

# Design & Implementation Plan Document

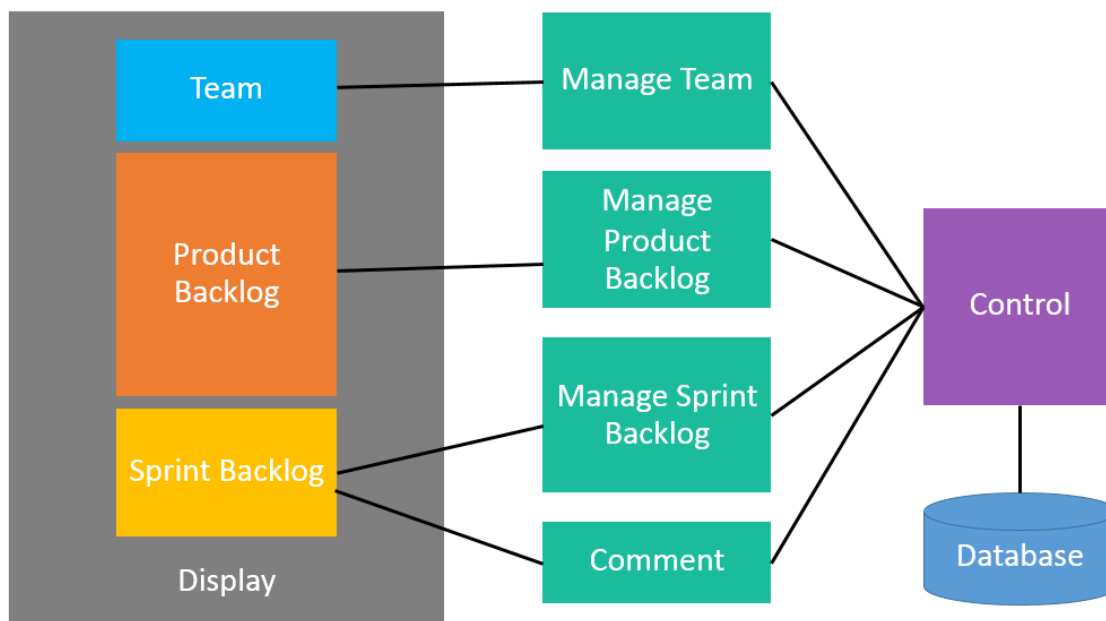
Class : Cloud Computing Fall 2014 @KMITL

ชื่อโปรเจกต์ : Scrum Framework

ชื่อกลุ่ม : Deadline-Driven-Development

ชื่อสมาชิกในกลุ่ม : 1. นายชานน จรัสสุทธิกุล รหัสนักศึกษา 55010280  
2. นายณัฐพงศ์ อมรบุญชรวะช รหัสนักศึกษา 55010371

## โครงสร้างภายในระบบ



### 1. Manage Team

เป็นส่วนที่บริหารจัดการทีม การเพิ่มสมาชิกเข้ามาในทีม, การลบสมาชิกออกจากทีม และ การให้ตำแหน่งสมาชิกในทีม

### 2. Manage Product Backlog

เป็นส่วนที่บริหารจัดการ Product Backlog โดย สามารถเพิ่ม user story, การลบ user story, การกำหนดลำดับความสำคัญของ user story, และการนำ user story ไปใส่ในแต่ละ sprint

### 3. Manage Sprint Backlog

เป็นส่วนที่บริหารจัดการ Sprint Backlog โดยสามารถเพิ่มงาน, ลบงาน, มอบงานให้กับสมาชิกในทีม, การย้ายสถานะของงาน (มี 3 สถานะ คือ To-do In progress และ Done) และ Comment

#### 4. Comment

เป็นส่วนที่สามารถใส่รายละเอียดหรือให้สมาชิกในทีมแสดงความคิดเห็นต่องานใน Sprint Backlog โดยสามารถ เพิ่มความคิดเห็น, แก้ไขความคิดเห็น และ ลบความคิดเห็น

### บริการ Cloud ที่ใช้ในการดำเนินการสร้าง Software

Microsoft Azure: Virtual Machines – ใช้ในการให้บริการซอฟต์แวร์ และสามารถปรับแต่งโครงสร้างภายในของระบบได้โดยง่าย

Microsoft Azure: Cloud Service - สำหรับการ Deploy Application และมี Cloud Service, DNS name และสามารถทำ load balance กับ virtual machines ได้

### Software ที่ใช้ในการสร้างระบบ

#### Font-end development

1. HTML5: เป็นโครงสร้างของ User Interface เบื้องต้นของระบบ
2. CSS3: ใช้ในการตกแต่ง User Interface ให้สวยงาม
3. Twitter Bootstrap: ตกแต่ง user interface หน้าเว็บไซต์ รวมถึงการทำให้เว็บไซต์นั้นรองรับการแสดงผลทั้งบน Mobile และ Desktop (responsive design)
4. AngularJS: ควบคุมการทำงานของส่วน user interface ให้มีการตอบสนองกับผู้ใช้ในเชิงตรรกะ เช่นการย้าย sprint backlog ไปตามกระดานต่างๆ

#### Back-end development

1. Laravel (PHP Framework): ใช้เพื่อทำให้ง่ายต่อการพัฒนาเว็บแอปพลิเคชัน โดยใช้ eloquent ORM ในการติดต่อฐานข้อมูล
2. MongoDB: สำหรับเก็บข้อมูลใน sprint backlog เป็นรูปแบบ document และใช้ในการ replicate, backup และ restore ข้อมูล

## Implementation plan

### 1. สร้างระบบ ปรับแต่ง และตั้งค่าระบบ Web Server

ทำการติดตั้งและตั้งค่าเบื้องต้นของ Virtual Machine ที่จะเป็น Web Server และตั้งค่า Web Server Application ที่ใช้ โดยในระบบนี้จะใช้ Nginx เป็น Web Server

### 2. สร้างระบบ database

ทำการติดตั้งและตั้งค่า database หลักเพื่อใช้ในการทดสอบและพัฒนาเบื้องต้น และตั้งค่าเพื่อรองรับการทำ Replication และ Sharding ต่อไป

### 3. ตั้งค่าระบบ Load Balance ของ Virtual Machine ที่ใช้

ทำการทำการตั้งค่า Load Balancer ของ Microsoft Azure เพื่อรองรับการขยายตัวของระบบและฐานผู้ใช้ในอนาคต

### 4. ตั้งค่าระบบ Replication ของ database

สร้าง Replication ของ database เพื่อใช้รองรับในกรณีที่ database มีความผิดพลาดขึ้น หรือในกรณีที่ผู้ใช้จำนวนมากเข้าใช้บริการพร้อมกันจำนวนมาก

### 5. ตั้งค่าทำระบบ Sharding ของ database

ทำการกระจาย database ออกเป็นหลาย ๆ ส่วนเพื่อให้สามารถทำการขยาย database เป็นไปได้โดยง่าย และรองรับการขยายตัวในด้านของข้อมูล หรือจำนวนผู้ใช้งาน

### 6. ตั้งค่าระบบ SSL และ https

ทำการเข้ารหัสช่องทางการสื่อสารในส่วนของระบบยืนยันตัวตน เพื่อเพิ่มความปลอดภัย และป้องกันการถูกดักข้อมูลในระหว่างทางของช่องทางสื่อสารข้อมูล

### 7. พัฒนาระบบยืนยันตัวตน

สร้างระบบยืนยันตัวตน และระบบสมัครสมาชิกสำหรับใช้งานแอปพลิเคชัน รวมถึงกำหนดค่าต่าง ๆ ที่จำเป็นต้องมีสำหรับผู้ใช้แต่ละคน

### 8. ออกแบบ User Interface

ทำการวางแผนและออกแบบส่วนติดต่อผู้ใช้ของแต่ละส่วนในแอปพลิเคชัน โดยคำนึงถึง User Experience เป็นส่วนสำคัญในการออกแบบ

### 9. สร้างส่วนของ User Interface

ทำการสร้างส่วนติดต่อผู้ใช้ตามที่ได้ออกแบบไว้ โดยใช้ AngularJS และ Twitter Bootstrap เป็นเครื่องมือในการช่วยพัฒนา

## 10. พัฒนาระบบ Team Management

สร้างส่วนจัดการสมาชิกในทีม – ประกอบไปด้วยส่วนของการสร้างทีม การเชิญผู้ใช้เข้าร่วมทีม การกำหนดตำแหน่งของสมาชิกในทีม การลบผู้ใช้ออกจากการเป็นสมาชิกทีม

## 11. พัฒนาระบบจัดการ Product Backlog

สร้างส่วนจัดการ Product Backlog – การเพิ่ม User Story, การจัดลำดับความสำคัญของ User Story การลบ User Story ออกจาก product backlog และการเลือก User story ที่จะทำการพัฒนาใน Sprint ที่ต้องการ

## 12. ทำระบบจัดการ Sprint Backlog

สร้างส่วนจัดการกับ Sprint Backlog – ประกอบไปด้วยระบบการเพิ่มงานที่จะทำ การย้ายงานไปอยู่ในรายการของ To-do, In progress, Done การมอบหมายงานให้สมาชิกในทีม และการลบงานที่เลือก

## 13. สร้างระบบ Comment

สร้างระบบแสดงความคิดเห็นในงานแต่ละงานบน Sprint Backlog โดยสามารถเพิ่มความคิดเห็น แก้ไข และลบความคิดเห็นได้

## 14. สร้าง REST API

สร้างส่วนติดต่อของแอปพลิเคชันในรูปแบบของ REST API โดยสามารถส่งข้อมูลของ Sprint Backlog ในรูปแบบของ JSON Object ได้

## 15. ทดสอบการทำงานและแก้ไขข้อผิดพลาด

ทำการทดสอบการทำงานของระบบในด้านของความถูกต้อง และความสามารถในการรองรับผู้ใช้งานจำนวนมากพร้อม ๆ กัน และความสามารถในการขยายตัวของระบบ

## ตารางแสดงแผนการดำเนินงาน

การดำเนินงาน/ระยะเวลาที่ดำเนินการ	13-19 ต.ค.	20 - 26 ต.ค.	27 ต.ค.- 2 พ.ย.	3 - 9 พ.ย.	10 - 16 พ.ย.
สร้างระบบ					
Web Server					
ออกแบบและสร้างระบบ database					
ตั้งค่าระบบ Load Balance ของ Virtual Machine ที่ใช้					
ตั้งค่าระบบ SSL					
ตั้งค่าระบบ Sharding ของ database					
ตั้งค่าระบบ Replication ของ database					
พัฒนาระบบ login & register					
ออกแบบ User Interface					
สร้างส่วนของ User Interface					
พัฒนาระบบ Team Management					
พัฒนาระบบจัดการ Product Backlog					
พัฒนาระบบจัดการ Sprint Backlog					
สร้างระบบ Comment					
สร้าง REST API					
ทดสอบการตั้งค่างานและแก้ไขข้อผิดพลาด					

## ผู้รับผิดชอบการดำเนินงานในแต่ละส่วน :

ชานน จรัสสุทธิกุล รับผิดชอบในส่วนของ

1. พัฒนาระบบ login & register
2. ออกแบบ User Interface
3. สร้างส่วนของ User Interface
4. พัฒนาระบบ Team Management
5. พัฒนาระบบจัดการ Product Backlog
6. พัฒนาระบบจัดการ Sprint Backlog
7. สร้างระบบ Comment

ณัฐพงศ์ อมรบุญชรวะช รับผิดชอบในส่วนของ

1. สร้างระบบ ปรับแต่ง และตั้งค่าระบบ Web Server
2. ออกแบบและสร้างระบบ database
3. พัฒนาระบบ Load Balance ของ Virtual Machine ที่ใช้
4. พัฒนาระบบ Replication ของ database
5. พัฒนาระบบ Sharding ของ database
6. พัฒนาระบบ SSL
7. สร้าง REST API
8. ทดสอบการพัฒนางานและแก้ไขข้อผิดพลาด