

[SOFTNORTH COMPANY]

Login System

[โครงการพัฒนาซอฟต์แวร์ ระบบการ Login]

6/10/2015

Project Plan	Confidential	Page 0 of 14
		Print Date: 10/6/15

DATE:	June,10,2015	Author Name :	Wacharin Naktubtim
--------------	--------------	----------------------	--------------------

Application Details:	
Application:	Login System

Version History				
Version	Date	Author	Reviewer	Description
0.1	June,10,2015	Wacharin Nak.	-	เก็บรวบรวมข้อมูล Requirement

Project Plan	Confidential	Page 1 of 14
		Print Date: 10/6/15

Table of Contents

1. Introduction	3
1.1 Purpose.....	3
1.2 Project Scope.....	3
2. Project Overviews	3
2.1 Product Feature.....	3
2.2 System Requirement.....	3
3. Project Schedule.....	4
3.1 Schedule	4
3.2 Task Planning	4
3.3 Module Planning	5
4. Codeing Standard and Naming Covention.....	5
4.1 PHP Source Files.....	5
4.2Methodnaming.....	5
DataDictionary.....	6
Class Diagram และ ชนิดตัวแปร.....	7
5. Class Register	8
6. Methods Detail	8
7. Class Member	10
8. Coding Standard and Naming Convention.....	10

บทนำ

เอกสารฉบับนี้เป็นส่วนหนึ่งของการพัฒนาระบบการเข้าใช้ระบบ หรือ การ Login โดยจะใช้ในการยืนยันผู้ใช้ ว่าชื่อผู้ใช้นี้ตรงกับฐานข้อมูลที่มีอยู่หรือไม่

1. Introduction

1.1 Purpose

เนื่องจากการได้มีการจัดทำระบบการ Login ซึ่งระบบของเรา ได้จัดทำขึ้นเฉพาะ admin โดยสามารถกรอกข้อมูลได้ ดังนี้ Username ซึ่งเป็นรหัสนักศึกษา และ Password เป็นวัน เดือน ปี เกิด

1.2 Project Scope

ระบบสมัครสมาชิก

1.2.1. ส่วนของการกรอกรายละเอียดในการเข้าใช้ระบบ

1.2.1.1 ผู้ใช้สามารถกรอกข้อมูลผู้ใช้ โดยการใส่ Username ซึ่งเป็นรหัสนักศึกษาในการเข้าสู่ระบบ

1.2.1.2 ผู้ใช้สามารถกรอกข้อมูลผู้ใช้ โดยการใส่ Password ซึ่งเป็นวัน เดือน ปี เกิด ในการเข้าสู่ระบบ

2. Product Overviews

2.1 Product Feature

2.1.1 มีระบบการเข้าสู่ระบบทั้งหมดโดยผู้ดูแล

- การ Login เข้าสู่ระบบ

2.2 System Requirement

เครื่องคอมพิวเตอร์ที่สามารถใช้งานระบบนี้ได้ต้องคุณสมบัติ ดังนี้

- Operating system : Windows XP, Windows 7, Windows 8
- Processor : Intel Core 2, Intel Core 5, Intel Core 7
- RAM Memory : 4 GB
- Free space on hard disk : 3 GB
- Web Server support PHP
- Keyboard
- Mouse

3. Project Schedule

3.1 Schedule

โครงการพัฒนาระบบจัดการสมัครสมาชิกนับตั้งแต่ 10 มิถุนายน 2558 – 10 พฤษภาคม 2558 รวมเป็นจำนวน 31 ชั่วโมง โดยมีแผนการดำเนินการโดยสังเขป

เดือน / ภารกิจ	09/06/2015				10/06/2015		
	5.00 PM	6.00 PM	7.00 PM	8.00 PM	9.00 PM	10.00 PM	11.00 PM
1. เก็บรวบรวมความต้องการและวิเคราะห์ความต้องการ	↔						
2. ออกแบบ Class diagram	↔						
3. จัดทำเอกสาร SRS		↔					
4. ออกแบบฐานข้อมูล							
5. เริ่มลงมือเขียน Code ด้วย PHP framework				↔	↔		
6. เริ่มทดสอบโปรแกรม							

ตารางแสดงเวลาการดำเนินการพัฒนาซอฟต์แวร์

3.2 Task Planning

ภารกิจ	ผู้รับผิดชอบ
1. เก็บรวบรวมความต้องการและวิเคราะห์ความต้องการ	นางสาว วิจิตตรา ปีกลม
2. ออกแบบ Class diagram	นาย วัชรินทร์ นาคทับทิม
3. จัดทำเอกสาร SRS	นาย วัชรินทร์ นาคทับทิม
4. ออกแบบฐานข้อมูล	นางสาว วิจิตตรา ปีกลม
5. เริ่มลงมือเขียน Code	นาย ปิ์ติวี นาย พงศ์ปกรณ์ ทองจินดา นาย ลายไท ปั่นอ้าย
6. เริ่มทดสอบโปรแกรม	นาย ลายไท ปั่นอ้าย

แผนการดำเนินงานในการพัฒนาระบบการจัดการสมัครสมาชิก

รุ่น 0.1 มีการจัดการหน้าที่การทำงานในกระบวนการแต่ละขั้นตอนการออกแบบ ไปจนถึงขั้นตอนการพัฒนาระบบ มีดังต่อไปนี้

3.3 ModulePlanning

แผนการดำเนินงานในการพัฒนาระบบการจัดการสมัครสมาชิกรุ่น 1.0 มีการจัดการความรับผิดชอบในกระบวนการในแต่ละส่วนของ module/Package ดังรายละเอียดดังต่อไปนี้

ภารกิจ	ผู้รับผิดชอบ
ระบบจัดการเข้าสู่ระบบทั้งหมดโดยผู้ดูแล - การเข้าสู่ระบบ Login	นาย ทยไทย์ ปิ่นอ้าย

ตารางแสดงหน้าที่ความรับผิดชอบในแต่ละ Module/Package

4. Coding Standard and Naming Convention

เอกสารฉบับนี้ถือเป็นข้อตกลงภายในองค์กรเกี่ยวกับแนวปฏิบัติ สำหรับรูปแบบการเขียนโปรแกรมด้วยภาษา PHP ซึ่งผู้พัฒนาต้องยอมรับมาตรฐานที่กำหนดไว้นี้ไปใช้งานทั้งนี้เพื่อประโยชน์ในการพัฒนา Software

4.1 PHP,Source,Files

เอกสาร PHP Source code อนุญาตให้มีได้โดยไม่จำกัดจำนวน เพียงแต่นับรวมและทั้งไฟล์ไม่เกิด 1,000 บรรทัด เนื่องจาก Source Code ที่มีมากเกินไปจะก่อให้เกิดความยุ่งยากต่อการควบคุม หากเกิน 1,000 บรรทัด ให้แตกออกมาเป็นอีกไฟล์หนึ่ง โดยใช้ชื่อไฟล์ที่เหมือนแตกออกมาแล้วด้วย .php เช่น Register.php เมื่อแตกออกมาจะเป็น Register1.php เป็นต้น

4.2 Methodnaming

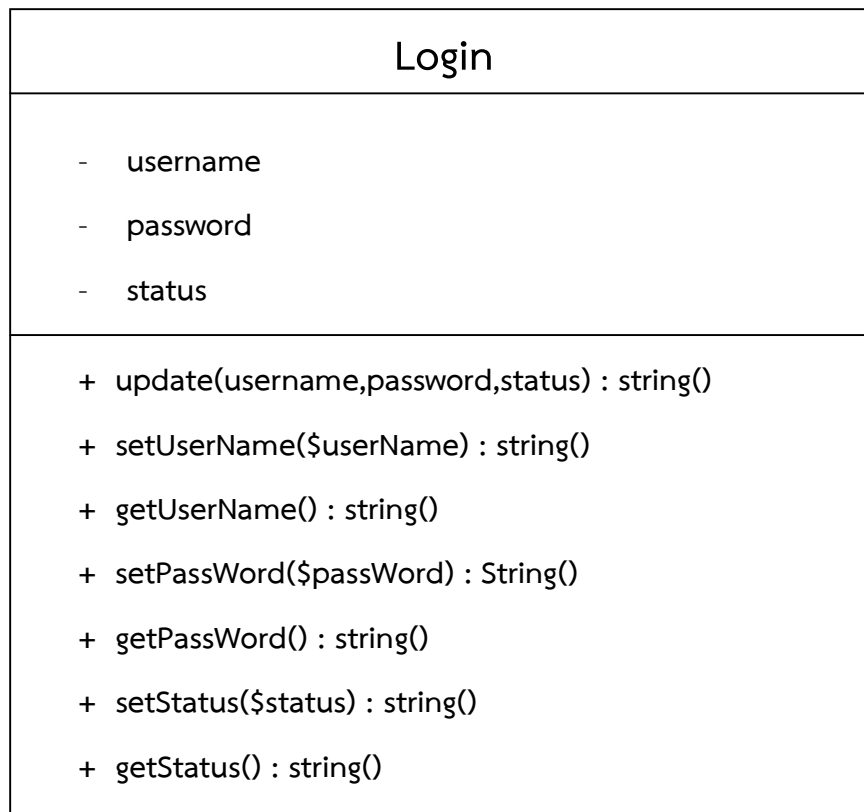
ชื่อของ Method ในภาษาพีเอชพี ให้ประกาศตามหลัก CamelCase และห้ามใช้ Underscore ในการตั้งชื่อ โดยทั่วไปการตั้งชื่อ Method มักจะเป็นคำกริยาหรือคำนาม เช่น findById() , findByIdAdd()

Data Dictionary

1. ตารางฐานข้อมูล

ชื่อฐานข้อมูล :		-	
ชื่อตาราง :		-	
คำอธิบาย :		เป็นการเข้าสู่ระบบ	
คีย์หลัก :		username	
คีย์นอก :		-	
ชื่อข้อมูล	ประเภท	คำอธิบาย	ตัวอย่างข้อมูล
username	varchar(50)	ชื่อเข้าระบบของสมาชิก	"B552531003"
password	varchar(15)	รหัสเข้าระบบของสมาชิก	"13041992"
status	enum('admin', 'user')	บอกสถานะว่าใครใช้งาน user หรือ admin	"enum('admin', 'user')"

Class Diagram



ชนิดตัวแปร

Variable	Type	Description
username	STRING	รับค่าชื่อเข้าสู่ระบบ
password	STRING	รับรหัสเข้าสู่ระบบ
status	STRING	บอกสถานะว่าเป็น user หรือ admin

5.Class Register

5.1 add(string: memberUsername, string: memberPassword, string: email:string: memberName, string: memberLastname, string: memberTel):boolean

เป็นฟังก์ชันการเพิ่ม ชื่อห้อง วันที่และเวลาการเข้าใช้งาน โดยชนิดของMethods จะใช้แบบ public เพื่อให้ทุก class เข้ามาเรียกใช้งานได้ โดยmethods นี้จะต้องส่งค่าเป็น string และreturn ค่าเป็นobject ชื่อว่า result

Example

5.2 add(string: memberUsername, string: memberPassword, string: email:string: memberName, string: memberLastname, string: memberTel):boolean

```
Public fuctionadd(username,password,name,lastname,email,tel)
{
    retrun true;
}
```

จะทำการส่งค่าข้อมูลการจองห้องไปใน database เป็นการเพิ่มคำสั่งเมื่อทำการสำเร็จจะคืนค่า true กลับมาเมื่อทำสำเร็จ

6. Methods Detail

Class Room

6.1 add(string: roomname, string: datein, string: timein):boolean

เป็นฟังก์ชันการเพิ่ม ชื่อห้อง วันที่และเวลาการเข้าใช้งาน โดยชนิดของMethods จะใช้แบบ public เพื่อให้ทุก class เข้ามาเรียกใช้งานได้ โดยmethods นี้จะต้องส่งค่าเป็น string และreturn ค่าเป็นobject ชื่อว่า result

Example

6.2 add(string: roomname, string: datein, string: timein):boolean

เช่น

```
Public fuction add(room1,05/06/2015,09.00น.-10.00น.)
```

```
{
```

```
    retrun true;
```

```
}
```

จะทำการส่งค่าข้อมูลการจองห้องไปใน database เป็นการเพิ่มคำสั่งเมื่อทำการสำเร็จจะคืนค่า true กลับมาเมื่อทำสำเร็จ

findByDate(\$date:string):query

เช่น

```
publicfunctionfindByDate(06/05/2015)
```

```
{
```

```
    return query
```

```
}
```

จะเป็นการค้นหาจาก datein หรือวันที่เข้าใช้งานเพื่อตรวจสอบวันเวลาใช้งานเมื่อทำการค้นหาแล้วจะนำค่าทั้งหมดที่หาได้จากฟิลล์ datein นั้นทุกๆ ฟิลล์กลับมาโดยเมื่อทำเสร็จจะคืนค่า query กลับมา

findAll():query

เช่น

Public function findByAll()

{

return query

}

จะเป็นการค้นหาจากข้อมูลทุกฟิลด์ที่มีอยู่ใน database ออกมานั้นทุกๆ ฟิลด์กลับมาโดยเมื่อทำเสร็จจะคืนค่า query กลับมา

7. Class Member

login(string:username,string:password):Boolean

return true

}

จะเป็นการใช้ username และ password ตรวจสอบจาก database และเมื่อตรวจสอบสิทธิว่าตรงแล้วจึงจะคืนค่ามาเป็น true

8. Coding Standard and Naming Convention

เอกสารฉบับนี้ถือเป็นข้อตกลงภายในองค์กรเกี่ยวกับแนวทางปฏิบัติสำหรับรูปแบบการเขียนโปรแกรม ด้วยภาษาพีเอชพี ซึ่งทีมพัฒนาทุกคนต้องยอมรับและนามาตรฐานที่กำหนดไว้นี้ไปใช้งาน ทั้งนี้ก็เพื่อประโยชน์ต่อการรูปแบบการพัฒนาซอฟต์แวร์ภายในองค์กรในหลายด้าน ดังเช่นตัวอย่างต่อไปนี้

- เพื่อรูปแบบของ source code ที่สามารถอ่านและทำความเข้าใจได้ง่ายยิ่งขึ้น
- เพื่อความสะดวกในขั้นตอนการบำรุงรักษา source code
- เพื่อง่ายต่อการทำความเข้าใจเมื่อมีความจำเป็นต้องส่งมอบ source code ให้ทีมพัฒนาอื่น

8.1 5.1 PHP Source Files

เอกสาร PHP Source code แต่ละไฟล์จะมี 1 Public Class ต่อ 1 ไฟล์เท่านั้น โดยจำนวนของ public Class นั้นอนุญาตให้มีได้โดยไม่ได้อำกัจำนวน เพียงแต่เมื่อนับรวมกันแล้วทั้งไฟล์ไม่ควรมีเกิน 2,000 บรรทัด ถ้าเกินให้ทำclassใหม่โดยใช้ underscore คั่นชื่อ class แล้วให้ใส่เลขกำกับเช่น เมื่อ class ชื่อ name.php ให้เป็น name_01.php

8.2 5.2. Source files naming

เอกสาร Source code ควรมีชื่อของเอกสารเป็นไปตามรูปแบบ

โดยชื่อของเอกสารจะต้องเป็นชื่อเดียวกับ Public Class ที่ประกาศไว้ใน Source code ของเอกสารนั้น โดยในทุกเอกสารต้องมีนามสกุลเป็น “.php” เท่านั้น

8.3 Naming Conventions

ข้อกำหนดของการตั้งชื่อ สำหรับส่วนประกอบต่างๆ ที่เกี่ยวข้องกับการเขียนโปรแกรมด้วย ภาษา PHP เช่น การประกาศ Package , Class , Interface ,Field,method , local variable ฯลฯ โดยมีข้อตกลงการตั้งชื่อทั้งหมด 2 รูปแบบคือ

- PascalCase คือ การตั้งชื่อโดยใช้ตัวอักษรแรกของแต่ละคำในชื่อที่จะตั้งเป็นตัวอักษรใหญ่ตัวอย่างเช่น AddName , EditLastName เป็นต้น

- CamelCase คือ การตั้งชื่อโดยใช้ตัวอักษรแรกของคำที่หนึ่งเป็นตัวเล็ก (lower case) และตั้งแต่คำที่สองเป็นต้นไป ในชื่อที่จะตั้งให้เป็นตัวอักษรใหญ่ (upper case) ตัวอย่างเช่น realName, findRoomByRoomId, getAddress เป็นต้น

8.4 Class naming

การประกาศชื่อของ Class ให้ยึดรูปแบบ Pascal Case โดยชื่อของ Class ต้อง

เป็นคำนำหรือ วลี ที่ตั้งตามวัตถุประสงค์ของการสร้าง Class และห้ามใช้ underscore ในการตั้งชื่อ เช่น Book,StudentName เป็นต้น

8.5 Field naming

การตั้งชื่อตัวแปร ควรตั้งชื่อให้สื่อถึงสิ่งที่ตัวแปรจะเก็บ และให้ใช้หลักการประกาศแบบ camelCase ตัวอย่างเช่น deleteName

8.6 Method naming

ชื่อของ Method ในภาษาพีเอชพี ให้ประกาศตามหลัก camel Case และห้ามใช้ underscore ในการตั้งชื่อ โดยทั่วไปการตั้งชื่อ Method มักจะเป็นคำกริยา หรือ คำนาม โดยควรเลือกให้เหมาะสมกับสิ่งที่ส่งกลับ หรือ หน้าที่ของ Method

5.6.1 Example

```
public function addName()
```

8.7 Indentation

การย่อหน้าในการเขียนโปรแกรมด้วยภาษา PHP ให้ใช้ tab โดยการ tab จะ tab ทุกครั้งที่ขึ้นต้นตัวอักษรแรกของบรรทัด โดยการ tab ของ function หรือ methods จะใช้ 2 tab ทุกครั้ง ส่วนการใช้ Braces { } จะเปิดหลัง methods นั้นๆ เสมอ เช่น function index() { เป็นต้น แล้วมีการใช้ 3 tab ในการเรียกใช้คำสั่งต่างๆ ภายใน methods และจะเพิ่ม 1 tab ไปเรื่อยๆ ถ้ามีคำสั่งต่างๆ เป็นคำสั่งย่อยๆ และจะใช้ Braces ปิดตรงคำแรกของบรรทัดเสมอ

8.7.1 Example

```
Public function add()  
{  
    if(i=0)  
    }  
    {  
        i=i++  
    }  
}
```

8.8การกำหนด modification ของ attribute

ทุก attribute ของ class ควรเป็น private และให้เขียนโปรแกรมเพื่อทำการ getter/setter ข้อมูลแทน โดยในขั้นตอนการ setter ทุกค่าเข้าไปให้ attribute ต้องมีการ validation input ก่อน