

Final Project Document

ชื่อกลุ่ม	Blue-O
ชื่อโครงการ	ระบบเช็คชื่อเข้าเรียน (Lab Checker)
ชื่อของ GitHub repository	: CE-KMITL-OOAD-2014/Lab_Checker
สมาชิก	นางสาวประภาพรณ กุลบุตร รหัสนักศึกษา 55010711 นายภวพล ภวปัญญากุล รหัสนักศึกษา 55010911
วิชา	การวิเคราะห์และออกแบบเชิงวัตถุ (Object Oriented Analysis and Design)
ภาคเรียนที่	1/2557

บทคัดย่อ :

Lab Checker เป็นเว็บไซต์ที่ใช้สำหรับการติดต่อสื่อสาร ระหว่างอาจารย์กับนักศึกษา โดยการใช้งานนั้น ทั้งอาจารย์และนักศึกษาจะต้องสมัครเป็นสมาชิกของเว็บไซต์ก่อน จึงจะสามารถใช้งานเว็บไซต์ได้

Lab Checker เป็นเครื่องมือที่เปรียบเสมือนกระดานเช็คชื่อของอาจารย์ แต่มีข้อแตกต่างคือ ป้องกันการสูญหาย ของข้อมูล สามารถตรวจสอบความผิดพลาดในการเช็คชื่อในแต่ละครั้งได้อีกด้วย ทั้งนี้อาจารย์มีสิทธิ์ในการเข้าถึงข้อมูลและแก้ไขข้อมูล พร้อมทั้งอัปโหลดไฟล์รายชื่อนักศึกษาในรายวิชาที่ตนรับผิดชอบได้แต่เพียงผู้เดียวและหากนักศึกษาคนใดต้องการที่จะทราบข้อมูล การเข้าเรียนหรือเข้าทำแลป จะทำได้เฉพาะวิชาที่ตนได้ลงทะเบียนเรียนไว้เท่านั้น ความสามารถอีกอย่างหนึ่งของเว็บไซต์ Lab Checker อาจารย์และนักศึกษาสามารถที่จะพูดคุยติดต่อสื่อสารกันได้ผ่าน หน้าเว็บไซต์ ในส่วนของกระดานพูดคุย เพื่อตอบปัญหาหรือข้อสงสัย หรือจะเป็นประกาศของรายวิชานั้นๆได้

เว็บไซต์ Lab Checker เป็นอีกทางเลือกหนึ่งของอาจารย์ ในการจัดการรายวิชาที่ตนเองสอน เพื่ออำนวยความสะดวก และง่ายต่อการจัดเก็บข้อมูล ป้องกันการสูญหายและการถูกทำลายได้อีกด้วย

บทนำและแรงจูงใจ :

การใช้ชีวิตประจำวันในปัจจุบันนี้ จะสังเกตเห็นได้ว่าการนำเอาเทคโนโลยีหลายๆอย่างมาประยุกต์ใช้กับชีวิตประจำวันกันอย่างแพร่หลาย เพื่ออำนวยความสะดวกในการใช้งานหลายๆอย่าง แม้กระทั่งการเรียนในชั้นเรียนเช่นกัน การเช็คชื่อเข้าชั้นเรียนนั้น จะเห็นได้ว่าการเช็คชื่อแต่ละครั้งนั้นอาจารย์บางท่านอาจจะเช็คชื่อลงในใบรายชื่อที่เป็นกระดาษ ซึ่งยุ่งยากต่อการจัดเก็บ และง่ายต่อการสูญหาย หรือถูกทำลายได้ อีกทั้งตัวนักศึกษาเอง ก็ไม่สามารถที่จะตรวจสอบได้ว่าตนนั้นได้รับการเช็คชื่อแล้วหรือไม่ ทำให้ฝ่ายใดฝ่ายหนึ่งเสียสิทธิ์ไปก็เป็นได้

ผู้จัดทำจึงเล็งเห็นว่าควรมีการนำเอาเทคโนโลยีมาประยุกต์ใช้กับการเช็คชื่อเข้าชั้นเรียนนี้ โดยทำเป็นเว็บไซต์ที่มีชื่อว่า Lab Checker โดยความสามารถของเว็บไซต์ Lab Checker มีดังต่อไปนี้

- ระบบสมัครเข้าใช้งานสำหรับอาจารย์และนักศึกษา
- นักศึกษาและอาจารย์ท่านอื่น สามารถเข้าดูและตรวจสอบความถูกต้องได้
- อาจารย์ประจำรายวิชาจะสามารถเขียนข้อมูลใหม่ หรือแก้ไขข้อมูลได้ และมีการอัปเดต

ข้อมูลตลอดเวลา

- มีการ Export ข้อมูลออกมาใน format csv
- Add announcement section + comments ไว้ประกาศ หรือสื่อสารกัน
- มี Upload รายชื่อวิชาและรายชื่อนักศึกษาในแต่ละรายวิชา

งานที่เกี่ยวข้องหรือคล้ายกับงานที่คุณจะทำในโปรเจก :

เว็บไซต์ Lab Checker มีแรงบันดาลใจมาจาก <https://www.ce.kmitl.ac.th> ซึ่งเป็นเว็บไซต์ของภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ซึ่งภายในเว็บไซต์จะประกอบด้วย หลักสูตรภาควิชา ประกาศต่างๆในสาขาวิชา และชื่อรายวิชาต่างๆ ซึ่งภายในก็จะประกอบไปด้วยไฟล์ประกอบการเรียนการสอนในวิชาเรียนนั้นๆ ประกาศรายวิชา กำหนดการเรียนการสอน เป็นต้น นอกจากนี้ยังมีข้อมูลติดต่อกับอาจารย์ในสาขาวิชาอีกด้วย

- **สิ่งที่คล้ายกับเว็บไซต์** <https://www.ce.kmitl.ac.th> มีระบบจัดการรายวิชา มีประกาศของแต่ละวิชา อาจารย์สามารถ Upload ไฟล์รายชื่อ และเอกสารประกอบการเรียนการสอนได้
- **สิ่งที่ต่างกัน** มีระบบเช็คชื่อออนไลน์ มีระบบพูดคุยกันระหว่างอาจารย์กับนักศึกษา

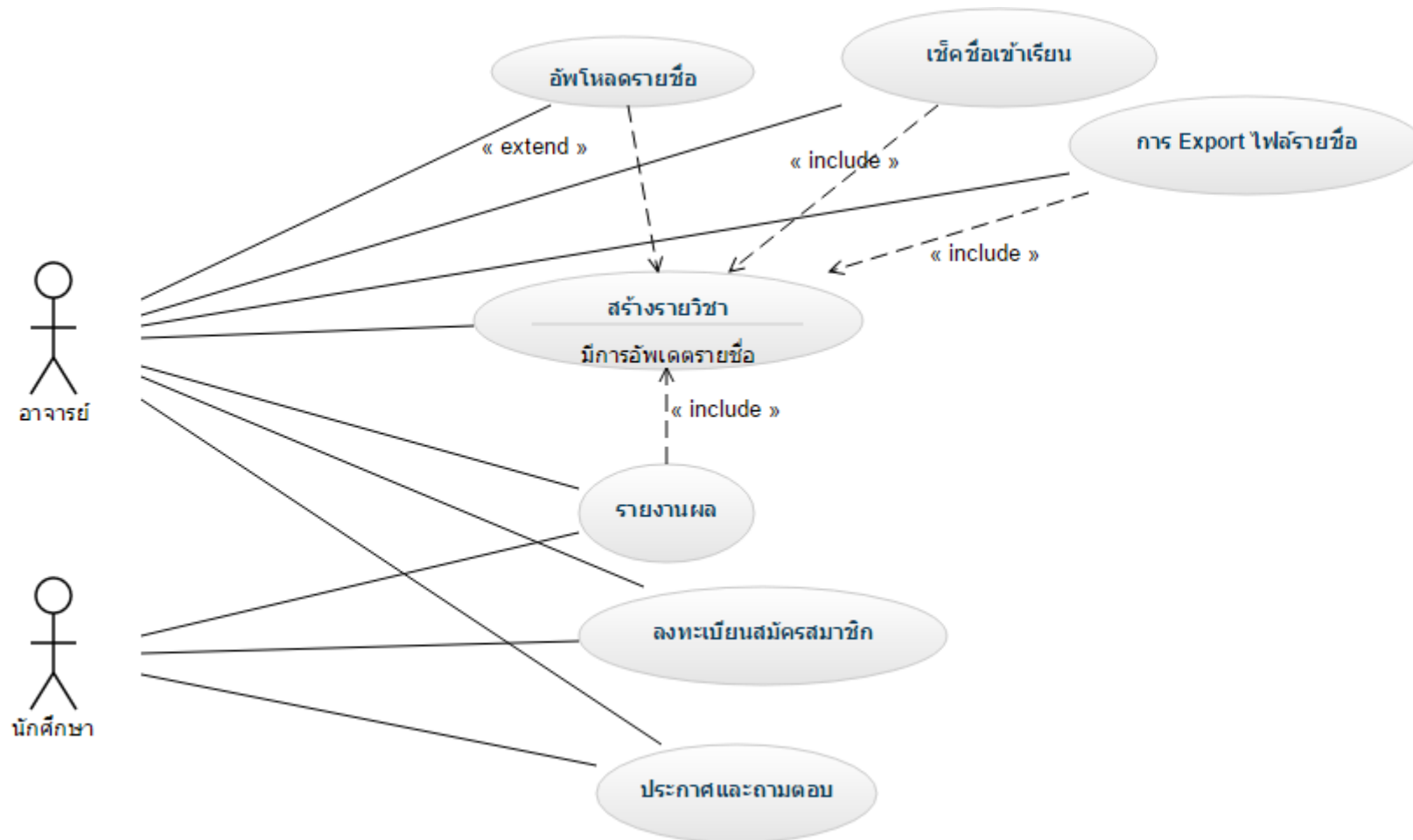
Functional Requirements:

- ลงทะเบียนสมัครสมาชิก (Register)
- การยืนยันตัวตนเข้าสู่ระบบ (Login) เพื่อเป็นการยืนยันว่าเป็นสมาชิกเว็บนี้ และเพื่อระบบว่าเป็นอาจารย์หรือนักศึกษา
- การจัดการรายวิชา (Create Class, Add Class) ใช้ในการสร้างชั้นเรียน การแก้ไขชั้นเรียน และการจัดการในชั้นเรียนเช่นการเช็คชื่อ การแจ้งข่าวสารรายวิชา
- การอัปโหลดรายชื่อ (Upload) ใช้สำหรับอัปโหลดรายชื่อนักศึกษา ซึ่งในรายชื่อจะต้องประกอบด้วยรหัสนักศึกษาและชื่อนักศึกษา เพื่อจะทำการให้สิทธิ์ในการดูประวัติการเช็คชื่อหรือการแจ้งข่าวสารรายวิชาแก่นักศึกษา โดยจะมีการเช็คว่ารหัสนักศึกษาที่ลงทะเบียนกับที่อัปโหลดมานั้นตรงกันหรือไม่หาก ไม่ตรงจะไม่ได้รับสิทธิ์ในการเข้าไปดูในรายวิชานั้นๆ
- รายงานผล (Report) เป็นหน้าเว็บที่รายงานผลการเช็คชื่อ ซึ่งในหน้านี้นี้จะสามารถดูได้เฉพาะคนที่มีสิทธิ์เข้าดูเท่านั้น
- บอร์ดประกาศและถามตอบ
- ระบบเช็คชื่อเข้าเรียนสำหรับอาจารย์ (Check)
- การ Export ไฟล์รายชื่อที่มีการเช็คแล้วหรือยังไม่เช็ค ออกมาเป็นไฟล์ .csv

Non-Functional Requirements:

- ระบบมีการอัปเดตแบบ Real time
- มีการรักษาความปลอดภัยของข้อมูล
- หน้าเว็บมีการออกแบบที่สวยงาม เหมาะสมกับการใช้งาน
- รองรับการแสดงผลทุกขนาดหน้าจอ
- สามารถเข้าระบบได้ตลอดเวลา
- เว็บไซต์มีเมนูที่เข้าใจง่าย สะดวกต่อการใช้งาน

Use Case Diagram:



ระบบเช็คชื่อเข้าเรียน (Lab Checker)

Use Case Specification (1)

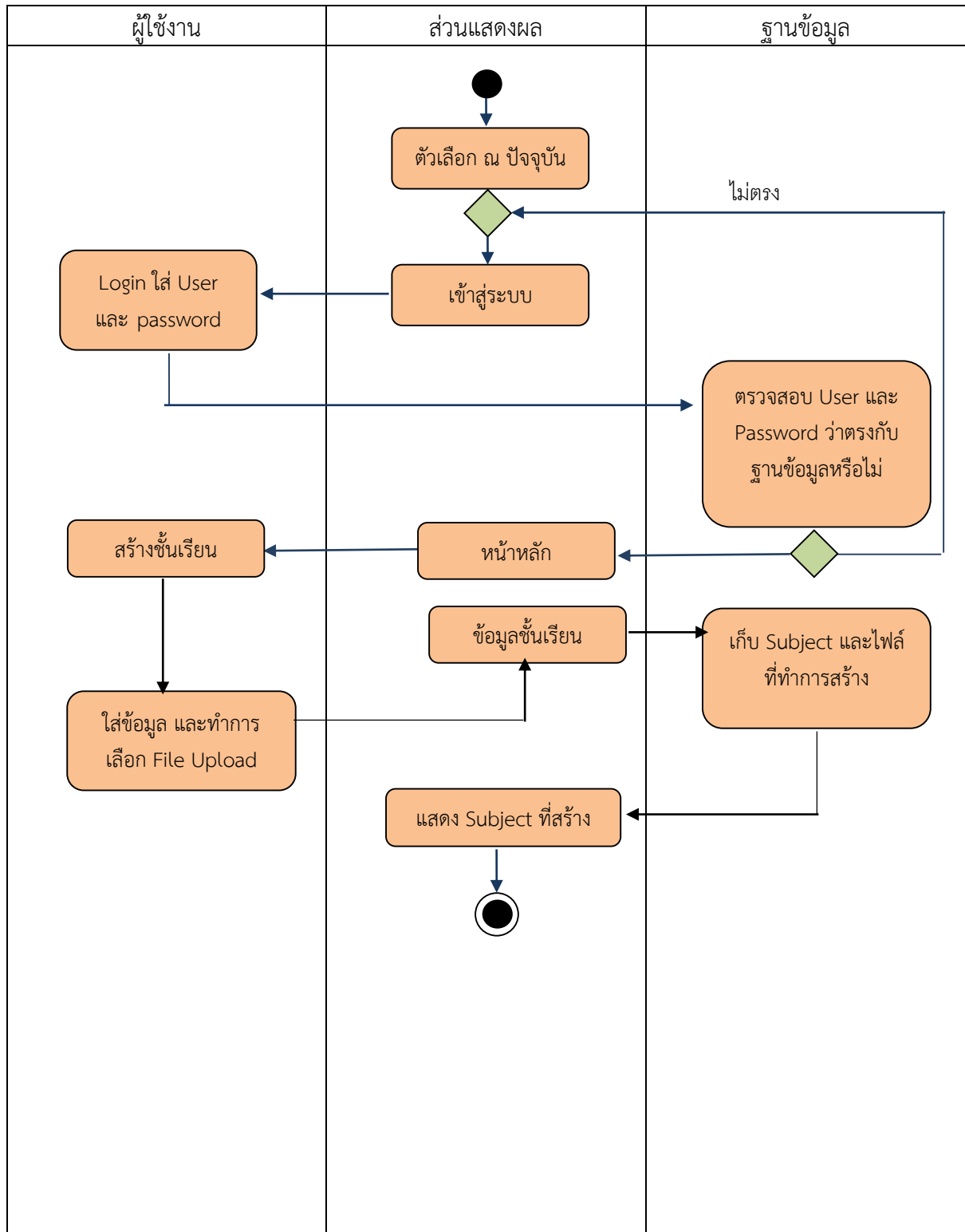
- Use Case Name:** สร้างรายวิชา
- Use Case Purpose:** มีไว้สำหรับสร้างรายวิชา เช่นการสร้างชั้นเรียน (Create Class) เกี่ยวกับรายวิชาเป็นต้น
- Actors:** อาจารย์
- Pre-conditions:** 1. ผู้ใช้บริการมีระบบอินเทอร์เน็ต
2. ผู้ใช้บริการทำการสมัครสมาชิกเรียบร้อยแล้ว
- Post-conditions:** ผู้ใช้ทำการเข้าสู่ระบบ
- Limitations:** สำหรับการสร้างชั้นเรียนจะทำได้เฉพาะอาจารย์เท่านั้น ส่วนนักเรียน และอาจารย์ที่มีสิทธิ์เข้าดูข้อมูลรายวิชานี้ จะสามารถเข้าดูข้อมูลได้เพียงอย่างเดียว ไม่สามารถแก้ไขหรือเพิ่มข้อมูลได้ (Read Only)
- Assumptions:** -
- Primary Scenario:** 1. ผู้ใช้เรียกเข้าสู่หน้าเว็บ
2. ผู้ใช้ทำการ Login เพื่อยืนยันตัวตน
3. ในกรณีที่ยังไม่มีชั้นเรียน ผู้ใช้จะต้องสร้างชั้นเรียนก่อน
4. ทำการอัปโหลดรายชื่อ
- Condition triggering an alternate scenario:** กรณีที่ผู้ใช้อัปโหลดไฟล์ชนิดอื่นที่ไม่ใช่ไฟล์ .csv
1. ระบบแจ้งเตือนว่า “กรุณาอัปโหลดไฟล์เป็นชนิด .CSV เท่านั้นค่ะ”
 2. ผู้ใช้ทำการอัปโหลดไฟล์ใหม่

Use Case Specification (2)

- Use Case Name:** Export ไฟล์รายชื่อ
- Use Case Purpose:** สำหรับอาจารย์ที่ต้องการข้อมูลการเช็คชื่อออกมา โดยระบบจะทำการ Export ออกมาเป็นไฟล์ .csv จากนั้นอาจารย์สามารถนำเอาข้อมูลนี้ไปประมวลผลข้อมูล
อย่างอื่นได้
- Actors:** อาจารย์
- Pre-conditions:**
1. ผู้ใช้บริการมีระบบอินเทอร์เน็ต
 2. ผู้ใช้บริการทำการสมัครสมาชิกเรียบร้อยแล้ว
 3. ผู้ใช้จะต้องทำการสร้างชั้นเรียนเรียบร้อยแล้ว
 4. ผู้ใช้จะต้องทำการอัปโหลดรายชื่อนักศึกษาเรียบร้อยแล้ว
- Post-conditions:** -
- Limitations:** อาจารย์เจ้าของวิชาเท่านั้นที่จะสามารถเข้าถึงได้
- Assumptions:** -
- Primary Scenario:**
1. ผู้ใช้เรียกเข้าสู่หน้าเว็บ
 2. ผู้ใช้ทำการ Login เพื่อยืนยันตัวตน
 3. เลือกรายวิชาที่ต้องการ Export
 4. เลือกเมนู “Export Data”
- Condition triggering an alternate scenario:** -

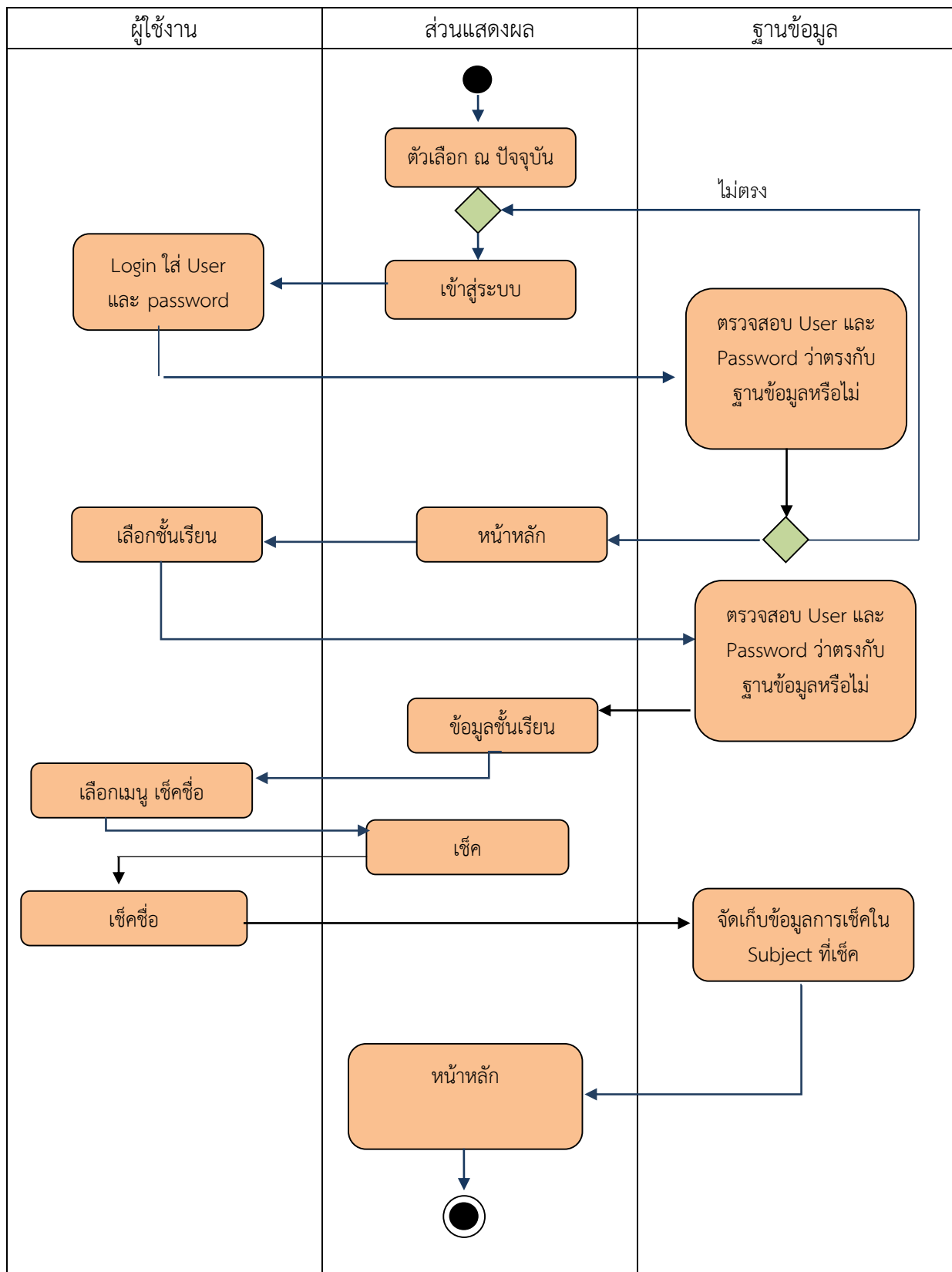
Activity Diagram

Use case : การอัปโหลดรายชื่อนักศึกษา



Activity Diagram

Use case : การเช็คชื่อนักศึกษา



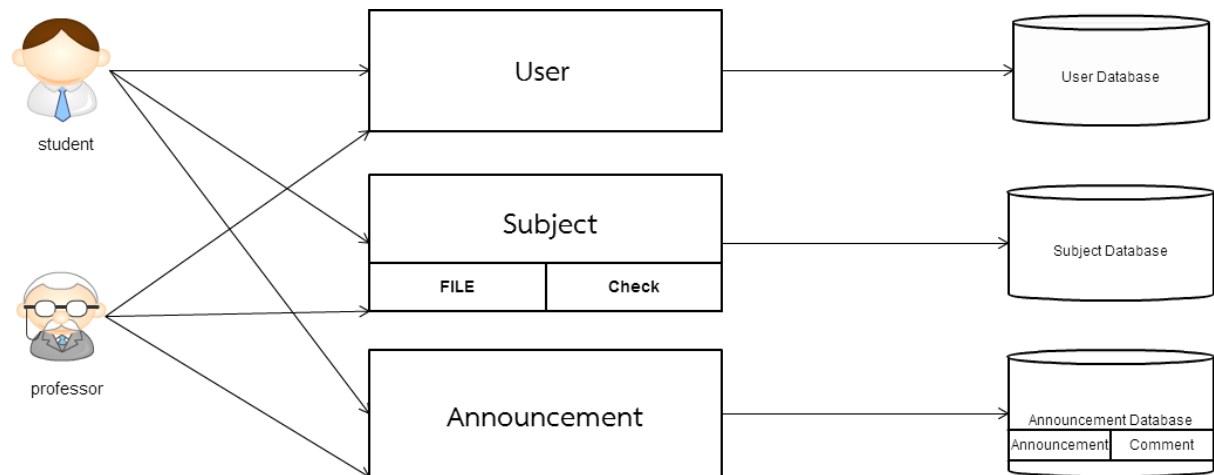
Design

■ Problem Analysis

☞ Abstraction

- **ผู้ใช้งาน** ซึ่งประกอบด้วย ชื่อ นามสกุล E-mail สถานะ(อาจารย์,นักศึกษา)
- **รายวิชา** ซึ่งประกอบด้วย ชื่อรายวิชา ชื่ออาจารย์ผู้สอน เวลาเรียน ห้องที่ใช้เรียน รายละเอียดรายวิชา ไฟล์รายวิชา
- **การ Check ชื่อ** ประกอบไปด้วย รายละเอียดการ Check และแสดงการ Check
- **การประกาศ** ประกอบไปด้วย หัวข้อการประกาศ และรายละเอียดของหัวข้อประกาศ
- **การแสดงความคิดเห็น** เป็นการแสดงความคิดเห็นในหัวข้อประกาศนั้นๆ

☞ Application Architecture



ภาพรวมของ Application Architecture

ส่วนการจัดการผู้ใช้งาน

ส่วนนี้จะประกอบด้วย การสมัครสมาชิก การเข้าสู่ระบบ และการแก้ไขข้อมูลของผู้ใช้งาน โดยส่วนจัดการผู้ใช้งานจะติดต่อกับฐานข้อมูลของผู้ใช้งาน และเชื่อมกับส่วนการจัดการรายวิชา ส่วนการประกาศ

ส่วนการจัดการรายวิชา

ส่วนนี้จะประกอบด้วย ส่วน Upload file, ส่วน Export file, ส่วนแก้ไขข้อมูล และส่วน View report โดยมีการติดต่อกับฐานข้อมูลของรายวิชา สำหรับ Permission ในการ Upload, Export และแก้ไขรายวิชา จะให้ Permission กับอาจารย์ประจำรายวิชาเท่านั้น แต่ในส่วน View Report จะให้สิทธิ์ทั้งอาจารย์และนักศึกษา

ส่วนประกาศและบอร์ดถาม-ตอบ

ส่วนนี้จะประกอบไปด้วย ส่วนการสร้างกระทู้ และส่วนการคอมเมนต์ตอบกระทู้ โดยผู้ใช้งานจะติดต่อกับฐานข้อมูลส่วนประกาศ โดยในส่วนนี้ทั้งอาจารย์ และนักศึกษา สามารถใช้งานได้

ส่วนการจัดการ Check ชื่อ

ส่วนนี้มีการติดต่อกับฐานข้อมูลผู้ใช้โดยจะให้ Permission เฉพาะอาจารย์ประจำวิชาเท่านั้น อีกทั้งยังมีการติดต่อกับฐานข้อมูลรายวิชา เพื่อจัดการในส่วนของการ Check ชื่อ

ฐานข้อมูลผู้ใช้

ประกอบด้วยชื่อผู้ใช้งาน, นามสกุล, รหัสผ่าน, E-mail, สถานะ, Student ID โดยจะเชื่อมต่อกับส่วนจัดการผู้ใช้งาน

ฐานข้อมูลรายวิชา

ในส่วนนี้ประกอบด้วย ชื่อวิชา, ชื่ออาจารย์ผู้สอน, วัน เวลาเรียนแต่ละวิชา, ห้องที่ใช้เรียน, รายละเอียดรายวิชา, File รายชื่อนักศึกษา โดยจะเชื่อมต่อกับส่วนการจัดการรายวิชา

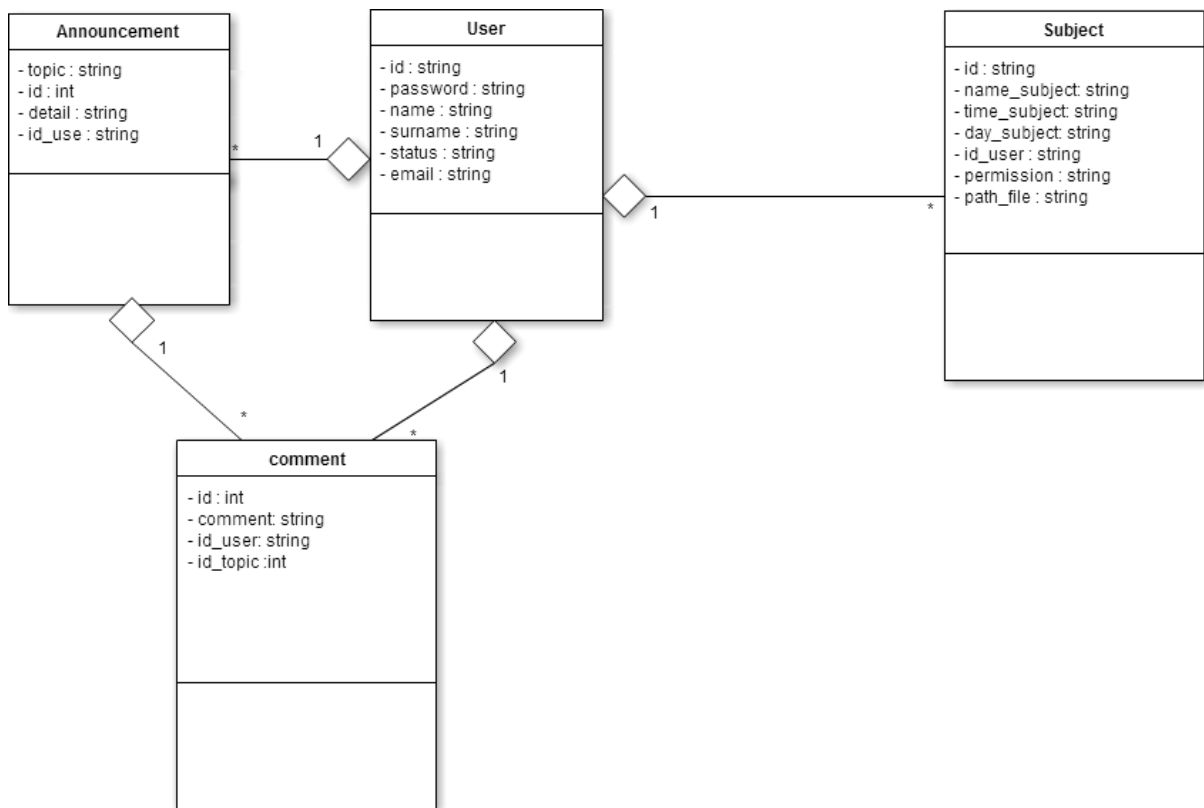
ฐานข้อมูลส่วนประกาศ

ในส่วนนี้จะประกอบไปด้วย หัวข้อกระทู้, ข้อมูลของกระทู้, Comment ในแต่ละกระทู้ โดยจะเชื่อมต่อกับส่วนประกาศและบอร์ดถาม-ตอบ

Subsystem/ Components

- ส่วนการจัดการผู้ใช้งาน ประกอบด้วยส่วนต่างๆดังต่อไปนี้
 - *การสมัครสมาชิก* โดยการสมัครสมาชิกจะต้องทำการระบุชื่อผู้ใช้งาน, รหัสผ่าน, E-mail, สถานะ, Student ID, ในส่วนนี้มีการติดต่อกับฐานข้อมูลผู้ใช้
 - *การเข้าสู่ระบบ* เป็นการยืนยันตัวตนในการเข้าใช้งานเว็บไซต์ เพื่อกำหนด Permission ในการเข้าถึงข้อมูลต่างๆในเว็บไซต์ ในส่วนนี้มีการติดต่อกับฐานข้อมูลผู้ใช้
- ส่วนการจัดการรายวิชา ประกอบด้วยส่วนต่างๆดังต่อไปนี้
 - *Create Subject* เป็นการสร้างรายวิชา เพื่อนำไปใช้ในการเช็คชื่อต่อไป
 - *Upload File* ส่วนนี้จะเป็นการ Upload File รายชื่อเพื่อนำไปเช็คชื่อในส่วนการจัดการเช็คชื่อต่อไป
 - *Export File* ส่วนนี้จะเป็นการ Export File ออกมาในรูปแบบ .csv ในกรณีที่ต้องการนำเอาไฟล์รายชื่อที่มีการเช็คแล้วไปใช้งานต่อ ในส่วนนี้มีการติดต่อกับฐานข้อมูลรายวิชา
 - *View Report* เป็นการรายงานผลการเช็ค ซึ่งส่วนนี้ทั้งนักศึกษา และอาจารย์สามารถที่จะดู Report ได้ และมีการติดต่อกับฐานข้อมูลรายวิชา
- ส่วนประกาศและบอร์ดถาม – ตอบ ประกอบด้วยส่วนต่างๆดังต่อไปนี้
 - *สร้างกระทู้* มีหน้าที่ในการสร้างกระทู้ หรือหัวข้อที่ต้องการจะประกาศ
 - *Comment* ส่วนนี้จะจัดการในส่วนที่ต้องการจะตอบกระทู้ หรือ Comment ในประกาศ
- ส่วนการจัดการ Check ชื่อ
ส่วนนี้จะเป็นส่วนที่ทำการเช็คชื่อ ตามที่มีการ Upload ไฟล์รายชื่อมา แล้วทำการเก็บข้อมูลไว้ในฐานข้อมูลรายวิชา ส่วนนี้จึงมีการติดต่อกับฐานข้อมูลรายวิชา ในส่วนนี้จะมีเฉพาะอาจารย์ประจำวิชานั้นๆเท่านั้นที่สามารถเข้าถึงได้

☞ Domain Classes



Class Diagram ของระบบ

- Class ของระบบประกอบด้วย 4 Class หลักๆ ดังนี้
 1. **User** จะเป็นส่วนที่จัดการเกี่ยวกับผู้ใช้งานทั้งหมด ทั้งฐานข้อมูลผู้ใช้งาน การให้ Permission แก่ผู้ใช้งานในการเข้าถึงส่วนต่างๆของระบบ อีกทั้งยังจัดเก็บข้อมูลผู้ใช้โดยส่งไปที่ฐานข้อมูลผู้ใช้งานอีกด้วย
 2. **Subject** ทำหน้าที่ในการจัดการรายวิชา โดยนำเอาข้อมูลจากฐานข้อมูลรายวิชามาจัดการให้ตรงกับความต้องการของผู้ใช้งาน เช่นการ Upload ไฟล์รายชื่อ สิทธิ์ในการแก้ไข หรือการสร้างชั้นเรียน รวมทั้งการดูผลการเช็คชื่อในชั้นเรียน
 3. **Announcement** ส่วนนี้จะเป็นการจัดการเกี่ยวกับบอร์ดประกาศ ทั้งการสร้างกระทู้ รวมทั้งการประกาศ จากนั้นทำการเก็บข้อมูลในฐานข้อมูลการประกาศ
 4. **comment** ส่วนนี้จะเป็นการจัดการเกี่ยวกับการแสดงความคิดเห็น จากนั้นทำการเก็บข้อมูลในฐานข้อมูลการประกาศ

Deployment

- ส่วนแสดงผล
 - ใช้ HTML, CSS, JavaScript, PHP ในการออกแบบส่วนแสดงผล
 - ใช้ Bootstrap เป็นตัวช่วยในการจัดการแสดงผล เพื่อรองรับการแสดงผลทุกขนาดหน้าจอ และ ช่วยในการออกแบบ Feature ในส่วนแสดงผลให้เว็บไซต์มีความน่าสนใจยิ่งขึ้น
 - ใช้ Laravel เป็น Framework ที่ใช้พัฒนา
- Server
 - ใช้บริการ Web site ของ Microsoft Azure
 - ใช้ VM 1 ตัว เพื่อ application server
- Database
 - ใช้ phpmyadmin เป็นตัวช่วยในการเข้าถึงการจัดการฐานข้อมูล
 - MySQL เป็นฐานข้อมูล

งานที่ทำ	ระยะเวลา	ผู้รับผิดชอบ
ศึกษาการออกแบบเว็บไซต์ โดยใช้ภาษา PHP และ HTML	19 ก.ย. 2557 – 3 ต.ค. 2557	ประภาพรณ กุลบุตร ภวพล ภวปัญญากุล
ศึกษา Laravel เพื่อใช้พัฒนาเว็บไซต์	3 ต.ค. 2557 – 27 ต.ค. 2557	ภวพล ภวปัญญากุล
ออกแบบส่วนแสดงผล และทดลองใช้ Bootstrap ในการรองรับการแสดงผลทุกหน้าจอ	13 ต.ค. 2557 – 17 ต.ค. 2557	ประภาพรณ กุลบุตร
ออกแบบและสร้างส่วนแสดงผล	17 ต.ค. 2557 – 10 พ.ย. 2557	ประภาพรณ กุลบุตร
สร้างและออกแบบฐานข้อมูล	17 ต.ค. 2557 – 27 ต.ค. 2557	ภวพล ภวปัญญากุล
เขียนส่วนจัดการผู้ใช้งาน	17 ต.ค. 2557 – 27 ต.ค. 2557	ภวพล ภวปัญญากุล
เขียนส่วนจัดการรายวิชา	28 ต.ค. 2557 – 31 ต.ค. 2557	ภวพล ภวปัญญากุล
เขียนส่วนการประกาศ	4 พ.ย. 2557 – 6 พ.ย. 2557	ภวพล ภวปัญญากุล
เขียนส่วนจัดการการ check ชื่อ	7 พ.ย. 2557 – 8 พ.ย. 2557	ประภาพรณ กุลบุตร
เขียนส่วนจัดการการให้ permission	8 พ.ย. 2557 – 10 พ.ย. 2557	ประภาพรณ กุลบุตร
ทำการ Test แต่ละส่วนของระบบ	10 พ.ย. 2557 – 14 พ.ย. 2557	ประภาพรณ กุลบุตร ภวพล ภวปัญญากุล

Test

1. ทดลองแก้ไขข้อมูลรายวิชา

/var/www/html/laravel/app/tests/ AddAnnouncementTest.php

2. ทดลองสร้างประกาศ

/var/www/html/laravel/app/tests/ EditSubjectTest.php

Evaluation

1. ทดลองแก้ไขข้อมูลรายวิชา

จุดประสงค์การทดลอง

- อาจารย์สามารถแก้ไขข้อมูลรายวิชาของตนเองได้ เพื่อแก้ไขข้อมูลของรายวิชาที่เกิดขึ้นจากความผิดพลาดในการสร้างรายวิชา หรือมีการเปลี่ยนแปลงคำอธิบายรายวิชา

สิ่งที่จะวัด

- รายละเอียดรายวิชามีการเปลี่ยนแปลงไปตามที่ได้ทำการแก้ไข

สิ่งที่ใช้ในการทดลอง

1. วิชา 1 วิชาสำหรับการทดลอง
2. รหัสสำหรับเข้าใช้งานของอาจารย์
3. Browser สำหรับเข้าใช้งานระบบ

วิธีการทดลอง

1. เข้าสู่ระบบในสถานะอาจารย์
2. เลือกเมนูชั้นเรียน ในแถบเมนูซ้ายมือ
3. เลือกรายวิชาจากกล่อง รายวิชาทั้งหมด
4. จากนั้นเลือกเมนู รายละเอียดรายวิชา
5. เลือกเมนู Edit
6. แก้ไขข้อมูลรายวิชาตามที่ต้องการ
7. กด submit

สิ่งที่ได้จากการทดลอง

ข้อมูลรายวิชานั้นๆถูกแก้ไขได้ถูกต้อง ตามที่ผู้ใช้งานได้ทำการแก้ไข

ผลการทดลอง

```
web@ubuntu:/var/www/html/laravel$ phpunit
PHPUnit 3.7.28 by Sebastian Bergmann.

Configuration read from /var/www/html/laravel/phpunit.xml

.

Time: 106 ms, Memory: 10.50Mb

OK (1 test, 6 assertions)
web@ubuntu:/var/www/html/laravel$
```

2. ทดลองสร้างประกาศ

จุดประสงค์การทดลอง

- สร้างประกาศขึ้นมา 1 หัวข้อ

สิ่งที่คาดว่าจะได้

- ผู้ใช้ทุกคนสามารถเข้าถึงประกาศนั้นได้

สิ่งที่ใช้ในการทดลอง

1. หัวข้อในการประกาศ
2. รหัสผู้ใช้งาน 1 รหัส
3. รายละเอียดในการประกาศ

วิธีการทดลอง

1. เข้าสู่ระบบ
2. เลือกเมนูบอร์ดประกาศ ถามตอบ ในแถบเมนูซ้ายมือ
3. เลือกเมนู New Topic
4. กรอกหัวข้อและรายละเอียด
5. กดยืนยัน
6. เลือกไปยังหัวข้อที่สร้างไว้
7. ตรวจสอบว่ามีรายละเอียดครบตามที่สร้างหรือไม่

สิ่งที่ได้จากการทดลอง

สามารถสร้างประกาศ และผู้ใช้งานสามารถเห็นได้ทุกคน

ผลการทดลอง

```
web@ubuntu:/var/www/html/laravel$ phpunit
PHPUnit 3.7.28 by Sebastian Bergmann.

Configuration read from /var/www/html/laravel/phpunit.xml

.

Time: 66 ms, Memory: 9.25Mb

OK (1 test, 5 assertions)
web@ubuntu:/var/www/html/laravel$
```

บทสรุป

เว็บไซต์ Lab Checker จัดทำขึ้นมามีวัตถุประสงค์เพื่อเป็นหนึ่งในตัวช่วยในการจัดการเรียนสอน เพื่อให้เกิดความสะดวกต่อการใช้งาน และนำเอาเทคโนโลยีในปัจจุบันมาประยุกต์ให้เข้ากับชีวิตประจำวัน อีกทั้งยังได้นำความรู้ที่ได้ศึกษาในวิชา Object Oriented Analysis and Design มาประยุกต์ใช้กับเว็บไซต์อีกด้วย

เว็บไซต์นี้มีแรงจูงใจมาจากการเข้าเรียนในแต่ละครั้งของผู้จัดทำ โดยจะสังเกตได้ว่าอาจารย์บางท่านมีการเช็คชื่อลงกระดาน ซึ่งมีความยุ่งยากในการจัดเก็บ และง่ายต่อการสูญหาย อีกทั้งหากมีการนำไปใช้ในการคำนวณคะแนน จะต้องมีการกรอกลงในระบบคอมพิวเตอร์อีก ซึ่งเป็นการยุ่งยากและเสียเวลาในการจัดทำ ทั้งนี้อีกหนึ่งแรงจูงใจคือ www.ce.kmitl.ac.th เว็บไซต์ของภาควิชา วิศวกรรมคอมพิวเตอร์ คณะ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่มีการจัดการรายวิชาต่างๆของ ภาควิชาไว้ในเว็บไซต์แห่งนี้ เช่น การประกาศข่าวสารต่างๆของภาควิชา การจัดการการเรียนการสอนในแต่ละ วิชา อัพโหลดเอกสารการเรียน เป็นต้น

สิ่งที่ผู้จัดทำได้รับจากการทำเว็บไซต์นี้ขึ้นมา คือประสบการณ์ใหม่ๆ ที่เป็นการนำเอาความรู้ใน ห้องเรียนมาประยุกต์ใช้กับชิ้นงาน ซึ่งชิ้นงานนี้สามารถที่จะนำไปประกอบอาชีพได้จริงในอนาคต และได้มี การศึกษาหาความรู้ด้วยตนเองในสิ่งที่ยังไม่เคยได้ศึกษามาก่อนหน้านี้ เช่นเรื่อง เว็บแอปพลิเคชัน การใช้งาน ฐานข้อมูล การใช้ใช้ Laravel เป็น Framework ที่ใช้พัฒนา เป็นต้น ซึ่งสิ่งที่ได้กล่าวมาข้างต้นนี้ ในช่วงแรกถือ ว่าเป็นเรื่องที่ยากและไกลตัวมาก และด้วยเวลาที่กระชั้นชิด ผู้จัดทำได้พยายามจนสุดท้าย ออกมาเป็นรูปร่าง ของเว็บไซต์ และสิ่งที่ได้จากการทำเว็บไซต์นี้ที่สำคัญมากๆอีกสิ่งหนึ่งคือ การทำงานเป็นทีม การคิด การ วางแผน การพัฒนา และได้เรียนรู้ว่ากว่าจะมาเว็บไซต์หนึ่ง นั้นต้องผ่านกระบวนการหลายขั้นตอน

ผู้จัดทำหวังว่า เว็บไซต์ Lab Checker จะเป็นอีกเครื่องมือหนึ่งที่ช่วยให้ การจัดการการเรียนการสอน ในชั้นเรียนนั้นง่ายขึ้น และมีความทันสมัยมากขึ้น และเป็นอีกหนึ่งช่องทางในการติดต่อสื่อสารกันระหว่าง อาจารย์กับนักศึกษา ช่วยประกาศข่าวสารในชั้นเรียนอีกช่องทางหนึ่ง อีกทั้งนักศึกษายังสามารถตรวจสอบผล การเข้าเรียนของตนเองได้ตลอดเวลา และสามารถทักท้วงอาจารย์ผู้สอนได้ หากข้อมูลมีความผิดพลาด และ หากผู้ปกครองต้องการทราบว่านักเรียนนั้นได้เข้าเรียนในแต่ละสัปดาห์หรือไม่ สามารถนำผลที่ได้ไปให้ ผู้ปกครองดูได้ เพื่อเป็นการยืนยันว่าตนได้เข้าเรียนในแต่ละสัปดาห์หรือไม่ สุดท้ายนี้หากเว็บไซต์นี้มีผิดพลาด ประการใด ต้องขออภัยไว้ ณ ที่นี้ด้วย

บรรณานุกรม

ส่วนแสดงผล

<http://bootstrapzero.com/>
getbootstrap.com
<http://bootswatch.com/>
startbootstrap.com
<http://www.codetukyang.com/>
<http://www.codetukyang.com/>

ส่วนระบบหลังบ้าน

<http://php.net/>
laravel.com
forums.phpfreaks.com/topic/279106-uploading-csv-files/
<https://ellislab.com/codeigniter/user-guide/>