

โครงการ

เรื่อง

ระบบจัดการการเรียน (LMS)

GitHub repository

CE-KMITL-OOAD-2014/Responsive\_LMS

จัดทำโดย

กลุ่ม TiefertNut

- |                         |               |             |
|-------------------------|---------------|-------------|
| 1. นาย วิษุวัตติ ชันเฮม | รหัส 55011163 | ชั้นปีที่ 3 |
| 2. นาย อรรถกร เทพสิทธิ์ | รหัส 55011432 | ชั้นปีที่ 3 |

เสนอ

ดร.อรรถ

สังข์เพชร

ดร.อภิญญา

สังข์เพชร

โครงการนี้เป็นส่วนหนึ่งของวิชา Object Oriented Analysis and Design

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

คณะวิศวกรรมศาสตร์ สาขาคอมพิวเตอร์ ชั้นปีที่ 3

ภาคเรียนที่ 1 ปีการศึกษา 2557

## บทคัดย่อ

เนื่องจากการศึกษาในระดับอุดมศึกษามีปฏิสัมพันธ์ระหว่างอาจารย์และนักเรียนมีน้อยกว่าในระดับมัธยมศึกษาทำให้เกิดปัญหาทั้งด้านการเรียนและการทำงานที่นักศึกษาได้รับมอบหมาย เพราะอาจารย์ไม่สามารถติดต่อกับนักศึกษาได้ครบทุกคน ขณะเดียวกันนักศึกษา也不能够เข้าพบอาจารย์เพื่อปรึกษาได้อย่างสะดวกมากนัก รวมถึงอาจารย์ไม่สามารถทราบได้ว่านักศึกษาเข้าใจเนื้อหาที่สอนหรือไม่สำหรับพิจารณาการสอนในเนื้อหาถัดไป ซึ่งเกิดจากปัจจัยหลายอย่างเช่น เวลา สถานที่ โอกาส ทำให้กลุ่มผมต้องการจัดทำระบบจัดการการเรียน(LMS) ที่จะช่วยให้อาจารย์และนักศึกษาสามารถติดต่อสื่อสารกันได้ง่ายขึ้น เพื่ออำนวยความสะดวกทั้งด้านการเรียน การสั่งงาน การส่งงาน การขอหรือให้คำปรึกษา การติดตามความเข้าใจของผู้เรียนในระหว่างการเรียนการสอน เป็นต้น

## บทนำและรายละเอียดการวิเคราะห์หัวข้อ

1. ปัญหาด้านการติดต่อสื่อสารระหว่างอาจารย์และนักศึกษา เนื่องจากปัจจัยหลายๆอย่างทำให้การขอคำปรึกษาหรือการสอบถามข้อสงสัยจากอาจารย์และนักศึกษามีความยากลำบาก เช่น นักศึกษามีข้อสงสัยเกี่ยวกับการบ้านที่ได้รับมอบหมาย แต่ไม่สามารถไปพบกับอาจารย์ที่ห้องพักอาจารย์ ตามวันและเวลาที่อาจารย์กำหนดได้ หรือ บางครั้งอาจารย์อาจต้องการชี้แนะแนวทางการทำการบ้านให้นักศึกษา แต่ไม่ทราบว่าจะติดต่อกับนักศึกษาทุกคนได้อย่างไร

2. ปัญหาขณะกำลังมีการเรียนการสอน นักศึกษาอาจมีข้อสงสัยในเนื้อหาที่กำลังเรียน แต่กลัวว่าจะขัดจังหวะการสอน ดังนั้นระบบนี้จะช่วยให้นักศึกษาสามารถส่งข้อความไปถามอาจารย์ได้เลย โดยไม่ต้องกลัวว่าจะรบกวนนักศึกษาคนอื่น

3. ปัญหาเรื่องการส่งงาน อาจารย์อาจกำหนดให้นักศึกษาส่งงานในวันและภายในเวลาที่กำหนด อาจเป็นปัญหาสำหรับนักศึกษาที่ไม่มีการเรียนการสอนในวันนั้นๆ แต่ต้องเดินทางจากบ้านเพื่อมาส่งงาน ดังนั้นระบบนี้จะช่วยให้นักศึกษาสามารถอัปโหลดงานส่งอาจารย์ได้จากที่บ้านได้ทันที อีกทั้งยังช่วยให้นักศึกษาไม่ต้องพิมพ์เอกสารอีกด้วย

4. ปัญหาเรื่องความเหมาะสมของงานที่ได้รับมอบหมาย ในบางครั้งอาจารย์อาจสั่งงานที่ไม่มีความเหมาะสมกับนักศึกษา อาจเกิดจากปัจจัยทั้งด้านเวลา คุณภาพงาน ปริมาณงาน เช่น งานที่ได้รับมอบหมายอาจครอบคลุมเนื้อหาที่นักศึกษายังไม่เคยเรียนมาก่อน แต่นักศึกษาได้รับความกดดันจากเรื่องเวลาทำให้ไม่สามารถศึกษาในเนื้อหาที่ไม่ทราบได้ ดังนั้นระบบนี้จะช่วยให้นักศึกษาสามารถประเมินงานหรือการสอนในแต่ละครั้งได้ เพื่อให้อาจารย์สามารถปรับหรือทำความเข้าใจกับนักศึกษาได้ง่ายยิ่งขึ้น

5. ปัญหาด้านภาษาและความซับซ้อนในระบบ LMS ในปัจจุบัน จะเห็นได้ว่าระบบ LMS ในปัจจุบันส่วนมากจะเป็นระบบภาษาอังกฤษและค่อนข้างมีความซับซ้อนให้การใช้งานระบบ ทำให้ไม่เป็นที่นิยมมากนัก ดังนั้นระบบนี้จะทำให้การใช้งานระบบจัดการการเรียนมีความสะดวกและใช้งานได้ง่ายขึ้น เพื่อให้บุคคลที่ไม่มีทักษะด้านภาษาและคอมพิวเตอร์สามารถใช้ระบบได้อย่างมีประสิทธิภาพมากที่สุด

## งานที่เกี่ยวข้อง

ระบบที่คล้ายกับกลุ่มผมคือ canvas ซึ่งเป็นระบบ LMS ซึ่งคล้ายกันที่ระบบสามารถ ดู/อัปโหลดเอกสาร ดู/ส่งงาน นักศึกษา ตรวจงานและให้คะแนน ขอ/ให้คำปรึกษา แต่จะมีความต่างกับระบบที่กลุ่มผมจะทำคือ

1. รูปแบบของระบบจะเป็นภาษาไทยเพื่อให้สะดวกและง่ายในการใช้งานสำหรับบุคคลทั่วไปที่ไม่จำเป็นต้องมีทักษะทางด้านคอมพิวเตอร์หรือภาษาอังกฤษมากนัก
2. ระบบจะทำงานออกแบบให้นักศึกษาแต่ละคนอยู่ในรายวิชาที่ลงทะเบียนไว้ทุกวิชา ทำให้สามารถรู้วัน เวลา ห้องเรียน ของ นักศึกษาทุกคนที่ลงทะเบียนในวิชานั้นๆ เพื่อง่ายสำหรับอาจารย์หากต้องการย้ายเวลาเรียน
3. ระบบส่งใบลาและอนุมัติใบลาเพื่อง่ายในการลาของนักศึกษาเมื่อเกิดเหตุจำเป็น อีกทั้งมีสถานะบอกว่าใบลาที่ส่งไป อาจารย์ รับทราบและอนุมัติหรือไม่
4. มีการแสดงสถานะขณะเรียน เช่น หากนักศึกษาไม่เข้าใจเนื้อหาที่อาจารย์กำลังสอนหรือตามไม่ทัน นักศึกษาสามารถแสดง สถานะเพื่อบอกอาจารย์ได้ เพราะสังคมไทยในปัจจุบัน นักเรียนนักศึกษามีความกล้าแสดงออกที่น้อยมาก หลายคนไม่กล้าที่จะยกมือถามเพราะกลัวเพื่อนจะล้อ หรือจะเป็นการทำให้ตนเองเป็นจุดสนใจมากเกินไป กลุ่มผมจึงมีความคิดว่าระบบนี้จะ สามารถช่วยลดปัญหาด้านนี้ได้
5. ระบบประเมินการสอนในแต่ละครั้ง จะเป็นระบบที่ช่วยให้นักศึกษาสามารถให้คำแนะนำหรืออธิบายเหตุผลเกี่ยวกับการ สอนครั้งนั้นๆได้

## ผลการวิเคราะห์ความต้องการของผู้ใช้ระบบ

### Functional requirements

- อาจารย์สามารถจัดการระบบห้องเรียนได้ (เปลี่ยนเวลาเรียน)
- อาจารย์สามารถให้คะแนนงานของนักศึกษาได้
- ระบบสามารถจัดการการติดต่อระหว่างอาจารย์และนักศึกษาได้ (ส่งข้อความ ส่งงาน ส่งงาน)
- นักศึกษาสามารถประเมินการเรียนการสอนในแต่ละครั้งได้
- อาจารย์สามารถดูสถานะภาพรวมที่นักศึกษาเพิ่มในชั้นเรียนได้

### Non-Functional requirements

- ระบบสามารถใช้งานได้พร้อมกันไม่ต่ำกว่า 10 คน
- ระบบมีการป้องกันการเข้าถึง URL โดยตรง โดยที่ผู้ใช้ไม่มีสิทธิ์ใช้งานหน้าเว็บนั้นๆ
- ระบบมีการเก็บรหัสผ่านการใช้งานระบบในรูปแบบที่เข้ารหัสแล้ว
- ระบบมีการแจ้งเตือนด้วยเครื่องหมายแจ้งเตือนแสดงสถานะ
- ส่วนต่างๆภายในเว็บมีการจัดเรียงตามขนาดหน้าต่างเว็บเบราว์เซอร์

## Use case diagram

## Use case specifications

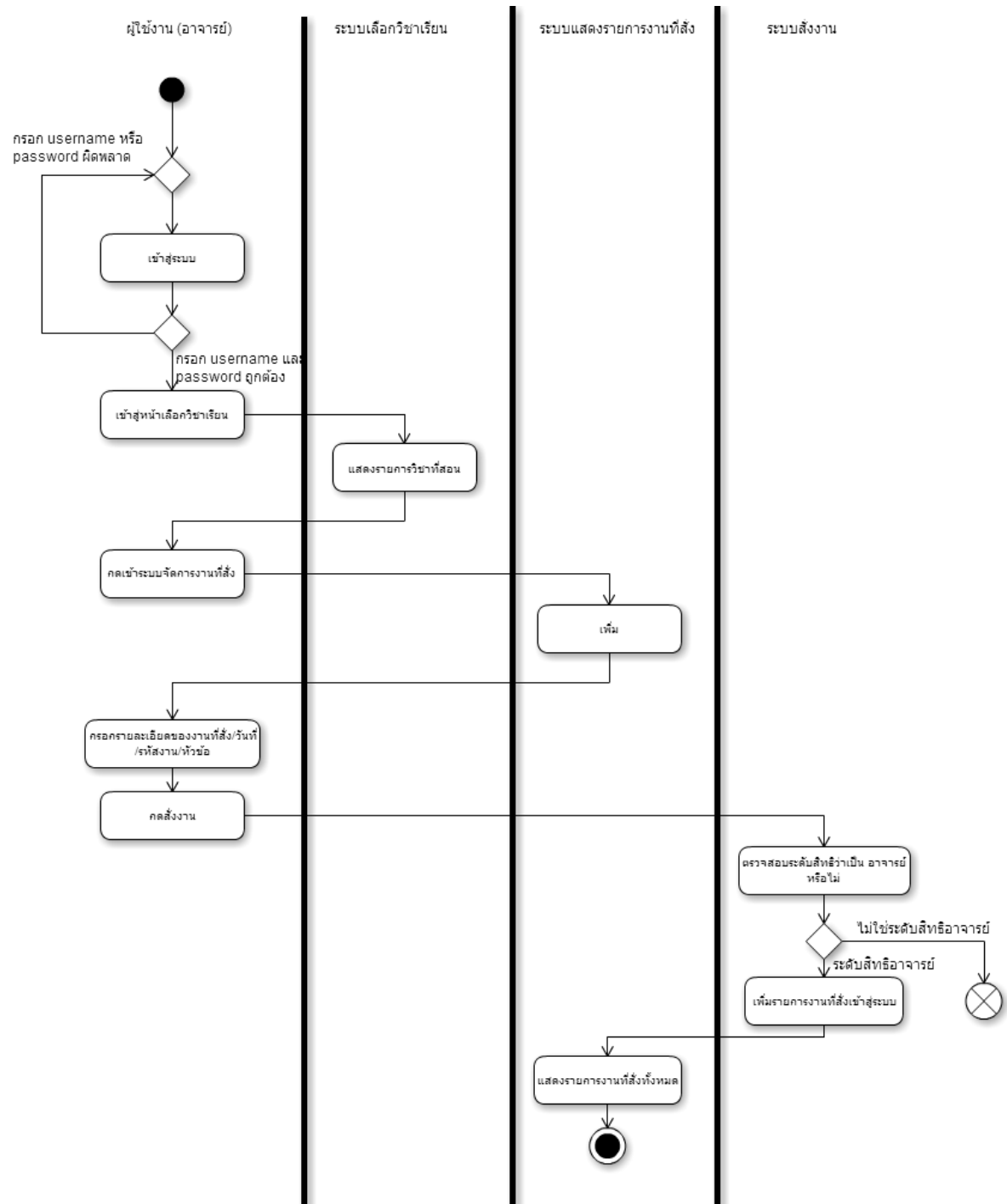
<b>Use case name</b>	ดูงานที่สั่ง
<b>Use case purpose :</b>	use case นี้สามารถดูงานทั้งหมดที่อาจารย์สั่งได้ โดยดูได้ทั้งงานที่ส่งแล้ว งานที่ยังไม่ได้ส่ง และสามารถค้นหาตามที่ต้องการเช่น ค้นหาตามวันที่ ค้นหาตามชื่องาน เพื่อให้ง่ายและสะดวกในการหางานที่ต้องทำส่ง
<b>Preconditions</b>	ต้องเข้าสู่ระบบด้วยสิทธิของอาจารย์หรือนักศึกษา
<b>Postconditions</b>	แสดงงานทั้งหมดที่อาจารย์สั่ง และมีการบอกสถานะของงานนั้นๆรวมทั้งรายละเอียด
<b>Limitations</b>	ระบบนี้มุ่งเน้นในการดูงานที่สั่ง จึงจะแจ้งให้งานที่สั่งล่าสุดแสดงก่อนเสมอ
<b>Assumptions</b>	ผู้ใช้เข้าสู่ระบบแล้ว ผู้ใช้เลือกวิชาที่ต้องการสั่งงานแล้ว
<b>Primary Scenario</b>	A. เข้าสู่ระบบด้วย Username Password และกรอก Captcha B. เลือกวิชาที่ต้องการดูงานที่สั่ง C. เข้าที่ระบบจัดการงาน D. เลือกดูงานที่สั่ง E. จบการทำงาน
<b>Alternate Scenario</b>	เงื่อนไขที่ 1 : กรอก Username ผิดพลาด A1. กรณีที่ใส่ Username ผิดพลาด ระบบจะไปยังหน้าที่แสดงข้อความ “Username นี้ไม่มีอยู่ในระบบ กรุณากรอกใหม่อีกครั้ง” A2. หลังจากนั้นระบบจะทำการเปลี่ยนไปหน้าเข้าสู่ระบบโดยอัตโนมัติ A3. กลับไป Primary Scenario ข้อ A เงื่อนไขที่ 2 : กรอก Password ผิดพลาด A1. กรณีที่ใส่ Password ผิดพลาด ระบบจะไปยังหน้าที่แสดงข้อความ “คุณกรอก Password ผิดพลาด กรุณากรอกใหม่อีกครั้ง” A2. หลังจากนั้นระบบจะทำการเปลี่ยนไปหน้าเข้าสู่ระบบโดยอัตโนมัติ A3. กลับไป Primary Scenario ข้อ A เงื่อนไขที่ 3 : กรอก Captcha ผิดพลาด A1. กรณีที่ใส่ Captcha ผิดพลาด ระบบจะไปยังหน้าที่แสดงข้อความ “คุณกรอก Captcha ผิดพลาด กรุณากรอกใหม่อีกครั้ง ” A2. หลังจากนั้นระบบจะทำการเปลี่ยนไปหน้าเข้าสู่ระบบโดยอัตโนมัติ A3. กลับไป Primary Scenario ข้อ A

Use case name	ส่งงาน
Use case purpose :	use case นี้ นักศึกษาสามารถส่งงานที่อาจารย์สั่ง ด้วยการแนบเอกสาร ไฟล์รูปภาพ หรือ ข้อความได้
Preconditions	ต้องเข้าสู่ระบบด้วยสิทธิของนักศึกษาและมีรายการงานที่ส่งจากอาจารย์
Postconditions	ส่งงานให้อาจารย์ที่สั่งงานในวิชานั้นๆได้
Limitations	เนื่องจากเป็นงานที่นักศึกษาต้องส่งให้อาจารย์ซึ่งอาจมีการให้คะแนนในงานชิ้นนั้นๆ จึงไม่สามารถใช้ไอดีของเพื่อนส่งงานได้ จำเป็นต้องเข้าสู่ระบบด้วยไอดีของนักศึกษาที่ต้องการส่งงานนั้นๆได้เท่านั้นเพื่อง่ายในการให้คะแนนของอาจารย์ ระบบไม่สามารถแก้ไขงานที่ส่งไปแล้วได้ ต้องส่งใหม่อีกครั้ง
Assumptions	ผู้ใช้เข้าสู่ระบบแล้ว ผู้ใช้เลือกวิชาที่ต้องการส่งงานแล้ว
Primary Scenario	A. เข้าสู่ระบบด้วย Username Password และกรอก Captcha B. เลือกวิชาที่ต้องการส่งงาน C. เข้าที่ระบบจัดการงาน D. เลือกส่งงาน ระบบจะแสดงหน้าส่งงานให้กรอกรายละเอียด E. เลือกงานที่ต้องการส่ง F. ใส่รายละเอียดต่างๆตามที่อาจารย์กำหนด และแนบไฟล์งาน รูปภาพ หรือข้อความที่ต้องการส่ง G. กดส่งงาน H. จบการทำงาน
Alternate Scenario	เงื่อนไขที่ 1 : กรอก Username ผิดพลาด A1. กรณีที่ใส่ Username ผิดพลาด ระบบจะไปยังหน้าที่แสดงข้อความ “Username นี้ไม่มีอยู่ในระบบ กรุณากรอกใหม่อีกครั้ง” A2. หลังจากนั้นระบบจะทำการเปลี่ยนไปหน้าเข้าสู่ระบบโดยอัตโนมัติ A3. กลับไป Primary Scenario ข้อ A เงื่อนไขที่ 2 : กรอก Password ผิดพลาด A1. กรณีที่ใส่ Password ผิดพลาด ระบบจะไปยังหน้าที่แสดงข้อความ “คุณกรอก Password ผิดพลาด กรุณากรอกใหม่อีกครั้ง” A2. หลังจากนั้นระบบจะทำการเปลี่ยนไปหน้าเข้าสู่ระบบโดยอัตโนมัติ A3. กลับไป Primary Scenario ข้อ A เงื่อนไขที่ 3 : กรอก Captcha ผิดพลาด A1. กรณีที่ใส่ Captcha ผิดพลาด ระบบจะไปยังหน้าที่แสดงข้อความ “คุณกรอก Captcha ผิดพลาด กรุณากรอกใหม่อีกครั้ง ” A2. หลังจากนั้นระบบจะทำการเปลี่ยนไปหน้าเข้าสู่ระบบโดยอัตโนมัติ A3. กลับไป Primary Scenario ข้อ A เงื่อนไขที่ 4 : กรณีต้องการส่งงานโดยส่งจากหน้าดูงานที่สั่ง C1. เลือกดูงานที่สั่ง C2. เลือกงานที่ต้องการส่งงาน

	<p>C3. เลือกลงงาน</p> <p>C4. กลับไป Primary Scenario ข้อ D</p> <p>เงื่อนไขที่ 5 : กรณีใส่รายละเอียดต่างๆตามที่อาจารย์กำหนดไม่ครบ</p> <p>G1. เมื่อกดส่งงานระบบจะขึ้นเตือนผู้ใช้งานระบบ</p> <p>“ท่านใส่รายละเอียดไม่ครบตามที่อาจารย์กำหนด กรุณาใส่ข้อมูลให้ครบ”</p> <p>G2. กลับไป Primary Scenario ข้อ F</p> <p>เงื่อนไขที่ 6 : กรณีเคยส่งงานไปแล้วอย่างน้อย1ครั้ง</p> <p>G1. เมื่อกดส่งงานระบบจะขึ้นเตือนผู้ใช้งานระบบ</p> <p>“เป็นงานที่เคยส่งแล้ว ท่านต้องการส่งใหม่หรือไม่”</p> <p>G2. ถ้ากดยืนยันกลับไป Primary Scenario ข้อ H</p>
--	---

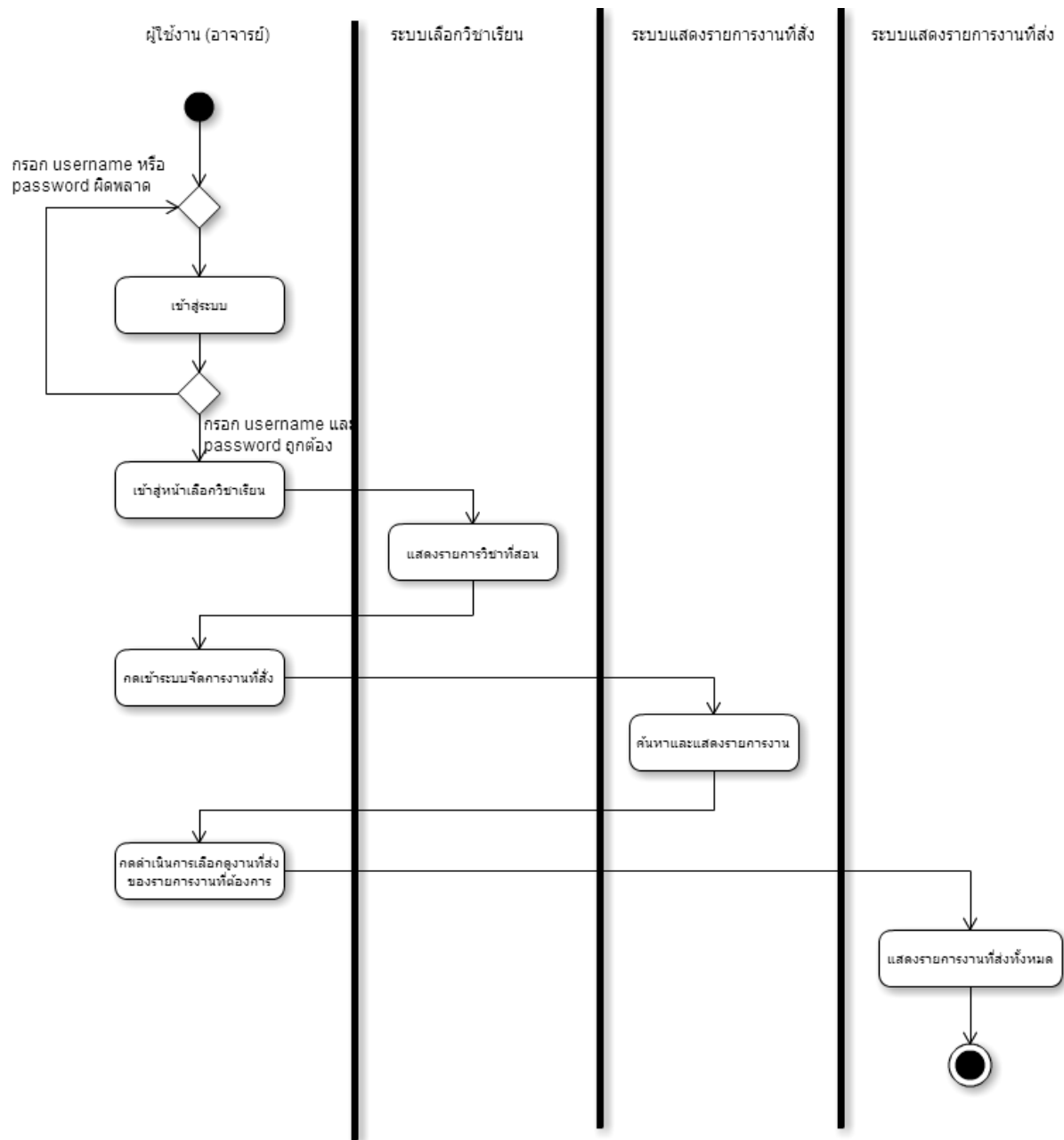
## Activity diagrams

### Use case สิ้นงาน





## Use case ดูงานที่ส่งงาน



## สถาปัตยกรรมของระบบ

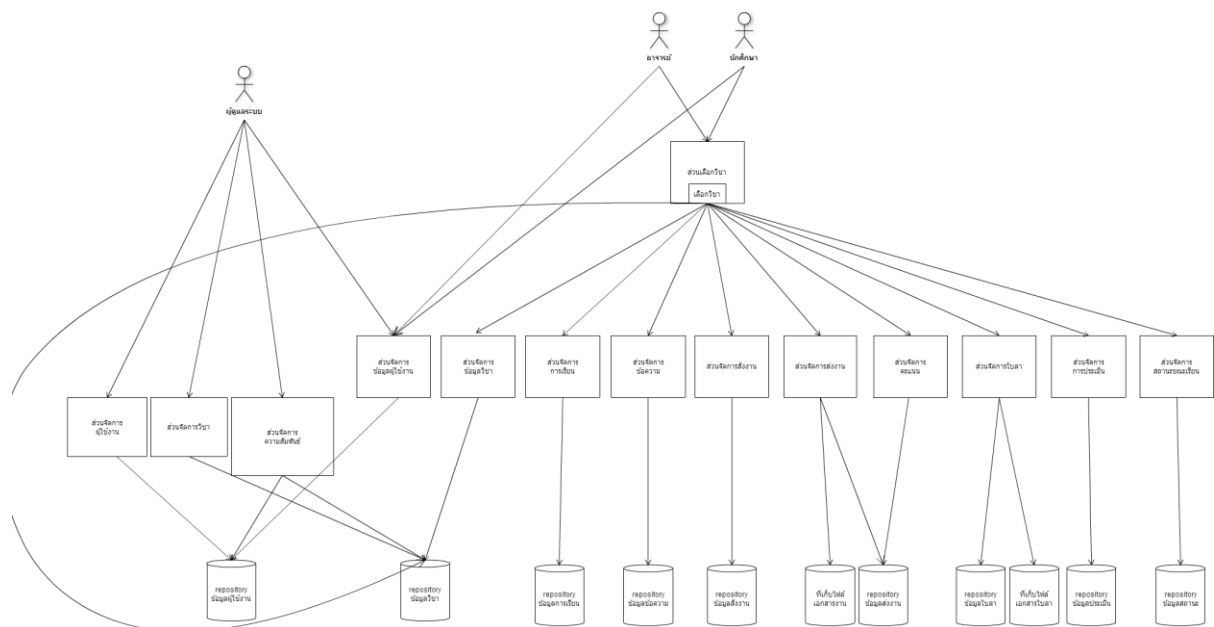
### Problem Analysis

#### Component

- ส่วนจัดการการเข้าสู่ระบบ
- ส่วนดูรายการนักศึกษา
- ส่วนเพิ่มนักศึกษา
- ส่วนลบนักศึกษา
- ส่วนดูรายละเอียดนักศึกษา
- ส่วนดูรายการอาจารย์
- ส่วนเพิ่มอาจารย์
- ส่วนลบอาจารย์
- ส่วนดูรายละเอียดอาจารย์
- ส่วนดูรายการผู้ดูแลระบบ
- ส่วนเพิ่มผู้ดูแลระบบ
- ส่วนลบผู้ดูแลระบบ
- ส่วนดูรายละเอียดผู้ดูแลระบบ
- ส่วนดูวิชา
- ส่วนเพิ่มวิชา
- ส่วนลบวิชา
- แก้ไขวิชา
- ส่วนเพิ่มอาจารย์ในวิชา
- ส่วนเพิ่มนักศึกษาในวิชา
- ส่วนลบอาจารย์ในวิชา
- ส่วนเพิ่มอาจารย์ในวิชา
- ส่วนดูข้อมูลผู้ใช้งาน
- ส่วนแก้ไขข้อมูลผู้ใช้งาน
- ส่วนดูข้อมูลวิชา
- ส่วนแก้ไขข้อมูลวิชา
- ส่วนดูตัวจัดการการเรียน
- ส่วนเปลี่ยนแปลงวันเรียน
- ส่วนดูข้อความ
- ส่วนส่งข้อความ
- ส่วนดูงานที่สั่ง
- ส่วนสั่งงาน
- ส่วนดูงานที่ส่ง
- ส่วนส่งงาน
- ส่วนดูคะแนน

- ส่วนเพิ่มคะแนน
- ส่วนแก้ไขคะแนน
- ส่วนดูใบลา
- ส่วนส่งใบลา
- ส่วนอนุมัติใบลา
- ส่วนดูผลการประเมิน
- ส่วนประเมิน
- ส่วนเพิ่มสถานะขณะเรียน
- ส่วนดูสถานะขณะเรียน

## Application Architecture



- ส่วนเลือกวิชา
  - ใช้เพื่อเลือกวิชาในการเข้าไปใช้งานระบบต่างๆได้
  - ในส่วนนี้จะประกอบไปด้วย ตัวเลือกวิชา โดยส่วนเลือกวิชานี้จะเป็นตัวติดต่อกับส่วนจัดการฐานข้อมูลวิชา และเชื่อมต่อกับส่วนจัดการข้อมูลวิชา ส่วนจัดการการเรียน ส่วนจัดการเอกสารการเรียน ส่วนจัดการข้อความ ส่วนจัดการงาน ส่วนจัดการคะแนน ส่วนจัดการใบลา ส่วนจัดการการประเมิน ส่วนจัดการสถานะขณะเรียน เพราะในการใช้งานระบบจะต้องมีการเลือกวิชาก่อนที่จะสามารถใช้งานได้ โดยที่อาจารย์ และนักเรียนสามารถใช้ส่วนนี้ได้

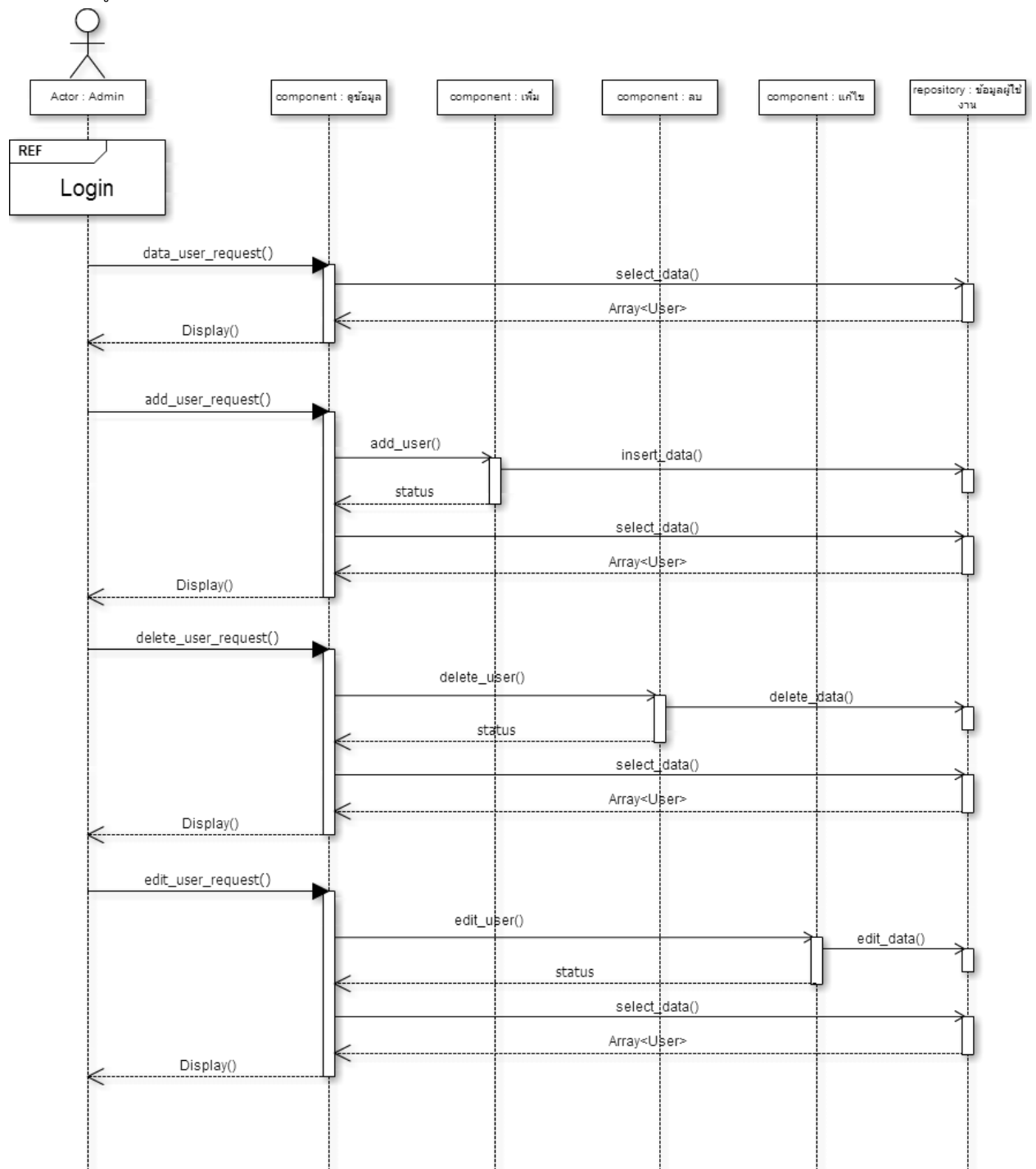
- ส่วนจัดการผู้ใช้งาน
  - ใช้เพื่อเพิ่มอาจารย์ นักศึกษา หรือผู้ดูแลระบบ เพื่อให้สามารถใช้งานในส่วนยืนยันตัวตน และได้รับระดับสิทธิ์ที่ถูกต้องในการใช้งาน
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงผู้ใช้งาน เพิ่มผู้ใช้งาน ลบผู้ใช้งาน แก้ไขข้อมูลผู้ใช้งาน โดยส่วนจัดการผู้ใช้งานนี้จะเป็นตัวติดต่อกับส่วนจัดการฐานข้อมูลผู้ใช้งาน โดยที่ผู้ดูแลระบบเท่านั้นที่สามารถใช้ส่วนนี้ได้
- ส่วนจัดการวิชา
  - ใช้เพื่อเพิ่มวิชาเรียนเข้าสู่ระบบ
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงวิชา เพิ่มวิชา ลบวิชา แก้ไขวิชา โดยส่วนจัดการวิชานี้จะเป็นตัวติดต่อกับส่วนจัดการฐานข้อมูลวิชา โดยที่ผู้ดูแลระบบเท่านั้นที่สามารถใช้ส่วนนี้ได้
- ส่วนจัดการความสัมพันธ์
  - ใช้ในการจัดความสัมพันธ์เพื่อให้รู้ว่าวิชาเรียนนี้ใครเป็นผู้สอนและมีนักศึกษาคนไหนบ้างที่เรียนในวิชานี้
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงความสัมพันธ์ของวิชา อาจารย์และนักศึกษา เพิ่มนักศึกษาในวิชาเรียน เพิ่มอาจารย์ผู้สอน ลบความสัมพันธ์ แก้ไขความสัมพันธ์ โดยส่วนจัดการความสัมพันธ์นี้จะเป็นตัวติดต่อกับส่วนจัดการฐานข้อมูลวิชา และฐานข้อมูลผู้ใช้งาน โดยที่ผู้ดูแลระบบเท่านั้นที่สามารถใช้ส่วนนี้ได้
- ส่วนจัดการข้อมูลผู้ใช้งาน
  - ใช้ในการดูข้อมูลส่วนตัวและแก้ไขข้อมูลส่วนตัวเพื่อให้ข้อมูลส่วนตัวของผู้ใช้งานระบบมีความถูกต้องและเป็นจริงตลอดเวลา
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงข้อมูลของผู้ใช้งาน แก้ไขข้อมูลของผู้ใช้งาน โดยส่วนจัดการข้อมูลผู้ใช้งานนี้จะตัวติดต่อกับส่วนจัดการฐานข้อมูลผู้ใช้งาน โดยที่ผู้ดูแลระบบ อาจารย์ และนักศึกษาสามารถใช้ส่วนนี้ได้
- ส่วนจัดการข้อมูลวิชา
  - ใช้ในการดูและแก้ไขรายละเอียดของวิชาเรียนเพื่อให้ นักศึกษาสามารถรู้ว่าวิชาเรียนนี้สอนเกี่ยวกับอะไรบ้าง และมีแนวทางการเรียนอย่างไร
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงรายละเอียดวิชา แก้ไขรายละเอียดของวิชาโดยส่วนจัดการข้อมูลวิชานี้จะเป็นตัวติดต่อกับส่วนจัดการฐานข้อมูลวิชา โดยที่อาจารย์สามารถดูและแก้ไขรายละเอียดของวิชาได้ ส่วนนักศึกษาสามารถดูรายละเอียดวิชาได้
- ส่วนจัดการการเรียน
  - ใช้ในการแจ้งนักศึกษาให้ทราบว่าวันไหนจะมีการจัดการเรียนการสอน หรือย้ายวันเวลาในการเรียน
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงรายละเอียดการเรียน(วัน เวลาเรียน) เปลี่ยนแปลงวันเรียน โดยส่วนจัดการข้อมูลการเรียนนี้จะตัวติดต่อกับส่วนจัดการฐานข้อมูลการเรียน โดยที่อาจารย์สามารถดูและเปลี่ยนแปลงวันเวลาเรียนได้ ส่วนนักศึกษาสามารถดูรายละเอียดการเรียนได้

- ส่วนจัดการข้อความ
  - ใช้ในการส่งคำถามหรือคำปรึกษาให้กับอาจารย์และใช้ในการตอบคำถามที่นักศึกษาถามมาได้
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงข้อความ ส่งข้อความ โดยส่วนจัดการข้อความนี้จะเป็นตัวติดต่อกับส่วนจัดการฐานข้อมูลข้อความ โดยที่อาจารย์และนักศึกษาสามารถใช้งานในส่วนนี้ได้
  
- ส่วนจัดการงาน
  - ใช้ในการส่งงานและส่งงาน เพื่อช่วยให้การส่งงานของอาจารย์ง่ายขึ้น
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงงานที่ส่ง ส่งงาน แสดงงานที่ส่ง ส่งงาน โดยส่วนจัดการงานนี้จะเป็นตัวติดต่อกับส่วนจัดการฐานข้อมูลส่งงานและฐานข้อมูลส่งงาน และเก็บเอกสารในที่เก็บไฟล์เอกสารงาน โดยที่อาจารย์สามารถดูงานที่ส่ง ส่งงาน ดูงานที่นักศึกษาส่งทั้งหมดได้ ส่วนนักศึกษาสามารถดูงานที่ส่ง ดูงานที่ส่ง ส่งงานได้
  
- ส่วนจัดการคะแนน
  - ใช้ในการจัดการคะแนนของนักศึกษา เพื่อให้ง่ายในการเก็บคะแนนและแจ้งให้นักศึกษาทราบคะแนนของตนเอง
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงคะแนน เพิ่มคะแนน ลบคะแนน แก้ไขคะแนน โดยส่วนจัดการคะแนนนี้จะตัวติดต่อกับส่วนจัดการฐานข้อมูลส่งงาน โดยที่อาจารย์สามารถดูคะแนนของนักศึกษา เพิ่มคะแนน ลบคะแนน แก้ไขคะแนนได้ ส่วนนักศึกษาสามารถดูคะแนนของตนเองได้
  
- ส่วนจัดการใบลา
  - ใช้ในการส่งใบลาให้กับอาจารย์ เพื่อช่วยอำนวยความสะดวกในการส่งใบลาของนักศึกษา
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงใบลา ส่งใบลา อนุมัติใบลา โดยส่วนจัดการใบลานี้จะเป็นตัวติดต่อกับส่วนจัดการฐานข้อมูลใบลาและเก็บเอกสารใบลาในที่เก็บไฟล์เอกสารใบลา โดยที่อาจารย์สามารถดูใบลาและอนุมัติใบลาได้ ส่วนนักศึกษาสามารถดูใบลาและส่งใบลาได้
  
- ส่วนจัดการการประเมิน
  - ใช้ในการประเมินอาจารย์เพื่อให้อาจารย์สามารถวิเคราะห์การสอนได้
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงการประเมิน ทำแบบประเมิน โดยส่วนจัดการการประเมินนี้จะตัวติดต่อกับส่วนจัดการฐานข้อมูลประเมินโดยที่อาจารย์สามารถดูผลการประเมินจากนักศึกษาได้ ส่วนนักศึกษาสามารถทำแบบประเมินได้
  
- ส่วนจัดการสถานะขณะเรียน
  - ใช้ในการแสดงสถานะของนักศึกษาขณะเรียนได้ อาจใช้เพื่อแจ้งให้อาจารย์ทราบว่าไม่เข้าใจบทเรียนในส่วนไหน หรือตามการสอนไม่ทัน
  - ในส่วนนี้จะประกอบไปด้วย ส่วนแสดงสถานะขณะเรียน เพิ่มสถานะขณะเรียน โดยส่วนจัดการสถานะขณะเรียนนี้จะตัวติดต่อกับส่วนจัดการฐานข้อมูลสถานะขณะเรียน โดยที่อาจารย์สามารถดูสถานะของนักศึกษาได้ ส่วนนักศึกษาสามารถเพิ่มสถานะขณะเรียนได้

## Subsystem / Component

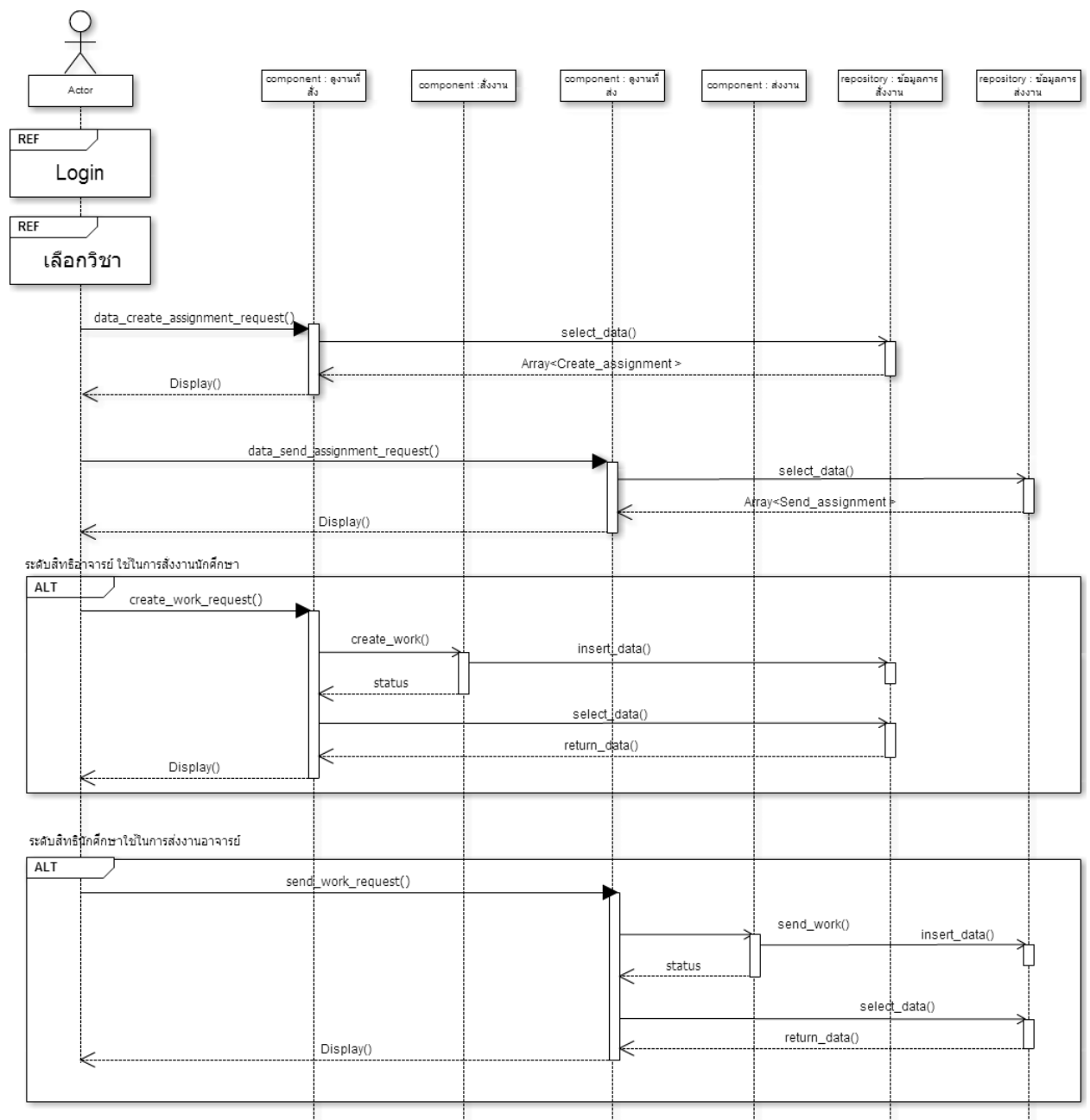
Sequence diagram 1

ส่วนจัดการผู้ใช้งาน



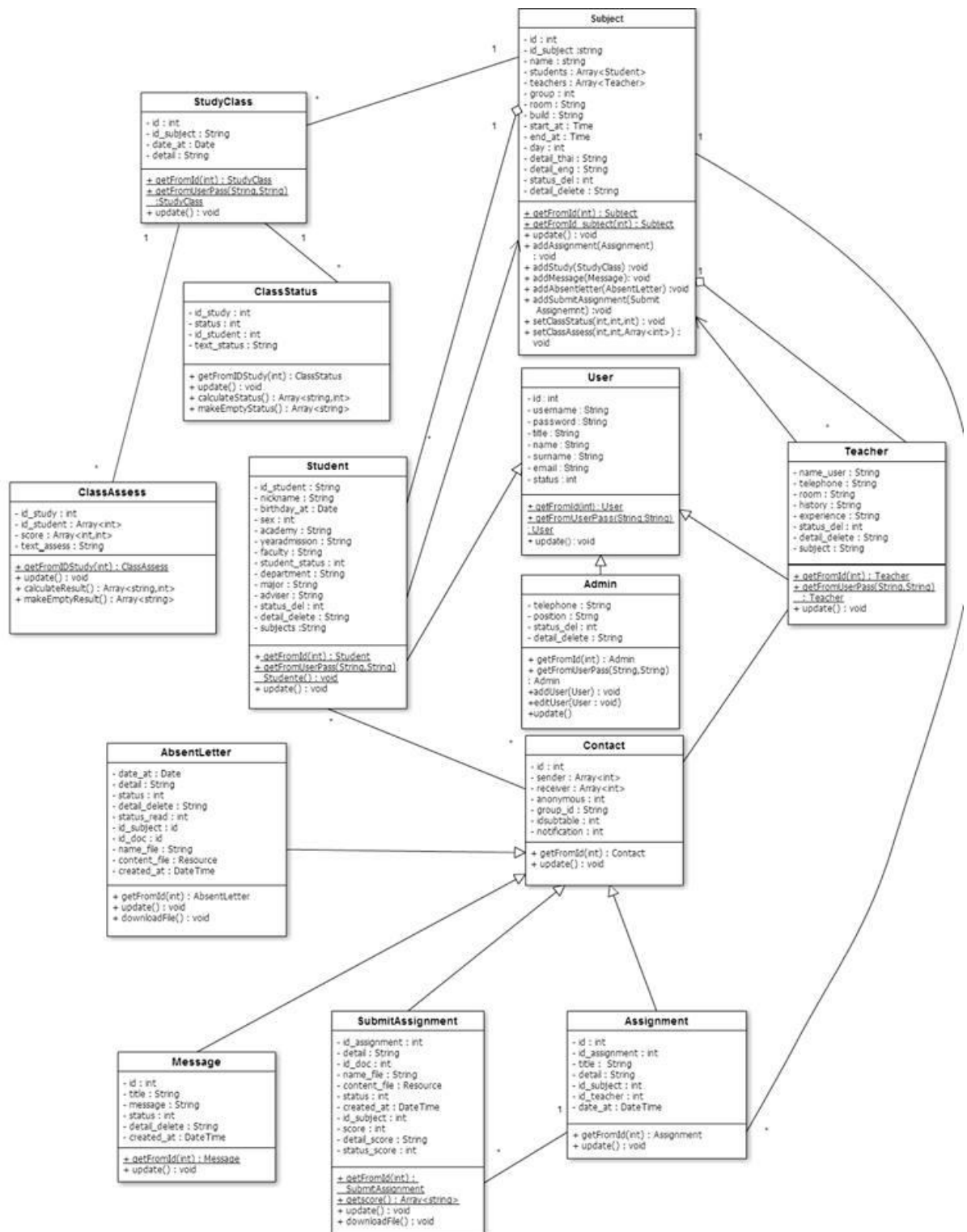
## Sequence diagram 2

### ส่วนจัดการงาน



## แผนภาพของคลาสหลัก

Class diagrams





## รายละเอียดการพัฒนาซอฟต์แวร์

### Deployment

ส่วนการแสดงผลและการติดต่อกับผู้ใช้

การแสดงผลแสดงในเว็บเบราว์เซอร์ ทางผู้พัฒนาเลือกใช้ HTML, CSS สำหรับการจัดการเรื่องโครงสร้างการแสดงผล และใช้ภาษา javascript สำหรับจัดการเรื่องการตอบสนองกับผู้ใช้ โดยมีการนำ front-end framework อย่าง Bootstrap มาเป็นตัวช่วยในการสร้างหน้าเว็บได้อย่างรวดเร็ว และตอบสนองกับอุปกรณ์ที่มีขนาดหน้าจอต่างกัน และ Library ของ javascript อย่าง JQuery มาช่วยในการเขียน javascript ให้สะดวกรวดเร็วมากขึ้น ส่วนนี้ทำงานบน Azure web site service ที่มี URL เป็น rpslms.azurewebsites.net

### ส่วนการประมวลผล

การประมวลผลทางผู้พัฒนาเลือกใช้บริการสร้าง web site ของ Microsoft azure ที่มี URL เป็น rpslms.azurewebsites.net ซึ่งได้รวมคุณสมบัติและซอฟต์แวร์ต่างๆที่จำเป็นต่อการใช้เป็น web server ไว้เรียบร้อยแล้ว และยังมีคุณสมบัติการ scale โดยมีบริการที่สามารถระบุจำนวนเครื่องที่ต้องการให้มีการแบ่งการประมวลผลออกไปได้โดยไม่ต้องเขียนโปรแกรมใหม่(scale out) ในส่วนของภาษาที่ใช้ประมวลผลทางผู้พัฒนาเลือกใช้ภาษา php ร่วมกับ MVC framework อย่าง Laravel เพื่อให้ง่ายต่อการพัฒนาและบำรุงรักษา

### ส่วนระบบฐานข้อมูล

ระบบฐานข้อมูลทางผู้พัฒนาเลือกใช้บริการ SQL database service ของ Microsoft azure ซึ่งมีคุณสมบัติในการ Scale ที่สามารถกำหนดได้โดยง่าย ในการเข้าถึงระบบฐานข้อมูลจะใช้ ElequentORM ที่มาพร้อมกับ Laravel framework ในการจัดการเพื่อให้ง่ายต่อการพัฒนา โดยใช้ server psparg25ep ที่ติดต่อด้วย URL [https://psparg25ep.database.windows.net/#\\$database=rpslmsdb](https://psparg25ep.database.windows.net/#$database=rpslmsdb)

### ส่วนระบบจัดเก็บข้อมูล

ระบบจัดเก็บข้อมูลทางผู้พัฒนาเลือกใช้บริการ storage service สำหรับเก็บไฟล์เอกสารที่ใช้ในระบบ ที่มี URL เป็น <https://rpslmsr.blob.core.windows.net/>

## Implementation plan

ลำดับ	การทำงาน	ผู้รับผิดชอบ	วันที่
1	ติดตั้งและตั้งค่าระบบ Azure website service, Laravel framework	นาย วิษุวัต ชันเฮม	22 กย - 25 กย
2	ออกแบบส่วนการแสดงผลหน้าเว็บของระบบทั้งหมด	นาย อรรถกร เทพลีธี	22 กย - 2 ตค
3	ทดลองใช้ Azure service(SQL database as a service, website service)	นาย วิษุวัต ชันเฮม	26 กย - 2 ตค
4	แปลง code ส่วนการแสดงผลเป็นรูปแบบ blade template ใน Laravel	นาย วิษุวัต ชันเฮม	13 ตค - 19 ตค
5	ออกแบบระบบฐานข้อมูลสำหรับใช้ในระบบทั้งหมด	นาย อรรถกร เทพลีธี	13 ตค - 19 ตค
6	เขียนส่วนการทำงานของระบบผู้ดูแลระบบ	นาย วิษุวัต ชันเฮม	20 ตค - 27 ตค
7	เขียนส่วนการทำงานของระบบอาจารย์	นาย อรรถกร เทพลีธี	20 ตค - 27 ตค
8	เขียนส่วนตอบสนองการทำงานในฝั่งผู้ใช้	นาย วิษุวัต ชันเฮม	28 ตค - 3 พย
9	เขียนส่วนการทำงานของระบบนักเรียน	นาย อรรถกร เทพลีธี	28 ตค - 3 พย
10	ทดสอบการทำงาน และแก้ไขข้อผิดพลาด	นาย วิษุวัต ชันเฮม	4 พย - 10 พย

ตารางแผนดำเนินการ รูปแบบที่

ลำดับ	การทำงาน	คนรับผิดชอบ	กันยายน										ตุลาคม										พฤศจิกายน																				
			22	23	24	25	26	27	28	29	30	1	2		13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10
1	ติดตั้งและตั้งค่าระบบ Azure website service, Laravel framework	นาย วิษุวัต ชันเฮม											สัปดาห์ภาค																														
2	ออกแบบส่วนการแสดงผลหน้าเว็บของระบบทั้งหมด	นาย อรรถกร เทพลีธี																																									
3	ทดลองใช้ Azure service(SQL database as a service, website service)	นาย วิษุวัต ชันเฮม																																									
4	แปลง code ส่วนการแสดงผลเป็นรูปแบบ blade template ใน Laravel	นาย วิษุวัต ชันเฮม																																									
5	ออกแบบระบบฐานข้อมูลสำหรับใช้ในระบบทั้งหมด	นาย อรรถกร เทพลีธี																																									
6	เขียนส่วนการทำงานของระบบผู้ดูแลระบบ	นาย วิษุวัต ชันเฮม																																									
7	เขียนส่วนการทำงานของระบบอาจารย์	นาย อรรถกร เทพลีธี																																									
8	เขียนส่วนตอบสนองการทำงานในฝั่งผู้ใช้	นาย วิษุวัต ชันเฮม																																									
9	เขียนส่วนการทำงานของระบบนักเรียน	นาย อรรถกร เทพลีธี																																									
10	ทดสอบการทำงาน และแก้ไขข้อผิดพลาด	นาย วิษุวัต ชันเฮม																																									

ตารางแผนดำเนินการ รูปแบบที่ 2

## ผลการทดสอบซอฟต์แวร์

### Test

1. ส่วนการทดสอบสถานะการเป็นผู้ดูแลระบบ  
File path คือ \laravel\app\controllers\Test.php (function usersAdmin())
2. ส่วนการทดสอบการค้นหาผู้ใช้โดยตัดผู้ใช้ที่มีสถานะ “ถูกลบออกจากระบบ” ออกไป  
File path คือ \laravel\app\controllers\Test.php (function searchExcludeDelUser())

### Evaluation

1. ทดลองให้อาจารย์สั่งงานไปถึงนักศึกษาทุกคนในวิชานั้นๆ

#### จุดประสงค์ของการทดลอง

อาจารย์สามารถสั่งงานนักศึกษาทุกคนในวิชาได้

#### สิ่งที่จะวัด

งานที่ส่งถูกส่งไปถึงนักศึกษาทุกคน และรายละเอียดของงานที่ส่งมีความถูกต้องครบถ้วน

#### สิ่งที่ต้องใช้ในการทดลอง

1. วิชา 1 วิชาสำหรับการทดลอง
2. รหัสสำหรับเข้าใช้งานระดับสิทธิของอาจารย์ 1 รหัส ที่สอนในวิชานั้นๆ
3. รหัสสำหรับเข้าใช้งานระดับสิทธิของนักศึกษา 5 รหัส ที่ลงทะเบียนในวิชานั้นๆ
4. Browser สำหรับเข้าใช้งานระบบ
5. งานสำหรับสั่งให้นักศึกษาทำ 1 งาน

#### วิธีทำการทดลอง

1. เข้าสู่ระบบในระดับสิทธิของอาจารย์
2. เลือกวิชาที่จะสั่งงาน
3. ไปที่ระบบสั่งงาน
4. เพ่งงานที่จะสั่ง
5. ใส่รายละเอียดของงาน
6. สั่งงานให้นักศึกษา
7. เข้าสู่ระบบในระดับสิทธิของนักศึกษา
8. เลือกวิชานั้นๆ
9. ไปที่ระบบสั่งงาน
10. ตรวจสอบว่ามีงานที่อาจารย์สั่งหรือไม่
11. กลับไปทำข้อ 7-11 ใหม่จนกว่าจะครบทั้ง 5 รหัส

#### ที่ได้จากการทดลอง

งานที่อาจารย์สั่งนักศึกษาด้วยระบบ นักศึกษาทุกคนในวิชาสามารถเห็นและดูรายละเอียดของงานที่สั่งได้อย่างถูกต้องครบถ้วน

## สรุปและสิ่งที่คุณได้จากการทดลองนี้

งานที่อาจารย์สั่งสามารถส่งไปถึงนักศึกษาทุกคนในวิชาจริง

### 2. ทดลองให้นักศึกษาส่งงาน

#### จุดประสงค์ของการทดลอง

นักศึกษาสามารถส่งงานตามที่อาจารย์สั่งได้

#### สิ่งที่จะวัด

งานที่นักศึกษาส่งให้อาจารย์มีความถูกต้องและครบถ้วน

#### สิ่งที่ต้องใช้ในการทดลอง

1. วิชา 1 วิชาสำหรับการทดลอง
2. รหัสสำหรับเข้าใช้งานระดับสิทธิของอาจารย์ 1 รหัส ที่สอนในวิชานั้นๆ
3. รหัสสำหรับเข้าใช้งานระดับสิทธิของนักศึกษา 5 รหัส ที่ลงทะเบียนในวิชานั้นๆ
4. Browser สำหรับเข้าใช้งานระบบ
5. งานที่อาจารย์สั่งให้ทำ
6. งานที่นักศึกษาต้องส่งให้อาจารย์ตามงานที่อาจารย์สั่งมา

#### วิธีทำการทดลอง

1. เข้าสู่ระบบในระดับสิทธิของนักศึกษา
2. เลือกวิชาที่จะส่งงาน
3. ไปที่ระบบส่งงาน
4. เลือกดำเนินการเพื่อส่งงาน
5. ใส่รายละเอียดของงาน
6. แนบเอกสารที่จะส่งงาน
7. ส่งงานให้อาจารย์
8. กลับไปทำข้อ 1 ใหม่จนครบทั้ง 5 รหัส
9. เข้าสู่ระบบในระดับสิทธิของอาจารย์
10. เลือกวิชานั้นๆ
11. ไปที่ระบบส่งงาน
12. เลือกดำเนินการเพื่อดูงานที่ส่งมา
13. ตรวจสอบว่ามีงานที่นักศึกษาส่งมาครบหรือไม่

#### ผลที่ได้จากการทดลอง

งานที่ส่งโดยนักศึกษาผ่านระบบสามารถใส่รายละเอียด และแนบเอกสารส่งให้อาจารย์ได้อย่างถูกต้อง

ครบถ้วน

## สรุปและสิ่งที่คุณได้จากการทดลองนี้

งานที่ส่งให้อาจารย์สามารถส่งไปถึงอาจารย์ได้จริง

## บทสรุป

จากการที่ทีมพัฒนาใช้เวลาพัฒนาระบบนี้เป็นเวลา 4 เดือน นับตั้งแต่การคิด requirement ของระบบที่ตรงกับความต้องการ ศึกษาการทำงานของระบบอื่นๆเพื่อหาจุดอ่อนของระบบเหล่านั้น ตลอดจนการออกแบบสถาปัตยกรรมระบบ รวมถึงการใช้ Object oriented เป็นเครื่องมือในการวิเคราะห์และออกแบบ

ในส่วนของการ Implement ผู้พัฒนานำ Design และ requirement ทั้งหมดที่วางแผนไว้ตอนแรกมาพัฒนาระบบ ทำให้รู้ถึงข้อดีข้อเสียของ Design ที่วางไว้ ซึ่งถือเป็นประสบการณ์ที่ดีในการปรับปรุงแก้ไขในโอกาสต่อไป นอกจากนี้ทางผู้พัฒนายังได้ประสบการณ์เกี่ยวกับการพัฒนาเว็บไซต์ด้วย Framework Laravel Cloud service ต่างๆ และการใช้ Git เป็น version control ซึ่งการพัฒนาระบบในสภาพแวดล้อมที่ไม่เคยทำมาก่อน

สุดท้ายทางผู้พัฒนาขอขอบคุณทุกคนที่มีส่วนร่วมให้งานนี้เสร็จลุล่วงไปได้ด้วยดี เพื่อน อาจารย์ที่ปรึกษา ที่ให้ความช่วยเหลือที่ดีมาโดยตลอด

## บรรณานุกรม

<https://leanpub.com/laravel-first-framework/read>

<http://laravel.com/docs/4.2>

<http://www.thaicreate.com/windows-azure.html>