate** button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	
modules\content\Content.php	create	<input checked="" type="checkbox"/>
modules\content\controllers\DefaultController.php	create	<input checked="" type="checkbox"/>
modules\content\views\default\index.php	create	<input checked="" type="checkbox"/>

รูปที่ 186 การ Generate module ด้วย Gii Generator

จากนั้นกดปุ่ม Generate จะเห็นว่ามีรายละเอียดแสดงดังนี้



www.programmerthailand.com

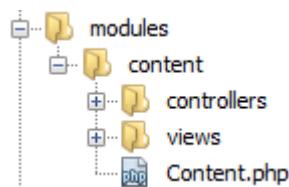
The module has been generated successfully.  
To access the module, you need to add this to your application configuration:

```
<?php  
.....  
'modules' => [  
    'content' => [  
        'class' => 'app\modules\content\Content',  
    ],  
],  
.....
```

```
Generating code using template "D:\wamp\www\yii2-basic\vendor\yiisoft\yii2-gii\generators\module/default"...  
generated modules\content\Content.php  
generated modules\content\controllers\DefaultController.php  
generated modules\content\views\default\index.php  
done!
```

รูปที่ 187 ข้อความหลัง Generate module เสร็จแล้ว

และจะเห็นว่ามีโครงสร้างไฟล์ต่างๆ เกิดขึ้นใน modules ดังนี้



รูปที่ 188 โครงสร้างไฟล์ต่างๆ ใน Module

เพิ่มการตั้งค่าในไฟล์ config/web.php

```
8     'basePath' => dirname(__DIR__),  
9     'bootstrap' => ['log'],  
10    'modules' => [  
11        'content' => [  
12            'class' => 'app\modules\content\Content',  
13        ],  
14    ],  
15    'components' => [  
16        'request' => [
```

รูปที่ 189 การเพิ่ม Module ใน Application

จากนั้นทดสอบเปิด URL สำหรับ Content Module index.php?r=content

This is the view content for action "index". The action belongs to the controller "app/modules/content/controllers/DefaultController" in the "content" module.

You may customize this page by editing the following file:

D:\xampp\www\yii2-basic\modules\content\views\default\index.php

© My Company 2014

Powered by Yii Framework

Yii Debugger | Yii 2.0.1 | PHP 5.5.12 | Status 200 Route content/default/index | Log 13 | Time 102 ms | Memory 5.6 MB | Asset Bundles 5 >

รูปที่ 190 ทดสอบเรียก URL สำหรับ Module

## การสร้าง Model ใน Module

การสร้าง Model ใน Module นั้นจะใช้ Gii Generator โดยเข้าที่เมนู Model Generator จากนั้นระบุ Table Name หรือชื่อตาราง Model Class หรือชื่อ Class Name Space หรือเนมสเปชของ Class และที่เก็บ Model และเลือก Generate Labels from DB Comments เพื่อนำ Comment ใน Field ต่างๆ ในตารางฐานข้อมูลมาทำเป็น Label ให้อัตโนมัติ ดังนี้

# Model Generator

This generator generates an ActiveRecord class for the specified database table.

**Table Name**

post

**Model Class**

Post

**Namespace**

app\modules\content\models

**Base Class**

yii\db\ActiveRecord

**Database Connection ID**

db

**Use Table Prefix**

**Generate Relations**

**Generate Labels from DB Comments**

รูปที่ 191 การตั้งค่าต่างๆ ในการ Generate Model ใน Module

จากนั้นกดปุ่ม Preview

Preview

Generate

Click on the above **Generate** button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	
modules\content\models\Post.php	create	<input checked="" type="checkbox"/>

รูปที่ 192 หลังจากกดปุ่ม Preview

และกดปุ่ม Generate



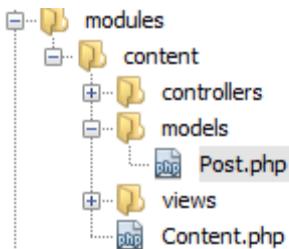
www.programmerthailand.com

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-basic\vendor\yiisoft\yii2-gii\generators\model/default"...
generated modules\content\models\Post.php
done!
```

รูปที่ 193 ข้อความหลังจากกดปุ่ม Generate

เมื่อต้องสร้างจะได้ดังนี้



รูปที่ 194 ตำแหน่งไฟล์ Model Post.php

## การสร้าง CRUD สำหรับ Module

สำหรับ CRUD นั้นจะใช้ Gii Generator เช่นเดียวกันกับการ Generate Model โดยเข้าเมนู CRUD Generator จากนั้นระบุที่อยู่ของ module ในการสร้าง CRUD ดังนี้

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update) data model.

## Model Class

app\modules\content\models\Post

## Search Model Class

app\modules\content\models\PostSearch

## Controller Class

app\modules\content\controllers\PostController

## View Path

@app/modules/content/views/post

## Base Controller Class

yii\web\Controller

## Widget Used in Index Page

GridView

รูปที่ 195 การตั้งค่าการ Generate CRUD ด้วย Gii Generator

Model Class คือ ที่อยู่ของ Model ในที่นี่จะระบุ

app\models\content\models\Post

Search Model Class คือ ชื่อ Model ที่จัดการการ Search ซึ่งจะถูก Generate ขึ้น ในที่นี่ระบุ

app\modules\content\models\PostSearch

Controller Class คือ ชื่อ Controller ที่ต้องการ ส่วนใหญ่จะระบุให้สอดคล้องกัน ในที่นี่ระบุ

app\modules\content\controllers\Post

ViewPath คือ ที่เก็บ View ที่จะถูก Generate ในที่นี่จะนำไปเก็บใน Module content จะระบุเป็น

@app/modules/content/views/post

จากนั้นกดปุ่ม Preview จะปรากฏรายละเอียดแสดงดังนี้



Preview    Generate

Click on the above **Generate** button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	<input checked="" type="checkbox"/>
modules\content\controllers\PostController.php	create	<input checked="" type="checkbox"/>
modules\content\models\PostSearch.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\_form.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\_search.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\create.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\index.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\update.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\view.php	create	<input checked="" type="checkbox"/>

รูปที่ 196 ไฟล์ต่างๆ หลังจากกดปุ่ม Preview

จากนั้นคลิกที่เป็น Generate จะปรากฏข้อความแสดงว่าสร้างเสร็จเรียบร้อยแล้วดังนี้

The code has been generated successfully.

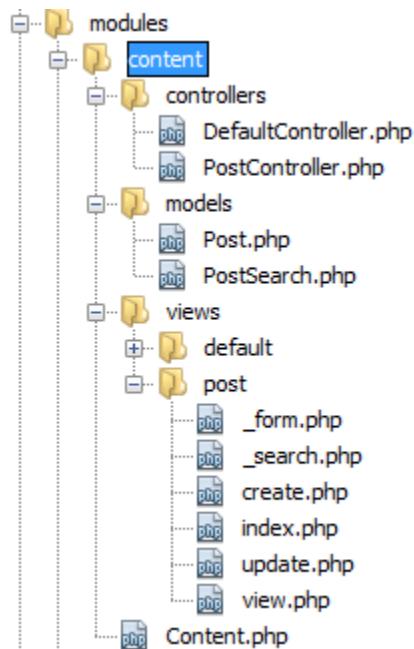
```
Generating code using template "D:\wamp\www\yii2-basic\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated modules\content\controllers\PostController.php
generated modules\content\models\PostSearch.php
generated modules\content\views\post\_form.php
generated modules\content\views\post\_search.php
generated modules\content\views\post\create.php
generated modules\content\views\post\index.php
generated modules\content\views\post\update.php
generated modules\content\views\post\view.php
done!
```

รูปที่ 197 ข้อความหลังจากกดปุ่ม Generate

เมื่อเปิดดูโครงสร้างไฟล์ต่างๆ ใน modules จะเห็นไฟล์ต่างๆ ดังนี้



www.programmerthailand.com



รูปที่ 198 โครงสร้างไฟล์ต่างๆ หลังจาก Generate Code แล้ว

ทำการทดลองเปิด Web Application ที่ URL `yii2-basic.dev/index.php?r=content/post` จะแสดงหน้า Web Application ดังนี้

ID	Content	Description
1	การพัฒนา Web Application สำหรับ Yii Framework 2 ตอน 2	Yii Framework 2 สำหรับ Web Application ตอน 2 คือการอธิบายและสอนวิธีการใช้งาน ของ Yii Framework 2 สำหรับ การพัฒนา Web Application ที่มีความซับซ้อน เช่น การจัดการฐานข้อมูล MySQL และการเชื่อมต่อ MySQL ผ่าน API RESTful
2	การพัฒนา Web Application สำหรับ Yii Framework version 2 ตอน 1	การพัฒนา Web Application สำหรับ Yii Framework version 2 ตอน 1 คือการอธิบายและสอนวิธีการใช้งาน ของ Yii Framework 2 สำหรับ การพัฒนา Web Application ที่มีความซับซ้อน เช่น การจัดการฐานข้อมูล MySQL และการเชื่อมต่อ MySQL ผ่าน API RESTful
3	ทดสอบ	ทดสอบ

รูปที่ 199 การเปิดหน้า Web Application content/post

# การสร้างโมดูลสำหรับ Yii2 Advanced Application Template

สำหรับ Yii2 Advanced Application Template นั้นเราต้องทำการ Generate 2 ส่วนนั่นก็คือ ส่วน Frontend และ Backend หรือเรียกว่ายๆ ว่าหน้าบ้านและหลังบ้านของ Web Application ของเรา นั่นเอง

## การสร้างโมดูลสำหรับ Frontend

สำหรับการสร้าง Module สำหรับ Frontend นั้นจะใช้ Gii Generator ในการสร้างโดยเข้าเมนู Module Generator จากนั้นตั้งค่าต่างๆ ดังนี้

## Module Generator

This generator helps you to generate the skeleton code needed by a Yii module.

### Module Class

frontend\modules\content\Content

### Module ID

content

### Code Template

default (D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\module/default )

**Preview**

รูปที่ 200 การสร้าง Module ใน Frontend

Module Class คือ ชื่อ Class ของ Module ที่ต้องการสร้างในที่นี่ระบุ

frontend\modules\content\Content

Module ID คือ Identity Module เป็นชื่อที่ไม่ซ้ำกันของ Module ในที่นี่ระบุ

content

จากนั้นกดปุ่ม Preview จะเห็นหน้าเว็บดังนี้



Preview

Generate

Click on the above **Generate** button  Create  Unchanged  Overwrite to generate the files selected below:

Code File	Action	<input checked="" type="checkbox"/>
modules\content\Content.php	create	<input checked="" type="checkbox"/>
modules\content\controllers\DefaultController.php	create	<input checked="" type="checkbox"/>
modules\content\views\default\index.php	create	<input checked="" type="checkbox"/>

รูปที่ 201 รายการไฟล์ต่างๆ หลังจากกดปุ่ม Preview

กด Checkbox ด้านหลัง Action เพื่อเลือกทั้งหมด ทำให้แน่ใจว่า Checkbox ที่ Create Unchanged และ Overwrite นั้นถูกเลือกอยู่ จากนั้นกดปุ่ม Generate เพื่อสร้าง Module จะปรากฏหน้าเว็บดังนี้

The module has been generated successfully.

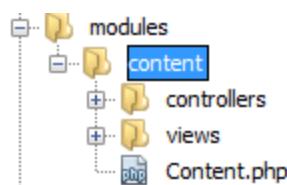
To access the module, you need to add this to your application configuration:

```
<?php  
.....  
'modules' => [  
    'content' => [  
        'class' => 'frontend\modules\content\Content',  
    ],  
.....
```

```
Generating code using template "D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\module/default"...  
generated modules\content\Content.php  
generated modules\content\controllers\DefaultController.php  
generated modules\content\views\default\index.php  
done!
```

รูปที่ 202 ข้อความหลังจากกดปุ่ม Generate

จะเห็นว่าที่ frontend/modules จะมีไฟล์ต่างๆ ดังนี้



รูปที่ 203 โครงสร้างไฟล์ใน Module

คัดลอกคำสั่งที่ได้เพิ่มใน frontend/config/main.php ในส่วน Module

```
'content' => [
    'class' => 'frontend\modules\content\Content',
],
```

โดยมีรูปแบบการวางคำสั่งดังนี้

```
14 'modules'=>[
15     'content' => [
16         'class' => 'frontend\modules\content\Content',
17     ],
18 ],
```

รูปที่ 204 การเพิ่ม Module ในไฟล์ config/main.php

ทดลองเปิด Web Application

```
http://yii2-advanced.dev/index.php?r=content หรือ
http://localhost/yii2-
advanced/frontend/web/index.php?r=content หรือ แบบ Pretty URL
http://yii2-advanced.dev/content หรือ
http://localhost/yii2-advanced/frontend/web/content
```

หมายเหตุ ถ้าเพิ่มเติมเรื่องการจัดการ URL แบบสวยงาม (Pretty URL) ในบทที่ 2 การติดตั้ง Yii2

จะได้หน้า Web Application ดังนี้





รูปที่ 205 ทดสอบเรียก URL สำหรับ Module

## การสร้าง Model ใน Module

หลังจากเข้า Gii Generator เรียบร้อยแล้วให้เข้าเมนู Model Generator โดยมีการกำหนดค่าต่างๆ ดังนี้

# Model Generator

This generator generates an ActiveRecord class for the specified database table.

## Table Name

post

## Model Class

Post

## Namespace

frontend\modules\content\models

## Base Class

yii\db\ActiveRecord

## Database Connection ID

db

## Use Table Prefix

## Generate Relations

## Generate Labels from DB Comments

รูปที่ 206 การตั้งค่าการสร้าง Model ใน Module

Table Name คือชื่อของตารางที่ต้องการสร้าง Model ในที่นี่คือ

post

Model Class คือชื่อของ Class ที่ใช้งานตารางที่กำหนดในที่นี่ให้ชื่อเดียวกันคือ

Post

Namespace คือที่อยู่ของ Model Class ที่ต้องการสร้างในที่นี่คือ

frontend/modules/content/models

ทำให้แน่ใจว่า Generate Relations ถูกเลือกอยู่ โดยเป็นการกำหนดให้สร้างความสัมพันธ์ไปยังตารางอื่นๆ หรือ Model อื่นๆ ด้วย และทำให้แน่ใจว่า Generate Labels from DB Comments ถูกเลือกอยู่ เป็นการระบุว่าให้นำ Comment ของ Field ต่างๆ ในตารางมาทำเป็น Label ซึ่งถ้าหากเราใส่เป็นภาษาไทยระบบจะ



นำมาสร้างเป็น Label ให้โดยอัตโนมัติ ด้วย Feature นี้ทำให้การพัฒนา Web Application มีความสะดวก และรวดเร็วมากยิ่งขึ้น

จากนั้นกดปุ่ม Preview

Preview    Generate

Click on the above **Generate** button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	
modules\content\models\Post.php	create	<input checked="" type="checkbox"/>

รูปที่ 207 รายการไฟล์หลังกดปุ่ม Preview

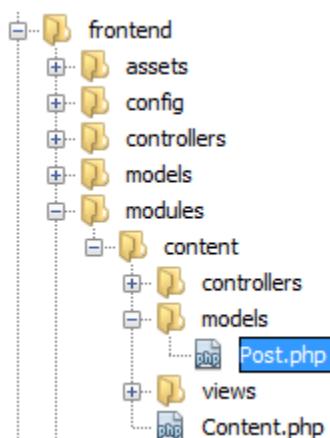
จากนั้นกดปุ่ม Generate

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\model/default"...
generated modules\content\models\Post.php
done!
```

รูปที่ 208 ข้อความหลังกดปุ่ม Generate

จะเห็นว่า Model Post ถูกสร้างขึ้นแล้วใน frontend/modules/content/models



รูปที่ 209 ไฟล์ Model Post.php หลัง Generate

## การสร้าง CRUD สำหรับ Module

สำหรับการสร้าง CRUD นั้นเป็นการใช้ Gii Generator ในการสร้างคำสั่งพื้นฐานในการแสดง เพิ่มแก้ไข และลบข้อมูล โดยหลังจากเข้า Gii Generator เรียบร้อยแล้วให้เข้าเมนู CRUD Generator แล้วตั้งค่าต่างๆ ดังนี้

## CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update) operations for a data model.

### Model Class

frontend\modules\content\models\Post

### Search Model Class

frontend\modules\content\models\PostSearch

### Controller Class

frontend\modules\content\controllers\PostController

### View Path

@frontend/modules/content/views/post

### Base Controller Class

yii\web\Controller

### Widget Used in Index Page

GridView

รูปที่ 210 การตั้งค่าสำหรับการสร้าง CRUD Generator

Model Class คือที่เก็บ Model ที่ต้องการสร้าง CRUD ในที่นี่กำหนด

frontend\modules\content\models\Post

Search Model Class คือการสร้าง Model สำหรับการจัดการการ Search ในที่นี่กำหนด

frontend\modules\content\models\PostSearch

Controller Class คือชื่อของ Controller ที่ต้องการสร้างในที่นี่กำหนด

frontend\modules\content\controllers\PostController



View Path เป็นการกำหนดที่เก็บของ View ในการทำ CRUD ในที่นี่กำหนด

@frontend/modules/content/views/post

จากนั้นกดปุ่ม Preview

Preview    Generate

Click on the above **Generate** button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	<input checked="" type="checkbox"/>
modules\content\controllers\PostController.php	create	<input checked="" type="checkbox"/>
modules\content\models\PostSearch.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\_form.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\_search.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\create.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\index.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\update.php	create	<input checked="" type="checkbox"/>
modules\content\views\post\view.php	create	<input checked="" type="checkbox"/>

รูปที่ 211 รายการไฟล์ต่างๆ หลังจากกดปุ่ม Preview

แล้วกดปุ่ม Generate

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated modules\content\controllers\PostController.php
generated modules\content\models\PostSearch.php
generated modules\content\views\post\_form.php
generated modules\content\views\post\_search.php
generated modules\content\views\post\create.php
generated modules\content\views\post\index.php
generated modules\content\views\post\update.php
generated modules\content\views\post\view.php
done!
```

รูปที่ 212 ข้อความหลังจากกดปุ่ม Generate

ทดสอบเปิด URL

http://yii2-advanced.dev/index.php?r=content/post หรือ  
http://localhost/yii2-advanced/frontend/web/index.php?r=content/post หรือ



<http://yii2-advanced.dev/index.php?r=content/post> หรือแบบ Pretty URL

<http://yii2-advanced.dev/content/post>

หมายเหตุ ดูเพิ่มเติมในส่วนของ การกำหนด Virtual Host และการกำหนด URL แบบสวยงาม (Pretty URL) ในบทที่ 2 การติดตั้ง Yii2 Advanced Application Template

จะได้หน้า Web Application ดังนี้

#	ID	หัวข้อ	รายละเอียด
1	1	เรียนรู้การสร้าง Web Application สำหรับ Yii Framework ของคุณ	ตอนนี้เรามาลงมือกัน ก็จะเริ่มต้นใช้เวลาอีกไม่นาน ก็จะสามารถทำเว็บไซต์ที่มีลักษณะแบบ Yii Framework ของคุณได้แล้ว
2	2	Yii Framework version 2 ทำให้งานง่ายขึ้น	การเปลี่ยนแปลงที่สำคัญที่สุดใน Yii Framework version 2 คือการเปลี่ยนชื่อภาษาอังกฤษเป็นภาษาไทย แต่ก็ยังคงความน่าเชื่อถือและมีประสิทธิภาพเหมือนเดิม



รูปที่ 213 หน้า Web Application เมื่อเรียก `content/post`

## การสร้างโมดูลสำหรับ Backend

สำหรับการสร้าง Module สำหรับ Frontend นั้นจะใช้ Gii Generator ในการสร้างโดยเข้าเมนู Module Generator จากนั้นตั้งค่าต่างๆ ดังนี้

# Module Generator

This generator helps you to generate the skeleton code needed by a Yii module.

## Module Class

backend\modules\content\Content

## Module ID

content

## Code Template

default (D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\module/default  
)

Preview

รูปที่ 214 การตั้งค่าการสร้าง Module

Module Class คือ ชื่อ Class และที่เก็บ Module ในที่นี่กำหนด

Backend\modules\content\Content

Module ID คือ ชื่อของ Module ที่ไม่ซ้ำกันใน Backend ในที่นี่กำหนด

content

จากนั้นกดปุ่ม Preview

Preview

Generate

Click on the above **Generate** button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	
D:\wamp\www\yii2-advanced\backend\modules\content\Content.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-advanced\backend\modules\content\controllers\DefaultController.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-advanced\backend\modules\content\views\default\index.php	create	<input checked="" type="checkbox"/>

รูปที่ 215 รายการไฟล์ต่างๆ หลังกดปุ่ม Preview

และกดปุ่ม Generate



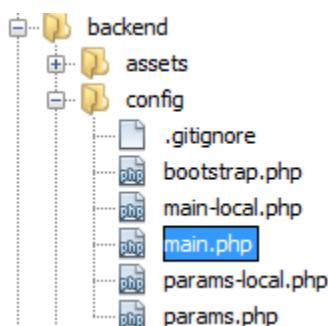
www.programmerthailand.com

The module has been generated successfully. You may try it now.

```
Generating code using template "D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\module/default"...
generated D:\wamp\www\yii2-advanced\backend\modules\content\Content.php
generated D:\wamp\www\yii2-advanced\backend\modules\content\controllers\DefaultController.php
generated D:\wamp\www\yii2-advanced\backend\modules\content\views\default\index.php
done!
```

รูปที่ 216 ข้อความหลังกดปุ่ม Generate

เปิดไฟล์ backend/config/main.php เพื่อแก้ไขการตั้งค่า โดยเพิ่มคำสั่งในการเรียกใช้งาน Module ที่เพิ่งสร้างขึ้นดังนี้



รูปที่ 217 ตำแหน่งไฟล์ config/main.php

จากนั้นเพิ่มคำสั่งดังนี้

```
14     'modules' => [
15         'content' => [
16             'class' => 'backend\modules\content\Content',
17         ],
18     ],
```

รูปที่ 218 การเพิ่ม Module ใน Web Application

ทดสอบเบิด URL

```
http://backend.yii2-advanced.dev/index.php?r=content หรือ
http://localhost/yii2-advanced/backend/web/index.php?r=content
หรือ แบบ Pretty URL
http://backend.yii2-advanced.dev/content
```

จะได้หน้า Web Application ดังนี้



รูปที่ 219 หน้าหลักของ Module

## การสร้าง Model ใน Module

หลังจากเข้า Gii Generator เรียบร้อยแล้วให้เข้าเมนู Model Generator โดยมีการกำหนดค่าต่างๆ ดังนี้

# Model Generator

This generator generates an ActiveRecord class for the specified database table.

**Table Name**

post

**Model Class**

Post

**Namespace**

backend\modules\content\models

**Base Class**

yii\db\ActiveRecord

**Database Connection ID**

db

**Use Table Prefix**

**Generate Relations**

**Generate Labels from DB Comments**

รูปที่ 220 การตั้งค่าการสร้าง Model สำหรับ Module

Table Name คือชื่อของตารางที่ต้องการสร้าง Model ในที่นี่คือ

post

Model Class คือชื่อของ Class ที่ใช้งานตารางที่กำหนดในที่นี้ให้ชื่อเดียวกันคือ

Post

Namespace คือที่อยู่ของ Model Class ที่ต้องการสร้างในที่นี่คือ

backend/modules/content/models

ทำให้แน่ใจว่า Generate Relations ถูกเลือกอยู่ โดยเป็นการกำหนดให้สร้างความสัมพันธ์ไปยังตารางอื่นๆ หรือ Model อื่นๆ ด้วย และทำให้แน่ใจว่า Generate Labels from DB Comments ถูกเลือกอยู่ เป็นการระบุว่าให้นำ Comment ของ Field ต่างๆ ในตารางมาทำเป็น Label ซึ่งถ้าหากเราใส่เป็นภาษาไทยระบบจะนำมาสร้างเป็น Label ให้โดยอัตโนมัติ ด้วย Feature นี้ทำให้การพัฒนา Web Application มีความสะดวกและรวดเร็วมากยิ่งขึ้น



จากนั้นกดปุ่ม Preview

Preview    Generate

Click on the above **Generate** button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	
D:\wamp\www\yii2-advanced\backend\modules\content\models\Post.php	create	<input checked="" type="checkbox"/>

รูปที่ 221 รายการไฟล์หลังกดปุ่ม Preview

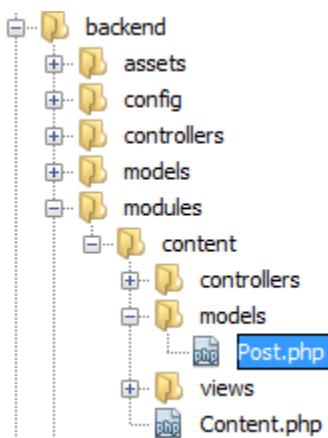
จากนั้นกดปุ่ม Generate

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\model/default"...
generated D:\wamp\www\yii2-advanced\backend\modules\content\models\Post.php
done!
```

รูปที่ 222 ข้อความหลังกดปุ่ม Generate

จะเห็นว่า Model Post ถูกสร้างขึ้นแล้วใน backend/modules/content/models



รูปที่ 223 ไฟล์ Model Post.php หลังการ Generate

## การสร้าง CRUD สำหรับ Module

สำหรับการสร้าง CRUD นั้นเป็นการใช้ Gii Generator ในการสร้างคำสั่งพื้นฐานในการแสดง เพิ่มแก้ไข และลบข้อมูล โดยหลังจากเข้า Gii Generator เรียบร้อยแล้วให้เข้าเมนู CRUD Generator และตั้งค่าต่างๆ ดังนี้

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update) operations for a data model.

## Model Class

backend\modules\content\models\Post

## Search Model Class

backend\modules\content\models\PostSearch

## Controller Class

backend\modules\content\controllers\PostController

## View Path

@backend/modules/content/views/post

## Base Controller Class

yii\web\Controller

## Widget Used in Index Page

GridView

รูปที่ 224 การตั้งค่าเพื่อสร้าง CRUD ด้วย Gii Generator

Model Class คือที่เก็บ Model ที่ต้องการสร้าง CRUD ในที่นี่กำหนด

backend\modules\content\models\Post

Search Model Class คือการสร้าง Model สำหรับการจัดการการ Search ในที่นี่กำหนด

backend\modules\content\models\PostSearch

Controller Class คือชื่อของ Controller ที่ต้องการสร้างในที่นี่กำหนด

backend\modules\content\controllers\PostController

View Path เป็นการกำหนดที่เก็บของ View ในการทำ CRUD ในที่นี่กำหนด

@backend/modules/content/views/post

จากนั้นกดปุ่ม Preview

[Preview](#) [Generate](#)

Click on the above [Generate](#) button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	
D:\wamp\www\yii2-advanced\backend\modules\content\controllers\PostController.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-advanced\backend\modules\content\models\PostSearch.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-advanced\backend\modules\content\views\post_form.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-advanced\backend\modules\content\views\post_search.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-advanced\backend\modules\content\views\post\create.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-advanced\backend\modules\content\views\post\index.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-advanced\backend\modules\content\views\post\update.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-advanced\backend\modules\content\views\post\view.php	create	<input checked="" type="checkbox"/>

รูปที่ 225 รายการไฟล์ต่างๆ หลังกดปุ่ม Preview

และกดปุ่ม Generate

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-advanced\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated D:\wamp\www\yii2-advanced\backend\modules\content\controllers\PostController.php
generated D:\wamp\www\yii2-advanced\backend\modules\content\models\PostSearch.php
generated D:\wamp\www\yii2-advanced\backend\modules\content\views\post_form.php
generated D:\wamp\www\yii2-advanced\backend\modules\content\views\post_search.php
generated D:\wamp\www\yii2-advanced\backend\modules\content\views\post\create.php
generated D:\wamp\www\yii2-advanced\backend\modules\content\views\post\index.php
generated D:\wamp\www\yii2-advanced\backend\modules\content\views\post\update.php
generated D:\wamp\www\yii2-advanced\backend\modules\content\views\post\view.php
done!
```

รูปที่ 226 ข้อความหลังกดปุ่ม Generate

ทดสอบเปิด URL

```
http://backend.yii2-advanced.dev/index.php?r=content/post หรือ
http://localhost/yii2-
advanced/backend/web/index.php?r=content/post หรือ
http://backend.yii2-advanced.dev/index.php?r=content/post หรือแบบ
Pretty URL
http://backend.yii2-advanced.dev/content/post
```



หมายเหตุ ดูเพิ่มเติมในส่วนของ การกำหนด Virtual Host และการกำหนด URL แบบสวยงาม (Pretty URL)  
ในบทที่ 2 การติดตั้ง Yii2 Advanced Application Template

จะได้หน้า Web Application ดังนี้

The screenshot shows a web browser window with the URL `backend.yii2-advanced.dev/content/post`. The page title is "My Company". The navigation bar includes "Home" and "Logout (admin)". Below the navigation, there are two tabs: "Home" and "Posts", with "Posts" being the active tab. A green button labeled "Create Post" is visible. The main content area displays a table titled "Posts" with the following data:

#	ID	หัวข้อ	รายละเอียด	操作
1	1	Yii Framework Web Application สำหรับ Yii Framework บล็อก	Yii Framework Web Application สำหรับ Yii Framework บล็อก สำหรับการเรียนรู้	
2	2	Yii Framework version 2 สำหรับ Yii Framework บล็อก	Yii Framework version 2 สำหรับ Yii Framework บล็อก สำหรับการเรียนรู้	



รูปที่ 227 การเปิด Web Application content/post

## บทที่ 5 รีม (Theme)

รีม (Theme) หรือ สีสันหน้าตาของ Web Application ในที่นี่จะพากท่านไปเรียนรู้แนวทางการสร้าง Theme ในแบบฉบับของ Programmer Thailand โดยจะสร้าง 2 แบบคือ Theme สำหรับ Basic Application Template และ Advanced Application Template

### รีมสำหรับ Yii2 Basic Application Template

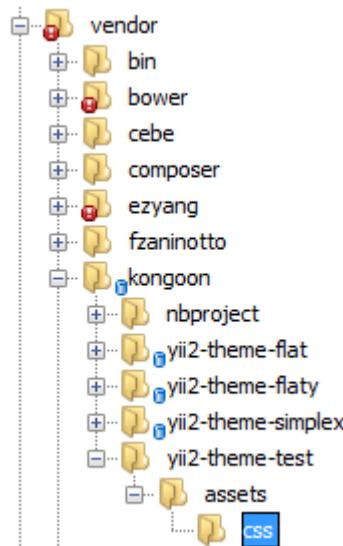
สำหรับ Theme ใน Yii2 Basic Application Template นั้นจะใช้ Theme จากเว็บไซต์ <http://bootswatch.com/simplex/> โดยดาวน์โหลด bootstrap.css มาใช้งาน โดยใช้ Bootstrap 3.3

#### ติดตั้งแบบไม่ใช้ Composer

สำหรับการติดตั้งแบบไม่ใช้ Composer นั้นถือได้ว่าเป็นการติดตั้งแบบ Manual กว่าได้เนื่องจากต้องไปแก้ไขไฟล์ต่างๆ ในระบบเพื่อให้สามารถโหลด Theme มาใช้งานในระบบได้ โดยมีขั้นตอนดังนี้

#### สร้าง Folder เก็บไฟล์ CSS

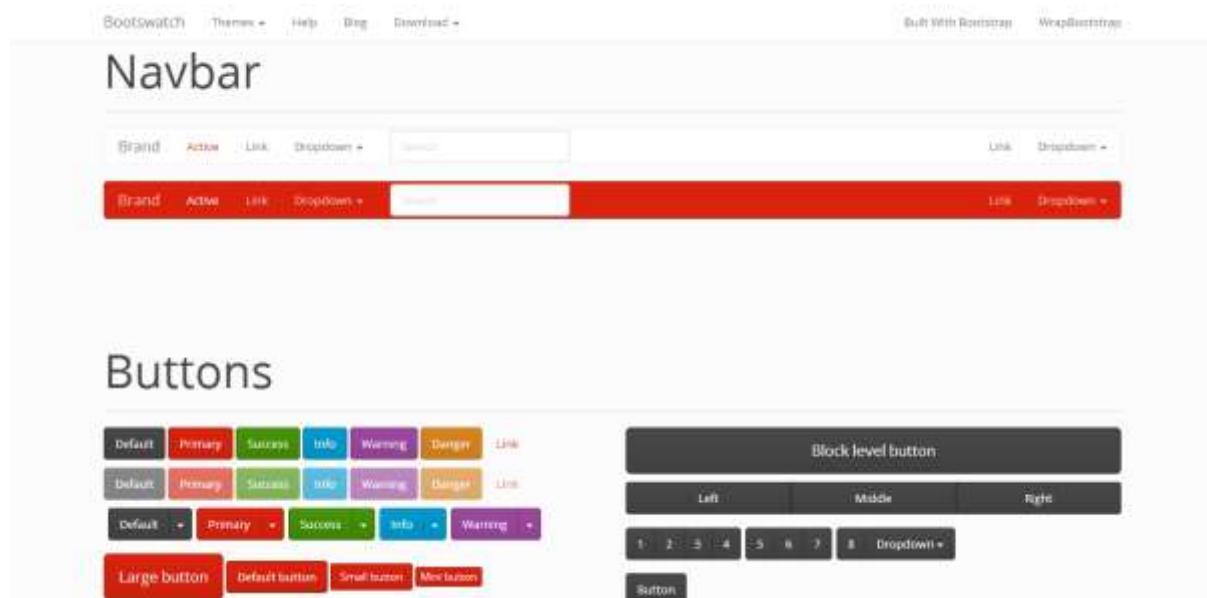
ในที่นี่จะสร้าง Folder เก็บ Theme ใน vendor/kongoon/yii2-theme-test ซึ่งภายในจะมี assets/css ดังนี้



รูปที่ 228 Folder เก็บไฟล์ CSS

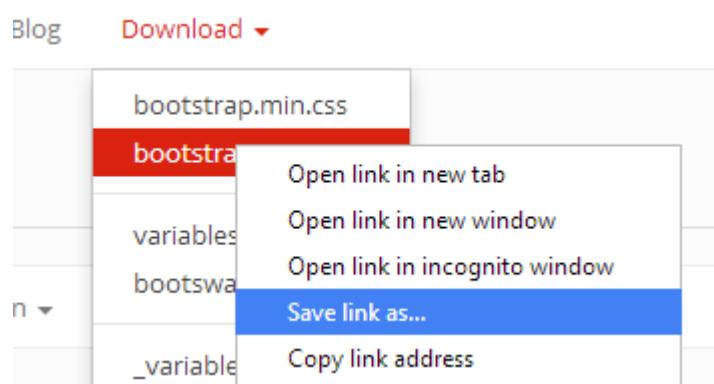
## ดาวน์โหลด CSS

ดาวน์โหลด Theme ที่ต้องการได้จากเว็บไซต์ <http://bootswatch.com/> หรือเว็บไซต์ที่ต้องการโดยใช้ Theme Bootstrap 3.3 ในที่นี่จะดาวน์โหลด Theme มาจากเว็บไซต์ <http://bootswatch.com/simplex/> ซึ่งมีหน้าเว็บที่มีสิ่งดังนี้



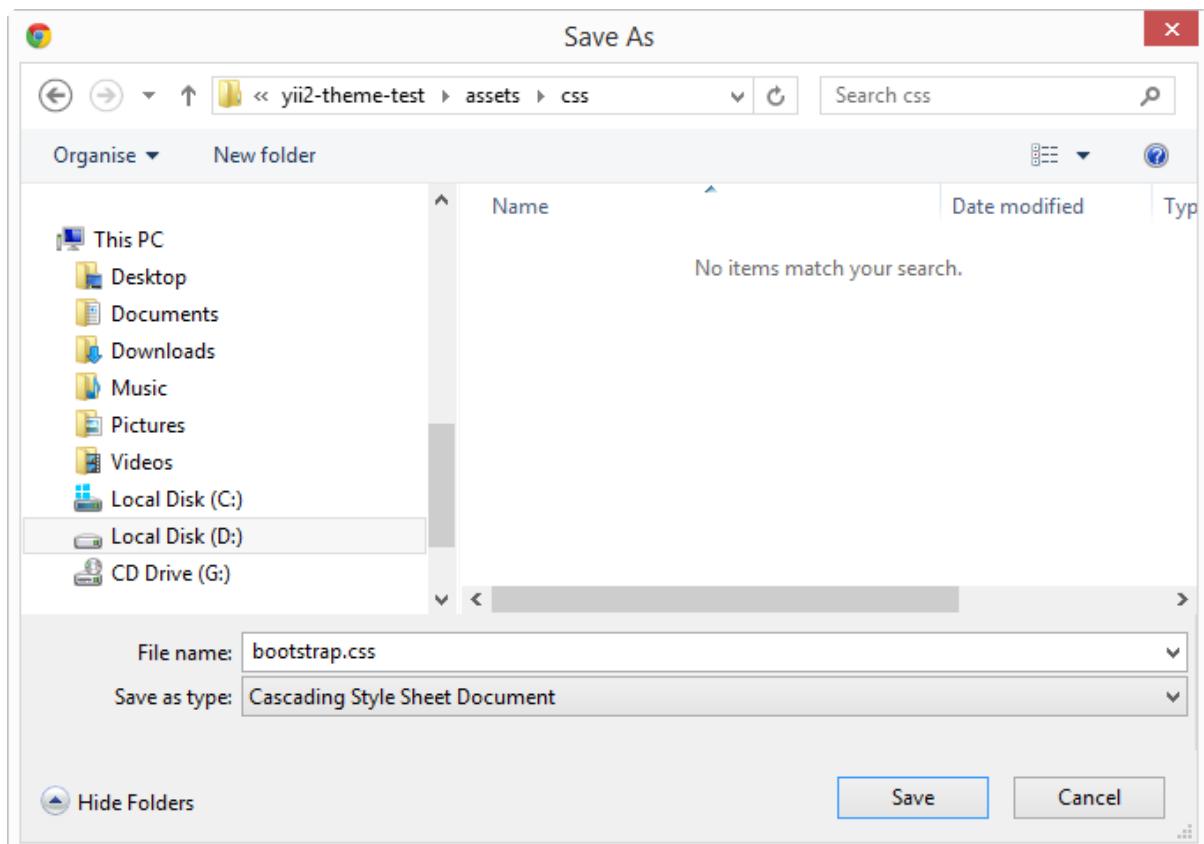
รูปที่ 229 หน้า Theme ที่นำมาสร้าง

คลิกที่เมนู Download แล้วคลิกขวาที่ bootstrap.css แล้วเลือก Save link as...



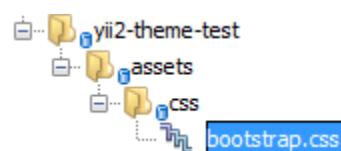
รูปที่ 230 การดาวน์โหลดไฟล์ CSS

จากนั้นนำไปวางใน vendor/kongoon/yii2-theme-test/assets/css



รูปที่ 231 การเลือกตำแหน่งที่เก็บไฟล์ CSS

จะได้ไฟล์ bootstrap.css ใน yii2-theme-test/assets/css เรียบร้อยดังนี้



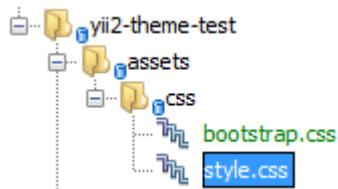
รูปที่ 232 ตำแหน่งที่เก็บไฟล์ CSS

## สร้าง CSS เพิ่มเติม

สร้างไฟล์ CSS เพิ่มเติมเพื่อเพิ่มข้อกำหนดให้กับ Theme ในที่นี่จะสร้างไฟล์ style.css โดยภายในใส่คำสั่ง padding-top:60px สำหรับ Tag body ดังนี้

```
body{  
    padding-top: 60px;  
}
```

โดยมีไฟล์ดังนี้



รูปที่ 233 การเพิ่มไฟล์ CSS

## สร้างไฟล์ Asset

ไฟล์ Asset เป็นไฟล์สำหรับใช้กำหนดการโหลด Theme และไฟล์ที่เกี่ยวข้อง เช่น JavaScript และ CSS อื่นๆ โดยส่วนใหญ่จะตั้งชื่อให้สอดคล้องกับชื่อ Theme ในที่นี้คือ TestAsset.php โดยเพิ่มข้อมูลดังนี้

```
<?php

namespace kongoon\theme\test; //กำหนด namespace

use yii\web\AssetBundle;

class TestAsset extends AssetBundle
{
    public $sourcePath='@vendor/kongoon/yii2-theme-test/assets';

    public $baseUrl = '@web';

    public $css=[
        'css/bootstrap.css',
        'css/style.css',
    ];

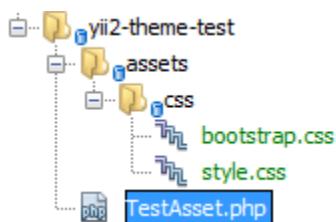
    public $js=[

    ];

    public $depends = [
        'yii\web\YiiAsset',
        'yii\bootstrap\BootstrapAsset',
        //'yii\bootstrap\BootstrapPluginAsset',
    ];
}
```

```
];  
  
public function init() {  
    parent::init();  
}  
}
```

จะได้โครงสร้างไฟล์ดังนี้



รูปที่ 234 ตำแหน่งที่ตั้งไฟล์ TestAsset.php

## เพิ่ม Theme ใน Layout

เปิดไฟล์ views/layouts/main.php และเพิ่มคำสั่งสำหรับการเรียกใช้งานดังนี้

```
<?php  
1   use yii\helpers\Html;  
2   use yii\bootstrap\Nav;  
3   use yii\bootstrap\NavBar;  
4   use yii\widgets\Breadcrumbs;  
5   //use app\assets\AppAsset;  
6   use kongoon\theme\test;  
7  
8  
9   /* @var $this \yii\web\View */  
10  /* @var $content string */  
11  
12  //AppAsset::register($this);  
13  test\TestAsset::register($this);  
14 ?>  
15  <?php $this->beginPage() ?>  
16  <!DOCTYPE html>  
17  <html lang="<?= Yii::$app->language ?>">
```

รูปที่ 235 การเรียกใช้งาน Theme ใหม่

## เพิ่มการ AutoLoad

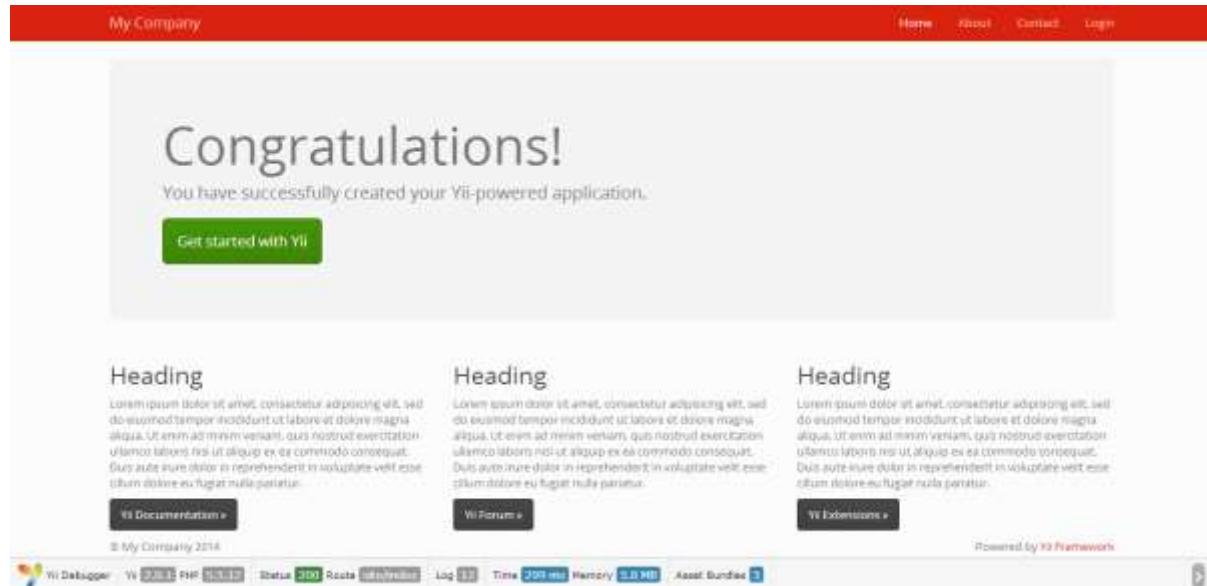
เนื่องจากไม่ได้ติดตั้งผ่าน Composer จะต้องเพิ่มการ AutoLoad ในไฟล์

vendor/composer/autoload\_psr4.php

```
'kongoon\\theme\\test\\' => array($vendorDir . '/kongoon/yii2-theme-test'),
```

## ทดสอบธีม (Theme)

ทดสอบโดยการเปิดหน้า Web Application จะได้รูปแบบดังนี้



รูปที่ 236 หน้า Web Application หลังการเปลี่ยน Theme แล้ว

## ติดตั้งแบบใช้ Composer

สำหรับวิธีการนี้ เป็นการติดตั้งผ่าน Composer โดยจะต้องสร้าง Repository แล้วนำไปวางใน

เว็บไซต์ Packagist.org ก่อน จึงจะสามารถติดตั้งได้ โดยดูวิธีการสร้างรีมอย่างและการติดตั้งผ่านรีมที่สร้างไว้แล้ว ดังนี้

## สร้าง Theme เอง

การติดตั้งแบบใช้ Composer นั้นจะเรียกได้ว่าเป็นการสร้าง Repository ในเว็บไซต์ที่ให้บริการใน ที่นี่จะใช้เว็บไซต์ Bitbucket.org และเพิ่ม Package ลงในเว็บไซต์ Packagist.org เพื่อให้สามารถติดตั้งผ่าน Composer ได้และยังสามารถเก็บไว้ติดตั้งในอนาคตได้ โดยมีขั้นตอนดังนี้



www.programmerthailand.com

หมายเหตุ หากไม่ต้องการสร้าง Theme เองให้ข้ามขั้นตอนนี้ แล้วไปดูในหัวข้อ **เลือกธีมที่มีอยู่แล้ว**

## เลือก Theme ที่ต้องการ

ในที่นี่จะใช้ Theme ชื่อ Superhero ซึ่งเป็น Bootstrap 3.3 โดยสามารถดาวน์โหลดได้ที่  
<http://bootswatch.com/superhero/>

## สร้าง Project ใน Bitbucket.org

เข้าสู่เว็บไซต์ <https://bitbucket.org/> และ Login จากนั้นเข้าเมนู Create



รูปที่ 237 ปุ่ม Create ใน Bitbucket.org

ใส่ชื่อ รายละเอียด และภาษาที่เขียน ที่สำคัญให้ยกเลิก Private repository ออก

### Create a new repository

The form for creating a new repository on Bitbucket.org. It includes fields for Name (set to 'yii2-theme-superhero'), Description ('Yii2 Framework Theme'), Access level (unchecked), Repository type (Git selected), Project management (Issue tracking and Wiki unchecked), Language ('PHP' selected), and Repository integrations (HipChat notifications unchecked). Buttons at the bottom include 'Create repository' and 'Cancel'.

Name *	<input type="text" value="yii2-theme-superhero"/>
Description	<input type="text" value="Yii2 Framework Theme"/>
Access level	<input type="checkbox"/> This is a private repository
Repository type	<input checked="" type="radio"/> Git <input type="radio"/> Mercurial
Project management	<input type="checkbox"/> Issue tracking <input type="checkbox"/> Wiki
Language	<input type="text" value="PHP"/>

### Repository integrations

HipChat  Enable HipChat notifications

**Create repository** **Cancel**

รูปที่ 238 การกำหนดข้อมูลสำหรับการสร้าง Repository



[www.programmerthailand.com](http://www.programmerthailand.com)

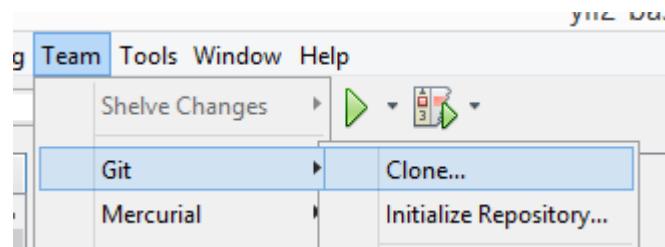
จากนั้นกดปุ่ม Create repository จะได้หน้าเว็บดังนี้

The screenshot shows the Bitbucket interface. On the left, there's a sidebar with options like 'Clone', 'Create branch', and 'Create pull request'. The main area is titled 'Repository setup' with the sub-section 'SourceTree'. It contains instructions to 'Get code into Bitbucket fast using Atlassian SourceTree or the command line'. Below this are two sections: 'SourceTree' with a 'Get started using the SourceTree client' link, and 'Command line' with links for 'I'm starting from scratch' and 'I have an existing project'. At the bottom, there's a note: 'Need a complete walkthrough? Visit Bitbucket 101 for a complete tutorial'.

รูปที่ 239 หน้าเว็บ Bitbucket.org หลังจากสร้าง Repository เรียบร้อยแล้ว

## Clone Project ลงในเครื่อง

เปิดโปรแกรม NetbeansIDE จากนั้นเข้าเมนู Team->Git->Clone



รูปที่ 240 การใช้ NetbeansIDE ในการ Clone

คลิกเมนู Clone ใน Bitbucket.org เพื่อคัด URL

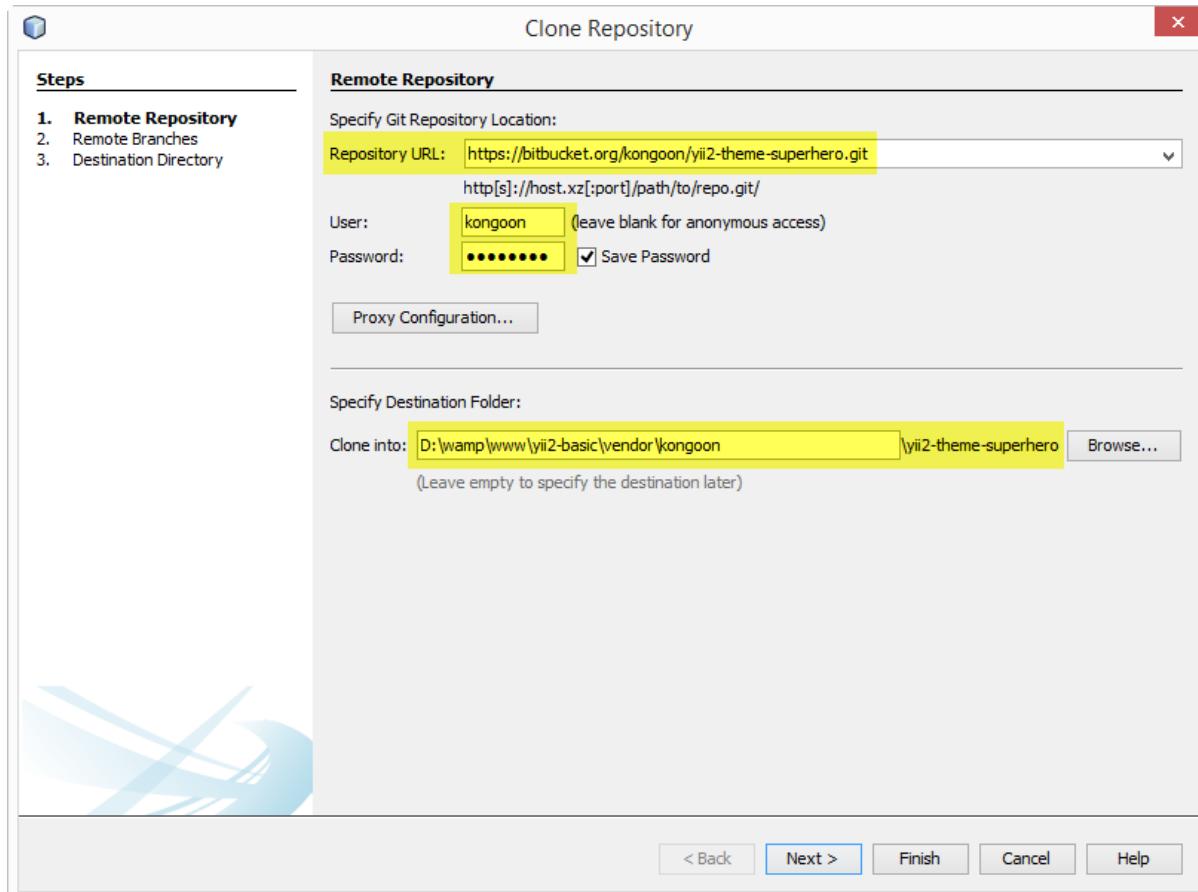
The screenshot shows the Bitbucket repository page for 'kongoon/yii2-theme-superhero'. In the 'ACTIONS' section, the 'Clone' button is highlighted. To its right, a tooltip shows the URL 'git clone https://kongoon@bitbucket.org/kongoon/yii2-theme-superhero.git'. A cursor is hovering over the URL text.

รูปที่ 241 การเบิดตู้ Link สำหรับ Clone

จากนั้นคัดลอกเฉพาะ URL ที่ต้องการคือ

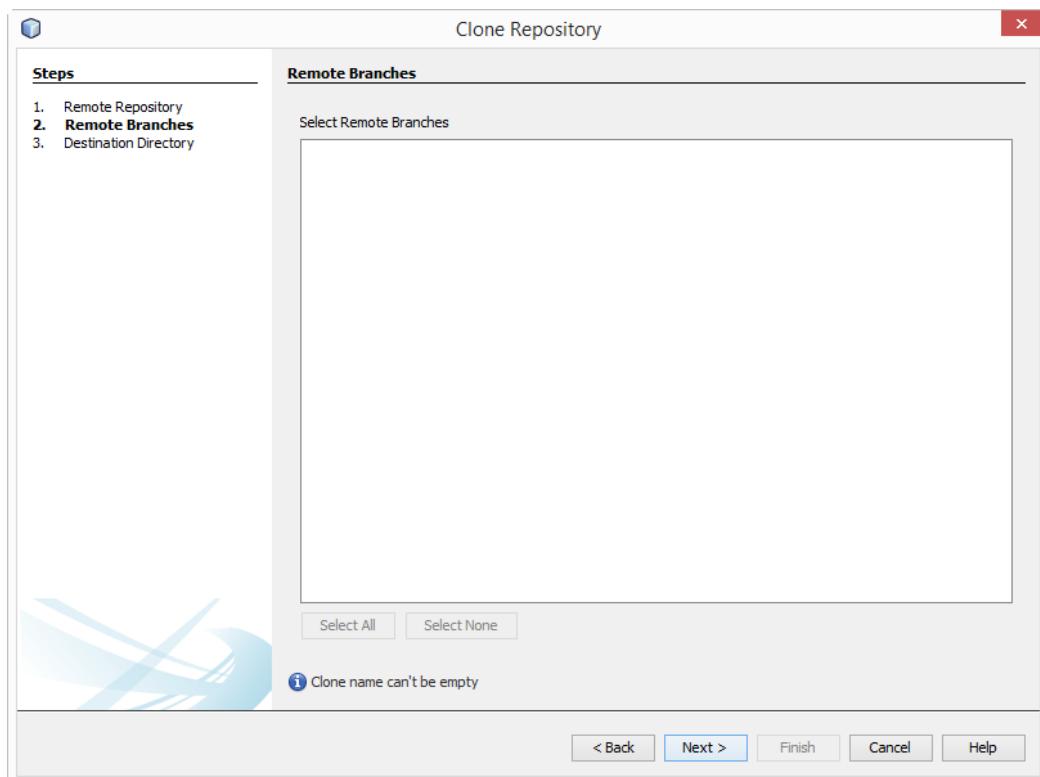
`https://bitbucket.org/kongoon/yii2-theme-superhero.git`

นำไปร่วงใน Netbeans IDE กรอก Username และ Password สำหรับ Login เข้าเว็บไซต์ Bitbucket.org และ กำหนดที่เก็บ



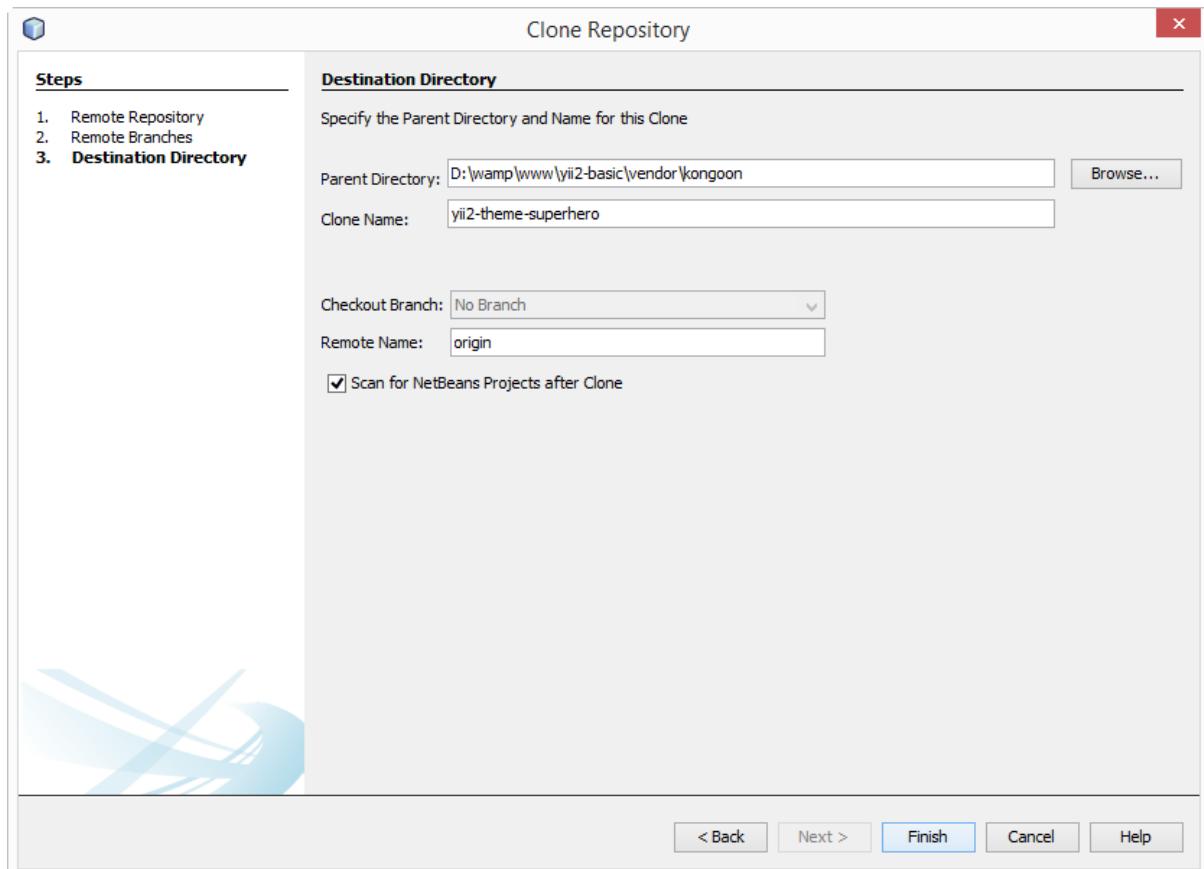
รูปที่ 242 การกำหนดการตั้งค่าไปยัง Repository

กดปุ่ม Next > จะปรากฏหน้าต่างดังนี้



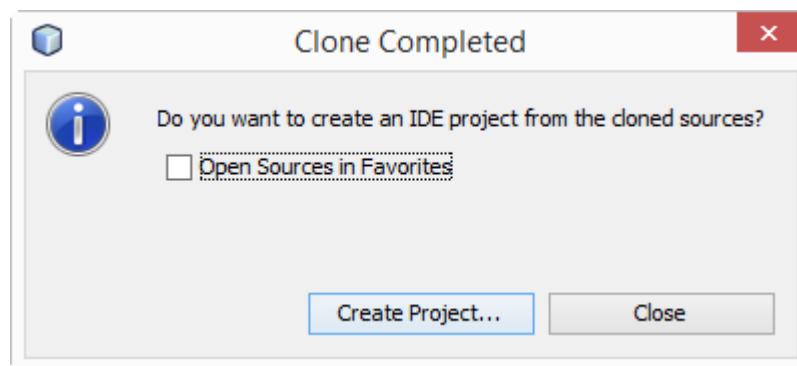
รูปที่ 243 การตั้งค่า Remote Branches

จากนั้นกดปุ่ม Next > อีกครั้ง จะปรากฏหน้าต่างการกำหนดที่เก็บ ดังนี้



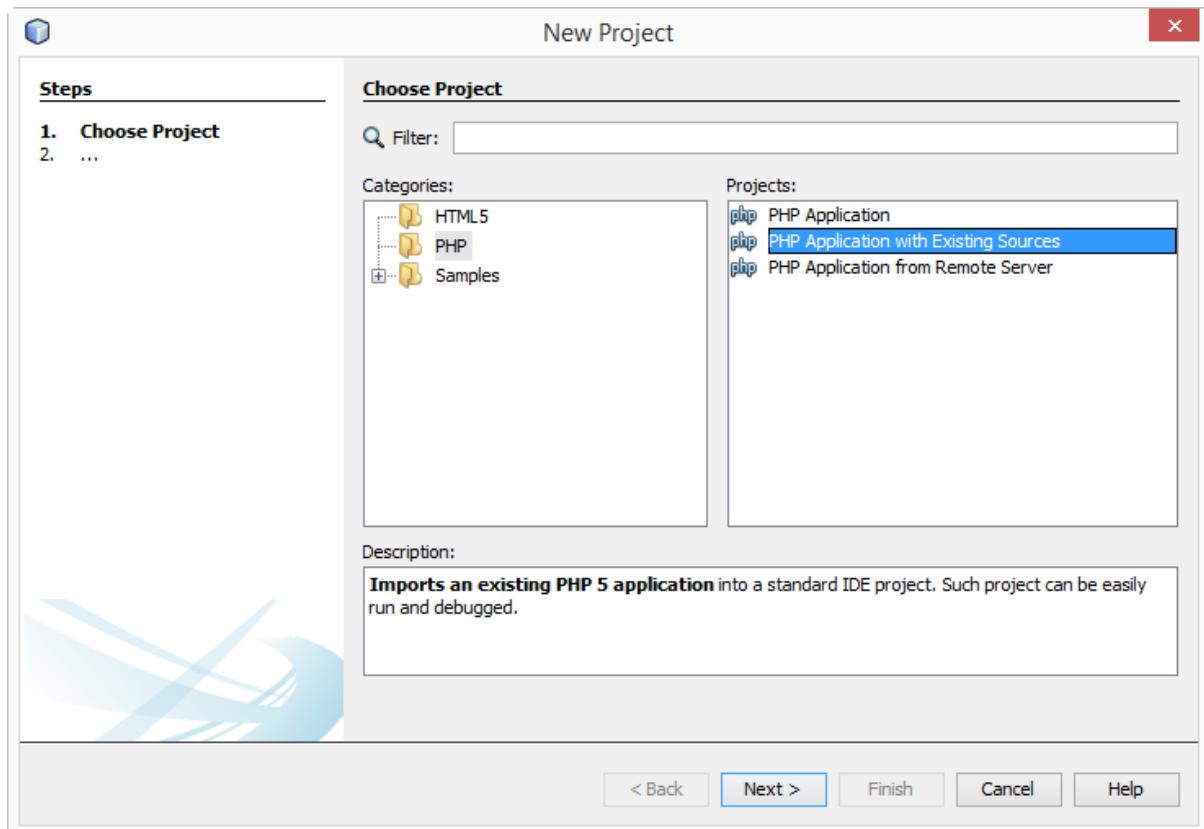
รูปที่ 244 การกำหนด Destination Directory

จากนั้นกดปุ่ม Finish จะปรากฏ Popup ดังนี้



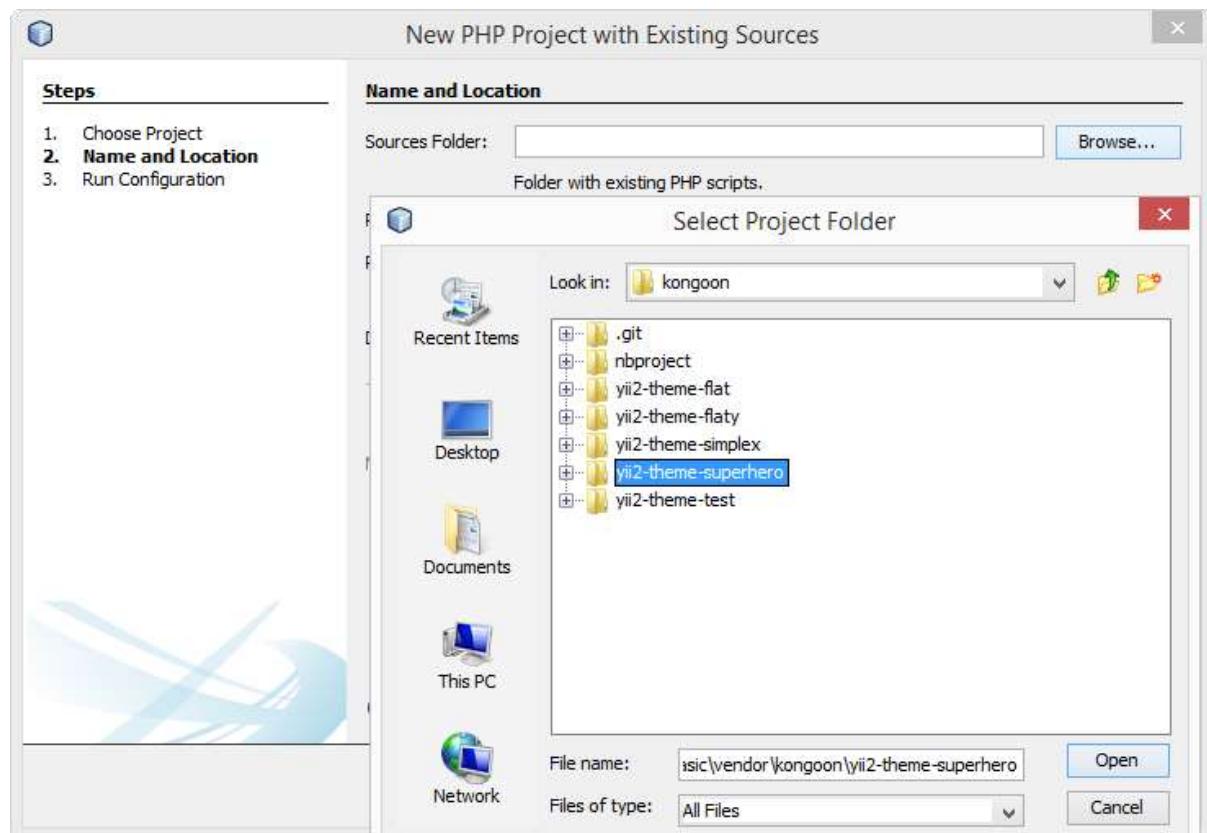
รูปที่ 245 POPUP หลังกดปุ่ม Finish

ให้กดปุ่ม Create Project... และเลือกแบบ PHP Application with Existing Sources จากนั้นกดปุ่ม Next >



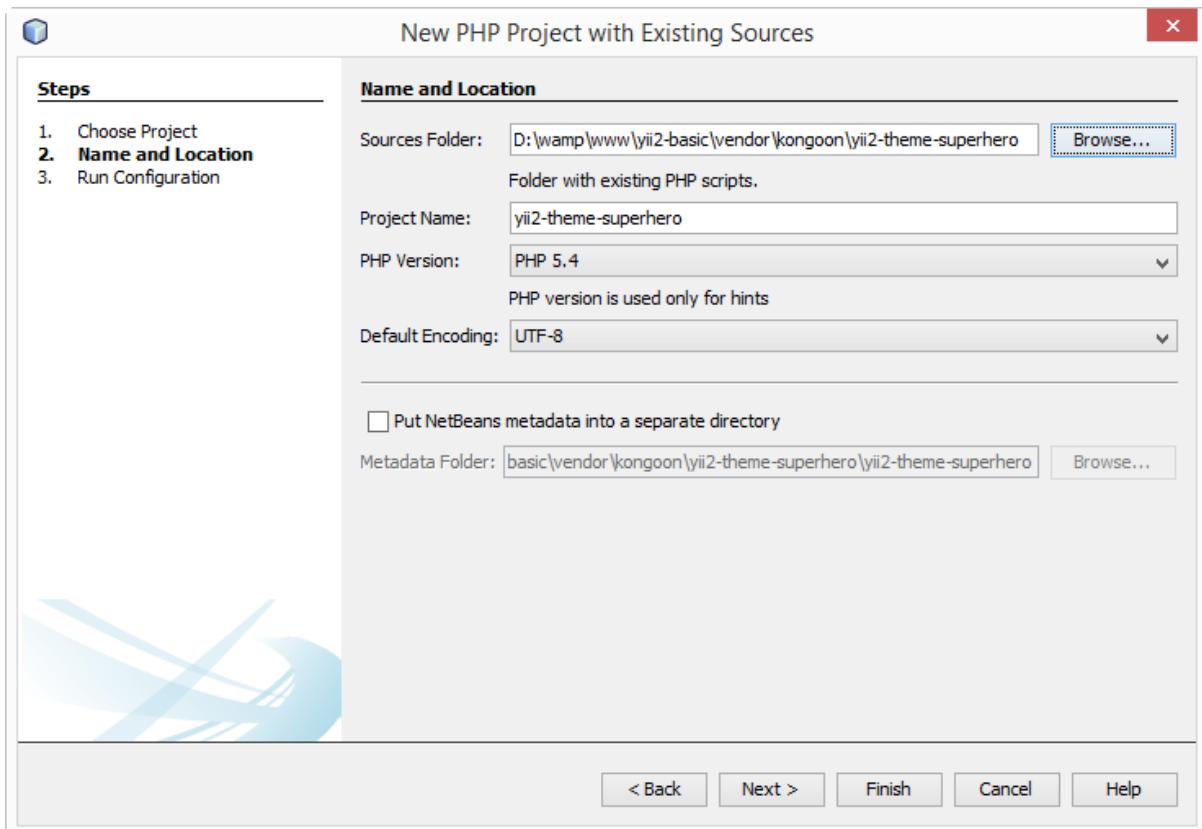
รูปที่ 246 การสร้าง Project ใหม่ใน NetbeansIDE

จะปรากฏหน้าต่างให้เลือก Folder ที่เก็บ Project ในที่นี่จะอยู่ที่ vendor/Kongoon/yii2-theme-superhero



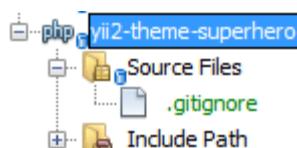
รูปที่ 247 การเลือก Project Folder

จากนั้นกดปุ่ม Open



รูปที่ 248 รายการการตั้งค่าสำหรับสร้าง Project ใหม่

และปุ่ม Finish ตามลำดับ จะเห็นว่ามี Project ใหม่เกิดขึ้นแล้ว



รูปที่ 249 Project ใน Netbeans IDE

## สร้างไฟล์ต่างๆ ที่เกี่ยวข้อง

สำหรับไฟล์ต่างๆ ที่สำคัญในการสร้าง Theme นั้นมีดังนี้

- README.md
- composer.json
- SuperheroAsset.php
- ไฟล์ CSS (bootstrap.css และ style.css)

## สร้างไฟล์ README.md

ไฟล์ README.md เป็นไฟล์อธิบายรายละเอียดของ Repository นี้ เช่น การติดตั้ง การตั้งค่า เป็นต้น โดยสามารถเพิ่มได้ตามต้องการ เช่น

```
Yii2 Superhero Theme
```

```
=====
```

```
Theme for Yii2 Web Application
```

```
Installation
```

```
-----
```

```
The preferred way to install this extension is through  
[composer] (http://getcomposer.org/download/) .
```

```
Either run
```

```
```
```

```
php composer.phar require --prefer-dist kongoon/yii2-theme-superhero "*"
```

```
```
```

```
or add
```

```
```
```

```
"kongoon/yii2-theme-superhero": "*"
```

```
```
```

```
to the require section of your `composer.json` file.
```

```
Usage
```

```
-----
```



```
Open your layout e.g. views/layouts/main.php and add  
```  
use kongoon\theme\superhero;  
  
superhero\SuperheroAsset::register($this);  
```
```

## สร้างไฟล์ composer.json

ไฟล์ composer.json เป็นไฟล์สำหรับโหลดการติดตั้งแบบอัตโนมัติโดยจำเป็นจะต้องกำหนดรูปแบบให้เป็นมาตรฐานของการติดตั้งใน Yii2 ดังนี้

```
{  
  
    "name": "kongoon/yii2-theme-superhero",  
  
    "description": "Theme for Yii2 Web Application",  
  
    "type": "yii2-extension",  
  
    "keywords": ["yii2", "extension", "theme", "superhero"],  
  
    "license": "MIT",  
  
    "authors": [  
        {  
            "name": "Programmer Thailand",  
            "email": "contact@programmerthailand.com"  
        }  
    ],  
  
    "require": {  
        "yiisoft/yii2": "*"  
    },  
  
    "autoload": {
```



```
"psr-4": {  
    "kongoon\\theme\\superhero\\": ""  
}  
}
```

## สร้างไฟล์ SuperheroAsset.php

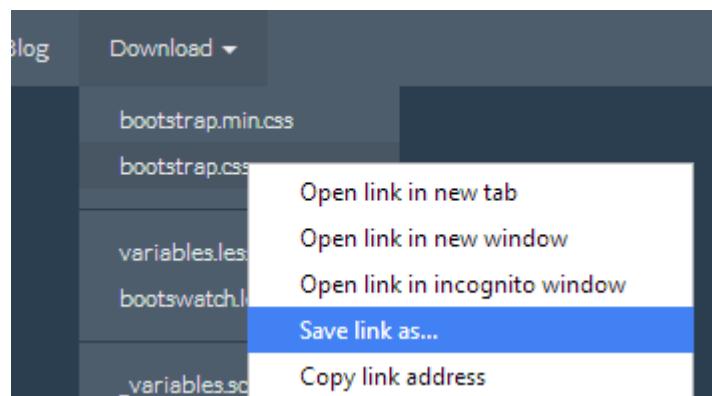
ไฟล์ SuperheroAsset.php เป็นไฟล์สำหรับโหลด Asset ต่างๆ เช่น CSS JavaScript ที่จำเป็นสำหรับ Theme

```
<?php  
/*  
 * 2014-12-22  
 * @author Programmer Thailand  
<contact@programmerthailand.com>  
 * http://bootswatch.com/superhero/  
 */  
  
namespace kongoon\theme\superhero;  
  
use yii\web\AssetBundle;  
class SuperheroAsset extends AssetBundle  
{  
    public $sourcePath='@vendor/kongoon/yii2-theme-superhero/assets';  
    public $baseUrl = '@web';  
  
    public $css=[  
        'css/bootstrap.css',  
        'css/style.css',  
    ];
```

```
public $js=[  
];  
  
public $depends = [  
    'yii\web\YiiAsset',  
    'yii\bootstrap\BootstrapAsset',  
    //'yii\bootstrap\BootstrapPluginAsset',  
];  
  
public function init() {  
    parent::init();  
}  
}
```

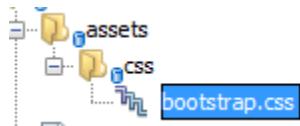
## ดาวน์โหลด bootstrap.css

เข้าเว็บไซต์ <http://bootswatch.com/superhero/> เพื่อดาวน์โหลด bootstrap.css โดยคลิกที่เมนู Download และคลิกขวาที่ bootstrap.css



รูปที่ 250 การดาวน์โหลด CSS

จากนั้นนำไฟล์ที่ vendor/kongoon/yii2-theme-superhero/assets/css



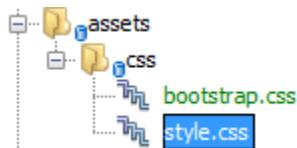
รูปที่ 251 ไฟล์ bootstrap.css ที่ดาวน์โหลด

## สร้างไฟล์ style.css

ไฟล์ style.css เป็นการกำหนดรูปแบบเว็บไซต์เพิ่มเติมในที่นี่จะกำหนด padding-top ให้กับ tag body โดยการเพิ่มข้อมูลดังนี้

```
body {  
    padding-top: 60px;  
}
```

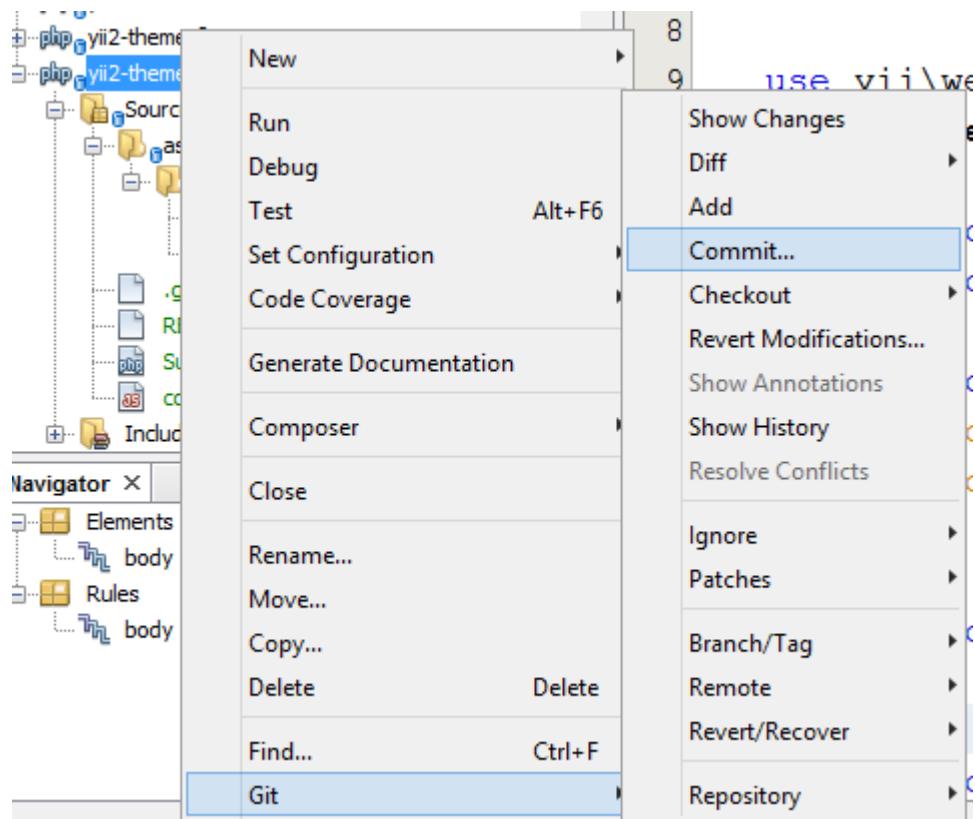
โดยมีโครงสร้างไฟล์ดังนี้



รูปที่ 252 ตำแหน่งไฟล์ style.css

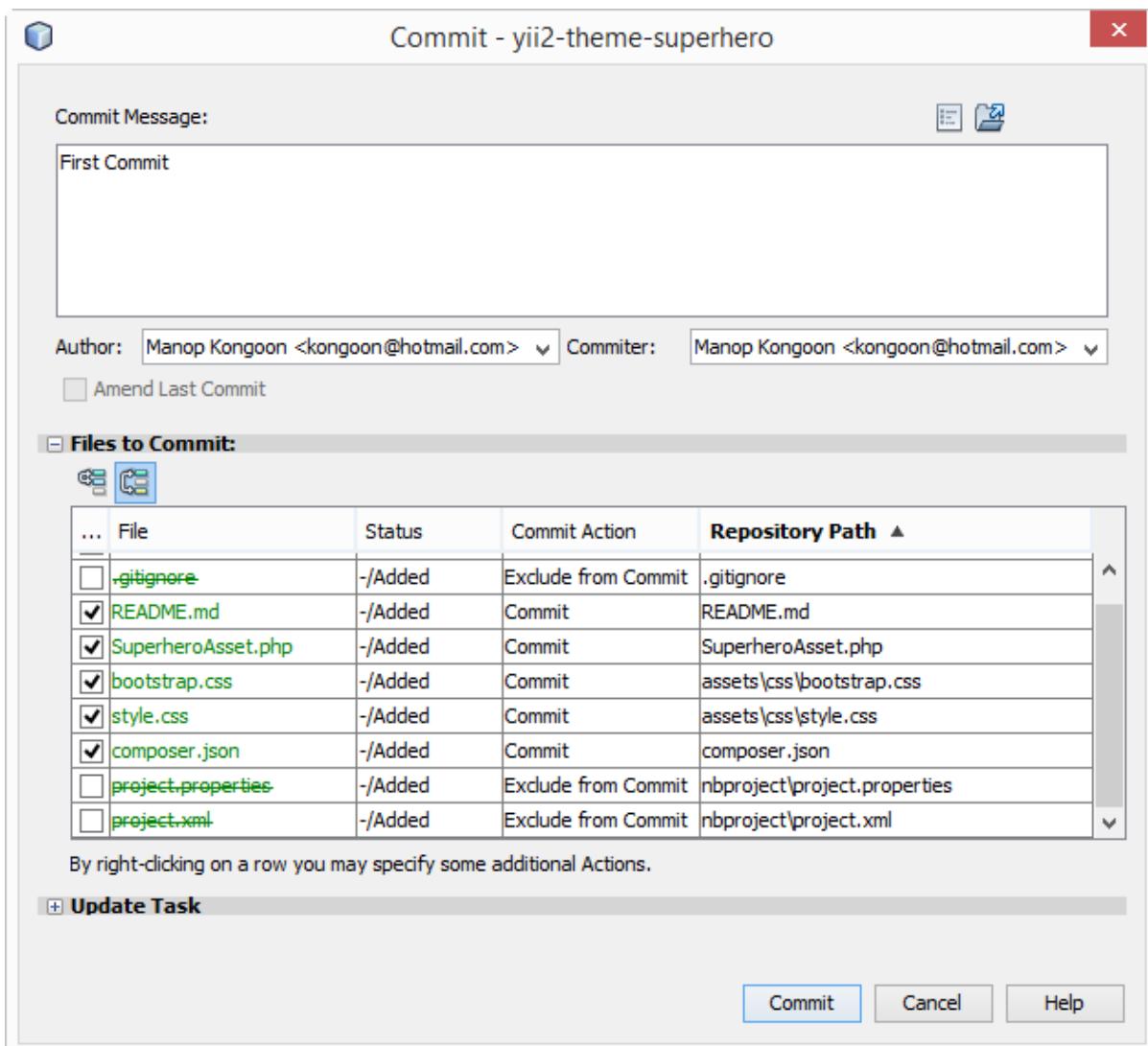
## Commit และ Push ไปที่ Bitbucket.org

ทำการ Commit ไปยัง Repository ที่ได้สร้างไว้โดยคลิกขวาที่ Project และเลือก Git และ Comit...



รูปที่ 253 การกำหนด Commit ไฟล์

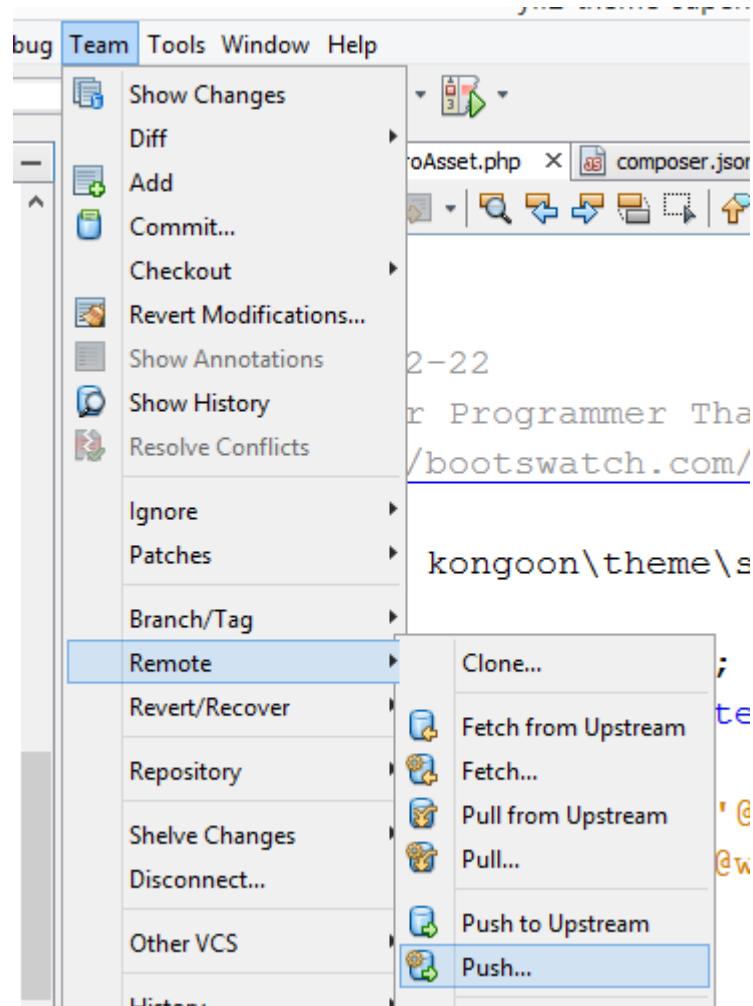
ยกเลิกไฟล์ที่ไม่เกี่ยวข้อง



รูปที่ 254 การลงทะเบียนไฟล์ที่จะ Commit

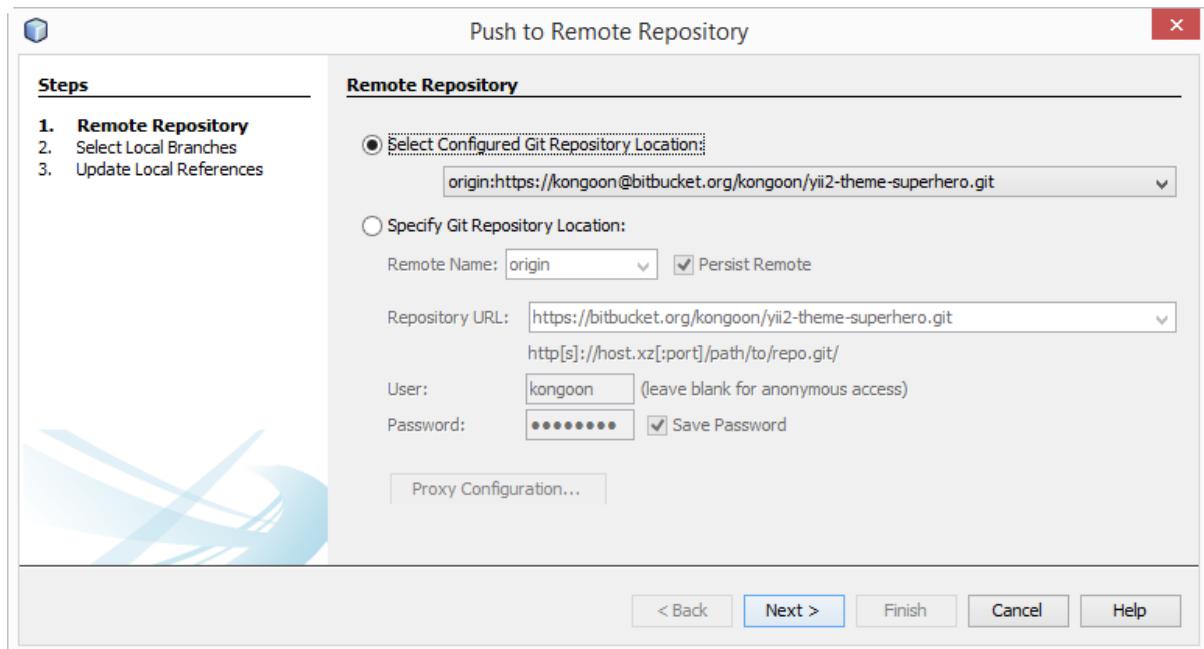
จากนั้นกดปุ่ม Commit แล้วหน้าต่างนี้จะหายไปซึ่งระบบได้ทำการ Commit ไว้แล้วเหลือ Pull ไปยัง Repository ใน Bitbucket.org ที่ได้สร้างไว้

ไปที่เมนู Team->Remote->Push



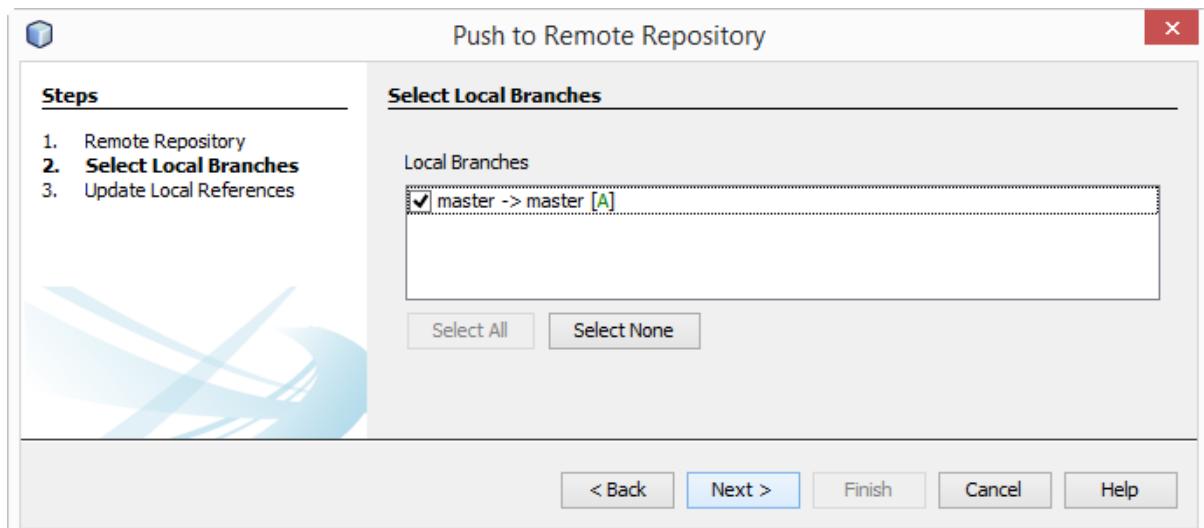
รูปที่ 255 การ Push ไฟล์ไปยัง Repository

จะปรากฏการตั้งค่าที่ตั้งค่าไว้ในตอนแรก



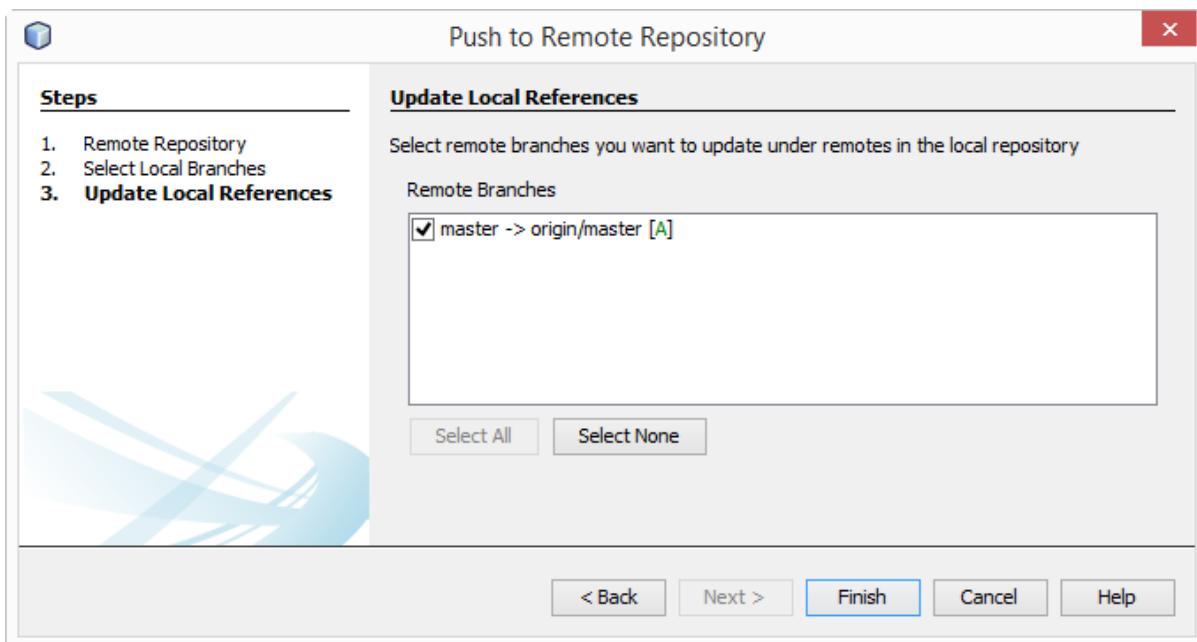
รูปที่ 256 การเลือก Repository

จากนั้นกดปุ่ม Next >



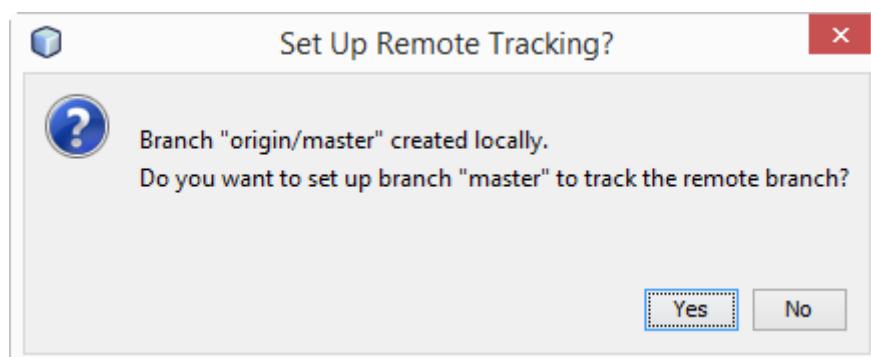
รูปที่ 257 การเลือก Local Branches

ให้คลิกที่ Check Box master และกดปุ่ม Next >



รูปที่ 258 การเลือก Update Local References

จากนั้นกดปุ่ม Finish จะมี Popup ให้กดปุ่ม Yes



รูปที่ 259 การ Set Up Remote Tracking

จากนั้นเปิด Bitbucket.org และเข้า Repository ที่สร้างจากนั้นไปที่เมนู Source จะโครงสร้างไฟล์ต่างๆ ดังนี้

The screenshot shows a GitHub repository page for 'kongoon/yii2-theme-superhero'. The repository has 160 stars. The 'Source' tab is selected, showing the repository's directory structure:

- assets/css
- nbproject
- .gitignore
- README.md
- SuperheroAsset.php
- composer.json

รูปที่ 260 ไฟล์ต่างๆ ที่ถูก Push ขึ้น Repository

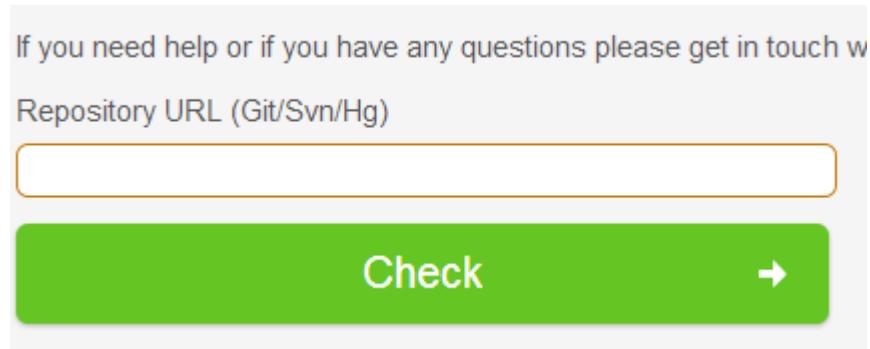
## เพิ่ม Package ใน Packagist.org

การเพิ่ม Repository ลงใน Packagist.org นั้นจะทำให้สามารถติดตั้ง Theme ผ่าน composer ได้โดยเข้าเว็บไซต์ <https://packagist.org/> และ Login เข้าระบบ จากนั้นคลิกที่ปุ่ม Submit Package



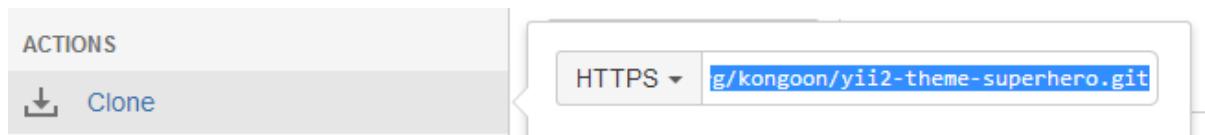
รูปที่ 261 บุํมสำหรับ Submit Package

จากนั้นจะปรากฏช่องสำหรับใส่ Link ไปยังไฟล์ .git



รูปที่ 262 การตรวจสอบไฟล์ .git ก่อนการ Submit

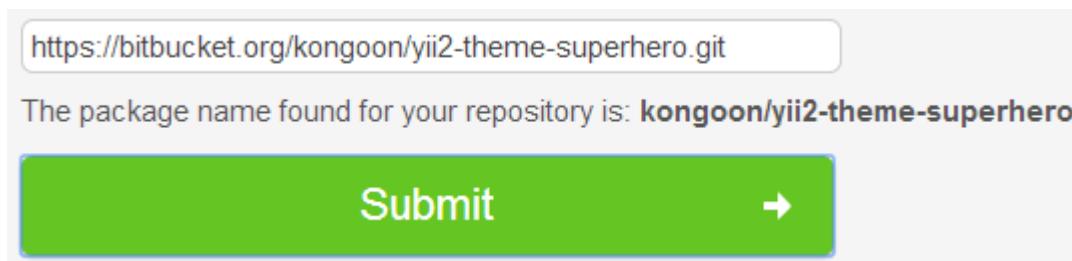
ให้นำ Link จาก Bitbucket.org มาใส่ในช่องนี้โดยคลิกที่ Clone จะปรากฏ Link ไปยังไฟล์ .git



จากนั้นปรับแต่ง Link เล็กน้อยโดยเอา username@ ออกก่อน จะได้เป็น

`https://bitbucket.org/kongoon/yii2-theme-superhero.git`

นำ Link มาวางแล้วกดปุ่ม Check ระบบจะตรวจสอบความถูกต้องจากนั้นกดปุ่ม Submit



รูปที่ 263 หลังจากตรวจสอบเสร็จแล้วจะมีปุ่ม Submit ให้กด

จะปรากฏรายการสรุป

☆ kongoon/yii2-theme-superhero  theme extension yii2 Abandon Delete Force Update Edit

This package is not auto-updated. Please set up the BitBucket POST Service for Packagist so that it gets updated whenever you push!

Theme for Yii2 Web Application

Maintainer: kongoon (add maintainer)  
Canonical: <https://bitbucket.org/kongoon/yii2-theme-superhero.git>

Overall: 0 installs  
30 days: 0 installs  
Today: 0 installs

**dev-master** reference: 3617365 Delete

require: "kongoon/yii2-theme-superhero": "dev-master"

**Author**  
• Programmer Thailand <[contact@programmerthailand.com](mailto:contact@programmerthailand.com)>

Requires	Requires (Dev)	Suggests
• <a href="#">yisoft/yii2</a> : *	None	None
Provides	Conflicts	Replaces
None	None	None

2014-12-22 19:52 UTC  
MIT

รูปที่ 264 รายละเอียด Package ที่ได้ Submit เรียบร้อย

จากนั้นไปตั้งค่า POST ดูรายละเอียดจาก Link BitBucket POST Service ในส่วนของ Link และ API Key

## Your API Token

Show API Token

You can use your API token to interact with the Packagist API.

## GitHub Service Hook

Enabling the Packagist service hook ensures that your package will always be updated instantly when changes are made to your GitHub repository. To do this, go to your GitHub repository, click the "Settings" button, then "Webhooks & Services". Add a "Packagist" service using the "Add service" button. Enter your API token (see above), plus your Packagist username. Check the "Active" box and submit the form. You're done! If Packagist removes the warning about the package not being auto-updated, you've successfully enabled the service hook.

## Bitbucket POST Service

To enable the Bitbucket service hook, go to your BitBucket repository, open the "Admin" tab and select "Service Hooks". Click the "Add Service" button and choose "Packagist". Select "POST" from the dropdown menu and enter the URL <https://packagist.org/api/bitbucket?username=kongoon&apiToken=...> in the "URL" field. Click "Save" and you're done!

รูปที่ 265 การกำหนดให้มีการ Hooks แบบ Auto

จากนั้นเปิด BitBucket.org ไปที่ Repository ที่สร้างจากนั้นไปที่เมนู Settings->Hooks->POST

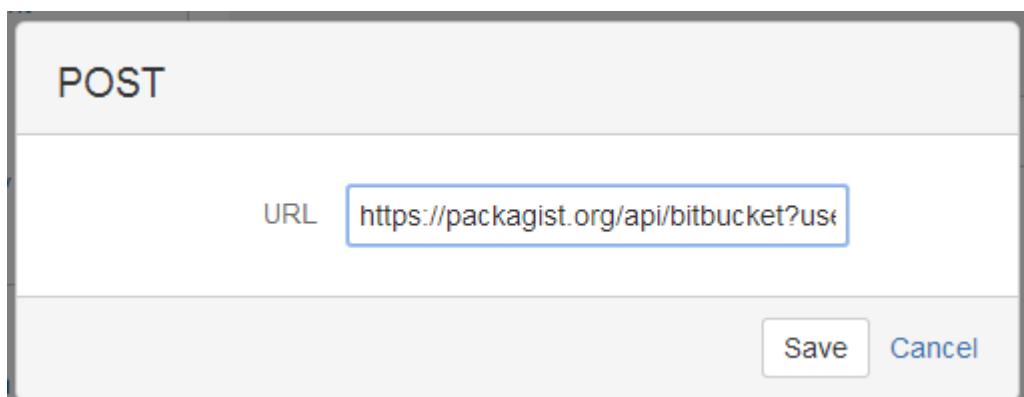


[www.programmerthailand.com](http://www.programmerthailand.com)

The screenshot shows the Bitbucket Settings page for a repository named 'kongoon yii2-theme-superhero'. The left sidebar includes actions like Clone, Create branch, Create pull request, Compare, Fork, and various navigation links such as Overview, Source, Commits, Branches, Pull requests, Downloads, and Settings. The 'Settings' link is highlighted with a yellow box. The main content area is titled 'Settings' and contains sections for GENERAL, INTEGRATIONS, ISSUES, and WIKI. The 'INTEGRATIONS' section is expanded, showing 'HipChat integration' and 'Hooks', which is also highlighted with a yellow box. A sub-section for 'Hooks' shows a dropdown menu set to 'POST' and a 'Add hook' button.

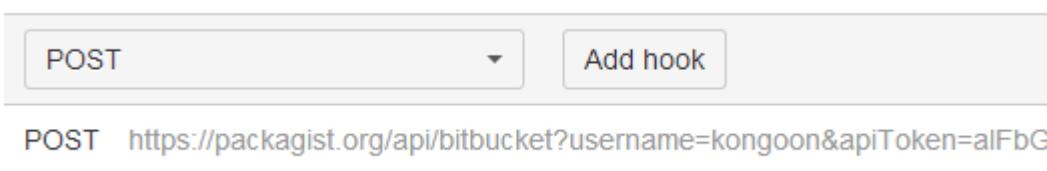
รูปที่ 266 การเพิ่ม Hooks แบบ POST

จากนั้นกดปุ่ม Add hook และใส่ Link ที่ได้จาก Packagist.org



รูปที่ 267 การกำหนด URL จาก packagist.org

จากนั้นกดปุ่ม Save ก็จะปรากฏ Link ดังนี้



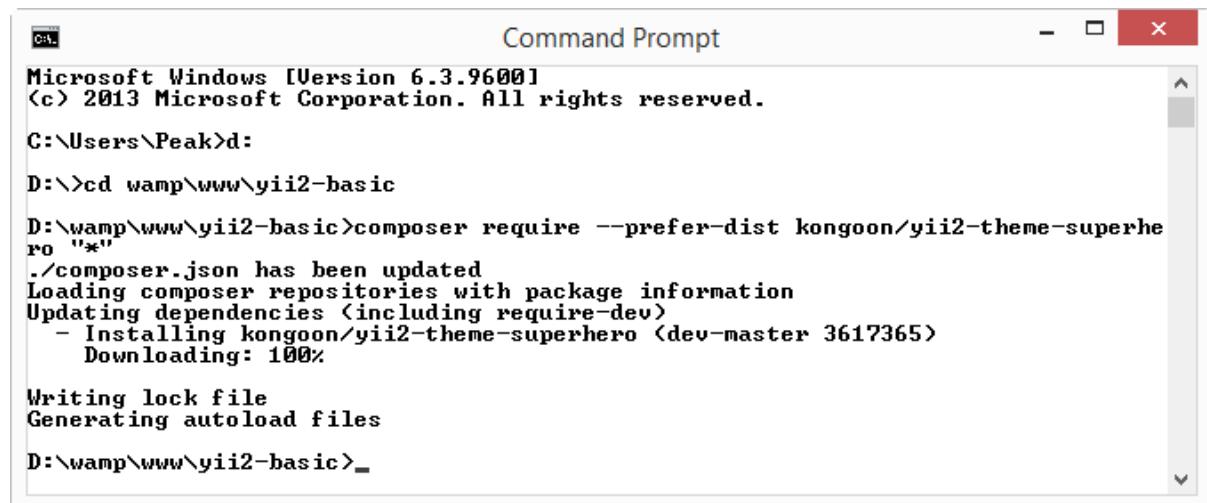
รูปที่ 268 รายการ URL หลังการเพิ่ม Hooks แบบ POST

## เลือกใช้ธีมที่มีอยู่แล้ว

เปิดโปรแกรม Command Prompt และไปที่ Project ในที่นี่คือ D:\wamp\www\yii2-basic  
จากนั้นพิมพ์คำสั่งในการติดตั้ง Theme ดังนี้

```
composer require --prefer-dist kongoon/yii2-theme-superhero  
"*"
```

หลังจากติดตั้งเสร็จเรียบร้อยจะได้ลักษณะนี้

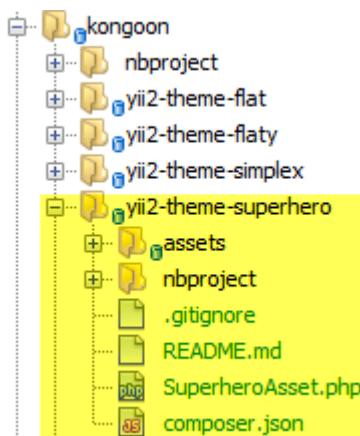


```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Peak>d:
D:>cd wamp\www\yii2-basic
D:\wamp\www\yii2-basic>composer require --prefer-dist kongoon/yii2-theme-superhero "*"
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
  - Installing kongoon/yii2-theme-superhero (dev-master 3617365)
    Downloading: 100%
Writing lock file
Generating autoload files
D:\wamp\www\yii2-basic>_
```

รูปที่ 269 การติดตั้ง Theme ด้วย Composer

จะเห็นว่ามีโครงสร้างไฟล์เรียบร้อย



รูปที่ 270 โครงสร้างไฟล์ Theme หลังการติดตั้ง

## เพิ่ม Theme ใน Layout

เปิดไฟล์ views/layouts/main.php เพื่อเพิ่ม Theme ใหม่เข้าไปใน Layout โดยแก้ไขการโหลด Asset ดังนี้

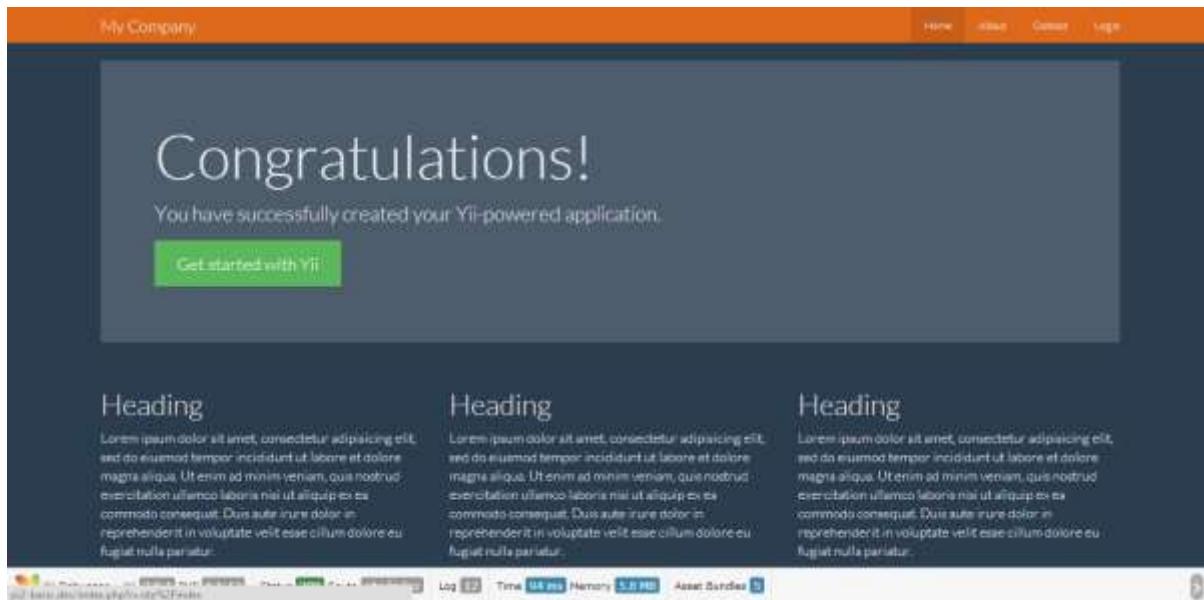


```
5  use yii\widgets\Breadcrumbs;
6  //use app\assets\AppAsset;
7  use kongoon\theme\superhero;
8
9  /* @var $this \yii\web\View */
10 /* @var $content string */
11
12 //AppAsset::register($this);
13 superhero\SuperheroAsset::register($this);
14 ?>
15 <?php $this->beginPage() ?>
16 <!DOCTYPE html>
```

รูปที่ 271 การเปลี่ยน Theme สำหรับ Layout

## ทดสอบธีม (Theme)

เปิดหน้า Web Application จะเห็นว่ามีการเปลี่ยนแปลงเป็น Theme ใหม่เรียบร้อยแล้ว



รูปที่ 272 หน้า Web Application หลังการเปลี่ยน Theme

## ธีมสำหรับ Yii2 Advanced Application Template

สำหรับ Theme ใน Yii2 Advanced Application Template นั้นมี 2 ส่วนคือ ส่วนของ Frontend และ ส่วนของ Backend โดยในตัวอย่างนี้จะเป็นการกำหนดให้ Frontend และ Backend ใช้ Theme คนละตัว

คงจะเสียสัก เพื่อให้เข้าใจการเปลี่ยนธีมมากยิ่งขึ้น โดยจะข้ามขั้นตอนของการสร้าง Theme ซึ่งดูเพิ่มเติมได้ที่ บทที่ 5 ชื่อ (Theme) ในหัวข้อ ติดตั้งแบบใช้ Composer หัวข้อย่อ สร้าง Theme เอง

## รีมสำหรับ Frontend

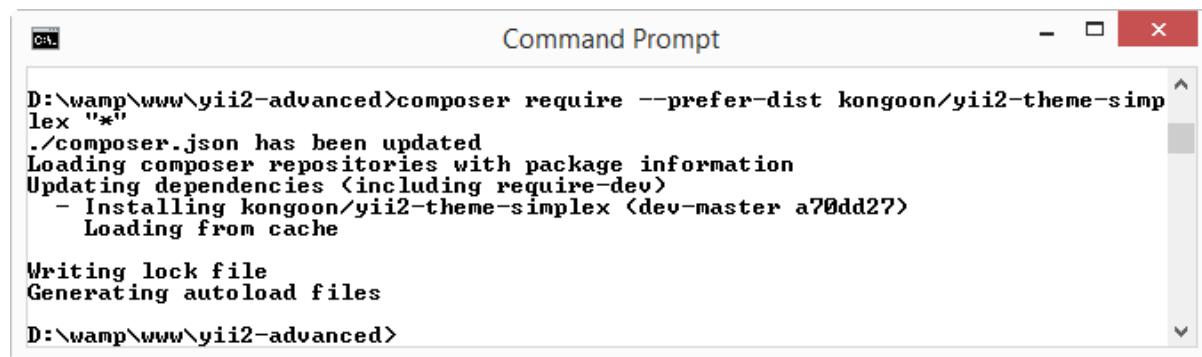
สำหรับ Frontend นั้นจะใช้ Theme ที่ได้สร้างไว้แล้วอยู่ที่

```
https://bitbucket.org/kongoon/yii2-theme-simplex
```

ติดตั้งผ่าน Composer โดยเข้าไฟล์เดอร์ Project พิมพ์คำสั่ง

```
composer require --prefer-dist kongoon/yii2-theme-simplex "*"
```

จะได้หน้า Command Prompt ดังนี้

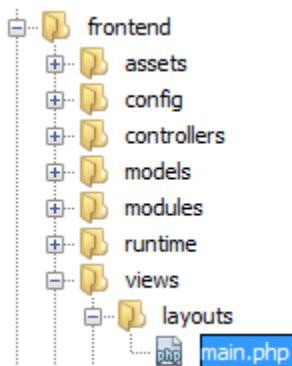


```
D:\wamp\www\yii2-advanced>composer require --prefer-dist kongoon/yii2-theme-simplex "*"
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
  - Installing kongoon/yii2-theme-simplex (dev-master a70dd27)
    Loading from cache

Writing lock file
Generating autoload files
D:\wamp\www\yii2-advanced>
```

รูปที่ 273 การติดตั้ง Theme ด้วย Composer

แก้ไขไฟล์ Layout ของ Web Application ในส่วน Frontend ใน frontend/views/layouts/main.php



รูปที่ 274 ตำแหน่งไฟล์ Layout/main.php

จากนั้นแก้ไขไฟล์ดังนี้

```
use kongoon\theme\simplex;
simplex\SimplexAsset::register($this);
```

โดย Comment คำสั่งเดิมแล้วเพิ่มคำสั่งใหม่ดังนี้

```
5   use yii\widgets\Breadcrumbs;
6   //use frontend\assets\AppAsset;
7   use kongoon\theme\simplex;
8   use frontend\widgets\Alert;
```

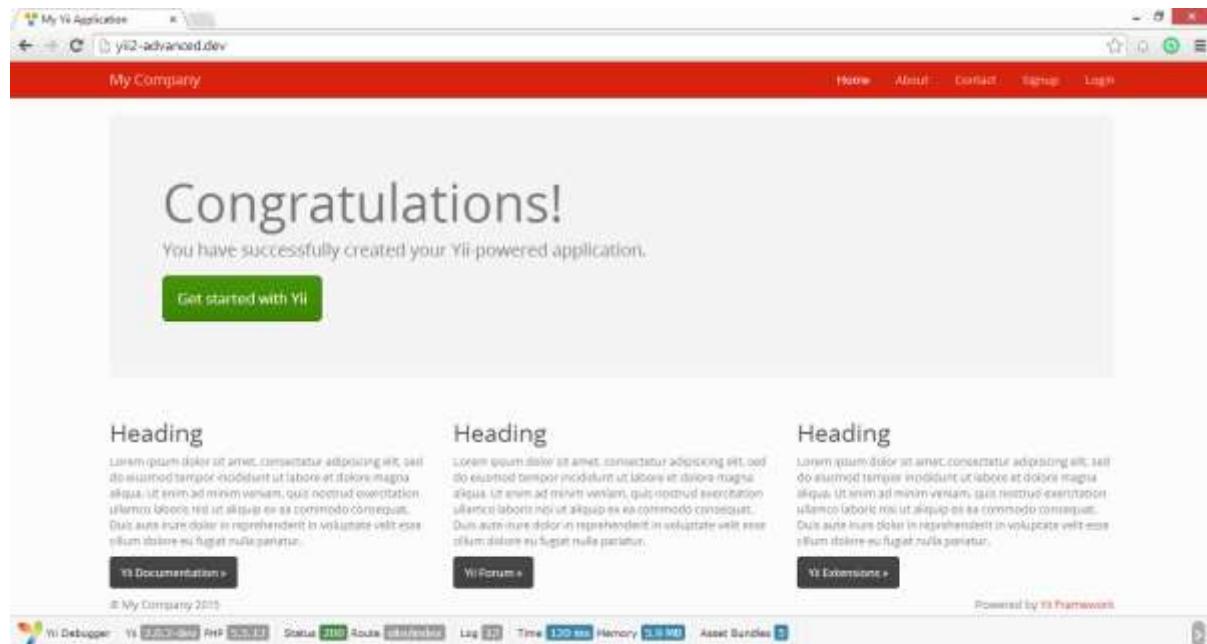
รูปที่ 275 การเพิ่ม Theme ใน Layout

และ

```
13  //AppAsset::register($this);
14  simplex\SimplexAsset::register($this);
```

รูปที่ 276 การเรียกใช้งาน Theme ด้วย register asset

จากนั้นทดสอบโดยการ Refresh Web Application จะได้หน้า Web Application ส่วน Frontend ดังนี้



รูปที่ 277 หน้า Web Application หลังการติดตั้ง Theme เรียบร้อยแล้ว

## รีมสำหรับ Backend

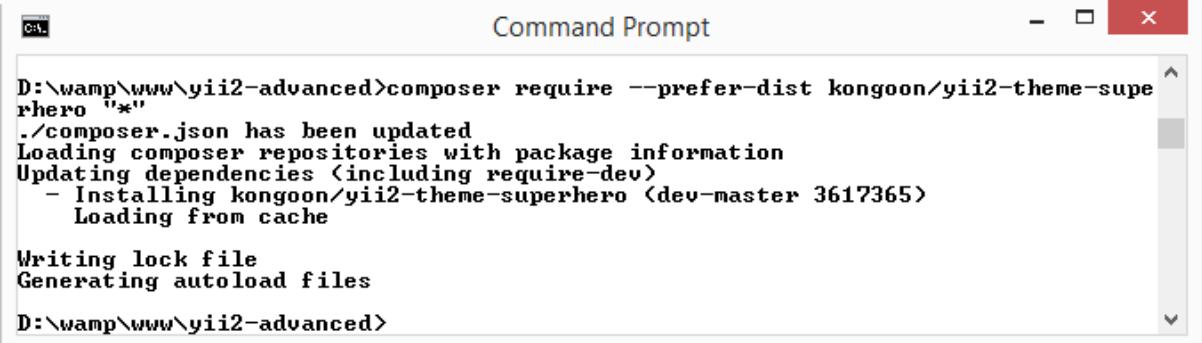
สำหรับ Backend นั้นจะใช้ Theme ที่ได้สร้างไว้แล้วอยู่ที่

<https://bitbucket.org/kongoon/yii2-theme-superhero>

ติดตั้งผ่าน Composer โดยเข้าไฟล์เดอร์ Project พิมพ์คำสั่ง

```
composer require --prefer-dist kongoon/yii2-theme-superhero "*"
```

จะได้หน้า Command Prompt ดังนี้

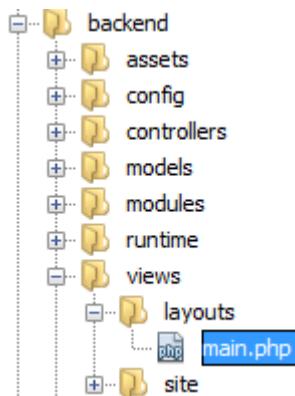


```
D:\wamp\www\yii2-advanced>composer require --prefer-dist kongoon/yii2-theme-superhero "*"
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
- Installing kongoon/yii2-theme-superhero (dev-master 3617365)
  Loading from cache

Writing lock file
Generating autoload files
D:\wamp\www\yii2-advanced>
```

รูปที่ 278 การติดตั้ง Theme ด้วย Composer

แก้ไขไฟล์ Layout ของ Web Application ในส่วน Frontend ใน backend/views/layouts/main.php



รูปที่ 279 ตำแหน่งไฟล์ Layout/main.php

จากนั้นแก้ไขไฟล์ดังนี้

```
use kongoon\theme\superhero;
simplex\SuperheroAsset::register($this);
```

โดย Comment คำสั่งเดิมแล้วเพิ่มคำสั่งใหม่ดังนี้

```
2 //use backend\assets\AppAsset;
3 use kongoon\theme\superhero;
```

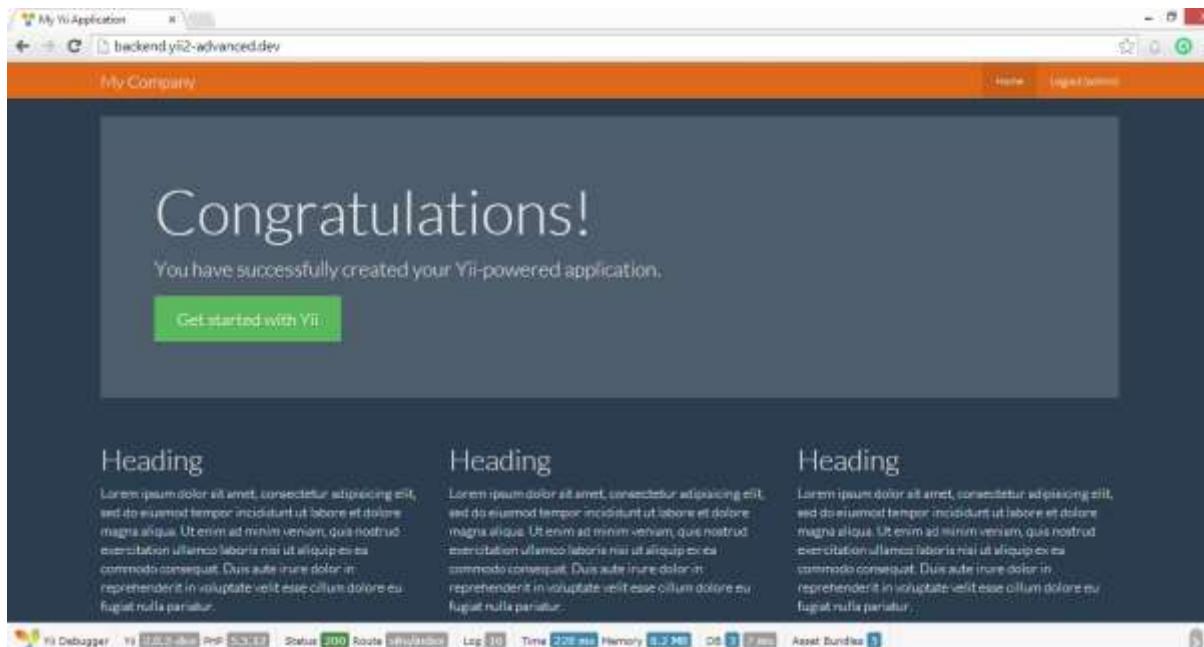
รูปที่ 280 การเพิ่ม Theme ใน Layout

และ

```
12 //AppAsset::register($this);  
13 superhero\SuperheroAsset::register($this);
```

รูปที่ 281 การเรียกใช้งาน Theme ด้วย register asset

จากนั้นทดสอบโดยการ Refresh Web Application จะได้หน้า Web Application ส่วน Backend ดังนี้



รูปที่ 282 หน้า Web Application หลังการติดตั้ง Theme เรียบร้อยแล้ว

## บทที่ 6 การติดตั้ง Extension เสริม

Extension หรือส่วนเสริมเป็นระบบต่างๆ ที่เพิ่มเข้ามาใน Yii2 เพื่อให้มีคุณสมบัติที่เพิ่มมากขึ้น เช่น ระบบกระดานข่าว ระบบซื้อขายสินค้า โดยมีผู้พัฒนาที่นอกเหนือจากทีมพัฒนา Yii2

### Yii2 User

Yii2 User เป็นระบบที่พัฒนาขึ้นเพื่อบริหารจัดการข้อมูล User ทั้งการสมัครสมาชิก ลืมรหัสผ่าน การยืนยันรหัสผ่าน การเข้าสู่ระบบและการแสดงข้อมูลผู้ใช้ ซึ่งง่ายต่อการใช้งานมาก ซึ่งจะอยู่ที่

```
https://github.com/dektrium/yii2-user และ
```

```
http://yii2-user.dmeroff.ru/
```

โดยจะใช้ Yii2 Advanced Application Template ในการติดตั้ง

### วิธีการติดตั้ง

เปิดไฟล์ composer.json และเพิ่มคำสั่งในการติดตั้ง Yii2 User ดังนี้

```
"require": {  
    "dektrium/yii2-user": "*"  
},
```

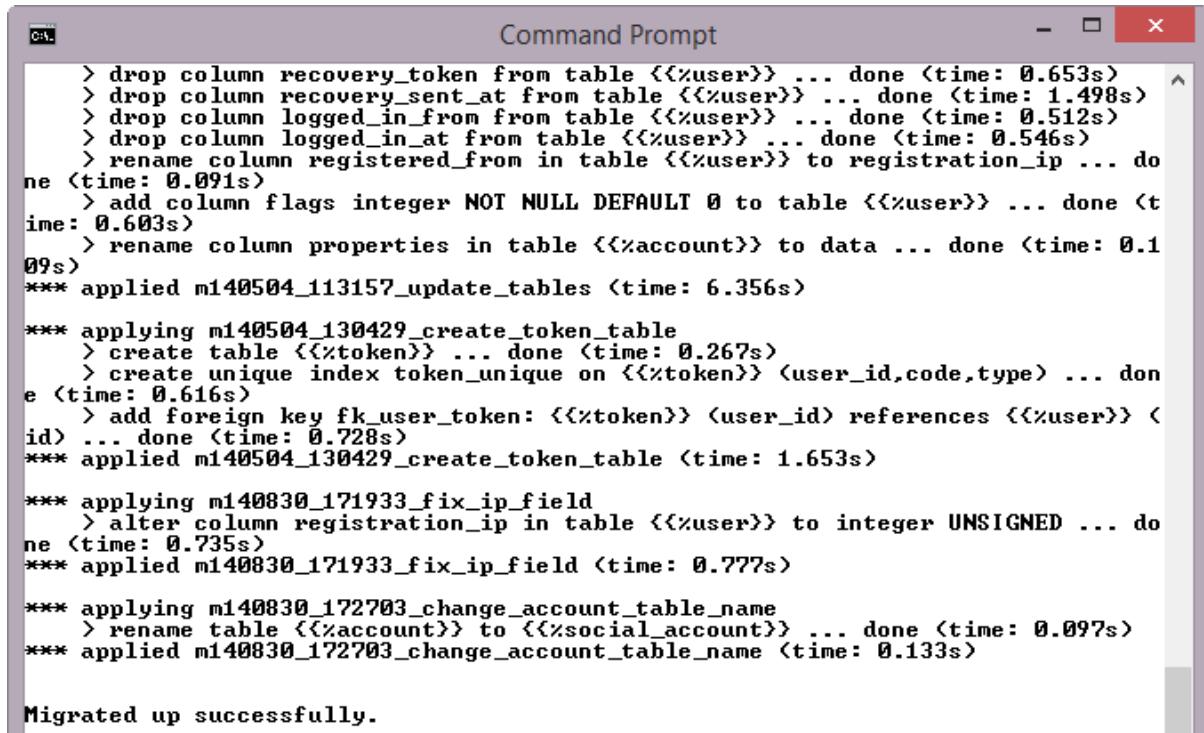
จากนั้นเปิด Command Prompt และเข้าไป Folder Project จากนั้นพิมพ์คำสั่ง เพื่อติดตั้ง Yii2 User ผ่าน Composer

```
composer update
```

จากนั้นติดตั้งตารางฐานข้อมูลซึ่งหากมีตาราง user ในฐานข้อมูลแล้วให้ลบออก และเรียกใช้คำสั่ง

```
yii migrate/up --migrationPath=@vendor/dektrium/yii2-  
user/migrations
```

หลังจากติดตั้งแล้วจะเห็นข้อความแจ้งดังนี้



```
Command Prompt
> drop column recovery_token from table <<%user>> ... done (time: 0.653s)
> drop column recovery_sent_at from table <<%user>> ... done (time: 1.498s)
> drop column logged_in_from from table <<%user>> ... done (time: 0.512s)
> drop column logged_in_at from table <<%user>> ... done (time: 0.546s)
> rename column registered_from in table <<%user>> to registration_ip ... done (time: 0.091s)
> add column flags integer NOT NULL DEFAULT 0 to table <<%user>> ... done (time: 0.603s)
> rename column properties in table <<%account>> to data ... done (time: 0.109s)
*** applied m140504_113157_update_tables (time: 6.356s)

*** applying m140504_130429_create_token_table
> create table <<%token>> ... done (time: 0.267s)
> create unique index token_unique on <<%token>> (user_id,code,type) ... done (time: 0.616s)
> add foreign key fk_user_token: <<%token>> (user_id) references <<%user>> (id) ... done (time: 0.728s)
*** applied m140504_130429_create_token_table (time: 1.653s)

*** applying m140830_171933_fix_ip_field
> alter column registration_ip in table <<%user>> to integer UNSIGNED ... done (time: 0.735s)
*** applied m140830_171933_fix_ip_field (time: 0.777s)

*** applying m140830_172703_change_account_table_name
> rename table <<%account>> to <<%social_account>> ... done (time: 0.097s)
*** applied m140830_172703_change_account_table_name (time: 0.133s)

Migrated up successfully.
```

รูปที่ 283 หน้า Command Prompt การ Migration Yii2 User

เมื่อเปิดดูตารางในฐานข้อมูลจะมีตารางต่างๆ เกิดขึ้นดังนี้

Table	Action	Rows	Type	Collation	Size	Overhead
migration	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	~8	InnoDB	utf8_general_ci	16 KiB	-
profile	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	~0	InnoDB	utf8_general_ci	16 KiB	-
social_account	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	~0	InnoDB	utf8_general_ci	48 KiB	-
token	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	~0	InnoDB	utf8_general_ci	16 KiB	-
user	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	~0	InnoDB	utf8_general_ci	48 KiB	-
5 tables	<a href="#">Sum</a>	8	InnoDB	utf8_general_ci	144 KiB	0 B

รูปที่ 284 ตารางต่างๆ เกิดจากการ Migration

## วิธีการตั้งค่า

แก้ไขไฟล์ config ของ Yii2 User โดยเปิดไฟล์ common/config/main.php และเพิ่มส่วน modules เข้าไปดังนี้

```
<?php
return [
    'vendorPath' => dirname(dirname(__DIR__)) . '/vendor',
    'components' => [
        'cache' => [
            'class' => 'yii\caching\FileCache',
        ],
    ],
    'modules' => [
        'user' => [
            'class' => 'dektrium\user\Module',
            'enableUnconfirmedLogin' => true,
            'confirmWithin' => 21600,
            'cost' => 12,
            'admins' => ['admin']
        ],
    ],
];
];
```

รูปที่ 285 การเพิ่ม Module ในระบบ

จากนั้นเรียก URL เพื่อทำการสมัครสมาชิกเข้าไปในระบบให้เป็น Admin

```
http://localhost/yii2-
project/backend/web/index.php?r=user/registration/register
```

จะปรากฏหน้าจอสำหรับสมัครสมาชิก ให้สมัครสมาชิก username: admin ซึ่งเป็นผู้มีสิทธิสูงสุดตามที่ได้ตั้งค่าในส่วน modules

### Sign up

**Username**

**Email**

**Password**

**Sign up**

[Already registered? Sign in!](#)

รูปที่ 286 การลงทะเบียนผู้ใช้งาน

หลังจากกดปุ่ม Sign up แล้วจะประมวล record ในตาราง user ในฐานข้อมูลดังนี้

id	username	email	password_hash	auth_key	confirmed_at
1	admin	contact@programmerthailand.com	\$2y\$12\$VmCAxahG2\$KnKIE6pDw\$huvxYhgtaf60knEf1IW/2L	ibMcF-ZG4rlu7YZYokMBz0SaYMGvmHa2	NULL

รูปที่ 287 ข้อมูลที่ถูกเพิ่มในตาราง user

ทดสอบเข้า URL เพื่อเข้าบริหารจัดการ User โดยไปที่ URL

```
http://localhost/yii2-
project/backend/web/index.php?r=user/admin
```

จะปรากฏรายการของ User ดังนี้

### Manage users

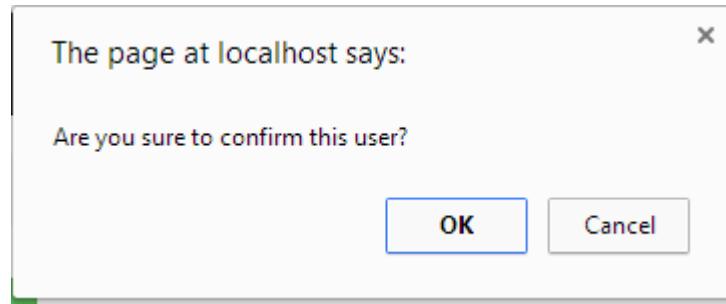
[Create a user account](#)

Username	Email	Registration ip	Registration time	Confirmation	Block status
admin	contact@programmerthailand.com	0.0.0.0	October 08, 2014 03:40	<a href="#">Confirm</a>	<a href="#">Block</a>

รูปที่ 288 การ Confirm ผู้ใช้ admin

ทำการ Confirm User ให้สามารถใช้งานได้ปกติ โดยกดที่ปุ่ม Confirm สีเขียวเพื่อ Confirm user จะประมวล Popup ให้ยืนยัน จากนั้นกดปุ่ม OK เพื่อยืนยันข้อมูล





รูปที่ 289 POPUP สำหรับ Confirm

จะปรากฏข้อความแสดงผลการยืนยันข้อมูลเรียบร้อยแล้ว

User has been confirmed  
Information  
Registered at October 08, 2014 03:40 from 0.0.0.0  
Confirmed at October 08, 2014 03:40

รูปที่ 290 ข้อความแสดงผลการ Confirm

เพียงเท่านี้ก็จะได้ระบบสมาชิกใน Web Application เรียบร้อยแล้ว

## Yii2 Admin

Yii2 Admin เป็น Extension ที่มีไว้สำหรับบริหารจัดการ RBAC โดยมี Feature การทำงานค่อนข้างคล่องคลุ่มเรื่อง RBAC และนอกจากนั้นยังมี GUI (Graphic User Interface) ให้พร้อม โดยเป็นการติดตั้งใน Yii2 Advanced Application Template ซึ่งสามารถดูรายละเอียดเพิ่มเติมได้ที่

<https://github.com/mdmsoft/yii2-admin>

โดยมีวิธีการติดตั้งและวิธีการตั้งค่าดังนี้

### วิธีติดตั้ง

ทำการติดตั้งผ่าน Composer โดยเข้าไปในโฟล์เดอร์ Project พิมพ์คำสั่งดังนี้

`composer require mdmsoft/yii2-admin "dev-master"`

จะได้หน้า Command Prompt ดังนี้

Command Prompt

```
D:\wamp\www\yii2-advanced>composer require mdmsoft/yii2-admin "dev-master"
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
- Installing bower-asset/jquery-ui (1.11.2)
  Loading from cache

- Installing yiisoft/yii2-jui (dev-master ce0f939)
  Cloning ce0f939a7c8745ebf8c995b6adf88d49976f??c8
  Failed to download yiisoft/yii2-jui from source: Failed to clone git@github.com:yiisoft/yii2-jui.git. git was not found, check that it is installed and in your PATH env.

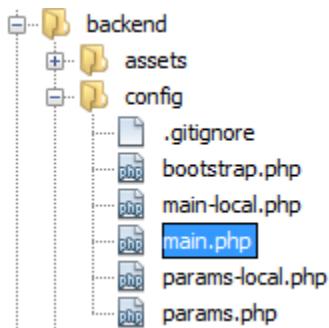
'git' is not recognized as an internal or external command,
operable program or batch file.

  Now trying to download from dist
- Installing yiisoft/yii2-jui (dev-master ce0f939)
```

รูปที่ 291 การติดตั้ง Yii2 Admin

## วิธีตั้งค่า

สำหรับการตั้งค่านี้นับเป็นไฟล์ backend/config/main.php



รูปที่ 292 ตำแหน่งที่ตั้งไฟล์ config/main.php

จากนั้นเพิ่มคำสั่งส่วน Modules

```
'admin' => [
    'class' => 'mdm\admin\Module',
    'layout' => 'left-menu', // default null. other available value 'right-menu' and 'top-menu'
    'controllerMap' => [
        'assignment' => [
            'class' =>
'mdm\admin\controllers\AssignmentController',
            'userClassName' => 'common\models\User',
        ]
    ]
]
```

```
        'idField' => 'id', // id field of model  
User  
    ]  
],  
],
```

จะได้เพิ่มส่วน admin ดังนี้

```
'modules' => [  
    'content' => [  
        'class' => 'backend\modules\content\Content',  
    ],  
    'admin' => [  
        'class' => 'mdm\admin\Module',  
        'layout' => 'left-menu', // default null. other available value  
        'controllerMap' => [  
            'assignment' => [  
                'class' => 'mdm\admin\controllers\AssignmentController',  
                'userClassName' => 'common\models\User',  
                'idField' => 'id', // id field of model User  
            ]  
        ],  
    ],  
],
```

รูปที่ 293 การเพิ่ม Module admin ในระบบ

เพิ่มคำสั่งส่วน Component

```
'authManager' => [  
    'class' => 'yii\rbac\DbManager',  
],
```

จะได้ส่วน Component ดังนี้

```
39 |     'authManager' => [  
40 |         'class' => 'yii\rbac\DbManager',  
41 |     ],
```

รูปที่ 294 การเพิ่ม authManager ในส่วน Component

เพิ่ม Array as access

```
'as access' => [
    'class' => 'mdm\admin\components\AccessControl',
    'allowActions' => [
        'admin/*', // add or remove allowed actions to
this list
    ]
],
```

จะได้ส่วน as access ดังนี้

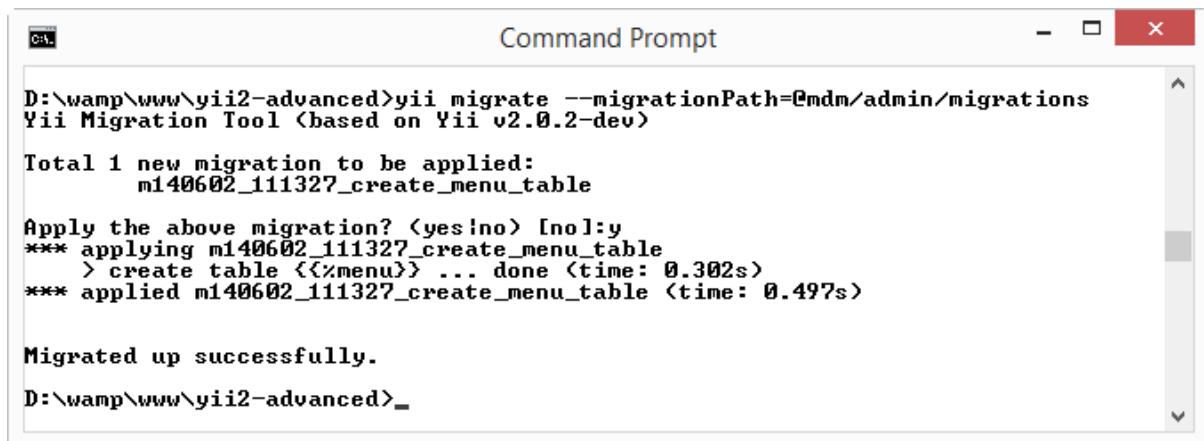
```
55     'as access' => [
56         'class' => 'mdm\admin\components\AccessControl',
57         'allowActions' => [
58             'admin/*', // add or remove allowed actions to this list
59         ]
60     ],
61     'params' => $params,
```

รูปที่ 295 การเพิ่ม as access ในระบบ

ทำการ Migrate เพื่อใช้งาน Menu โดยพิมพ์คำสั่ง

```
yii migrate --migrationPath=@mdm/admin/migrations
```

จะได้หน้า Command Prompt โดยระบบจะให้ยืนยันโดยกดปุ่ม y และกดปุ่ม Enter ดังนี้

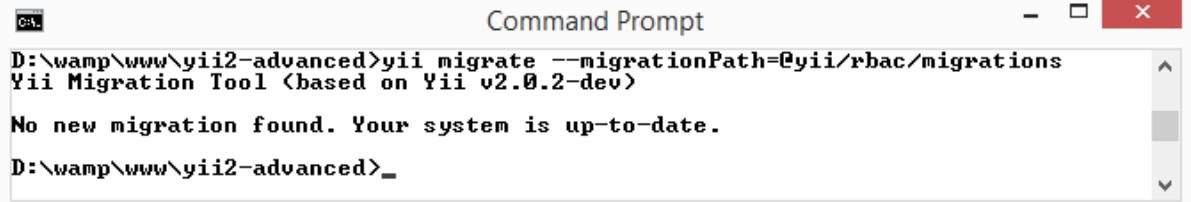


รูปที่ 296 การทำ Migration สำหรับ Yii2 Admin

หากยังไม่ได้สร้างตารางสำหรับ RBAC ให้สร้างโดยพิมพ์คำสั่ง

```
yii migrate --migrationPath=@yii/rbac/migrations
```

จะได้หน้า Command Prompt ดังนี้ (กรณีสร้าง RBAC table แล้ว) หากยังไม่ได้สร้าง ระบบจะให้ยืนยันโดยกดปุ่ม y และกดปุ่ม Enter



```
D:\wamp\www\yii2-advanced>yii migrate --migrationPath=@yii/rbac/migrations
Yii Migration Tool <based on Yii v2.0.2-dev>

No new migration found. Your system is up-to-date.

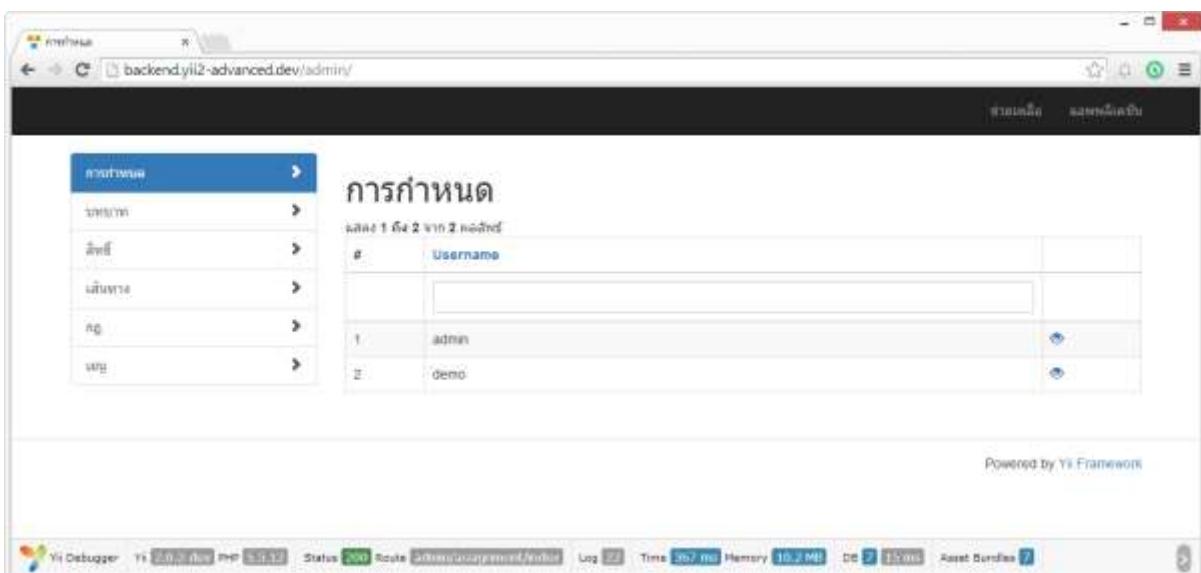
D:\wamp\www\yii2-advanced>_
```

รูปที่ 297 การทำ Migration สำหรับสร้างตารางของ RBAC

เรียก URL

`http://backend.yii2-advanced.dev/admin/`

จะได้หน้า Web Application ดังนี้



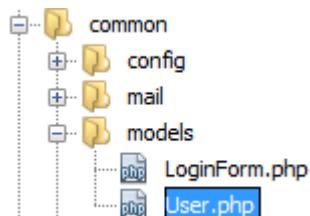
รูปที่ 298 การเรียกหน้า Web Application

## การใช้งาน

สำหรับการใช้งาน Yii2 Admin นั้นจะออกล่าเวย์เบื้องต้นเท่านั้น ทั้งนี้สามารถดูรายละเอียดเพิ่มเติมได้ที่ <https://github.com/mdmsoft/yii2-admin> หรือ <http://www.yiiframework.com/extension/yii2-admin> โดยมีวิธีการใช้งานเบื้องต้นดังนี้

## กำหนดลิทรี

ในขั้นตอนนี้เป็นการกำหนดสิทธิ์ให้แก่ User ว่ามีกี่สิทธิ์โดยกำหนดเป็นค่า Constant ในไฟล์ Model ที่อยู่ใน common/models/User.php



รูปที่ 299 ตำแหน่งไฟล์ User.php

จากนั้นกำหนด Constant ดังนี้

```
25     class User extends ActiveRecord
26 {
27     const STATUS_DELETED = 0;
28     const STATUS_ACTIVE = 10;
29     const ROLE_USER = 1;
30     const ROLE_MANAGER = 5;
31     const ROLE_ADMIN = 10;
32 }
```

รูปที่ 300 การกำหนด Constant

โดยกำหนดค่าแต่ละค่าแทนสิทธิ์แต่ละสิทธิ์ซึ่งค่านี้จะถูกบันทึกในตาราง user ที่ Field role

```
const ROLE_USER = 1;
const ROLE_MANAGER = 5;
const ROLE_ADMIN = 10;
```

เปิดตาราง user ด้วย phpMyAdmin จากนั้นกำหนดสิทธิ์ให้แก่ user ต่างๆ เช่น

ID	Username	Auth Key	Password Hash	Password Reset Token	Email	Role	Status
1	admin	gt2Cjh09PDJD-DlRnhF5Ym9KSxwS-	\$2y\$13\$8eEbB12axoAjZBf7JGKXnay0isng.tak66fQ6t0ix...		contact@programmerthailand.com	10	1
2	demo	61ur5L83uWBT_LWAA_kjIUUVzex9O7YE	\$2y\$13\$5JFUGelomtSV4GH0KAduu3PquWz5jEjeRFf0nRpu...		demo@demo.com	5	1

รูปที่ 301 การกำหนด Role ให้แก่ User

กำหนดให้ User

```
admin มี Role เท่ากับ 10 มีสิทธิ์เป็น Admin
```

demo มี Role เท่ากับ 5 มีสิทธิเป็น Manager

## สร้างกฎ (Rule)

สำหรับ กฎ นั้นเป็น Class ที่ถูกสร้างขึ้นโดยเก็บไว้ใน common/rbac/ ในที่นี้จะสร้างไฟล์ กฎ ในชื่อ UserRoleRule.php โดยเขียนโปรแกรมดังนี้

```
<?php

namespace common\rbac;

use yii\rbac\Rule;
use yii\helpers\ArrayHelper;
use common\models\User;

class UserRoleRule extends Rule{

    public $name = 'userRole';

    public function execute($user, $item, $params) {
        $user = ArrayHelper::getValue($params, 'user',
User::findOne($user));
        if ($user) {
            $role = $user->role;
            if ($item->name === 'admin') {
                return $role == User::ROLE_ADMIN;
            } elseif ($item->name === 'manager') {
                return $role == User::ROLE_ADMIN || $role ==
User::ROLE_MANAGER;
            } elseif ($item->name === 'user') {
                return $role == User::ROLE_ADMIN || $role ==
User::ROLE_MANAGER || $role == User::ROLE_USER;
            }
        }
        return false;
    }
}
```

}

## เลือกเส้นทาง (Routing)

เป็นการเลือกเส้นทางไว้ก่อนเพื่อทำการกำหนดให้กับสิทธิ์ (Permission) และบทบาท (Role) ต่อไป โดย เข้าเมนู เส้นทาง จากนั้นคลิกที่ เส้นทาง (Routes) ที่ต้องการแล้วกดปุ่ม >> เพื่อเลือก



รูปที่ 302 เส้นทาง (Route) ที่ระบบสร้างขึ้น

## สร้างสิทธิ์ (Permission)

สิทธิ์คือส่วนที่เราสามารถเลือกเส้นทางมาใส่แล้วกำหนดให้กับบทบาท (Role) ต่อไปโดยจะสร้างสิทธิ์ต่างๆ โดยเข้าเมนู สิทธิ์ (Permission) ดังนี้

1	AccessDebug	เข้าถึง Debug		
2	adminAdmin	จัดการ RBAC		
3	AdminPost	จัดการโพสสำหรับ Administrator		
4	ManagerPost	สิทธิ์ของ Manager ในการ index,create,view,update โพส		
5	SiteAccess	เข้าถึง site/*		

รูปที่ 303 รายการสิทธิ์ที่สร้างในระบบ

## AccessDebug

มีรายการ เส้นทาง หรือ Routes ต่างๆ ดังนี้

## AccessDebug

บันทึก | ลบ

ชื่อ	AccessDebug
รายละเอียด	เข้าถึง Debug
ชื่อกฎ	(ไม่ได้ตั้ง)
ข้อมูล	(ไม่ได้ตั้ง)

ที่มีอยู่:

ที่กำหนดให้:

**Permission**  
adminAdmin  
AdminPost  
ManagerPost  
SiteAccess

**Routes**  
/\*  
/admin/\*  
/admin/assignment/\*

>>  
<<

**Routes**  
/debug/\*  
/debug/default/\*  
/debug/default/download-mail  
/debug/default/index  
/debug/default/toolbar  
/debug/default/view

รูปที่ 304 ลิทต์ AccessDebug

## adminAdmin

มีรายการ เส้นทาง หรือ Routes ต่างๆ ดังนี้

## adminAdmin

ปั้นปุ่ง    ลบ

ชื่อ	adminAdmin
รายละเอียด	จัดการ RBAC
ข้อคุย	(ไม่ได้ตั้ง)
ข้อมูล	(ไม่ได้ตั้ง)

ที่มีอยู่: [ ]

Permission
AccessDebug
AdminPost
ManagerPost
SiteAccess

Routes
/*
/content/*
/content/default/*
/content/default/index
/content/post/*
/content/post/create
/content/post/delete

>>  
<<

ที่กำหนดให้: [ ]

Routes
/admin/*
/admin/assignment/*
/admin/assignment/assign
/admin/assignment/index
/admin/assignment/role-search
/admin/assignment/view
/admin/default/*
/admin/default/index
/admin/menu/*
/admin/menu/create
/admin/menu/delete
/admin/menu/index

รบกู้ 305 ลิทว์ adminAdmin

## AdminPost

มีรายการ เส้นทาง หรือ Routes ต่างๆ ดังนี้



www.programmerthailand.com

## AdminPost

ปรับปรุง

ลบ

ชื่อ	AdminPost
รายละเอียด	จัดการโพสสำหรับ Administrator
ชื่อคุณ	(ไม่ได้ตั้ง)
ข้อมูล	(ไม่ได้ตั้ง)

ที่มีอยู่:

**Permission**

- AccessDebug
- adminAdmin
- ManagerPost
- SiteAccess

**Routes**

- /\*

ที่กำหนดให้:

**Routes**

- /content/post/\*
- /content/post/create
- /content/post/delete
- /content/post/index
- /content/post/update
- /content/post/view

รูปที่ 306 ลิทธี AdminPost

## ManagerPost

มีรายการ เส้นทาง หรือ Routes ต่างๆ ดังนี้

## ManagerPost

ปรับปรุง

ลบ

ชื่อ	ManagerPost
รายละเอียด	สิทธิ์ของ Manager ในการ index,create,view,update โพส
ชื่อคุณ	(ไม่ได้ตั้ง)
ข้อมูล	(ไม่ได้ตั้ง)

ที่มีอยู่:

- /admin/rule/create
- /admin/rule/delete
- /admin/rule/index
- /admin/rule/update
- /admin/rule/view
- /content/\*
- /content/default/\*
- /content/default/index
- /content/post/\*

ที่กำหนดให้:

**Permission**

- AccessDebug
- SiteAccess

**Routes**

- /content/post/create
- /content/post/index
- /content/post/update
- /content/post/view

รูปที่ 307 ลิทธี ManagerPost



## SiteAccess

มีรายการ เส้นทาง หรือ Routes ต่างๆ ดังนี้

## SiteAccess

The screenshot shows the SiteAccess configuration interface. At the top, there are two buttons: 'ปรับปรุง' (Edit) in blue and 'ลบ' (Delete) in red. Below this is a table with four rows:

ชื่อ	SiteAccess
รายละเอียด	เข้าถึง site/*
ชื่อคุณ	(ไม่ได้ตั้ง)
ข้อมูล	(ไม่ได้ตั้ง)

Below the table are two lists with arrows for moving items between them:

- ที่มีอยู่:** [Permission] AccessDebug, adminAdmin, AdminPost, ManagerPost, Routes /\*
- ที่กำหนดให้:** [Routes] /site/\*, /site/error, /site/index, /site/login, /site/logout

At the bottom center is the text "รูปที่ 308 ลิทธี SiteAccess".

## กำหนดบทบาท (Role)

สำหรับบทบาทนี้จะกำหนดสิทธิ์ให้แก่แต่ละบทบาท โดยบทบาทนี้จะต้องให้สอดคล้องกับ Constant ที่ได้กำหนดไว้ ในที่นี้มี 3 บทบาท คือ user, manager และ admin โดยมีการกำหนดบทบาท ต่างๆ ดังนี้

## บทบาท

The screenshot shows the Role management interface. At the top, there is a green button labeled 'สร้างบทบาท' (Create Role). Below this is a table with three columns:

แสดง 1 ถึง 3 จาก 3 ผลลัพธ์

#	ชื่อ	รายละเอียด
1	admin	ผู้ดูแลระบบ
2	manager	ผู้จัดการ
3	user	(ไม่ได้ตั้ง)

รูปที่ 309 การกำหนดบทบาท

## admin

มีการกำหนดสิทธิ์ให้ดังนี้

## admin

ปัจจุบัน | ลบ

ชื่อ	admin
รายละเอียด	ผู้ดูแลระบบ
ชื่อคุณ	userRole
ข้อมูล	(ไม่ได้ตั้ง)

ที่มีอยู่: [ ]

**Roles**  
user

**Permission**  
ManagerPost

**Routes**  
/\*  
/admin/\*

ที่กำหนดให้: [ ]

**Roles**  
manager

**Permission**  
AccessDebug  
adminAdmin  
AdminPost  
SiteAccess

>>  
<<

รูปที่ 310 การกำหนดสิทธิ์ให้แก่ทบทวน admin

## manager

มีการกำหนดสิทธิ์ให้ดังนี้

## manager

ปัจจุบัน | ลบ

ชื่อ	manager
รายละเอียด	ผู้จัดการ
ชื่อคุณ	userRole
ข้อมูล	(ไม่ได้ตั้ง)

ที่มีอยู่: [ ]

**Roles**  
admin

**Permission**  
adminAdmin  
AdminPost

**Routes**  
/\*

ที่กำหนดให้: [ ]

**Roles**  
user

**Permission**  
AccessDebug  
ManagerPost  
SiteAccess

>>  
<<

รูปที่ 311 การกำหนดสิทธิ์ให้แก่ทบทวน manager

## user

มีการกำหนดสิทธิ์ให้ดังนี้

## user

บันทึก ลบ

ชื่อ	user
รายละเอียด	(ไม่ได้ตั้ง)
ชื่อคุณ	userRole
ข้อมูล	(ไม่ได้ตั้ง)

ที่มีอยู่:

**Roles**  
admin  
manager  
**Permission**  
AccessDebug

ที่กำหนดให้:

**Permission**  
SiteAccess



รูปที่ 312 การกำหนดสิทธิ์ให้แก่บุคลากร user

## บทที่ 7 ระบบจัดการงาน (Task Manager)

ระบบจัดการงานเป็นระบบจัดการงานที่จะทำในโครงการนึงๆ เพื่อให้ทีมงานเห็นภาพรวมงานในโครงการร่วมกัน ทำให้เดินหน้าไปด้วยกันได้

### วิเคราะห์ระบบ

การวิเคราะห์ระบบจะแบ่งการออกแบบตามสิทธิ์ของผู้ใช้งานโดยมี 2 ส่วนคือ ผู้ดูแลระบบและเจ้าหน้าที่

### ผู้ดูแลระบบ

- สามารถจัดการโครงการทั้งหมด
- สามารถจัดการงานทั้งหมด

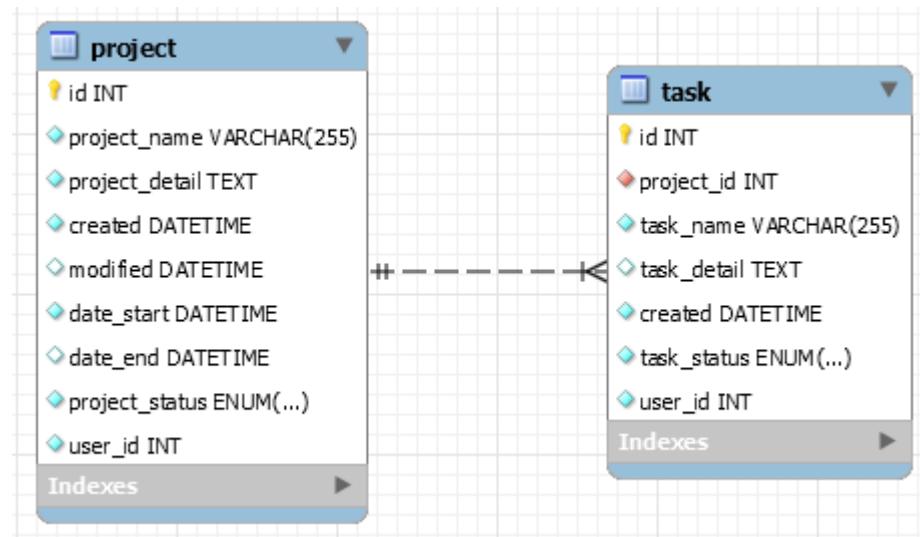
### เจ้าหน้าที่

- สามารถจัดการโครงการของตัวเอง
- สามารถจัดการงานของตัวเอง

### ออกแบบฐานข้อมูล

### สร้าง ER Diagram

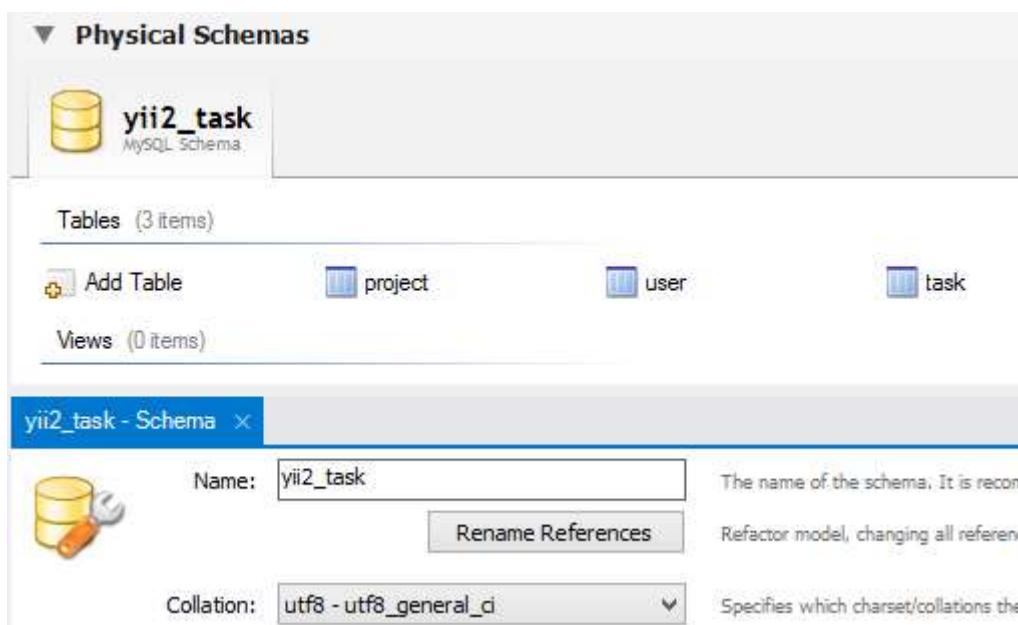
สำหรับการออกแบบฐานข้อมูลจะใช้โปรแกรม MySQL Workbench ในการออกแบบ Entity Relationship Diagram หรือ ER Diagram โดยไม่รวมตารางที่เกิดจากการ Migrate



รูปที่ 313 ER Diagram ระบบจัดการงาน

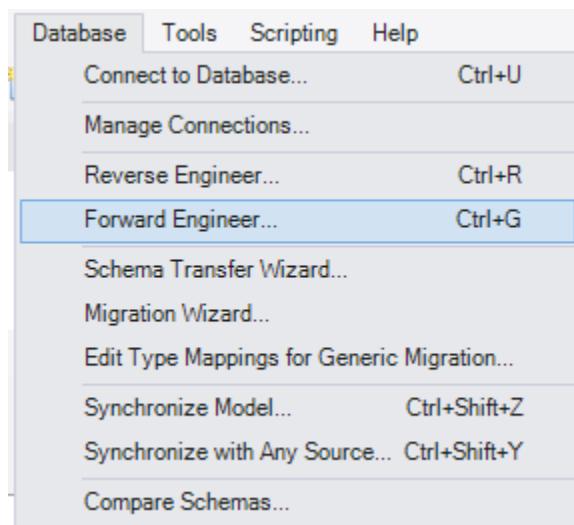
## การ Forward Engineer ไปยังฐานข้อมูล

ก่อน Forward Engineer ให้เปลี่ยนชื่อ Schema หรือชื่อฐานข้อมูล โดยตั้งชื่อเป็น yii2\_task ดังนี้



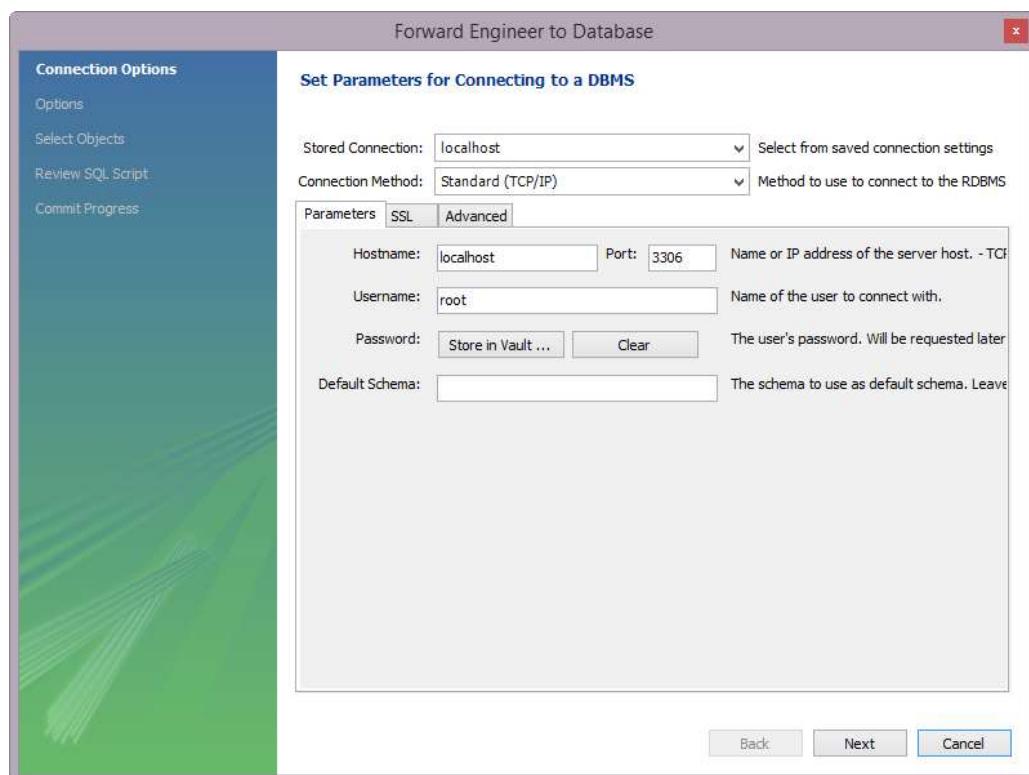
รูปที่ 314 การกำหนดชื่อ Schema

จากนั้นไปที่เมนู Database -> Forward Engineer



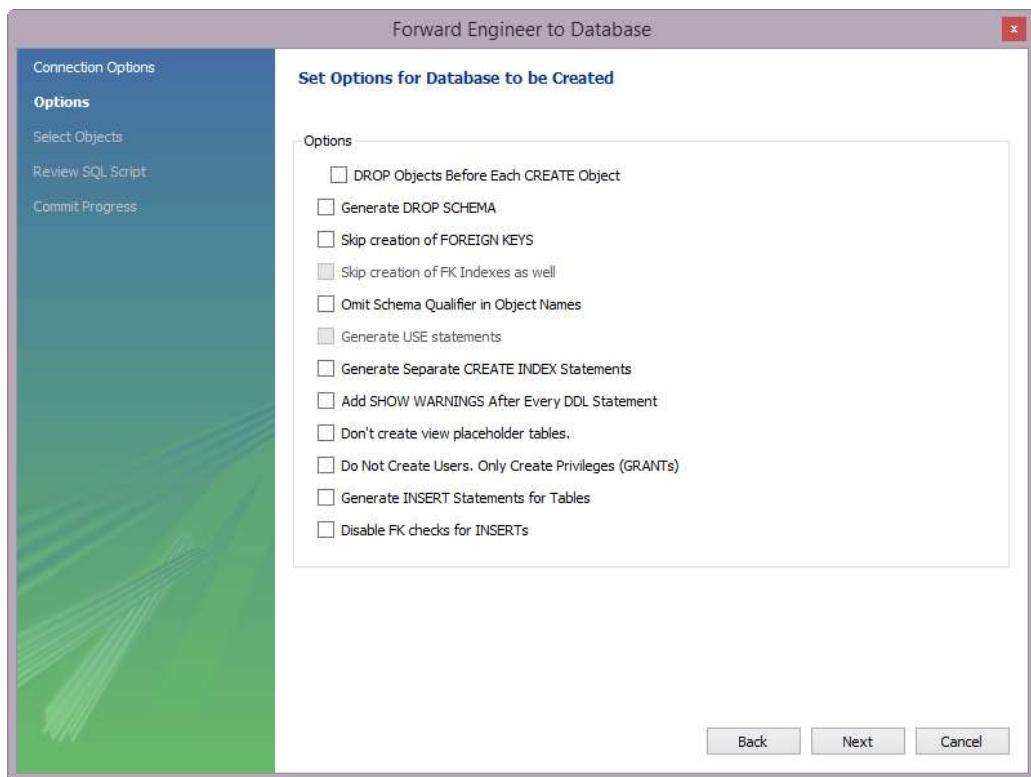
รูปที่ 315 การเลือกเมนู Forward Engineer

จะปรากฏหน้าต่างสำหรับเชื่อมต่อไปยังฐานข้อมูล MySQL ในขั้นตอนนี้กดปุ่ม Next



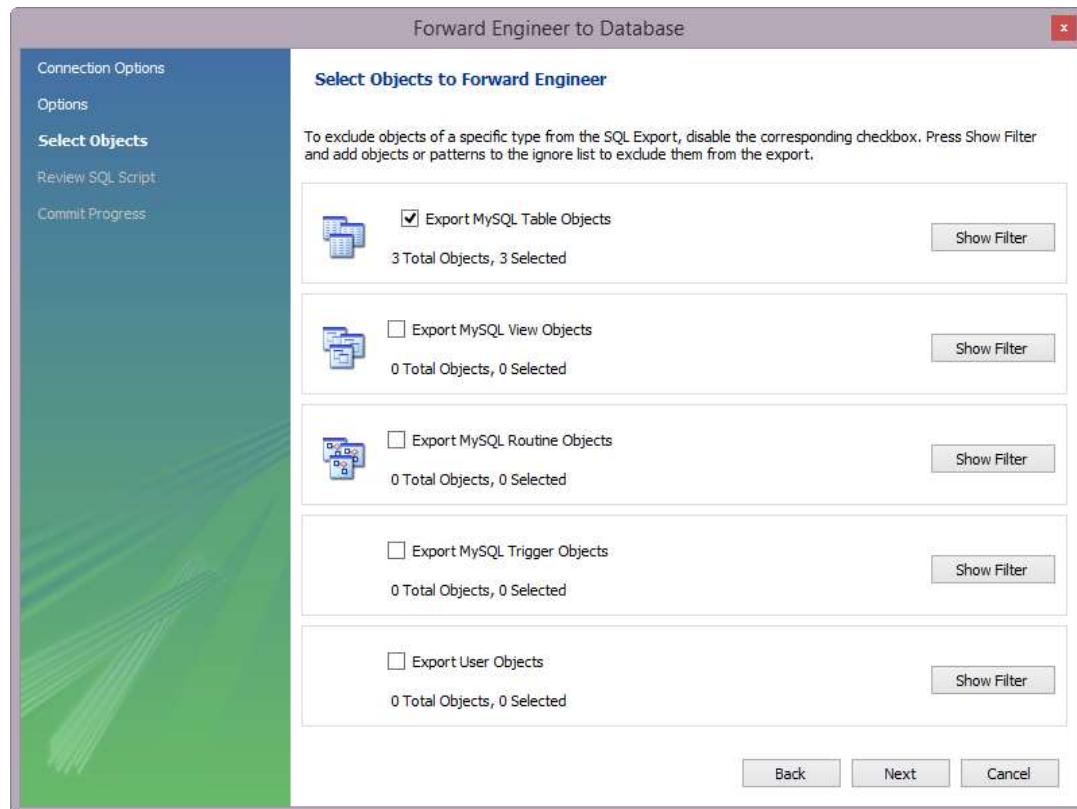
รูปที่ 316 การเลือกการเชื่อมต่อไปยังฐานข้อมูล

จะปรากฏหน้าต่าง Option สำหรับ Forward Engineer ขั้นตอนนี้กดปุ่ม Next อีกครั้ง



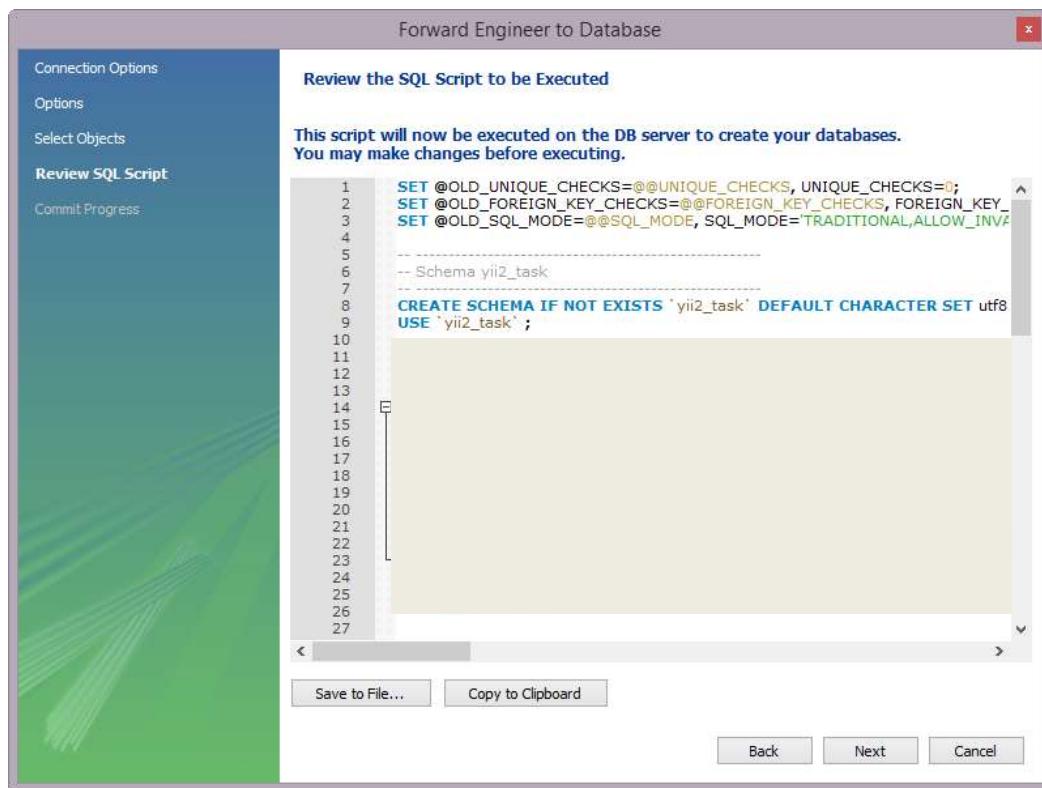
รูปที่ 317 การกำหนดตัวเลือกแล้วไปคลิกปุ่ม Next อีกครั้ง

จะปรากฏหน้าต่างดังตัวเลือก Object ในขั้นตอนนี้กดปุ่ม Next อีกครั้ง



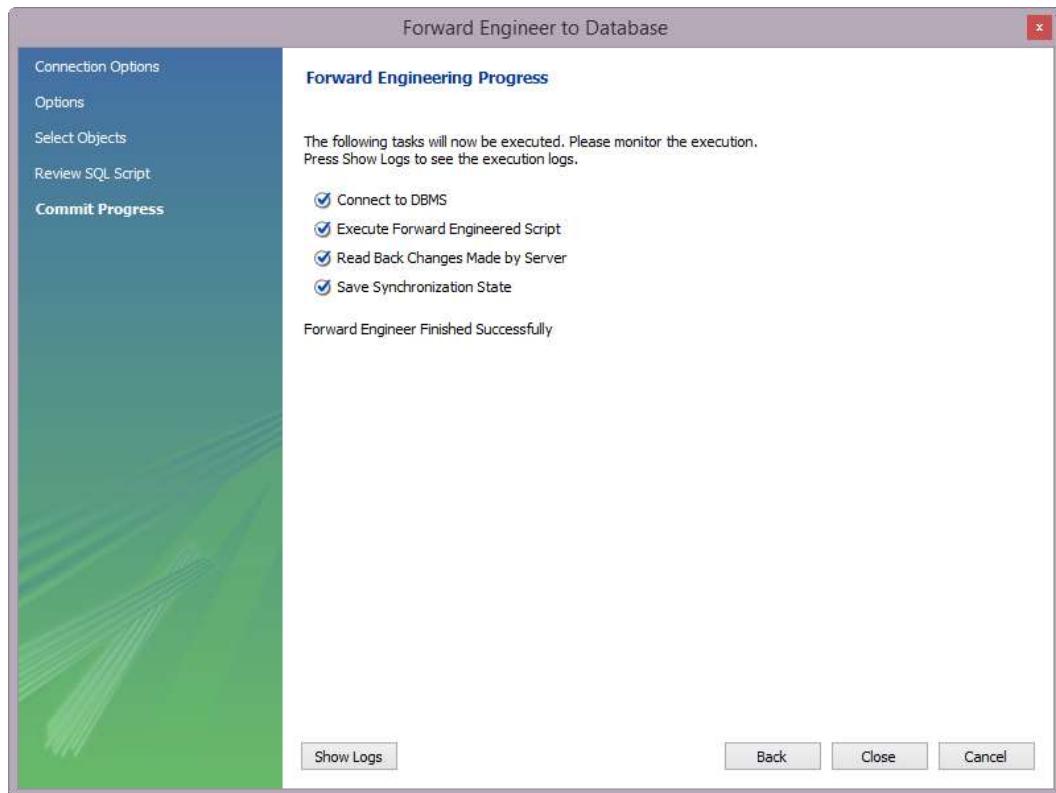
รูปที่ 318 การกรอง Object ที่จะ Forward Engineer

จะปรากฏหน้าต่างแสดงรายละเอียดของคำสั่ง SQL กดปุ่ม Next อีกครั้ง



รูปที่ 319 รายละเอียดคำสั่ง SQL

จะปรากฏหน้าต่างของระบบที่จะประมวลผลคำสั่งที่ได้กำหนดก่อนหน้านี้ จากนั้นกดปุ่ม Close เป็นอันเสร็จสิ้นขั้นตอนการ Forward Engineer



รูปที่ 320 การ Forward Engineer เสร็จเรียบร้อย

ทดลองเปิด phpMyAdmin และเข้าไปที่ฐานข้อมูล yii2\_task จะเห็นตารางต่างๆ ดังนี้

A screenshot of the phpMyAdmin interface. The left sidebar shows databases: 'New', '265\_rubber', '276\_member', 'information\_schema', 'mysql', 'performance\_schema', 'test', 'tmp\_task', 'y02' (which is expanded to show 'y02\_advanced', 'y02\_basic', 'y02\_education', 'y02\_school', 'y02\_task' which is further expanded to 'New', 'project', 'task', 'user'), and 'y02basic', 'y02education'. The main panel shows the 'task' table under the 'project' database. The table has columns: id, name, status, and created\_at. A modal window titled 'Create table' is open, showing fields and a 'Number of columns' input field set to 3. The top navigation bar includes tabs for 'SQL', 'phpMyAdmin', 'Tables', 'Data', 'Structure', 'Operations', 'Collation', 'Indexes', 'Keys', 'Comments', 'Triggers', 'Events', 'Privileges', 'Character Set', and 'Collation'.

รูปที่ 321 ตารางต่างๆ ที่เกิดขึ้น

## พจนานุกรมข้อมูล (Data Dictionary)

พจนานุกรมข้อมูลหรือ Data Dictionary นั้นเป็นส่วนแสดงรายละเอียดของการจัดเก็บข้อมูลใน Field ต่างๆ จากฐานข้อมูล

### ตาราง project

หมายเหตุของตาราง: โครงการ

ส่วน	ชนิด	ว่าง เปล่า (null)	ค่าปริยาย	หมายเหตุ
<b>id</b>	int(11)	ไม่		
<b>project_name</b>	varchar(255)	ไม่		โครงการ
<b>project_detail</b>	text	ไม่		รายละเอียด โครงการ
<b>created</b>	datetime	ไม่		สร้างเมื่อ
<b>modified</b>	datetime	ใช่	NULL	แก้ไขเมื่อ
<b>date_start</b>	datetime	ไม่		วันที่เริ่ม
<b>date_end</b>	datetime	ใช่	NULL	วันที่สิ้นสุด
<b>project_status</b>	enum('เปิด', 'หยุดชั่วคราว', 'ยกเลิก', 'เสร็จ สมบูรณ์')	ไม่	เปิด	
<b>user_id</b>	int(11)	ไม่		ผู้บันทึก

ดังนี้

ชื่อคีร์	ชนิด	เอกลักษณ์	กลุ่ม	ส่วน	เอกเทศ	การ	ว่าง เปล่า (null )	หมายเหตุ
<b>PRIMARY</b>	BTRE E	ใช่	ไม่	<b>id</b>	0	A	ไม่	
<b>fk_project_user1_id</b> <b>x</b>	BTRE E	ไม่	ไม่	<b>user_id</b>	0	A	ไม่	

## ตาราง task

หมายเหตุของตาราง: งาน

ส่วน	ชนิด	ว่างเปล่า (null)	ค่าปริยาย	หมายเหตุ
<b>id</b>	int(11)	ไม่		
<b>project_id</b>	int(11)	ไม่		โครงการ
<b>task_name</b>	varchar(255)	ไม่		งาน
<b>task_detail</b>	text	ใช่	NULL	รายละเอียดงาน
<b>created</b>	datetime	ไม่		สร้างเมื่อ
<b>task_status</b>	enum('เปิด', 'หยุดชั่วขณะ', 'ยกเลิก', 'เสร็จสมบูรณ์')	ไม่	เปิด	สถานะ
<b>user_id</b>	int(11)	ไม่		ผู้บันทึก

ด้วย

ชื่อคีย์	ชนิด	เอกลักษณ์	กลุ่ม	ส่วน	เอกเทศ	การ	ว่างเปล่า (null)	หมายเหตุ
<b>PRIMARY</b>	BTRE E	ใช่	ไม่	id	0	A	ไม่	
<b>fk_task_project_id</b>	BTRE E	ไม่	ไม่	project_id	0	A	ไม่	
<b>fk_task_user1_idx</b>	BTRE E	ไม่	ไม่	user_id	0	A	ไม่	

รูปที่ 322 การแทรกผู้ใช้

## สร้าง Project Skeleton

Project Skeleton เป็นโครงสร้างของ Yii2 ซึ่งจะใช้ Template แบบ Basic เป็นแม่แบบในการสร้างโดย Project นี้จะใช้ชื่อว่า yii2-task และจะใช้ Command Prompt ในการสร้าง โดยเข้าไปที่ Folder web root ในที่นี่คือ d:\wamp\www โดยพิมพ์คำสั่ง



www.programmerthailand.com

d:

แล้วกดปุ่ม Enter จากนั้นพิมพ์

```
cd wamp\www
```

แล้วกดปุ่ม Enter จากนั้นพิมพ์คำสั่งในการสร้าง Project Skeleton ดังนี้

```
composer create-project --prefer-dist yiisoft/yii2-app-advanced yii2-task
```

แล้วกดปุ่ม Enter yii2-task คือชื่อ Project ที่จะสร้างใน www หลังจากกดปุ่ม Enter จะเห็นว่า Composer กำลังติดตั้งส่วนที่เกี่ยวข้องต่างๆ ดังนี้

```
D:\wamp\www>composer create-project --prefer-dist yiisoft/yii2-app-advanced yii2-task
Installing yiisoft/yii2-app-advanced (2.0.1)
- Installing yiisoft/yii2-app-advanced (2.0.1)
  Downloading: connection...      Downloading: 0%
  Downloading: 5%
  Downloading: 15%
  Downloading: 20%
  Downloading: 25%
  Downloading: 30%
  Downloading: 35%
  Downloading: 40%
  Downloading: 50%
  Downloading: 55%
  Downloading: 65%
  Downloading: 70%
  Downloading: 75%
  Downloading: 85%
  Downloading: 90%
  Downloading: 95%
  Downloading: 100%
Created project in yii2-task
Loading composer repositories with package information
```

รูปที่ 323 การติดตั้ง Yii2 Advanced Application Template

เมื่อติดตั้งเสร็จเรียบร้อยจะเห็นเป็นดังนี้

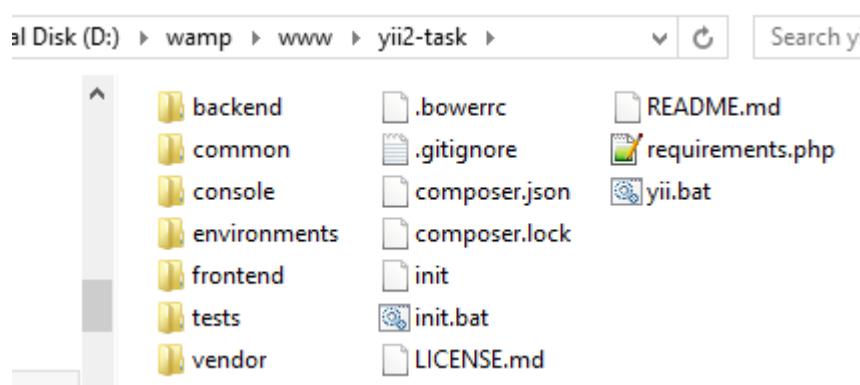


```
Command Prompt
- Installing yiisoft/yii2-bootstrap <2.0.1>
  Loading from cache
- Installing yiisoft/yii2-debug <2.0.1>
  Loading from cache
- Installing bower-asset/typeahead.js <v0.10.5>
  Loading from cache
- Installing phpspec/php-diff <v1.0.2>
  Loading from cache
- Installing yiisoft/yii2-gii <2.0.1>
  Loading from cache
- Installing fzhaninotto/faker <v1.4.0>
  Loading from cache
- Installing yiisoft/yii2-faker <2.0.1>
  Loading from cache

Writing lock file
Generating autoload files
D:\wamp\www>_
```

รูปที่ 324 หลังการติดตั้งเสร็จ

เมื่อเข้าไปดูใน d:\wamp\www\yii2-task จะเห็นรายการไฟล์ต่างๆ ดังนี้

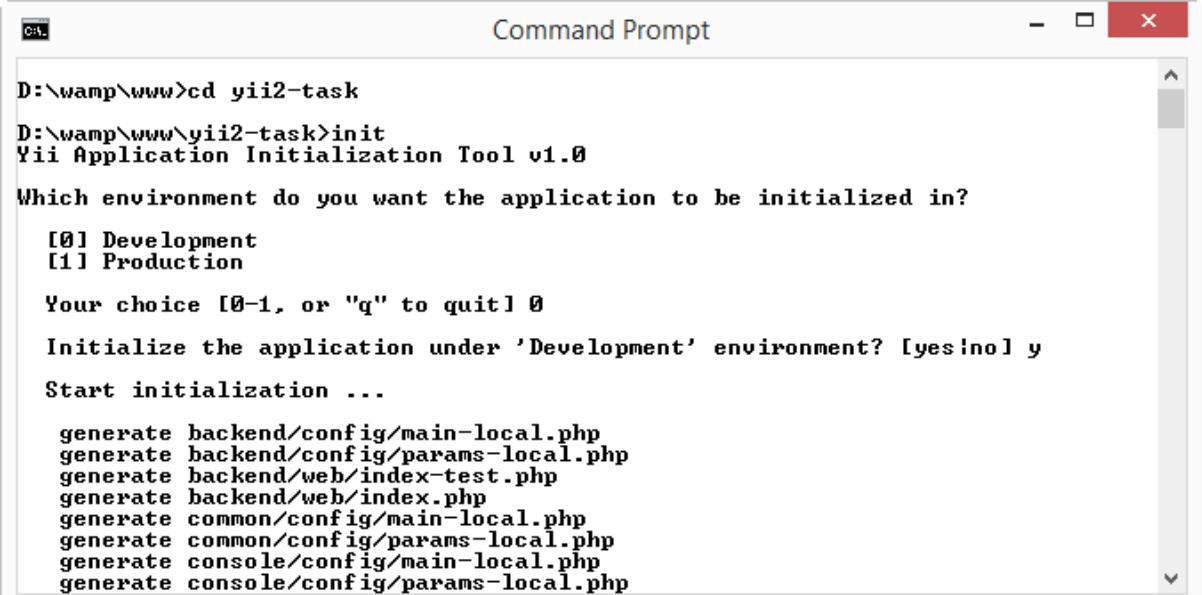


รูปที่ 325 ไฟล์ต่างๆ หลังการติดตั้งเสร็จ

ทำการ init เพื่อเลือกแบบ Developer โดยเข้า Folder Project yii2-task และพิมพ์คำสั่งดังนี้

```
init
```

แล้วทำการกดเลือก 0 และกดปุ่ม Enter จากนั้น Yii2 จะให้ยืนยันอีกครั้งโดยกดปุ่ม y และกดปุ่ม Enter จะเห็นดังนี้



```
D:\wamp\www>cd yii2-task
D:\wamp\www\yii2-task>init
Yii Application Initialization Tool v1.0

Which environment do you want the application to be initialized in?

[0] Development
[1] Production

Your choice [0-1, or "q" to quit] 0

Initialize the application under 'Development' environment? [yes|no] y

Start initialization ...

generate backend/config/main-local.php
generate backend/config/params-local.php
generate backend/web/index-test.php
generate backend/web/index.php
generate common/config/main-local.php
generate common/config/params-local.php
generate console/config/main-local.php
generate console/config/params-local.php
```

รูปที่ 326 การเลือก Development Environment

## ติดตั้ง Extension

### ติดตั้ง Date Picker

Datepicker จะใช้ yii2-date-picker-widger ดูเพิ่มเติมได้ที่ <https://github.com/2amigos/yii2-date-picker-widget>

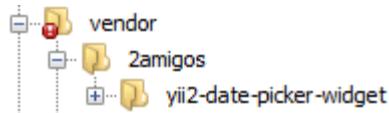
พิมพ์คำสั่งในการติดตั้ง composer require “2amigos/yii2-date-picker-widget” “\*” และกดปุ่ม Enter

Command Prompt

```
D:\wamp\www\yii2-task>composer require "2amigos/yii2-date-picker-widget" "*"
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Reading bower.json of bower-asset/jquery (2.1.3)
Reading bower.json of bower-asset/jquery (2.1.2)
Reading bower.json of bower-asset/jquery (2.1.1)
Reading bower.json of bower-asset/jquery (2.1.0)
Reading bower.json of bower-asset/jquery (2.0.3)
Reading bower.json of bower-asset/jquery (2.0.2)
Reading bower.json of bower-asset/jquery (2.0.1)
Reading bower.json of bower-asset/jquery (2.0.0)
Reading bower.json of bower-asset/jquery (1.11.2)
Reading bower.json of bower-asset/jquery (1.11.1)
Reading bower.json of bower-asset/jquery (1.11.0)
Reading bower.json of bower-asset/jquery (1.10.2)
Reading bower.json of bower-asset/jquery (1.10.1)
Reading bower.json of bower-asset/jquery (1.10.0)
Reading bower.json of bower-asset/jquery (1.9.1)
Reading bower.json of bower-asset/jquery (1.9.0)
Reading bower.json of bower-asset/jquery (1.8.3-patch1)
Reading bower.json of bower-asset/jquery (1.8.3)
Reading bower.json of bower-asset/jquery (1.8.2)
Reading bower.json of bower-asset/jquery (1.8.1)
```

รูปที่ 327 การติดตั้ง DatePicker

จะเห็นโครงสร้างไฟล์ใน vendor ดังนี้



รูปที่ 328 ที่เก็บ DatePicker Widger

## เขียนโปรแกรม

ในส่วนนี้เป็นการเขียนโปรแกรม โดยจะแบ่งการเขียนเป็นแบบ Step by Step เพื่อให้เข้าใจการทำงานแต่ละขั้นตอน

## สร้าง Virtual host

สำหรับการตั้งค่า Virtual host นั้นให้ทำการเปิดไฟล์ httpd-vhosts.conf ที่อยู่ใน

D:\wamp\bin\apache\apache2.4.9\conf\extra จากนั้นเพิ่มคำสั่งในการสร้าง VirtualHost ดังนี้

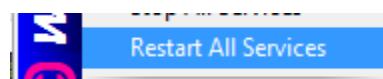
```
<VirtualHost *:80>
    ServerAdmin webmaster@yii2-task.dev
    DocumentRoot "d:/wamp/www/yii2-task/frontend/web"
    ServerName yii2-task.dev
    ErrorLog "logs/yii2-basic.log"
```



```
CustomLog "logs/yii2-basic.log" common
</VirtualHost>

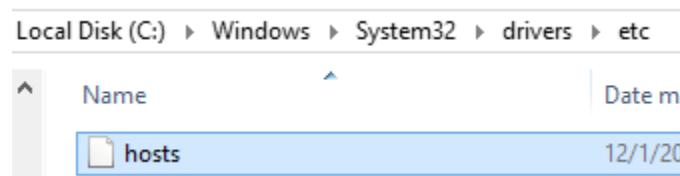
<VirtualHost *:80>
    ServerAdmin webmaster@yii2-task.dev
    DocumentRoot "d:/wamp/www/yii2-task/backend/web"
    ServerName backend.yii2-task.dev
    ErrorLog "logs/yii2-basic.log"
    CustomLog "logs/yii2-basic.log" common
</VirtualHost>
```

จากนั้นทำการ Restart WAMP Server



รูปที่ 329 การ Restart All Services ของ WAMP Server

แล่ทำการเปิดไฟล์ hosts ที่อยู่ใน C:\Windows\System32\drivers\etc

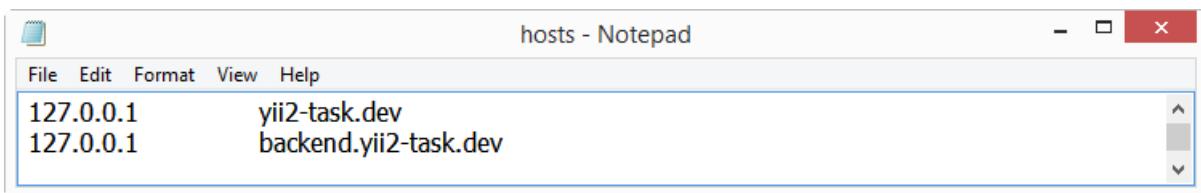


รูปที่ 330 ตำแหน่งไฟล์ hosts

ทำการเปิด Notepad แบบ Administrator โดยค้นหาที่ช่อง Search จากนั้นคลิกขวาเลือก Run as administrator

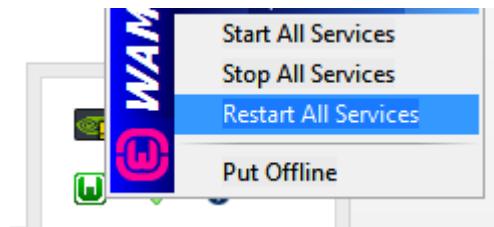


รูปที่ 331 การเปิด Notepad แบบ Administrator



รูปที่ 332 การเพิ่ม yii2-task.dev และ backend.yii2-task.dev ในไฟล์ hosts

ทำการ Restart All Services ใน WAMP โดยคลิก W เลือก Restart All Services



รูปที่ 333 การ Restart All Services

ทดสอบการแสดงผลโดยพิมพ์ yii2-task.dev ที่ URL Address ของ Web Browser



## Task

ระบบจัดการงานในโครงการต่างๆ

ໂຄຮກງານລອດວານ



รูปที่ 334 หน้าหลักของ Yii2-task

## แก้ไขการตั้งค่า Yii2

### การตั้งค่าการเชื่อมต่อฐานข้อมูล

เปิดไฟล์ common/config/main-local.php จากนั้นกำหนดการเชื่อมต่อฐานข้อมูลโดยระบบ dbname, username และ password ในการเชื่อมต่อฐานข้อมูล MySQL

```
2     return [
3         'components' => [
4             'db' => [
5                 'class' => 'yii\db\Connection',
6                 'dsn' => 'mysql:host=localhost;dbname=yii2_task',
7                 'username' => 'root',
8                 'password' => '',
9                 'charset' => 'utf8',
10            ],
11        ],
12    ];
13 
```

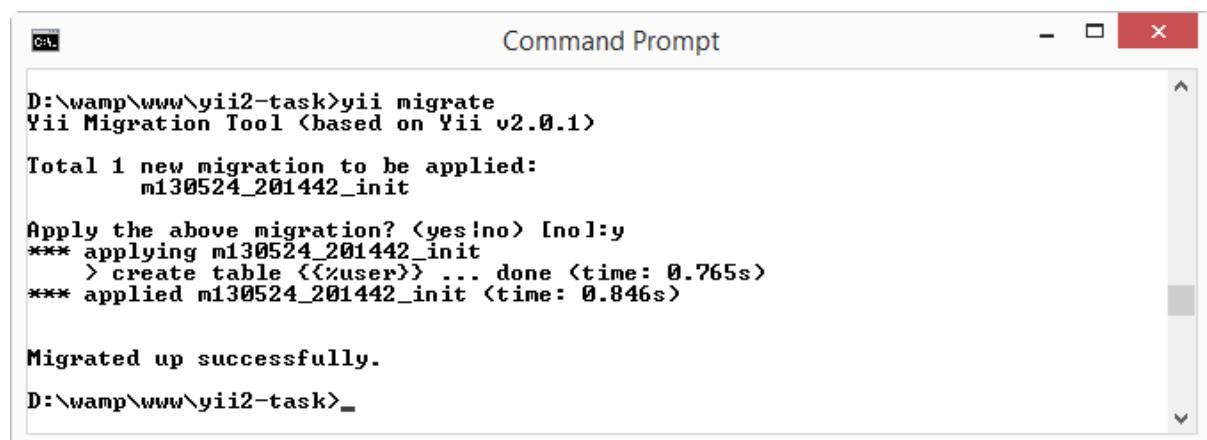
รูปที่ 335 การเชื่อมต่อฐานข้อมูล

### สร้าง Migrate

ทำการสร้าง Migrate โดยเปิด Command Prompt และเข้าไป Folder Project จากนั้นพิมพ์คำสั่งดังนี้

```
yii migrate
```

จะได้ผลลัพธ์ดังนี้



```
D:\wamp\www\yii2-task>yii migrate
Yii Migration Tool (based on Yii v2.0.1)

Total 1 new migration to be applied:
  m130524_201442_init

Apply the above migration? <yes|no> [no]:y
*** applying m130524_201442_init
  > create table `{{%user}}` ... done (time: 0.765s)
*** applied m130524_201442_init (time: 0.846s)

Migrated up successfully.

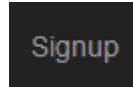
D:\wamp\www\yii2-task>_
```

รูปที่ 336 การใช้คำสั่ง migrate

และจะมีตาราง user เกิดขึ้นในฐานข้อมูล

## เพิ่ม user ในตาราง user

เปิด URL <http://localhost/yii2-task/frontend/web/> จากนั้นกดลิ้งค์ SignUp ที่เมนูหลัก



รูปที่ 337 เมนู Signup

จะพบหน้า Web Application แล้วกรอกข้อมูลดังนี้

# Signup

Please fill out the following fields to signup:

**Username**

admin

**Email**

contact@programmerthailand.com

**Password**

.....

**Signup**

รูปที่ 338 เพิ่ม admin user

Username: admin

Password: 1234567890

แล้วกดปุ่ม Signup จากนั้นระบบจะ Redirect ไปหน้า site/index พร้อมเข้าระบบให้เรียบร้อย

แล้วเพิ่ม user demo เพื่อทดสอบ

# Signup

Please fill out the following fields to signup:

**Username**

demo

**Email**

demo@demo.com

**Password**

.....

**Signup**

รูปที่ 339 เพิ่ม demo user

Username: demo

Password: 1234567890

เมื่อเปิดดูข้อมูลในตาราง user จะปรากฏข้อมูลดังนี้

id	username	auth_key	password_hash	password_reset_token	email	role
1	admin	JduBtIEAc8BvGqQV7A-fkdk1Ky9GBK	\$2y\$13\$5TgAVhox6SFVKMzQVLuIHO/tbyRd6dzbtDvHUpaqGE	NULL	contact@programmerthailand.com	10
2	demo	sFAijyFehSC38CwhzgVJBxBkEoYmY	\$2y\$13\$2g9c36opqjtbuJ5Cw1szVqoiHDYfOHvoE	NULL	demo@demo.com	10

รูปที่ 340 ข้อมูลที่ถูกเพิ่มในตาราง user

## กำหนด Role

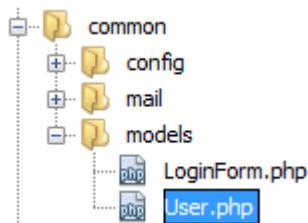
กำหนด Role ใหม่โดยกำหนดให้ admin มี role เป็น 10 และ user มี role เป็น 1 ดังนี้

id	username	auth_key	password_hash	password_reset_token	email	role
1	admin	JduBtIEAc8BvGqQV7A-fkdk1Ky9GBK	\$2y\$13\$5TgAVhox6SFVKMzQVLuIHO/tbyRd6dzbtDvHUpaqGE	NULL	contact@programmerthailand.com	10
2	demo	sFAijyFehSC38CwhzgVJBxBkEoYmY	\$2y\$13\$2g9c36opqjtbuJ5Cw1szVqoiHDYfOHvoE	NULL	demo@demo.com	1

รูปที่ 341 การกำหนด role ให้กับ user

## แก้ไขไฟล์ User Model

เปิดไฟล์ common/models/User.php เพื่อทำการแก้ไข



รูปที่ 342 ตำแหน่งไฟล์ User.php

กำหนด Constant เพิ่มเติมดังนี้

```
29 |     const ROLE_USER = 1;
30 |     const ROLE_ADMIN = 10;
```

รูปที่ 343 กำหนด Constant เพิ่มเติม

กำหนด role เพิ่มเติมในส่วน range

```
['role', 'default', 'value' => self::ROLE_USER],
['role', 'in', 'range' => [self::ROLE_USER, self::ROLE_ADMIN]],
```

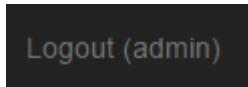
รูปที่ 344 การเพิ่ม range ใน role

กำหนดความสัมพันธ์ ไปยังตาราง Project และตาราง Task ในลักษณะ One to Many

```
48 | /**
49 | * @return \yii\db\ActiveQuery
50 | */
51 | public function getProjects() {
52 |     return $this->hasMany(Project::className(), ['user_id' => 'id']);
53 | }
54 |
55 | /**
56 | * @return \yii\db\ActiveQuery
57 | */
58 | public function getTasks() {
59 |     return $this->hasMany(Task::className(), ['user_id' => 'id']);
60 | }
```

รูปที่ 345 กำหนด Relation

ทดสอบการ Login ถ้าใส่ username และ password ที่ถูกต้องจะแสดงเมนูดังนี้



รูปที่ 346 ลิ้งค์สำหรับออกจากระบบ

หากเข้าระบบไม่ถูกต้องจะแสดงดังนี้

Username  
admin

Password  
\*\*\*\*

Incorrect username or password.

รูปที่ 347 หากเข้าระบบไม่ถูกต้องจะแสดงข้อความแจ้งข้อผิดพลาด

## เพิ่มข้อมูลเข้าในตาราง project และ task

เพิ่มข้อมูลลงในตาราง project

ID	project_name	project_detail	created	modified	date_start	date_end	project_status	user_id
โครงการ	รายละเอียดโครงการ	สร้างเมื่อ	แก้ไขเมื่อ	วันที่เริ่ม	วันที่สิ้นสุด	สถานะ	ผู้บันทึก	
1	โครงการพัฒนา ระบบใน สำนักงาน	โครงการพัฒนา ระบบสำหรับใช้ งานในสำนักงาน	2014-10-16 18:29:33	NULL	2014-10-17 09:00:00	NULL	เปิด	1

รูปที่ 348 แทรกโครงการในตาราง project

เพิ่มข้อมูลลงในตาราง task

ID	project_id	task_name	task_detail	created	task_status	user_id
โครงการ	งาน	รายละเอียดงาน	สร้างเมื่อ	สถานะ	ผู้บันทึก	
1	1	สอนความความต้องการระบบที่ ต้องการใช้งานเร่งด่วน	ติดต่อสอบถามจากเจ้าหน้าที่ฝ่าย ต่างๆ เพื่อความความต...	2014-10-17 10:13:13	เปิด	1

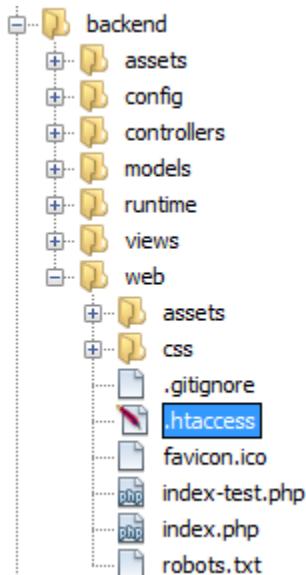
รูปที่ 349 แทรกงานในตาราง task

## เพิ่ม urlManager

สำหรับ urlManager นั้นเป็นการเพิ่มเพื่อให้สามารถสร้าง Link ข้ามไปมาระหว่าง Backend และ Frontend ได้โดยกำหนดดังนี้

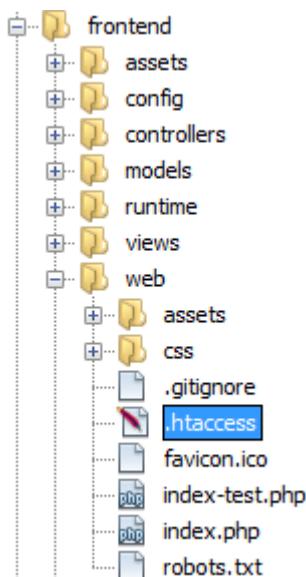
## เพิ่มไฟล์ .htaccess

เพิ่ม .htaccess ใน frontend/web และ backend/web โดยตำแหน่งไฟล์ .htaccess สำหรับ backend จะอยู่ที่



รูปที่ 350 ตำแหน่งไฟล์ .htaccess สำหรับ backend

และตำแหน่งไฟล์ .htaccess สำหรับ frontend อยู่ที่



รูปที่ 351 ตำแหน่งไฟล์ .htaccess สำหรับ frontend

โดยให้ทั้งสองไฟล์มีข้อมูลเหมือนกันดังนี้

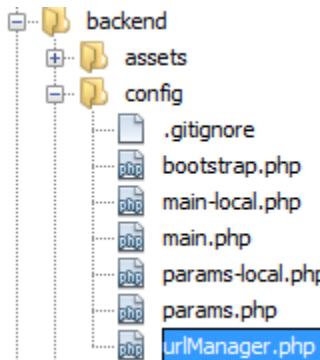
```
1 RewriteEngine on
2 # if a directory or file exists, use
3 RewriteCond %{REQUEST_FILENAME} !-f
4 RewriteCond %{REQUEST_FILENAME} !-d
5
6 # otherwise forward it to index.php
7 RewriteRule . index.php
```

รูปที่ 352 ข้อมูลใน .htaccess

## สร้างไฟล์ urlManager.php

สร้างไฟล์ urlManager.php ใน frontend/config และ backend/config

โดยตำแหน่งไฟล์ urlManager.php สำหรับ backend อยู่ที่



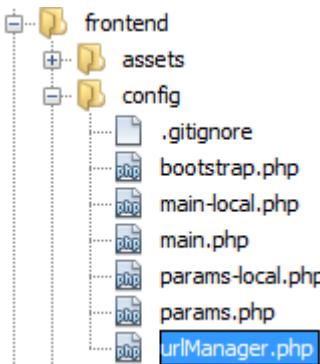
รูปที่ 353 ตำแหน่งไฟล์ urlManager.php สำหรับ backend

โดยใหม่ข้อมูลดังนี้

```
<?php
return [
    'class' => 'yii\web\urlManager',
    'baseUrl' => '//backend.yii2-task.dev',
    'showScriptName' => false,
    'enablePrettyUrl' => true,
    'rules' => [
        ],
];
;
```

รูปที่ 354 ข้อมูลในไฟล์ urlManager.php สำหรับ backend

และตำแหน่งไฟล์สำหรับ frontend จะอยู่ที่



รูปที่ 355 ตำแหน่งไฟล์ urlManager.php สำหรับ frontend

โดยเพิ่มข้อมูลดังนี้

```
<?php
2 return [
3     'class' => 'yii\web\urlManager',
4     'baseUrl' => '//yii2-task.dev',
5     'showScriptName' => false,
6     'enablePrettyUrl' => true,
7     'rules' => [
8
9 ],
10 ];
```

รูปที่ 356 ข้อมูลในไฟล์ urlManager.php สำหรับ frontend

## แก้ไขไฟล์ config สำหรับ frontend

เปิดไฟล์ frontend/config/main.php จากนั้นแก้ไข โดยเพิ่มข้อมูลดังนี้

```
$urlManagerBackend = require(__DIR__ .
'../../backend/config/urlManager.php');

$urlManager = require(__DIR__ . '/urlManager.php');
```

จะได้ Code ดังนี้

```
5     require(__DIR__ . '/params.php'),
6     require(__DIR__ . '/params-local.php')
7 );
8 $urlManagerBackEnd = require(__DIR__ . '/../backend/config/urlManager.php');
9 $urlManager = require(__DIR__ . '/urlManager.php');
10
11 return [
12     'id' => 'app-frontend',

```

รูปที่ 357 การเพิ่มตัวแปรอ้างอิงไปยังไฟล์ urlManager.php

จากนั้นเพิ่ม array ใน components ดังนี้

```
'urlManager' => $urlManager,
'urlManagerBackEnd' => $urlManagerBackEnd,
```

จะได้ Code ดังนี้

```
30     'errorHandler' => [
31         'errorAction' => 'site/error',
32     ],
33     'urlManager' => $urlManager,
34     'urlManagerBackEnd' => $urlManagerBackEnd,
35 ],
36     'params' => $params,
```

รูปที่ 358 การเพิ่ม array ใน components

## แก้ไขไฟล์ config สำหรับ backend

เปิดไฟล์ backend/config/main.php จากนั้นแก้ไข โดยเพิ่มข้อมูลดังนี้

```
$urlManagerBackEnd = require(__DIR__ .
'../../frontend/config/urlManager.php');

$urlManager = require(__DIR__ . '/urlManager.php');
```

จะได้ Code ดังนี้

```
5     require(__DIR__ . '/params.php'),
6     require(__DIR__ . '/params-local.php')
7 );
8
9 $urlManagerFrontEnd = require(__DIR__ . '/../frontend/config/urlManager.php');
10 $urlManager = require(__DIR__ . '/urlManager.php');
11
12 return [
13     'id' => 'app-backend',
14     'basePath' => dirname(__DIR__),

```

รูปที่ 359 การเพิ่มตัวแปรอ้างอิงไปยังไฟล์ urlManager.php

จากนั้นเพิ่ม array ใน components ดังนี้

```
'urlManager' => $urlManager,
'urlManagerFrontEnd' => $urlManagerFrontEnd,
```

จะได้ Code ดังนี้

```
32     'errorHandler' => [
33         'errorAction' => 'site/error',
34     ],
35     'urlManager' => $urlManager,
36     'urlManagerFrontEnd' => $urlManagerFrontEnd,
37 ],
38     'params' => $params,
```

รูปที่ 360 การเพิ่ม array ใน components

## แก้ไขหน้า index

เปิดไฟล์ frontend/views/site/index.php และแก้ไขเพื่อสร้างบุ้มไปยัง backend/views/dashboard/index ดังนี้

```
1 <?php
2 /* @var $this yii\web\View */
3 $this->title = 'Project Task';
4
5 use yii\helpers\Html;
6 ?>
7 <div class="site-index">
8     <div class="jumbotron">
9         <h1>Task</h1>
10
11        <p class="lead">ระบบจัดการงานในโครงการต่างๆ</p>
12        <a href="= Yii::$app-&gt;urlManagerBackend-&gt;baseUrl . '/dashboard/index' ; ?&gt;"&gt;
13            โครงการและงาน
14        &lt;/a&gt;
15    &lt;/div&gt;
16 &lt;/div&gt;
17</pre
```

รูปที่ 361 แก้ไขหน้าแรกของ Web Application

เมื่อแสดงผลจะได้หน้า Web Application ดังนี้



รูปที่ 362 แสดงผลหน้าแรก

## สร้าง Model Task และ Project

สร้าง Model โดยใช้ Gii Generator ไปที่ <http://localhost/yii2-task/web/index.php?r=gii> จากนั้นเข้าเมนู Model Generator จากนั้นใส่ชื่อตารางในที่นี่คือ task และกดปุ่ม Preview

# Model Generator

This generator generates an ActiveRecord class for the specified database table.

## Table Name

task

## Model Class

Task

## Namespace

backend\models

## Base Class

yii\db\ActiveRecord

## Database Connection ID

db

**Use Table Prefix**

**Generate Relations**

**Generate Labels from DB Comments**

รูปที่ 363 การสร้าง Model Task

จะเห็นหน้า Web Application ลักษณะนี้ จากนั้นกด Generate

**Preview** **Generate**

Click on the above **Generate** button to generate the files selected below:

**Create**  **Unchanged**  **Overwrite**

Code File	Action	
D:\wamp\www\yii2-task\backend\models\Task.php	create	<input checked="" type="checkbox"/>

รูปที่ 364 รายการไฟล์หลังกดปุ่ม Preview

จะเห็นหน้าสรุปดังนี้



www.programmerthailand.com

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-task\vendor\yiisoft\yii2-gii\generators\model/default"...
generated D:\wamp\www\yii2-task\backend\models\Task.php
done!
```

รูปที่ 365 ข้อความหลังสร้าง Model Task เสร็จแล้ว

สำหรับตาราง project ก็ทำเช่นเดียวกัน

## Model Generator

This generator generates an ActiveRecord class for the specified database table.

### Table Name

project

### Model Class

Project

### Namespace

backend\models

### Base Class

yii\db\ActiveRecord

### Database Connection ID

db

### Use Table Prefix

### Generate Relations

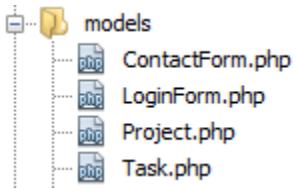
### Generate Labels from DB Comments

รูปที่ 366 สร้าง Model Project

กดปุ่ม Preview และ Generate ตามลำดับ จากนั้นจะเห็น Task.php และ Project.php ใน Folder backend\models ดังนี้



www.programmerthailand.com



รูปที่ 367 รายการ Model ที่ได้สร้างไว้

## แก้ไข Relation ใน Model Task

เปิดไฟล์ backend/models/Task.php และทำการแก้ไขส่วนการเชื่อมโยงไปยังตาราง user ใน Method getUser() ดังนี้

```
public function getUser()
{
    return $this->hasOne(\common\models\User::className(), ['id' => 'user_id']);
}
```

จะได้ Code ลักษณะดังนี้

```
64
65     public function getUser()
66     {
67         return $this->hasOne(\common\models\User::className(), ['id' => 'user_id']);
}
```

รูปที่ 368 การแก้ไขความล้มเหลวใน Model Task

## สร้าง CRUD ด้วย Gii Generator

เปิด URL สำหรับ Gii Generator <http://yii2-task.dev/gii>

## สร้าง CRUD สำหรับ Project

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update) for your data model.

### Model Class

backend\models\Project

### Search Model Class

backend\models\ProjectSearch

### Controller Class

backend\controllers\ProjectController

### View Path

@backend/views/project

รูปที่ 369 สร้าง CRUD สำหรับ Project

### กดปุ่ม Preview

[Preview](#) [Generate](#)

Click on the above [Generate](#) button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\controllers\ProjectController.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\models\ProjectSearch.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\project\_form.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\project\_search.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\project\create.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\project\index.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\project\update.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\project\view.php	create	<input checked="" type="checkbox"/>

รูปที่ 370 ไฟล์ต่างๆ หลังกดปุ่ม Preview

### กดปุ่ม Generate



www.programmerthailand.com

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-task\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated D:\wamp\www\yii2-task\backend\controllers\ProjectController.php
generated D:\wamp\www\yii2-task\backend\models\ProjectSearch.php
generated D:\wamp\www\yii2-task\backend\views\project\_form.php
generated D:\wamp\www\yii2-task\backend\views\project\_search.php
generated D:\wamp\www\yii2-task\backend\views\project\create.php
generated D:\wamp\www\yii2-task\backend\views\project\index.php
generated D:\wamp\www\yii2-task\backend\views\project\update.php
generated D:\wamp\www\yii2-task\backend\views\project\view.php
done!
```

รูปที่ 371 ข้อความหลัง Generate เสร็จแล้ว

ทดลองเข้า URL <http://backend.yii2-task/project> ทำการ login จะพบ Web Application ดังนี้

The screenshot shows a Yii2 application interface. At the top, there is a navigation bar with links for Home, Project, Contact, and Logout (ออก). Below the navigation bar, the title 'My Company' is displayed. Underneath, there is a breadcrumb trail: Home > Projects. The main content area is titled 'Projects'. A green button labeled 'Create Project' is visible. Below it, a table displays one item with the following data:

#	ID	หัวข้อการ	รายละเอียดหัวข้อการ	สถานะ	แก้ไข
1	1	บริษัท จำกัด จำกัด จำกัด	บริษัท จำกัด จำกัด จำกัด จำกัด	2014-10-16 18:20:33	[Edit]



รูปที่ 372 ทดสอบเปิด project

## สร้าง CRUD สำหรับ Task

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update) operations for a data model.

### Model Class

backend\models\Task

### Search Model Class

backend\models\TaskSearch

### Controller Class

backend\controllers\TaskController

### View Path

@backend/views/task

รูปที่ 373 การสร้าง CRUD สำหรับ Task

กดปุ่ม Preview

[Preview](#) [Generate](#)

Click on the above [Generate](#) button to generate the files selected below:

Create  Unchanged  Overwrite

Code File	Action	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\controllers\TaskController.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\models\TaskSearch.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\task\_form.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\task\_search.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\task\create.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\task\index.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\task\update.php	create	<input checked="" type="checkbox"/>
D:\wamp\www\yii2-task\backend\views\task\view.php	create	<input checked="" type="checkbox"/>

รูปที่ 374 รายการไฟล์หลังกดปุ่ม Preview

และ Generate



www.programmerthailand.com

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-task\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated D:\wamp\www\yii2-task\backend\controllers\TaskController.php
generated D:\wamp\www\yii2-task\backend\models\TaskSearch.php
generated D:\wamp\www\yii2-task\backend\views\task\_form.php
generated D:\wamp\www\yii2-task\backend\views\task\_search.php
generated D:\wamp\www\yii2-task\backend\views\task\create.php
generated D:\wamp\www\yii2-task\backend\views\task\index.php
generated D:\wamp\www\yii2-task\backend\views\task\update.php
generated D:\wamp\www\yii2-task\backend\views\task\view.php
done!
```

รูปที่ 375 ข้อความหลังกดปุ่ม Generate

ทดสอบเข้า URL <http://backend.yii2-task.dev/task> ทำการ login จะพบ Web Application ดังนี้

The screenshot shows a Yii2 application interface. At the top, there is a navigation bar with the text 'My Company' and links for 'Home', 'About', 'Contact', and 'Logout (admin)'. Below the navigation bar, the main content area has a breadcrumb trail 'Home > Tasks'. The title 'Tasks' is displayed above a table. A green button labeled 'Create Task' is visible above the table. The table has columns: #, ID, ภารกิจ, สถานะ, วิธีแก้ไขภารกิจ, and สร้างเมื่อ. It contains one row with ID 1, task name 'ทดสอบระบบจัดการภารกิจ', status 'ดำเนินการ', edit link, and creation date '2014-10-17 10:13:13'.



รูปที่ 376 ทดสอบเปิด Task

## แก้ไขหน้าแสดงผล Task

### แก้ไข view index

เปิดไฟล์ backend/views/task/index.php จากนั้นแก้ไขให้แสดงผลในส่วนที่ต้องการดังนี้

```
22      <?= GridView::widget([
23          'dataProvider' => $dataProvider,
24          'filterModel' => $searchModel,
25          'columns' => [
26              ['class' => 'yii\grid\SerialColumn'],
27              // 'id',
28              [
29                  'attribute'=>'project_id',
30                  'value'=>'project.project_name'
31              ],
32              'task_name',
33              // 'task_detail:ntext',
34              'created',
35              'task_status',
36              'user.username',
37
38              ['class' => 'yii\grid\ActionColumn'],
39          ],
40      ]); ?>
```

รูปที่ 377 แก้ไขการแสดงผล

เมื่อทดสอบจะได้รายการดังนี้

The screenshot shows a Yii2 application interface. At the top, there is a dark header bar with the text "My Company" on the left and "Logout (demo)" on the right. Below the header, the URL "หน้าหลัก / Tasks" is visible. The main content area has a title "Tasks". A green button labeled "Create Task" is located at the top left of the table. The table has a header row with columns: #, โครงการ, สถานะ, ผู้จัดมือ, วันที่, และ User Username. There is one data row shown:

#	โครงการ	สถานะ	ผู้จัดมือ	วันที่	User Username
1	โครงการพัฒนาระบบใบอนุญาต	ดำเนินการอยู่	ผู้ดูแลระบบ	2014-10-17 10:13:13	admin

รูปที่ 378 แสดงรายการ Task

แก้ไข view view

เปิดไฟล์ backend/views/task/view.php จากนั้นแก้ไขส่วนแสดงผลตามที่ต้องการดังนี้

ส่วนของ title แก้ไขดังนี้

```
9     $this->title = $model->task_name;
10    $this->params['breadcrumbs'][] = [
    รูปที่ 379 การแก้ไข $this->title
```

จากนั้นแก้ไขส่วน DetailView ดังนี้

```
28    <?= DetailView::widget([
29        'model' => $model,
30        'attributes' => [
31            'id',
32            'project.project_name',
33            'task_name',
34            'task_detail:ntext',
35            'created',
36            'task_status',
37            'user.username',
38        ],
39    ]) ?>
```

รูปที่ 380 การแก้ไขการแสดงผลใน view.php

จะได้การแสดงผลดังนี้

Home / Tasks / สอนความความต้องการระบบที่ต้องการใช้งานเร่งด่วน

## สอนความความต้องการระบบที่ต้องการใช้งานเร่งด่วน

[Update](#) [Delete](#)

ID	1
โครงการ	โครงการพัฒนาระบบในสำนักงาน
งาน	สอนความความต้องการระบบที่ต้องการใช้งานเร่งด่วน
รายละเอียดงาน	ติดต่อสอบถามจากเจ้าหน้าที่ฝ่ายต่างๆ เพื่อความความต้องการพัฒนาระบบที่เร่งด่วนก่อน ให้สามารถใช้งานได้
สร้างเมื่อ	2014-10-17 10:13:13
สถานะ	เปิด
User Username	admin

รูปที่ 381 หน้าแสดงผล view สำหรับ task



www.programmerthailand.com

## แก้ไขฟอร์มเพิ่มข้อมูล

สำหรับฟอร์มเพิ่มข้อมูลนั้นส่วนใหญ่เป็นการแก้ไขการทำ DropDownList จากตารางอื่นๆ ที่มีการเชื่อมโยงข้อมูลกับแบบ One to Many เช่น หมวดหมู่สินค้ากับสินค้า เป็นต้น ในที่นี้จะทำ DropDownList ให้กับ Task โดยดึงข้อมูลมาจาก Project นั้นเอง

### เปิดไฟล์ *views/project/\_form.php*

ฟอร์ม Project นั้นเป็นฟอร์มสำหรับการเพิ่มโครงการโดยเขียนโปรแกรมดังนี้

```
<?php  
use yii\helpers\Html;  
use yii\widgets\ActiveForm;  
use dosamigos\datepicker\DatePicker;  
?  
<div class="project-form">  
    <?php $form = ActiveForm::begin(); ?>  
    <?= $form->field($model, 'project_name') ->  
    >textInput(['maxlength' => 255]) ?>  
    <?= $form->field($model, 'project_detail') ->  
    >textarea(['rows' => 6]) ?>  
    <?= $form->field($model, 'date_start')->widget(  
        DatePicker::className(), [  
            // inline too, not bad  
            'inline' => true,  
            // modify template for custom rendering  
            'template' => '<div class="well well-sm"  
style="background-color: #fff; width:250px">{input}</div>',  
            'clientOptions' => [  
                'autoclose' => true,  
                'format' => 'yyyy-mm-dd'  
            ]  
        ]) ?>
```

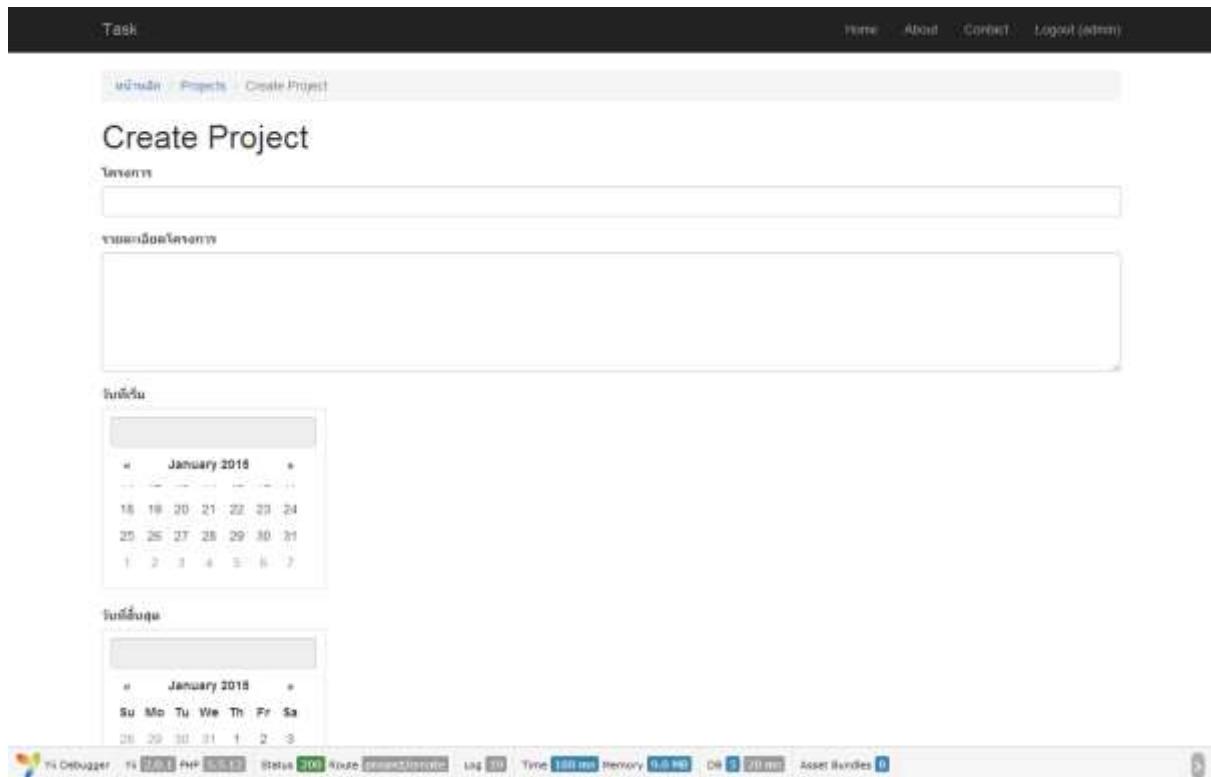
```
<?= $form->field($model, 'date_end')->widget( [DatePicker::className(), [ // inline too, not bad 'inline' => true, // modify template for custom rendering 'template' => '<div class="well well-sm" style="background-color: #fff; width:250px">{input}</div>', 'clientOptions' => [ 'autoclose' => true, 'format' => 'yyyy-mm-dd' ] ]) ?>

<?= $form->field($model, 'project_status')->dropDownList([ 'ปิด' => 'ปิด', 'หยุดชั่วคราว' => 'หยุดชั่วคราว', 'ยกเลิก' => 'ยกเลิก', 'เสร็จสมบูรณ์' => 'เสร็จสมบูรณ์', ], ['prompt' => '']) ?>

<div class="form-group">
    <?= Html::submitButton($model->isNewRecord ? 'Create' : 'Update', ['class' => $model->isNewRecord ? 'btn btn-success' : 'btn btn-primary']) ?>
</div>

<?php ActiveForm::end(); ?>
```

ເນື້ອແສດງຜລຈະໄດ້ລັກໝານນີ້



รูปที่ 382 แบบฟอร์มเพิ่ม Project

## เปิดไฟล์ *views/task/\_form.php*

ฟอร์ม Task นี้เป็นฟอร์มสำหรับการเพิ่มรายการงานต่างๆ โดยต้องมีข้อมูลระบุว่าเป็นโครงการไหน โดยเขียนโปรแกรมดังนี้

```
<?php  
use yii\helpers\Html;  
use yii\widgets\ActiveForm;  
?  
<div class="task-form">  
    <?php $form = ActiveForm::begin(); ?>  
    <?= $form->field($model, 'project_id') ->dropDownList(yii\helpers\ArrayHelper::map(backend\models\Project::find()->all(), 'id', 'project_name')) ;?>  
    <?= $form->field($model, 'task_name') ->textInput(['maxlength' => 255]) ?>
```

```
<?= $form->field($model, 'task_detail')->textarea(['rows' => 6]) ?>

<?= $form->field($model, 'task_status')->dropDownList([
    'เปิด' => 'ปิด', 'หยุดชั่วขณะ' => 'หยุดชั่วขณะ', 'ยกเลิก' => 'ยกเลิก', 'เสร็จสมบูรณ์' => 'เสร็จสมบูรณ์', ], ['prompt' => '']) ?>

<div class="form-group">

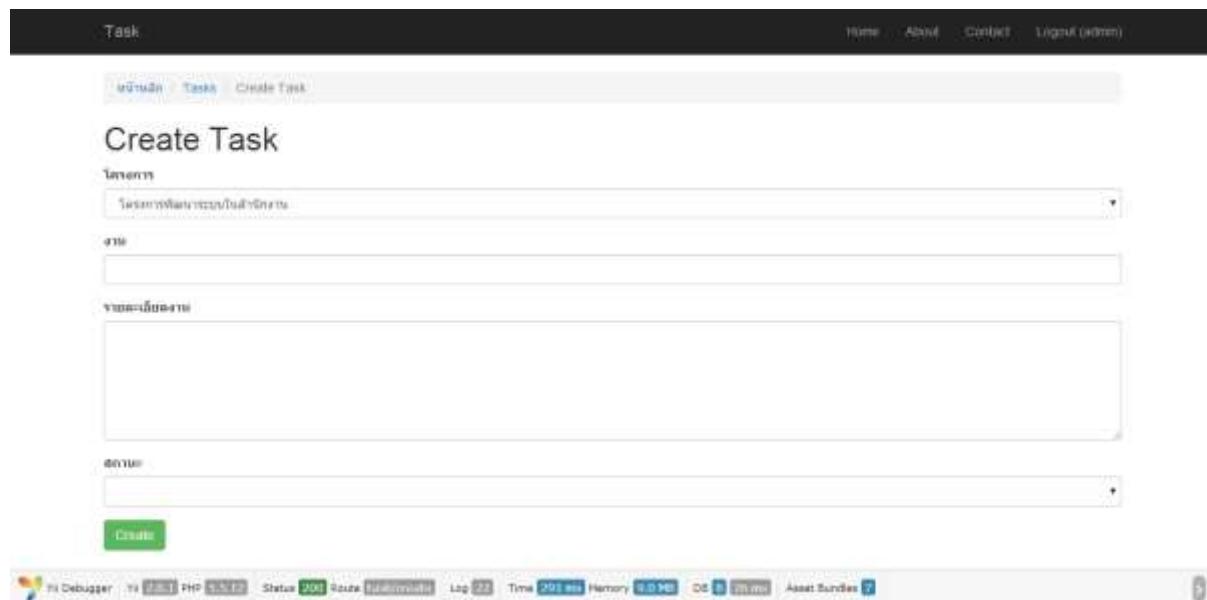
    <?= Html::submitButton($model->isNewRecord ? 'Create' : 'Update', ['class' => $model->isNewRecord ? 'btn btn-success' : 'btn btn-primary']) ?>

</div>

<?php ActiveForm::end(); ?>

</div>
```

เมื่อแสดงผลจะได้ดังนี้



รูปที่ 383 แบบฟอร์มเพิ่ม Task

## เพิ่ม Access Control สำหรับ Controller

ในไฟล์ controllers/ProjectController.php และ controllers/TaskController.php ให้เพิ่ม Access Control ใน Method behaviors() ทั้ง 2 ตัวนี้

```
public function behaviors() {
19    return [
20        'verbs' => [
21            'class' => VerbFilter::className(),
22            'actions' => [
23                'delete' => ['post'],
24            ],
25        ],
26        'access' => [
27            'class' => AccessControl::className(),
28            'only' => ['create', 'update', 'delete'],
29            'rules' => [
30                [
31                    'allow' => false,
32                    'verbs' => ['POST'],
33                ],
34                [
35                    'allow' => true,
36                    'roles' => ['@'],
37                ]
38            ],
39        ],
40    ];
41}
```

รูปที่ 384 เพิ่ม Access Control สำหรับ Controller

## เขียนโปรแกรมหน้า Dashboard

### สร้าง *DashboardController.php*

ไปที่ <http://localhost/yii2-task/web/index.php?r=gii> จากนั้นเข้าเมนู Controller Generator ใส่ Controller ID และ Action IDs ตามภาพ แล้วกด Preview

# Controller Generator

This generator helps you to quickly generate a new controller class, one or several controllers corresponding views.

## Controller ID

dashboard

## Action IDs

index

## Controller Namespace

app\controllers

## Base Class

yii\web\Controller

## Code Template

default (D:\wamp\www\yii2-task\vendor\yiisoft\yii2-gii\generators\controller/default)

Preview

รูปที่ 385 Generate DashboardController.php

จะปรากฏรายการการตั้งภาพ แล้วกด Generate

Preview    Generate

Click on the above <b>Generate</b> button to generate the files selected below:			
Code File	Action	Create	Unchanged
controllers\DashboardController.php	create	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
views\dashboard\index.php	create	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

รูปที่ 386 รายการไฟล์หลังกดปุ่ม Preview

จะปรากฏรายละเอียดดังนี้



www.programmerthailand.com

The controller has been generated successfully. You may [try it now](#).

```
Generating code using template "D:\wamp\www\yii2-task\vendor\yiisoft\yii2-gii\generators\controller/default"...
generated controllers\DashboardController.php
generated views\dashboard\index.php
done!
```

รูปที่ 387 ข้อความหลังกดปุ่ม Generate

เมื่อกด try it now จะเห็นหน้า Web Application ดังนี้

[localhost/yii2-task/web/index.php?r=dashboard%2Findex](http://localhost/yii2-task/web/index.php?r=dashboard%2Findex)

My Company

## dashboard/index

You may change the content of this page by modifying the file <D:\wamp\www\yii2-task\views\dashboard\index.php>.

รูปที่ 388 ทดลองเปิด dashboard/index

## แก้ไข Dashboard

สำหรับหน้า dashboard/index นั้น จะต้องเข้าสู่ระบบจึงจะสามารถเข้าถึงหน้านี้ได้ โดยไปเพิ่ม method behaviors() เพื่อจัดการกับ Access Control ดังนี้



[www.programmerthailand.com](http://www.programmerthailand.com)

```
5  class DashboardController extends \yii\web\Controller {  
6      public function behaviors() {  
7          return [  
8              'access'=>[  
9                  'class' => \yii\filters\AccessControl::className(),  
10                 'only' => ['index'], //กำหนด action ที่จะให้ใช้ Access Control  
11                 'rules' => [  
12                     [  
13                         'allow' => false,  
14                         'verbs' => ['POST'] //ไม่อนุญาตให้มีการโพส  
15                     ],  
16                     [  
17                         'allow'=>true,  
18                         'roles'=>['@'], //ให้เฉพาะผู้ที่เข้าระบบเข้าถึงได้  
19                     ]  
20                 ]  
21             ];  
22         }  
23     }
```

รูปที่ 389 เพิ่ม Access Control สำหรับกำหนดลิทีร์

## เขียนโปรแกรมเรียกข้อมูล

เขียนโปรแกรมเรียกข้อมูลจากตาราง task ทั้งหมดโดยเรียงจาก id มากไปน้อย นั่นคือให้แสดง task ล่าสุด โดยเปิดไฟล์ controllers/DashboardController.php และเพิ่มคำสั่งใน Method actoinIndex() ดังนี้

```
3      namespace app\controllers;  
4      use app\models\Task;  
5      use app\models\Project;
```

รูปที่ 390 การใช้ use เพื่อ Import Class ที่ต้องการ

```
29     public function actionIndex() {
30         $tasks = new ActiveDataProvider([
31             'query'=> Task::find(),
32             'sort'=>[
33                 'defaultOrder'=>[
34                     'id'=>SORT_DESC,
35                 ]
36             ]
37         ]);
38         return $this->render('index', ['tasks'=>$tasks]);
39     }
}
```

รูปที่ 391 การดึงข้อมูลแบบ ActiveDataProvider()

จากนั้นเขียนโปรแกรมแสดงผลแบบ GridView โดยเปิดไฟล์ views/dashboard/index.php

```
1 <?php
2 $this->title = 'รายการงาน';
3 use yii\grid\GridView;
4 ?>
5 <h1>รายการงาน</h1>
6 <?php
7 echo GridView::widget([
8     'dataProvider'=>$tasks,
9     'columns'=>[
10         'id',
11         'project.project_name',
12         'task_name',
13         'created',
14         'task_status',
15         'user.firstname',
16         'user.lastname',
17     ]
18 ]);
```

รูปที่ 392 การเขียนโปรแกรมแสดงผลแบบ Grid

เปิด URL ดู <http://localhost/yii2-task/web/index.php?r=dashboard%2Findex>



The screenshot shows a Yii application interface. At the top, there's a navigation bar with links for Home, About, Contact, and Logout (Logout). Below the navigation, the title "รายการงาน" (Work Log) is displayed. A message "Showing 1-1 of 1 item" is shown above a single-row grid view. The grid has columns for ID, เรื่องงาน (Work Item), สถานะ (Status), ผู้รับผิดชอบ (Responsible Person), วันที่สร้าง (Creation Date), ผู้บันทึก (Recorder), สถานะ (Status), and หมายเหตุ (Notes). The single row contains the following data:

ID	เรื่องงาน	สถานะ	ผู้รับผิดชอบ	วันที่สร้าง	ผู้บันทึก	สถานะ	หมายเหตุ
1	เรียบเรียกตรวจสอบไฟฟ้าบ้าน	ดำเนินการอยู่	นายสมชาย ใจดี	2014-10-17 10:13:13	admin	ดำเนินการ	ตรวจสอบไฟฟ้าบ้าน

รูปที่ 393 ตัวอย่างการแสดงผลแบบ GridView

## สร้าง RBAC

### ตั้งค่า AuthManager

เปิดไฟล์ common/config/main.php เพิ่มการตั้งค่าส่วน components ดังนี้

```
9   'authManager' => [
10  |   'class' => 'yii\rbac\DbManager',
11  | ],
```

รูปที่ 394 การตั้งค่า authManager

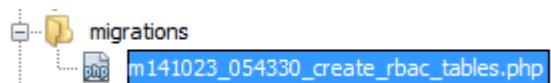
### สร้างไฟล์ migrate create\_rbac\_tables

เปิด command prompt แล้วไปที่ D:\wamp\www\yii2-task จากนั้นพิมพ์คำสั่ง

```
yii migrate/create create_rbac_tables
```

แล้วกดปุ่ม yes แล้วกดปุ่ม Enter

เปิดไฟล์ใน migrations ที่สร้างไว้



รูปที่ 395 การสร้าง migration

แก้ไขคำสั่งในการ Dump ไฟล์ฐานข้อมูลใน method up() ดังนี้

```
public function up()
{
    $this->execute(
        file_get_contents(
            Yii::getAlias('@yii/rbac/migrations/schema-mysql.sql')
        );
}
```

รูปที่ 396 การแก้ไข migration

จากนั้นพิมพ์คำสั่ง migrate เพื่อ update ลงในฐานข้อมูลดังนี้

```
yii migrate
```

จากนั้นกดปุ่ม y และกดปุ่ม Enter

A screenshot of a terminal window titled 'Select Command Prompt'. The window shows the command 'D:\wamp\www\yii2-task>yii migrate' being run. The output is as follows:

```
D:\wamp\www\yii2-task>yii migrate
Yii Migration Tool (based on Yii v2.0.0)

Total 1 new migration to be applied:
  m141023_054330_create_rbac_tables

Apply the above migration? <yes|no> [no]:y
*** applying m141023_054330_create_rbac_tables
  > execute SQL: /**
* Database schema required by \yii\rbac\DbManager.
*
* @author Qiang Xue <qiang.xue@gmail.com>
* @author Alexander Kochetov <creocoder@gmail.com>
* @link http://www.yiiframework.com/
* @copyright 2008 Yii Software LLC
* @license http://www.yiiframework.com/license/
* @since 2.0
*/
drop table if exists `auth_assignment`;
drop table if exists `auth_item_child`;
drop table if exists `auth_item`;
drop table if exists `auth_rule`;
```

รูปที่ 397 การเรียกใช้คำสั่ง yii migrate

เมื่อสร้างเสร็จแล้วจะมีลักษณะดังนี้

Select Command Prompt

```
'parent'          varchar(64) not null,
'child'           varchar(64) not null,
primary key ('parent', 'child'),
foreign key ('parent') references 'auth_item' ('name') on delete cascade on update cascade,
foreign key ('child') references 'auth_item' ('name') on delete cascade on update cascade
) engine InnoDB;

create table 'auth_assignment'
(
  'item_name'        varchar(64) not null,
  'user_id'          varchar(64) not null,
  'created_at'       integer,
primary key ('item_name', 'user_id'),
foreign key ('item_name') references 'auth_item' ('name') on delete cascade on update cascade
) engine InnoDB;
... done (time: 4.320s)
*** applied m141023_054330_create_rbac_tables (time: 4.404s)

Migrated up successfully.

D:\wamp\www\yii2-task>
```

รูปที่ 398 ข้อความเมื่อทำการ migrate เสร็จแล้ว

เมื่อเปิดดูตารางในฐานข้อมูลจะพบว่ามีตารางต่างๆ เกิดขึ้นดังนี้



รูปที่ 399 ตารางต่างๆ ที่เกิดขึ้นในฐานข้อมูล

## สร้าง OwnerRule Class

ทำการสร้างไฟล์ OwnerRule.php ใน common/rbac/OwnerRule.php จากนั้นเขียนโปรแกรมดังนี้

```
<?php

namespace common\rbac;

use yii\rbac\Item;

use yii\rbac\Rule;

class OwnerRule extends Rule

{

    public $name = 'ownItemRule';
```

```
public function execute($user, $item, $params)
{
    $attribute = isset($params['attribute']) ? $params['attribute'] : 'user_id';

    return isset($params['model']) && $user && $user === $params['model']->getAttribute($attribute);
}
```

## สร้าง UserGroupRule Class

สร้าง Class UserGroupRule ใน common/rbac/UserGroupRule.php จากนั้นเขียนโปรแกรมดังนี้

```
<?php

namespace common\rbac;

use common\models\User;
use Yii;
use yii\helpers\ArrayHelper;
use yii\rbac\Rule;

class UserGroupRule extends Rule
{
    public $name = 'userGroup';

    public function execute($user, $item, $params)
    {
        $user = ArrayHelper::getValue($params, 'user',
User::findOne($user));

        if ($user) {
            $role = $user->role;
            if ($item->name === 10) {
                return $role == User::ROLE_ADMIN;
            }
        }
    }
}
```

```
elseif ($item->name === 1) {  
    return $role == User::ROLE_ADMIN || $role ==  
User::ROLE_USER;  
}  
}  
return false;  
}  
}
```

## สร้าง RbacController

สร้าง Class RbacController ใน console/controllers/RbacController.php

```
<?php  
  
namespace console\controllers;  
  
use Yii;  
use yii\console\Controller;  
use common\rbac\OwnerRule;  
  
class RbacController extends Controller{  
    public function actionInit() {  
        $auth = Yii::$app->authManager;  
  
        // เพิ่มสิทธิ์ "createTask"  
        $createTask = $auth->createPermission('createTask');  
        $createTask->description = 'เพิ่ม Task';  
        $auth->add($createTask);  
  
        // เพิ่มสิทธิ์ "updateTask"  
        $updateTask = $auth->createPermission('updateTask');
```



```
$updateTask->description = 'ແກ້ໄຂ Task';
$auth->add($updateTask);

// ເພີ່ມທຳກຳ "owner" ແລະ ກໍານົດໃຫ້ກັບສິຫຼື "createTask"
$owner = $auth->createRole('owner');
$auth->add($owner);
$auth->addChild($owner, $createTask);

// ເພີ່ມທຳກຳ "admin" ແລະ ກໍານົດໃຫ້ກັບສິຫຼື "updateTask"
$admin = $auth->createRole('admin');
$auth->add($admin);
$auth->addChild($admin, $updateTask);
$auth->addChild($admin, $owner);

// ເພີ່ມກົດໝາຍ
$rule = new OwnerRule;
$auth->add($rule);

// ເພີ່ມສິຫຼື "updateOwnTask" ແລະ ເຂົ້າມໂຢງໄປໜັງກົດ
$updateOwnTask = $auth->createPermission('updateOwnTask');

$updateOwnTask->description = 'ແກ້ໄຂ Task ຂອງຕ້ວເອງ';
$updateOwnTask->ruleName = $rule->name;
$auth->add($updateOwnTask);

// "updateOwnTask" ເປັນລູກຂອງ "updatePost"
$auth->addChild($updateOwnTask, $updateTask);

// ຂອມໃຫ້ "owner" ແກ້ໄຂ Task ຂອງຕ້ວເອງ
```

```
$auth->addChild($owner, $updateOwnTask);  
  
// กำหนดค่าให้กับ user id=1 คือ admin id=2 คือ demo  
$auth->assign($owner, 2);  
$auth->assign($admin, 1);  
}
```

จากนั้นเรียก Action init โดยพิมพ์คำสั่งใน Command Prompt ดังนี้

```
yii rbac/init
```

จะเห็นว่ามีແລກต่างๆ เกิดขึ้นในตารางในฐานข้อมูล

Table	Action	Rows	Type	Collation
auth_assignment	Browse Structure Search Insert Empty Drop	~2	InnoDB	utf8_general_ci
auth_item	Browse Structure Search Insert Empty Drop	~5	InnoDB	utf8_general_ci
auth_item_child	Browse Structure Search Insert Empty Drop	~5	InnoDB	utf8_general_ci
auth_rule	Browse Structure Search Insert Empty Drop	~1	InnoDB	utf8_general_ci

รูปที่ 400 ภาพที่เกิดขึ้นหลังการใช้คำสั่ง init

## แก้ไขการ Update

เปิดไฟล์ backend/controllers/TaskController.php จากนั้นเพิ่มโปรแกรมตรวจสอบการเข้าถึงใน actionUpdate() ดังนี้

```
public function actionUpdate($id) {  
  
    $model = $this->findModel($id);  
  
    if (Yii::$app->user->can('updateTask', ['model' => $model])) {  
  
        if ($model->load(Yii::$app->request->post()) &&  
        $model->save()) {  
  
            return $this->redirect(['view', 'id' =>  
            $model->id]);  
        } else {  
  
            return $this->render('update', [  
                'model' => $model,  
                'formName' => 'updateForm'  
            ]);  
        }  
    }  
}
```

```
'model' => $model,  
]);  
}  
} else {  
    throw new \yii\web\ForbiddenHttpException("ไม่สามารถ  
แก้ไขโพสของคนอื่นได้");  
}  
}
```

## ทดสอบ RBAC

ทดสอบการเข้าถึงโดยทดสอบ Permission ในการแก้ไข Post ของตัวเองและแก้ไข Post ของคนอื่น โดยใช้ User 2 ระดับคือ

Username: admin

Password: 1234567890

Username: demo

Password: 1234567890

ทำการเข้า URL สำหรับ Web Application โดยใช้ Username: demo <http://backend.yii2-task.dev/task/index> จะเห็นหน้า Web Application ดังนี้

My Company

home | Logout (demo)

Tasks

Create Task

#	Task Name	Date	Start Date	End Date	User Username	Action
1	Test Task 1	2014-10-17	2014-10-17 10:13:13		admin	
2	Test Task 2	2015-01-11	2015-01-11 06:54:24		admin	
3	Test Task 3	2015-01-11	2015-01-11 07:01:42		demo	



รูปที่ 401 แสดงหน้าแรกของ Task

ทดลองคลิกที่ Icon รูปดินสอเพื่อแก้ไขงานของ admin จะพบว่ามีข้อความแสดงข้อผิดพลาดดังนี้

## Forbidden (#403)

ไม่สามารถแก้ไขงานของคนอื่นได้

The above error occurred while the Web server was processing your request.

Please contact us if you think this is a server error. Thank you.

รูปที่ 402 ข้อความแสดงข้อผิดพลาดเมื่อแก้ไขงานของคนอื่น

แต่ถ้าหากแก้ไขงานของตัวเองจะสามารถแก้ไขได้



รูปที่ 403 สามารถแก้ไขงานของตัวเองได้

## บทสรุป

สำหรับระบบจัดการงานนั้นเป็นการจำลองการเขียนโปรแกรมแบบ Step by Step ซึ่งเป็นแนวทางให้สามารถนำไปประยุกต์และพัฒนาต่อไป โดยมีส่วนสำคัญที่จะต้องทำความเข้าใจ คือ ส่วนของการใช้ Gii Generator ช่วยในการ Generate Code ต่างๆ ทำให้shedดาวและรวดเร็วมากยิ่งขึ้น และส่วนของการทำงานแบบ MVC ที่จะต้องทำความเข้าใจในการส่งค่าต่างๆ ระหว่าง Controller และ View และการทำงานกับข้อมูลผ่าน Model และที่สำคัญที่สุดคือส่วนของ RBAC ซึ่งมีความซับซ้อนที่อาจต้องใช้เวลาในการทำความเข้าใจ

บทเรียนเรื่องระบบจัดการงานนี้คงเป็นแนวทางหนึ่งในการเรียนรู้ Yii Framework 2 และคงทำให้ท่านผู้อ่านได้แนวทางในการพัฒนา Web Application ด้วย Yii Framework 2 ได้อีกแนวทางหนึ่งที่จะช่วยในการพัฒนา Web Application ในระดับองค์กรได้ง่ายดายมากยิ่งขึ้น

## บทที่ 8 ระบบสารสนเทศทางภูมิศาสตร์ (GIS) ด้วย Google Map



### วิเคราะห์ระบบ

การวิเคราะห์ระบบนั้นจะแบ่งตามลักษณะผู้ใช้งานโดยมี User เพียงสิทธิ์เดียวคือ ผู้ดูแลระบบ หรือ Administrator ที่จะสามารถเพิ่ม ลบ แก้ไข ข้อมูลพิกัดตำแหน่งได้ ส่วนบุคคลทั่วไปนั้นสามารถดูแผนที่ได้อย่างเดียวตามเงื่อนไขต่างๆ

### เจ้าหน้าที่

- จัดการข้อมูลโรงเรียน
- จัดการข้อมูลระดับชั้นที่เปิดสอน
- จัดการข้อมูลเครือข่าย
- จัดการข้อมูลประเภทโรงเรียน

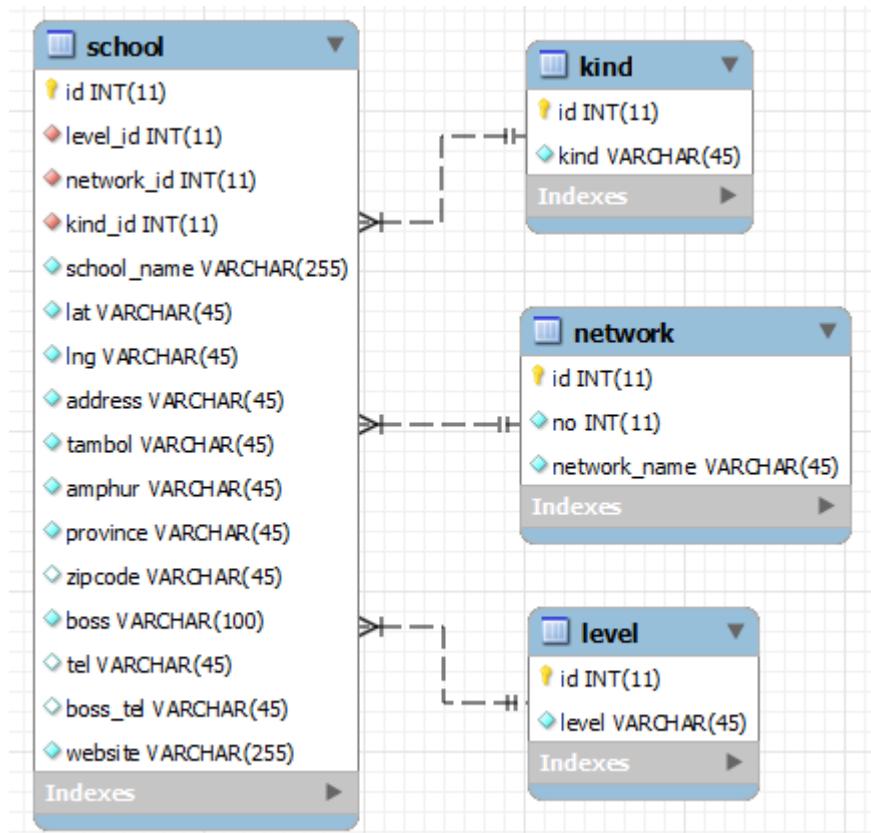
### บุคคลทั่วไป

- ดูแผนที่ตามเงื่อนไข อำเภอ
- ดูแผนที่ตามเงื่อนไข ระดับที่เปิดสอน

## ออกแบบฐานข้อมูล

### ER Diagram

สำหรับ ER Diagram นี้เป็นโครงสร้างของระบบหลัก โดยไม่รวมตารางที่ Yii Framework ได้สร้างให้ในขั้นตอนการ Migrate



รูปที่ 404 ER Diagram

## พจนานุกรมข้อมูล (Data Dictionary)

### kind

หมายเหตุของตาราง: ประเภทโรงเรียน

ส่วน	ชนิด	ว่าง เปล่า (null)	ค่าปริยาย	หมายเหตุ

<b>id</b>	int(11)	ไม่		รหัสประเภทโรงเรียน
<b>kind</b>	varchar(45)	ไม่		ประเภทโรงเรียน

ดังนี้

ชื่อคีย์	ชนิด	เอกสารักษณ์	กลุ่ม	ส่วน	เอกสาร	การ ตรวจสอบ	ว่างเปล่า (null)	หมาย เหตุ
<b>PRIMARY</b>	BTREE	ใช่	ไม่	id	3	A	ไม่	

## level

หมายเหตุของตาราง: ระดับการเปิดสอน

ส่วน	ชนิด	ว่าง เปล่า (null)	ค่าปริยาย	หมายเหตุ
<b>id</b>	int(11)	ไม่		รหัสระดับการเปิดสอน
<b>level</b>	varchar(45)	ไม่		ระดับการเปิดสอน

ดังนี้

ชื่อคีย์	ชนิด	เอกสารักษณ์	กลุ่ม	ส่วน	เอกสาร	การ ตรวจสอบ	ว่างเปล่า (null)	หมาย เหตุ
<b>PRIMARY</b>	BTREE	ใช่	ไม่	id	4	A	ไม่	

## network

หมายเหตุของตาราง: เครือข่าย

ส่วน	ชนิด	ว่าง เปล่า (null)	ค่าปริยาย	หมายเหตุ
<b>id</b>	int(11)	ไม่		รหัสเครือข่าย
<b>no</b>	int(11)	ไม่		หมายเลขเครือข่าย
<b>network_name</b>	varchar(45)	ไม่		ชื่อเครือข่าย

ดังนี้

ชื่อคีย์	ชนิด	เอกสารักษณ์	กลุ่ม	ส่วน	เอกสาร	การ ตรวจสอบ	ว่างเปล่า (null)	หมาย เหตุ



PRIMARY	BTREE	ใช่	ไม่	id	28	A	ไม่	
---------	-------	-----	-----	----	----	---	-----	--

## school

หมายเหตุของตาราง: โรงเรียน

ส่วน	ชนิด	ว่าง เปล่า (null)	ค่าประยุ	หมายเหต
<b>id</b>	int(11)	ไม่		รหัสโรงเรียน
<b>level_id</b>	int(11)	ไม่		ระดับที่เปิดสอน
<b>network_id</b>	int(11)	ไม่		เครือข่าย
<b>kind_id</b>	int(11)	ไม่		ประเภทโรงเรียน
<b>school_name</b>	varchar(255)	ไม่		ชื่อโรงเรียน
<b>lat</b>	varchar(45)	ไม่		ละติจูด
<b>lng</b>	varchar(45)	ไม่		ลองติจูด
<b>address</b>	varchar(45)	ไม่		ที่อยู่
<b>tambol</b>	varchar(45)	ไม่		ตำบล
<b>amphur</b>	varchar(45)	ไม่		อำเภอ
<b>province</b>	varchar(45)	ไม่		จังหวัด
<b>zipcode</b>	varchar(45)	ใช่	NULL	รหัสไปรษณีย์
<b>boss</b>	varchar(100)	ไม่		ผู้อำนวยการ
<b>tel</b>	varchar(45)	ใช่	NULL	เบอร์โทรศัพท์
<b>boss_tel</b>	varchar(45)	ใช่	NULL	เบอร์มือถือผอ.
<b>website</b>	varchar(255)	ไม่		เว็บไซต์

ด้านนี้

ชื่อคีย์	ชนิด	เอกลักษณ์	กลุ่ม	ส่วน	เอกเทศ	การ	ว่าง	หมายเหตุ
PRIMARY	BTRE E	ใช่	ไม่	id	287	A	ไม่	

fk_schools_levels	BTRE E	ไม่	ไม่	level_id	287	A	ไม่	
fk_schools_network s1	BTRE E	ไม่	ไม่	network_i d	143	A	ไม่	
fk_schools_kinds1	BTRE E	ไม่	ไม่	kind_id	143	A	ไม่	

## สร้าง Project Skeleton

เพื่อความเข้าใจมากยิ่งขึ้นสามารถดูเรื่องการติดตั้ง Yii2 เพิ่มเติมก่อนทำขั้นตอนนี้

Project Skeleton เป็นโครงสร้างของ Yii2 ซึ่งจะใช้ Template แบบ Basic เป็นแม่แบบในการสร้าง โดย Project นี้จะใช้ชื่อว่า yii2-gis และจะใช้ Command Prompt ในการสร้าง โดยเข้าไปที่ Folder web root ในที่นี่คือ d:\wamp\www โดยพิมพ์คำสั่ง

```
d:
```

แล้วกดปุ่ม Enter จากนั้นพิมพ์

```
cd wamp\www
```

แล้วกดปุ่ม Enter จากนั้นพิมพ์คำสั่งในการสร้าง Project Skeleton ดังนี้

```
composer create-project --prefer-dist yiisoft/yii2-app-basic  
yii2-gis
```

แล้วกดปุ่ม Enter yii2-gis คือชื่อ Project ที่จะสร้างใน www หลังจากกดปุ่ม Enter จะเห็นว่า Composer กำลังติดตั้งส่วนที่เกี่ยวข้องต่างๆ ดังนี้



```
D:\wamp\www>composer create-project --prefer-dist yiisoft/yii2-app-basic yi2-gis
Installing yiisoft/yii2-app-basic (2.0.0)
- Installing yiisoft/yii2-app-basic (2.0.0)
  Loading from cache

Created project in yi2-gis
Loading composer repositories with package information
Installing dependencies (including require-dev)
- Installing yiisoft/yii2-composer (2.0.0)
  Loading from cache
- Installing swiftmailer/swiftmailer (v5.3.0)
  Loading from cache
- Installing bower-asset/jquery (2.1.1)
  Loading from cache
- Installing bower-asset/yii2-pjax (v2.0.1)
  Loading from cache
- Installing bower-asset/punycode (v1.3.1)
  Loading from cache
- Installing bower-asset/jquery.inputmask (3.1.32)
```

รูปที่ 405 การสร้าง Project

เมื่อติดตั้งเสร็จเรียบร้อยจะเห็นเป็นดังนี้

```
- Installing yiisoft/yii2-debug (2.0.0)
  Loading from cache

- Installing bower-asset/typeahead.js (v0.10.5)
  Loading from cache

- Installing phpspec/php-diff (v1.0.2)
  Loading from cache

- Installing yiisoft/yii2-gii (2.0.0)
  Loading from cache

- Installing fzhaninotto/faker (v1.4.0)
  Loading from cache

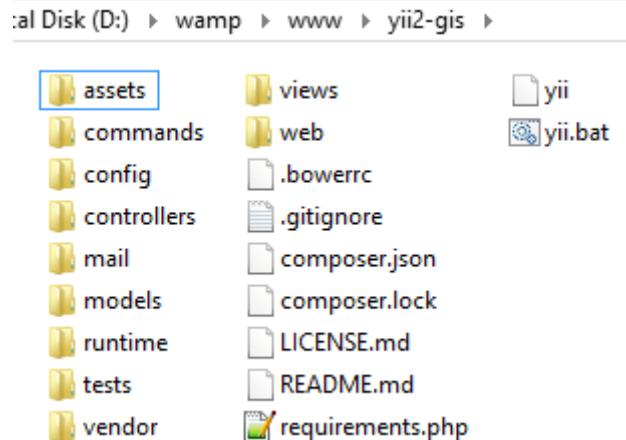
- Installing yiisoft/yii2-faker (2.0.0)
  Loading from cache

Writing lock file
Generating autoload files
chmod('runtime', 0777)...done.
chmod('web/assets', 0777)...done.
chmod('yii', 0755)...done.

D:\wamp\www>
```

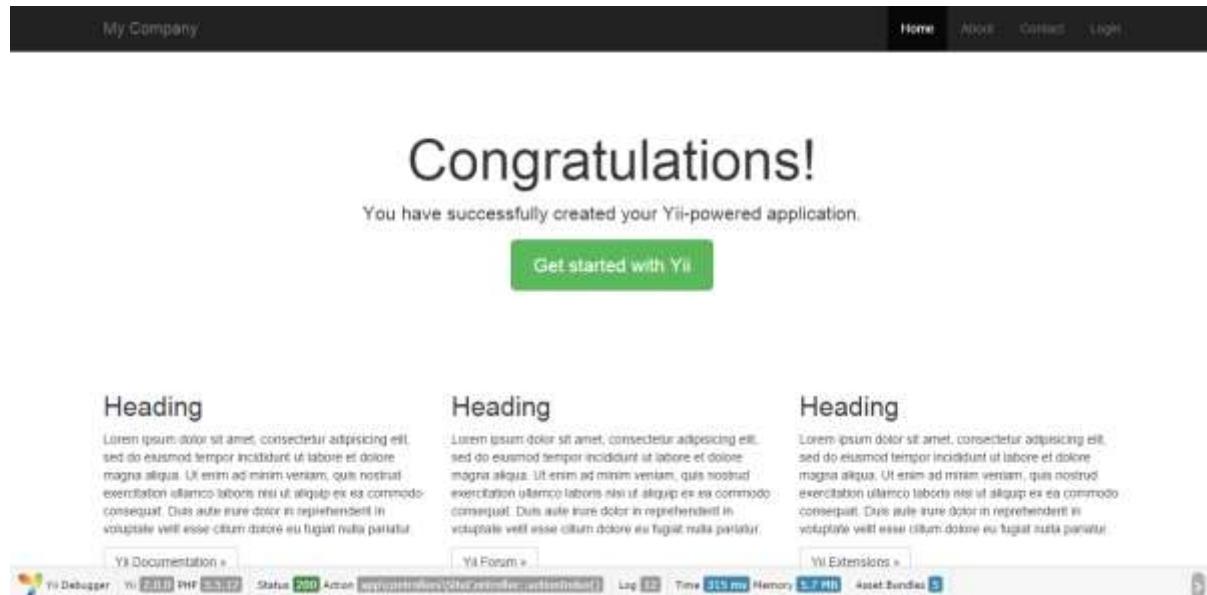
รูปที่ 406 หลังจากสร้าง Project เสร็จแล้ว

เมื่อเข้าไปดูใน d:\wamp\www\yii2-gis จะเห็นรายการไฟล์ต่างๆ ดังนี้



รูปที่ 407 รายการไฟล์ต่างๆ ใน Project

ทดสอบเรียก URL เพื่อเปิดดูหน้า Web Application โดยเปิด URL <http://localhost/yii2-gis/web> จะได้หน้า Web Application ดังนี้



รูปที่ 408 หน้าเริ่มต้นของ Project

## ติดตั้งส่วนเสริม

สำหรับส่วนเสริมเป็นส่วนที่จะติดตั้งเพิ่มเติมเข้าไปใน Web Application ที่สร้างขึ้นโดยจะติดตั้งส่วนเสริมชื่อ yii2-user สามารถดูรายละเอียดเพิ่มเติมได้ที่ <https://github.com/dektrium/yii2-user>

## Yii2 User

เปิดไฟล์ composer.json และเพิ่มคำสั่งด้านล่างในส่วน require

```
"dektrium/yii2-user": "*"
```

ดังภาพด้านล่าง

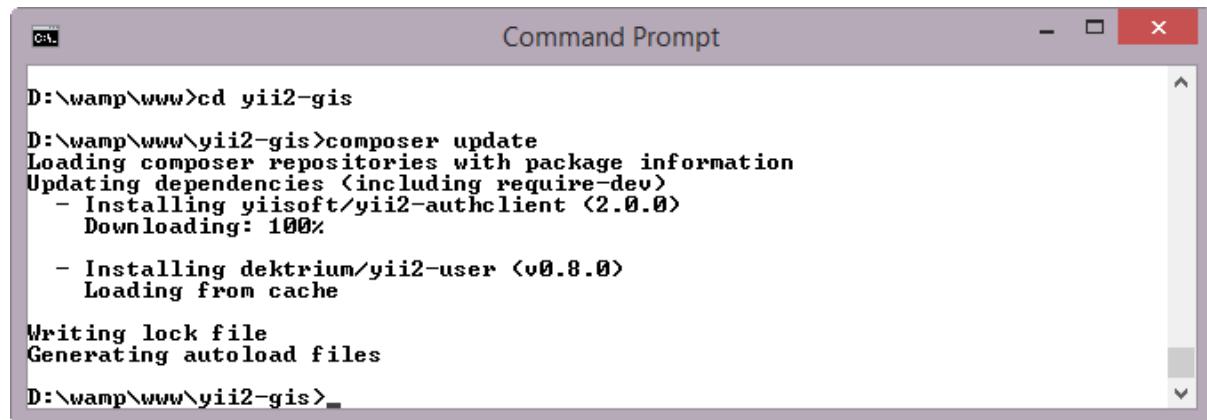
```
16     "require": {  
17         "php": ">=5.4.0",  
18         "yiisoft/yii2": "*",  
19         "yiisoft/yii2-bootstrap": "*",  
20         "yiisoft/yii2-swiftmailer": "*",  
21         "dektrium/yii2-user": "*"  
22     },
```

รูปที่ 409 การเพิ่มคำสั่งใน composer.json

จากนั้นพิมพ์คำสั่ง

```
composer update
```

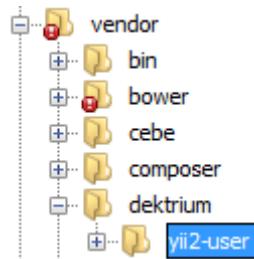
แล้วกดปุ่ม Enter รอสักครู่จะปรากฏการติดตั้ง yii2-authclient และ yii2-user ดังนี้



```
D:\wamp\www>cd yii2-gis  
D:\wamp\www\yii2-gis>composer update  
Loading composer repositories with package information  
Updating dependencies (including require-dev)  
- Installing yiisoft/yii2-authclient (2.0.0)  
  Downloading: 100%  
  
- Installing dektrium/yii2-user (v0.8.0)  
  Loading from cache  
  
Writing lock file  
Generating autoload files  
D:\wamp\www\yii2-gis>_
```

รูปที่ 410 การใช้คำสั่ง composer update

เมื่อเปิดดูโครงสร้างไฟล์จะเห็น Folder ใน vendor ดังนี้



รูปที่ 411 yii2-user ถูกติดตั้งเรียบร้อยแล้ว

ตั้งค่าใน config/web.php โดยเพิ่มส่วนของ modules ดังนี้



```
8     'bootstrap' => ['log'],
9     'modules' => [
10         'user' => [
11             'class' => 'dektrium\user\Module',
12             'enableUnconfirmedLogin' => true,
13             'enableRegistration' => false, //ปิดการลงทะเบียน
14             'confirmWithin' => 21600,
15             'cost' => 12,
16             'admins' => ['admin']
17         ],
18     ],
19     'components' => [
20         'request' => [
21     ]]
```

รูปที่ 412 การตั้งค่า module

## Yii2 Google Map Library

สามารถเพิ่มเติมได้ที่ <http://www.yiiframework.com/extension/yii2-google-maps-library/>

เปิดไฟล์ composer.json เพิ่มคำสั่ง

```
"2amigos/yii2-google-maps-library" : "*"
```

ดังนี้

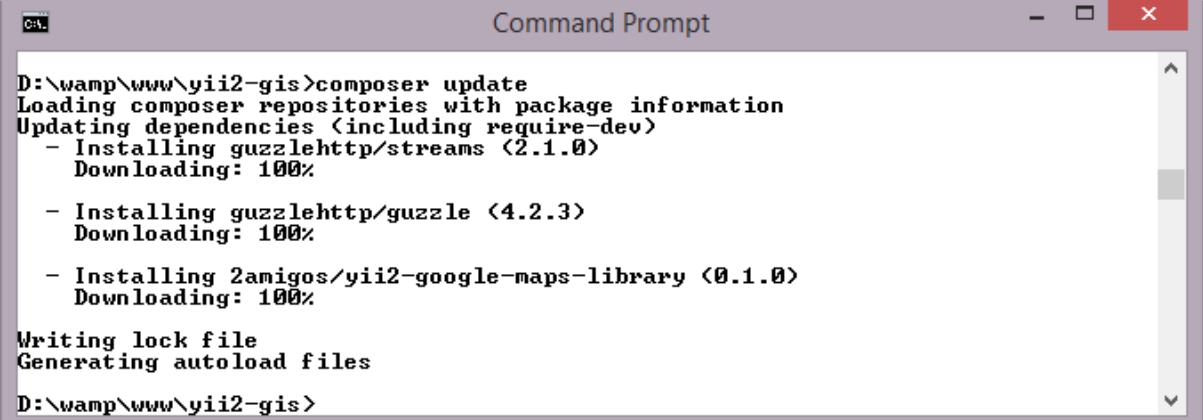
```
16     "require": {
17         "php": ">=5.4.0",
18         "yiisoft/yii2": "*",
19         "yiisoft/yii2-bootstrap": "*",
20         "yiisoft/yii2-swiftmailer": "*",
21         "dektrium/yii2-user": "*",
22         "2amigos/yii2-google-maps-library" : "*"
23     },
```

รูปที่ 413 การเพิ่มคำสั่งใน composer.json

จากนั้นเข้าไปใน d:\wamp\www\yii2-gis และพิมพ์คำสั่ง

```
composer update
```

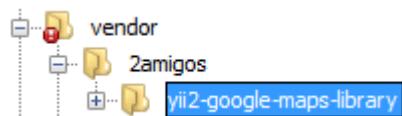
แล้วกดปุ่ม Enter จะได้การแสดงผลดังนี้



```
D:\wamp\www\yii2-gis>composer update
Loading composer repositories with package information
Updating dependencies (including require-dev)
- Installing guzzlehttpstreams (2.1.0)
  Downloading: 100%
- Installing guzzlehttpguzzle (4.2.3)
  Downloading: 100%
- Installing 2amigos/yii2-google-maps-library (0.1.0)
  Downloading: 100%
Writing lock file
Generating autoload files
D:\wamp\www\yii2-gis>
```

รูปที่ 414 การติดตั้ง google maps library

เมื่อเปิดดูโครงสร้างไฟล์จะเห็น Folder ใน vendor ดังนี้



รูปที่ 415 ที่เก็บ yii2-google-maps-library

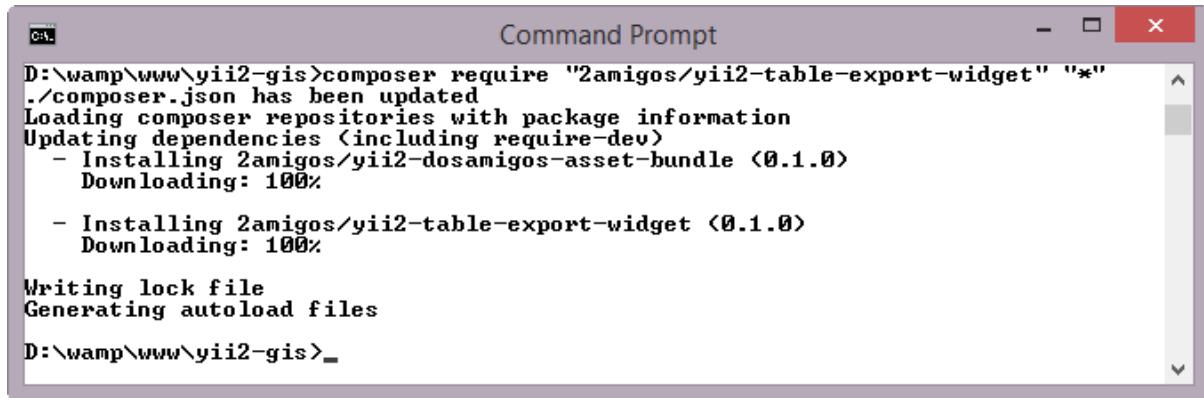
## Yii2 Export Table

สามารถเพิ่มเติมได้ที่ <https://github.com/2amigos/yii2-table-export-widget> โดยมีขั้นตอนในการติดตั้งดังนี้

เรียกใช้คำสั่ง ในการติดตั้งดังนี้

```
composer require "2amigos/yii2-table-export-widget" "*"
```

แล้วกดปุ่ม Enter จะได้การแสดงผลดังนี้



```
D:\wamp\www\yii2-gis>composer require "2amigos/yii2-table-export-widget" "*"
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
- Installing 2amigos/yii2-dosamigos-asset-bundle (0.1.0)
  Downloading: 100%
- Installing 2amigos/yii2-table-export-widget (0.1.0)
  Downloading: 100%
Writing lock file
Generating autoload files
D:\wamp\www\yii2-gis>_
```

รูปที่ 416 การติดตั้ง yii2-table-export-widget

เมื่อเปิดดูโครงสร้างไฟล์จะเห็น Folder ใน vendor ดังนี้



รูปที่ 417 ที่เก็บหลังการติดตั้ง

## เขียนโปรแกรม

### แก้ไขการตั้งค่า Yii2

#### การตั้งค่าการเชื่อมต่อฐานข้อมูล

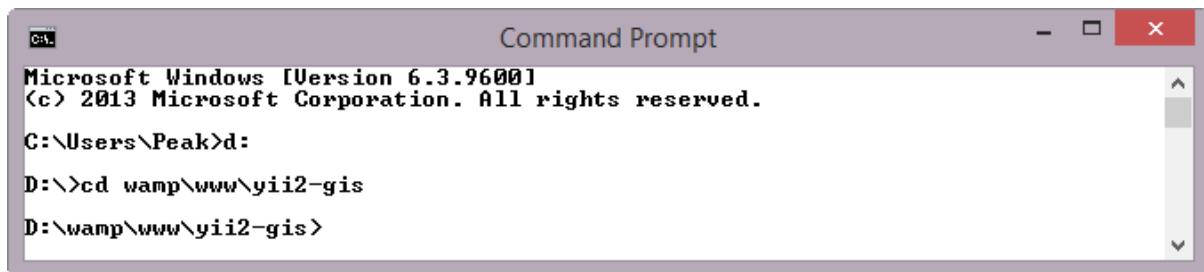
เปิดไฟล์ config/db.php เพื่อตั้งค่าการเชื่อมต่อฐานข้อมูลกำหนด dbname คือชื่อฐานข้อมูลในที่นี่คือ yii2\_gis และกำหนด username และ password สำหรับเชื่อมต่อฐานข้อมูล MySQL ดังนี้

```
<?php  
2  
3     return [  
4         'class' => 'yii\db\Connection',  
5         'dsn' => 'mysql:host=localhost;dbname=yii2_gis',  
6         'username' => 'root',  
7         'password' => '',  
8         'charset' => 'utf8',  
9     ];
```

รูปที่ 418 การเขียนต่อไปยังฐานข้อมูล

## สร้าง Migrate

เปิด Command Prompt จากนั้นเข้าไปที่ folder project ในที่นี่คือ d:\wamp\www\yii2-gis ดังนี้

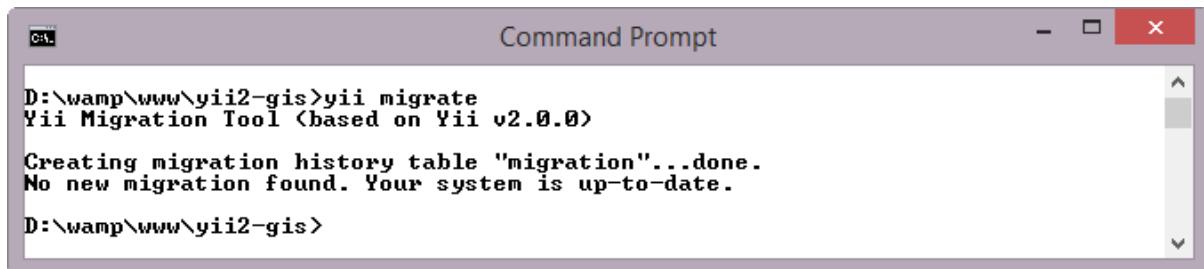


รูปที่ 419 เข้า Folder project ผ่าน command prompt

จากนั้นพิมพ์คำสั่ง

```
yii migrate
```

แล้วกดปุ่ม Enter



รูปที่ 420 การใช้คำสั่ง yii migration

ทำการ migration จาก yii2-user โดยพิมพ์คำสั่งต่อไปนี้

```
yii migrate/up --migrationPath=@vendor/dektrium/yii2-  
user/migrations
```

แล้วกดปุ่ม Enter จากนั้นโปรแกรมจะถามให้ Confirm ให้กด y และกด enter

```
D:\wamp\www\yii2-gis>yii migrate/up --migrationPath=@vendor/dektrium/yii2-user/migrations
Yii Migration Tool (based on Yii v2.0.0)

Total 6 new migrations to be applied:
m140209_132017_init
m140403_174025_create_account_table
m140504_113157_update_tables
m140504_130429_create_token_table
m140830_171933_fix_ip_field
m140830_172703_change_account_table_name

Apply the above migrations? (yes|no) [no]:y
*** applying m140209_132017_init
    > create table `{{%user}}` ... done (time: 1.749s)
    > create unique index user_unique_username on `{{%user}}` (username) ... done
```

รูปที่ 421 การ migration yii2-user

เมื่อติดตั้งเสร็จเรียบร้อยจะเห็นลักษณะนี้

```
> add foreign key fk_user_token: {{%token}} (user_id) references {{%user}} (id) ... done (time: 2.792s)
*** applied m140504_130429_create_token_table (time: 5.114s)

*** applying m140830_171933_fix_ip_field
    > alter column registration_ip in table `{{%user}}` to integer UNSIGNED ... done (time: 4.690s)
*** applied m140830_171933_fix_ip_field (time: 4.951s)

*** applying m140830_172703_change_account_table_name
    > rename table `{{%account}}` to `{{%social_account}}` ... done (time: 0.906s)
*** applied m140830_172703_change_account_table_name (time: 1.077s)

Migrated up successfully.

D:\wamp\www\yii2-gis>_
```

รูปที่ 422 ข้อความเมื่อ migrate เสร็จเรียบร้อยแล้ว

เมื่อเปิดดูตารางในฐานข้อมูลจะมีตารางต่างๆ เพิ่มขึ้นดังนี้

รายการ	การดำเนินการ	แมก	รหัส	การตรวจสอบ	ขนาด	เก็บความเข้าบัน
kind	เปิดดู   โหลดสร้าง   ค้นหา   แก้ไข   ลบข้อมูล   โอนเท็ง	->	InnoDB	utf8_general_ci	16 กิกะไบต์	-
level	เปิดดู   โหลดสร้าง   ค้นหา   แก้ไข   ลบข้อมูล   โอนเท็ง	->	InnoDB	utf8_general_ci	16 กิกะไบต์	-
migration	เปิดดู   โหลดสร้าง   ค้นหา   แก้ไข   ลบข้อมูล   โอนเท็ง	->	InnoDB	utf8_general_ci	16 กิกะไบต์	-
network	เปิดดู   โหลดสร้าง   ค้นหา   แก้ไข   ลบข้อมูล   โอนเท็ง	->	InnoDB	utf8_general_ci	16 กิกะไบต์	-
profile	เปิดดู   โหลดสร้าง   ค้นหา   แก้ไข   ลบข้อมูล   โอนเท็ง	->	InnoDB	utf8_general_ci	16 กิกะไบต์	-
school	เปิดดู   โหลดสร้าง   ค้นหา   แก้ไข   ลบข้อมูล   โอนเท็ง	->	InnoDB	utf8_general_ci	176 กิกะไบต์	-
social_account	เปิดดู   โหลดสร้าง   ค้นหา   แก้ไข   ลบข้อมูล   โอนเท็ง	->	InnoDB	utf8_general_ci	48 กิกะไบต์	-
token	เปิดดู   โหลดสร้าง   ค้นหา   แก้ไข   ลบข้อมูล   โอนเท็ง	->	InnoDB	utf8_general_ci	16 กิกะไบต์	-
user	เปิดดู   โหลดสร้าง   ค้นหา   แก้ไข   ลบข้อมูล   โอนเท็ง	->	InnoDB	utf8_general_ci	48 กิกะไบต์	-

รูปที่ 423 รายการตารางต่างๆ หลังจาก migrate

เปิด URL เพื่อเพิ่มข้อมูล admin ไปที่ <http://localhost/yii2-gis/web/index.php?r=user/registration/register> จากนั้นใส่ข้อมูลในฟอร์มโดยใช้ชื่อ username เป็น admin รหัสผ่านตามต้องการโดยมีความยาวอย่างน้อย 6 ตัวอักษร ในที่นี้จะใช้ adminadmin และกดปุ่ม Sign Up

Sign up

**Username**  
admin

**Email**  
kongoon@hotmail.com

**Password**  
.....

**Sign up**

รูปที่ 424 หน้าสมัครสมาชิก

จะปรากฏข้อมูลให้แสดงการยืนยันทาง Email

We need to confirm your email address  
Please check your email and click the confirmation link to complete your registration. The email can take a few minutes to arrive. But if you are having troubles, you can request a new one by clicking the link below.  
[Request new confirmation message](#)

รูปที่ 425 ข้อความหลังจากสมัครสมาชิกแล้ว

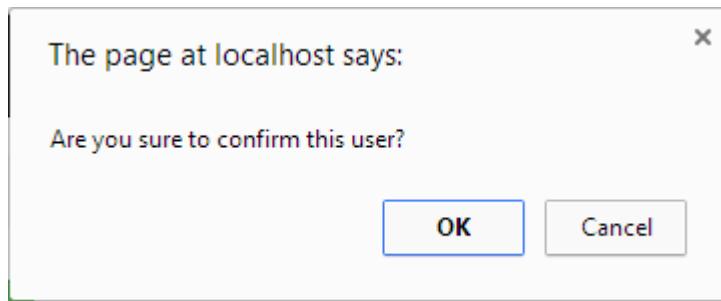
เข้าบริหารจัดการ User โดยไปที่ URL <http://localhost/yii2-gis/web/index.php?r=user/admin> จากนั้นระบบจะให้ Login ให้ใช้ username และ password ที่สมัครไว้ก่อนหน้านี้ จะปรากฏรายการของ User ดังนี้

## Manage users [Create a user account](#)

Username	Email	Registration ip	Registration time	Confirmation	Block status
admin	contact@programmerthailand.com	0.0.0.0	October 08, 2014 03:40	<a href="#">Confirm</a>	<a href="#">Block</a>

รูปที่ 426 หน้าจัดการผู้ใช้งาน

ทำการ Confirm User ให้สามารถใช้งานได้ปกติ โดยกดที่ปุ่ม Confirm สีเขียวเพื่อ Confirm user จะปรากฏ Popup ให้ยืนยัน จากนั้นกดปุ่ม OK เพื่อยืนยันข้อมูล



รูปที่ 427 PopUp แสดงการยืนยันการเปิดใช้งาน

จะปรากฏข้อความแสดงผลการยืนยันข้อมูลเรียบร้อยแล้ว

admin

User has been confirmed

Information

Registered at October 08, 2014 03:40 from 0.0.0.0  
Confirmed at October 08, 2014 03:40

รูปที่ 428 ข้อความหลัง Confirm User แล้ว

## สร้าง Model ด้วย Gii Generator

เรียก URL <http://localhost/yii2-gis/web/index.php?r=Yii2-Gii> เพื่อสร้าง Model ด้วย Gii Generator จากนั้นใส่เครื่องหมาย \* ที่ช่อง Table Name เพื่อสร้าง Model ในทุกๆ ตารางในฐานข้อมูลครั้งเดียว ที่สำคัญคือ อย่าลืมคลิกที่ Generate Labels from DB Comments แล้วกดปุ่ม Preview

# Model Generator

This generator generates an ActiveRecord class for the specified database table.

## Table Name

\*

## Namespace

app\models

## Base Class

yii\db\ActiveRecord

## Database Connection ID

db

## Use Table Prefix

## Generate Relations

## Generate Labels from DB Comments

## Enable I18N

รูปที่ 429 การสร้าง Model ด้วย Gii Generator

จะได้รายการของการสร้าง Model ดังนี้ และเลือก Generate Model เฉพาะที่ต้องการคือ Kind, Level, Network และ School และกดปุ่ม Generate

[Preview](#)
[Generate](#)

Click on the above [Generate](#) button to generate the files selected below:

Code File	Action	
models\Kind.php	create	<input checked="" type="checkbox"/>
models\Level.php	create	<input checked="" type="checkbox"/>
models\Migration.php	create	<input type="checkbox"/>
models\Network.php	create	<input checked="" type="checkbox"/>
models\Profile.php	create	<input type="checkbox"/>
models\School.php	create	<input checked="" type="checkbox"/>
models\SocialAccount.php	create	<input type="checkbox"/>
models\Token.php	create	<input type="checkbox"/>
models\User.php <a href="#">diff</a>	overwrite	<input type="checkbox"/>

รูปที่ 430 การเลือก Model ที่ต้องการสร้าง

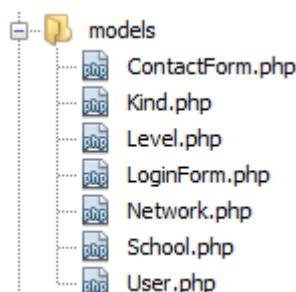
จะได้หน้า Web Application แสดงข้อความดังนี้

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-gis\vendor\yiisoft\yii2-gii\generators\model/default"...
generated models\Kind.php
generated models\Level.php
skipped models\Migration.php
generated models\Network.php
skipped models\Profile.php
generated models\School.php
skipped models\SocialAccount.php
skipped models\Token.php
skipped models\User.php
done!
```

รูปที่ 431 ข้อความหลังสร้าง Model เสร็จเรียบร้อยแล้ว

เมื่อเปิดดูที่ models จะเห็นไฟล์ Model ที่ถูกสร้างขึ้นดังนี้



รูปที่ 432 รายการไฟล์ Model

## สร้าง CRUD ด้วย Gii Generator

สร้าง CRUD สำหรับ Kind

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update, Delete) for a model.

#### Model Class

app\models\Kind

#### Search Model Class

app\models\KindSearch

#### Controller Class

app\controllers\KindController

#### Base Controller Class

yii\web\Controller

รูปที่ 433 การสร้าง CRUD สำหรับ Kind

กดปุ่ม Preview และกดปุ่ม Generate

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-gis\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated controllers\KindController.php
generated models\KindSearch.php
generated views\kind\_form.php
generated views\kind\_search.php
generated views\kind\create.php
generated views\kind\index.php
generated views\kind\update.php
generated views\kind\view.php
done!
```

รูปที่ 434 ข้อความหลังจากสร้าง CRUD เสร็จเรียบร้อย



www.programmerthailand.com

## สร้าง CRUD สำหรับ Network

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update, Delete) for a model.

### Model Class

app\models\Network

### Search Model Class

app\models\NetworkSearch

### Controller Class

app\controllers\NetworkController

### Base Controller Class

yii\web\Controller

รูปที่ 435 การสร้าง CRUD สำหรับ Network

กดปุ่ม Preview และกดปุ่ม Generate

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-gis\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated controllers\NetworkController.php
generated models\NetworkSearch.php
generated views\network\_form.php
generated views\network\_search.php
generated views\network\create.php
generated views\network\index.php
generated views\network\update.php
generated views\network\view.php
done!
```

รูปที่ 436 ข้อความหลังจากสร้าง CRUD เสร็จแล้ว



www.programmerthailand.com

สร้าง CRUD สำหรับ Level

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update) model.

## Model Class

app\models\Level

## Search Model Class

app\models\LevelSearch

## Controller Class

app\controllers\LevelController

## Base Controller Class

yii\web\Controller

รูปที่ 437 การสร้าง CRUD สำหรับ Level

กดปุ่ม Preview และกดปุ่ม Generate

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-gis\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated controllers\LevelController.php
generated models\LevelSearch.php
generated views\level\_form.php
generated views\level\_search.php
generated views\level\create.php
generated views\level\index.php
generated views\level\update.php
generated views\level\view.php
done!
```

รูปที่ 438 ข้อความหลังจากสร้าง CRUD เสร็จแล้ว



www.programmerthailand.com

สร้าง CRUD สำหรับ School

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update, Delete) operations for a model.

## Model Class

app\models\School

## Search Model Class

app\models\SchoolSearch

## Controller Class

app\controllers\SchoolController

## Base Controller Class

yii\web\Controller

รูปที่ 439 การสร้าง CRUD สำหรับ School

กดปุ่ม Preview และกดปุ่ม Generate

The code has been generated successfully.

```
Generating code using template "D:\wamp\www\yii2-gis\vendor\yiisoft\yii2-gii\generators\crud/default"...
generated controllers\SchoolController.php
generated models\SchoolSearch.php
generated views\school\_form.php
generated views\school\_search.php
generated views\school\create.php
generated views\school\index.php
generated views\school\update.php
generated views\school\view.php
done!
```

รูปที่ 440 ข้อความหลังการสร้าง CRUD เสร็จแล้ว

ทดลองเปิด URL <http://localhost/yii2-gis/web/index.php?r=school> จะได้หน้า Web Application ดังนี้



[www.programmerthailand.com](http://www.programmerthailand.com)

#	รหัสโรงเรียน	ชื่อสถานศึกษา	เลขที่บ้าน	ที่อยู่ของโรงเรียน	วันที่แก้ไข
1	1	2	3	4	5
2	2	1	3	4	5
3	3	2	3	4	5
4	4	2	3	4	5
5	5	2	3	4	5

รูปที่ 441 ทดลองเปิด URL สำหรับ School

## เขียนโปรแกรมแสดงแผนที่

การแสดงแผนที่นั้นจะแสดงเมื่อเปิด Web Application ขึ้นมา นั่นก็คือ Controller Site และ Action Index ดังนั้น ต้องไปแก้ไขโปรแกรมที่ไฟล์ controllers/SiteController.php และส่วนการแสดงผลที่ views/site/index.php

### แก้ไข controllers/SiteController.php

เปิดไฟล์ controllers/SiteController.php จากนั้นเพิ่มคำสั่งในการตรวจสอบและการเรียกข้อมูลจากฐานข้อมูลดังนี้ ซึ่งมี 2 ส่วน

ส่วนการตรวจสอบการส่งข้อมูลและส่วนการเรียกข้อมูล ซึ่งหากมีการส่งข้อมูลมาแสดงว่ามีการเลือกชุดข้อมูล ในที่นี้จะเลือก 2 แบบ นั่นคือเลือกแบบอำเภอ (amphur) กับเลือกแบบระดับการสอน (level)

```
if (isset($_GET['type'])) {
```

เป็นการตรวจสอบว่ามีการส่งข้อมูลหรือไม่ ถ้ามีการส่งข้อมูลเข้ามาให้ใช้เงื่อนไขในการเรียกข้อมูล

```
if ($_GET['type'] == 'amphur') { // ถ้าเลือกอำเภอ
    $w['amphur'] = $_GET['amphur'];
}

if ($_GET['type'] == 'level') { // ถ้าเลือกระดับที่เปิดสอน
    $w['level_id'] = $_GET['level'];
```

```
}
```

```
$schools = new \yii\data\ActiveDataProvider([
    'query' => \app\models\School::find()
->where($w), //เรียกข้อมูลตามเงื่อนไข $w แบบ ActiveDataProvider ส่ง
ให้ GridView
]);

$schools2 = \app\models\School::find()->where($w)-
>with('network')->each(); //เรียกข้อมูลตามเงื่อนไข $w แบบ array ส่งให้ foreach
```

แต่ถ้าไม่มีการเลือกข้อมูลหรือเป็นการเปิดเข้ามาครั้งแรกจะเรียกข้อมูลทั้งหมดไปแสดงผล

```
$schools = new \yii\data\ActiveDataProvider([
    'query' => \app\models\School::find//เรียกข้อมูล แบบ
ActiveDataProvider ส่งให้ GridView
];

$schools2 = \app\models\School::find()-
>with('network')->each(); //เรียกข้อมูลตามเงื่อนไข $w แบบ array ส่งให้ foreach
```

เมื่อเขียนคำสั่งแบบเต็มจะได้รูปแบบดังนี้



```
public function actionIndex() {
    $w = [];  
    if (isset($_GET['type'])) { //ตรวจสอบว่ามีการเลือกข้อมูลหรือไม่  
        if ($_GET['type'] == 'amphur') { //ถ้าเลือกอำเภอ  
            $w['amphur'] = $_GET['amphur'];  
        }  
        if ($_GET['type'] == 'level') { //ถ้าเลือกระดับที่เปิดสอน  
            $w['level_id'] = $_GET['level'];  
        }  
  
        $schools = new \yii\data\ActiveDataProvider([  
            'query' => \app\models\School::find()  
                ->where($w), //เรียกข้อมูลตามเงื่อนไข $w แบบ array  
        ]);  
        $schools2 = \app\models\School::find()  
            ->where($w)  
            ->with('network')  
            ->each(); //เรียกข้อมูลตามเงื่อนไข $w แบบ array  
    } else { //ถ้าไม่มีการเลือกข้อมูลให้แสดงทั้งหมด  
        $schools = new \yii\data\ActiveDataProvider([  
            'query' => \app\models\School::find() //เรียกข้อมูล  
        ]);  
        $schools2 = \app\models\School::find()  
            ->with('network')  
            ->each(); //เรียกข้อมูลตามเงื่อนไข $w แบบ array  
    }  
  
    return $this->render('index', ['schools' => $schools,  
        'schools2' => $schools2]);  
}
```

รูปที่ 442 การแก้ไข SiteController.php

## แก้ไข views/site/index.php

โหลด Class ที่ต้องการใช้งาน

```
1 <?php  
2     $this->title = 'Yii2 GIS'; //กำหนด Title  
3  
4     use yii\helpers\Html;  
5     use yii\grid\GridView;  
6     //Google Library  
7     use dosamigos\google\maps\LatLng;  
8     use dosamigos\google\maps\overlays\InfoWindow;  
9     use dosamigos\google\maps\overlays\Marker;  
10    use dosamigos\google\maps\Map;  
11 ?>
```

รูปที่ 443 การแก้ไข View index.php

สร้าง Jumbotron ของ Bootstrap

```
12 <div class="site-index">  
13  
14     <div class="jumbotron">  
15         <h1>GISEdu</h1>  
16  
17         <p class="lead">ระบบสารสนเทศทางภูมิศาสตร์  
18             (Geographic Information System) </p>  
19  
20     </div>
```

รูปที่ 444 การเพิ่ม Jumbotron

สร้างปุ่มกด

```
22 <div class="body-content">
23
24     <div class="row">
25         <div class="col-md-2">
26             <?= Html::a('ເໝັ້ນອຸປະກອນລາຊົ້ານີ້', ['index', 'type' => 'amphur',
27                 'amphur' => 'ເໝັ້ນອຸປະກອນລາຊົ້ານີ້'],
28                 ['class' => 'btn btn-primary btn-block']); ?>
29             <?= Html::a('ມ່ານສໍາມລົມ', ['index', 'type' => 'amphur',
30                 'amphur' => 'ມ່ານສໍາມລົມ'],
31                 ['class' => 'btn btn-primary btn-block']); ?>
32             <hr>
33             <?= Html::a('ປະກມຕິກາ', ['index', 'type' => 'level',
34                 'level' => 2],
35                 ['class' => 'btn btn-success btn-block']); ?>
36             <?= Html::a('ໝາຍບົກກັດ', ['index', 'type' => 'level',
37                 'level' => 1], ['class' => 'btn btn-success btn-block']); ?>
38         </div>
```

ຮູບທີ 445 ການສ້າງບຸ້ມກົດດ້ານຫ້າຍມືອງ

ຕັ້ງຄ່າແຜນທີ່

```
39         <div class="col-md-10">
40             <?php
41             //print_r($schools);
42             $coord = new LatLng(['lat' => 15.39745881, 'lng' => 104.72004496];
43             $map = new Map([
44                 'center' => $coord,
45                 'zoom' => 10,
46                 'width' => '100%',
47                 'height' => '500',
48             ]);
```

ຮູບທີ 446 ການຕັ້ງຄ່າແຜນທີ່

ວນ Loop ຊ້ອມລຸ ຈາກຕັ້ງແປຣ \$schools2 ທີ່ສ່າງມາຈາກ Controller

```
49     foreach ($schools2 as $s) {
50         //echo $s->school_name;
51         $coords = new LatLng(['lat' => $s->lat, 'lng' => $s->lng]);
52         // ສ້າງ marker ໃນແຜນທີ່
53         $marker = new Marker([
54             'position' => $coords,
55         ]);
```

ຮູບທີ 447 ການ Loop ຊ້ອມເພື່ອສ້າງຈຸດໃນແຜນທີ່

ກຳທັນດີ Info Window ເນື້ອກົດ Marker

```
57 // กำหนด InfoWindow ให้กับ Marker  
58 $marker->attachInfoWindow(  
59     new InfoWindow([  
60         'content' => '<h4>{$ชื่อร้าน}' . $s->school_name . ''  
61             . '</h4><table class="table table-bordered">'  
62             . '<tr></tr><tr><td>ที่อยู่</td><td>' . $s->address . '</td>  
63             <td rowspan="9" align="center" valign="top">  
64                 <br>  
65                 <strong>' . $s->boss . '</strong><br>  
66                 ผู้อำนวยการ</td>  
67             </tr>  
68             <tr><td>ตำบล</td><td>' . $s->tambol . '</td>  
69             </tr>  
70             <tr><td>อำเภอ</td><td>' . $s->amphur . '</td>  
71             </tr>  
72             <tr><td>จังหวัด</td><td>' . $s->province . '</td>  
73             </tr>  
74             <tr><td>รหัสไปรษณีย์</td><td>' . $s->zipcode . '</td>  
75             </tr>  
76             <tr><td>หมายเลขโทรศัพท์</td><td>' . $s->tel . '</td>  
77             </tr>  
78             <tr><td>ละติจูด</td><td>' . $s->lat . '</td>  
79             </tr>  
80             <tr><td>ลองจิจูด</td><td>' . $s->lng . '</td>  
81             </tr>  
82             <tr><td>Website</td><td><a href="' . $s->website . ''  
83                 . '" target="_blank">' . $s->website . '</a></td>  
84             </tr>  
85             </table>'  
86         ])  
87     );
```

รูปที่ 448 การกำหนด InfoWindows

นำ Marker ที่ได้ใส่ลงในแผนที่ \$map

```
89 // นำ Marker ที่ได้ใส่ในแผนที่  
90 $map->addOverlay($marker);  
91 }
```

รูปที่ 449 การเพิ่มจุดพิกัดในแผนที่

กำหนดให้แสดงแผนที่

```
93 // กำหนดให้แสดงแผนที่  
94 echo $map->display();
```

รูปที่ 450 การกำหนดให้แสดงแผนที่



ด้านล่างของแผนที่ นำตัวแปร \$schools จาก ActiveDataProvider ใน SiteController มาแสดงผลแบบ GridView

```
96 | echo GridView::widget([
97 |     'dataProvider' => $schools,
98 |     'columns' => [
99 |         'school_name',
100 |         'level.level',//ดึงมาจากตาราง level
101 |         'network.network_name',//ดึงมาจากตาราง network
102 |         'kind.kind',//ดึงมาจากตาราง kind
103 |         'address',
104 |         'tambol',
105 |         'amphur',
106 |         'boss',
107 |     ],
108 | ]);
109 | ?>
110 |         </div>
111 |     </div>
112 |
113 | </div>
114 | </div>
```

รูปที่ 451 การแสดงรายการตาราง

## กำหนดสิทธิ์ให้เฉพาะเจ้าหน้าที่ สามารถเพิ่ม แก้ไข และลบข้อมูล

ในการกำหนดสิทธิ์การเข้าใช้งานนั้นให้กำหนดใน Controller ของทุกๆ Controller ที่ต้องการกำหนดสิทธิ์ ซึ่งกำหนดในส่วนของ Method behaviors() โดยเพิ่มคำสั่งดังนี้

```
'access' => [
    'class' =>
\yii\filters\AccessControl::className(),
    'only' => ['create', 'update', 'delete'],
    'rules' => [
        // deny all POST requests
    ]
]
```

```
'allow' => false,  
'verbs' => ['POST']  
,  
// allow authenticated users  
[  
    'allow' => true,  
    'roles' => ['@'],  
,  
// everything else is denied  
,  
],
```

เมื่อตั้ง Method behaviors จะได้ดังนี้

```
public function behaviors() {
    18     return [
    19         'verbs' => [
    20             'class' => VerbFilter::className(),
    21             'actions' => [
    22                 'delete' => ['post'],
    23             ],
    24         ],
    25         'access' => [
    26             'class' => \yii\filters\AccessControl::className(),
    27             'only' => ['create', 'update', 'delete'],
    28             'rules' => [
    29                 // deny all POST requests
    30                 [
    31                     'allow' => false,
    32                     'verbs' => ['POST']
    33                 ],
    34                 // allow authenticated users
    35                 [
    36                     'allow' => true,
    37                     'roles' => ['@'],
    38                 ],
    39                 // everything else is denied
    40             ],
    41         ],
    42     ];
    43 }
```

รูปที่ 452 การกำหนดลิฟต์การเข้าถึง

## บทสรุป

ระบบสารสนเทศทางภูมิศาสตร์นั้นเป็นระบบที่สร้างขึ้นเพื่อให้ช่วยให้เรียนรู้ Yii2 Basic Application ได้ง่ายดายมากยิ่งขึ้น ซึ่งมีขั้นตอนการเรียนรู้แบบ Step by Step โดยการเชื่อมต่อกับ Google Map เพื่อแสดงข้อมูลในรูปแบบแผนที่ และมีการใช้ Extension เสริมเพิ่มเติมเข้ามาในระบบเพื่อช่วยให้การพัฒนาระบบ เป็นไปตามความต้องการมากขึ้น เช่นการติดตั้ง Yii2 User, Yii2 Export Table และ Yii2 Google Map Library เป็นต้น และเรียนรู้การเรียกข้อมูลมาแสดงในแผนที่และแบบตารางตามเงื่อนไขต่างๆ

ทั้งนี้ผู้เขียนหวังว่าระบบสารสนเทศทางภูมิศาสตร์นี้จะช่วยให้ผู้อ่านเรียนรู้และเข้าใจการประยุกต์ใช้ Yii Framework 2 ในการแสดงผลข้อมูลออกมานในรูปของแผนที่จาก Google Map และแสดงผลเป็น

ตารางข้อมูล และเรียนรู้การเรียกข้อมูลตามเงื่อนไขต่างๆ ได้ และสามารถนำระบบไปประยุกต์ใช้ในหน่วยงานต่อไปได้



## คำขอบคุณ

ท้ายนี้ต้องขอขอบพระคุณท่านผู้อ่านทุกๆ ท่านที่ช่วยสนับสนุนพวกเรา Programmer Thailand หากว่าหนังสือเล่มนี้มีข้อขัดแย้งหรือส่วนใดทางทีมงานต้องกราบขออภัยมา ณ ที่นี่ และหากจะเป็นประโยชน์มากขึ้นทางทีมงานขอเชิญผู้อ่านให้คำแนะนำพວກเราได้ที่ contact@programmerthailand.com ซึ่งเราจะนำไปปรับปรุงและแก้ไขต่อไป

## ទារចន៍

Access Control, 101, 102, 103, 226, 229  
action, 71, 72, 94, 101, 102, 103  
Alias, 76, 84  
APC Extension, 9  
Application Component, 70, 71  
Array, 74, 83, 176  
backend, 23, 48, 58, 59, 60, 64, 65, 66, 77, 129, 131, 132, 133, 134, 168, 172, 173, 175, 178  
Backend, 57, 60, 61, 64, 119, 127, 128, 165, 167, 169  
Command Prompt, 22, 24, 43, 67, 104, 164, 166, 168, 170, 174, 177, 178, 196, 203, 246, 253  
Component, 42, 64, 71, 104, 176  
composer, 22, 24, 25, 43, 67, 71, 141, 149, 150, 151, 160, 164, 166, 168, 170, 174, 197, 246, 248, 249, 250, 251  
composer.json, 71, 149, 150, 151, 170, 248, 250  
Controller, 34, 69, 70, 71, 73, 78, 81, 92, 94, 95, 101, 116, 125, 133, 226, 227, 263, 267, 269  
Database, 27, 45, 189  
Filter, 70, 101  
frontend, 23, 44, 47, 51, 54, 55, 56, 58, 62, 63, 77, 119, 120, 121, 123, 124, 125, 126, 166  
Frontend, 57, 61, 62, 119, 127, 165, 166, 167, 168  
Model, 31, 32, 34, 52, 53, 54, 69, 70, 71, 77, 81, 90, 91, 113, 115, 116, 122, 123, 124, 125, 130, 131, 132, 133, 206, 213, 256, 257, 258  
Module, 75, 111, 112, 113, 115, 116, 119, 120, 121, 122, 125, 127, 128, 129, 130, 132, 175  
MVC, 69  
namespace, 32, 55, 73, 76, 77, 91, 139, 152  
Namespace, 76, 123, 131  
PHP Framework, 7, 69  
Project Skeleton, 44, 196, 197, 246  
Relation, 90  
Request, 70  
Routing, 71, 73  
Template, 23, 24, 25, 30, 38, 42, 43, 51, 57, 61, 76, 77, 103, 111, 119, 127, 135, 136, 165, 174, 196, 246  
View, 34, 57, 69, 70, 71, 72, 79, 81, 82, 94, 116, 126, 133  
WAMP, 9, 11, 12, 15, 17, 18, 41, 61  
Web Application, 6, 7, 9, 23, 24, 25, 26, 30, 36, 38, 39, 40, 42, 43, 44, 48, 51, 52, 59, 60, 62, 66, 67, 69, 70, 71, 72, 76, 90, 100, 101, 111, 118, 119, 121, 124, 127, 129, 131, 135, 136, 141, 151, 165, 166,

- 167, 168, 169, 174, 178, 213, 214, 218,  
220, 229, 248, 258, 262, 263
- Yii Framework, 6, 7, 18, 30, 54, 57, 71, 90
- Yii2, 6, 7, 8, 9, 23, 24, 25, 38, 43, 57, 67, 69,  
70, 71, 72, 73, 76, 77, 103, 111, 119,  
121, 127, 135, 136, 150, 151, 165, 170,  
171, 174, 196, 203, 246, 248, 250, 251,  
252
- Yii2 Advanced, 23, 43, 57, 103, 119, 127,  
135, 165, 174
- Yii2 Basic, 23, 24, 38, 111, 136  
 yii2-advanced, 43, 44, 45, 47, 48, 51, 56,  
 58, 59, 60, 64, 66, 104, 121, 126, 127,  
 129, 134, 178  
 ฐานข้อมูล, 9, 26, 27, 29, 45, 46, 47, 48, 50,  
 52, 69, 70, 77, 84, 88, 91, 113, 170, 171,  
 173, 188, 189, 190, 194, 195, 203, 233,  
 234, 243, 252, 254, 256, 263  
 แม่แบบ, 23, 196, 246  
 โมดูล, 111, 119, 127