

รายงานการปฏิบัติงานสหกิจศึกษา
เรื่อง
เว็บแอปพลิเคชันประเมินความสามารถเบื้องต้นของผู้
สมัครงาน
Pre-Employment Testing

ปฏิบัติงาน ณ บริษัท ไชเงิน

โดย

นายศตวรรษ ธิติสุภกุล
รหัสประจำตัว 60070093

รายงานนี้เป็นส่วนหนึ่งของการศึกษารายวิชา สหกิจศึกษา
สาขาวิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 1 ปีการศึกษา 2563
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

รายงานการปฏิบัติงานสหกิจศึกษา
เว็บแอปพลิเคชันประเมินความสามารถเบื้องต้นของผู้
สมัครงาน

Pre-Employment Testing

ปฏิบัติงาน ณ บริษัท ไซเงิน

โดย

นายศตวรรษ ชิติสกุล
รหัสประจำตัว 60070093

ปฏิบัติงาน ณ บริษัท ไซเงิน
65/60 ชั้น 6 อาคารชานาญเพ็ญชาติปิสน์สเซ็นเตอร์ ถนนพระราม 9 เขตห้วยขวาง,
กรุงเทพมหานคร 10310

Web site : <https://zygencenter.com/>

Pre-Employment Testing

Satawat Thtisupakul

**A REPORT SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENT FOR COOPERATIVE EDUCATION PROGRAM
THE DEGREE OF BACHELOR OF SCIENCE PROGRAM IN
INFORMATION TECHNOLOGY
FACULTY OF INFORMATION TECNOLOGY
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

1/ 2020

COPYRIGHT 2020

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF ECHNOLOGY LADKRABANG

วันที่ xx พฤศจิกายน พ.ศ. 2563

เรื่อง ขอส่งรายงานการปฏิบัติงานสหกิจศึกษา

เรียน ดร.สุพัฒนดา โชติพันธ์

ที่ปรึกษาสหกิจศึกษาในสาขาเทคโนโลยีสารสนเทศ

ตามที่ข้าพเจ้า นายศตวรรษ ธิติสุภกุล นักศึกษาสาขาวิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ได้ปฏิบัติงานสหกิจศึกษาระหว่างวันที่ 1 มิถุนายน พ.ศ. 2563 ถึงวันที่ 30 กันยายน พ.ศ. 2563 ในตำแหน่ง Full Stack Developer ณ สถานประกอบการชื่อ บริษัท ไซเงิน และได้รับมอบหมายจากพนักงานที่ปรึกษาให้ศึกษาและจัดทำรายงานเรื่อง เว็บไซต์พลิเคชันประเมินความสามารถเบื้องต้นของผู้สมัครงาน

บัดนี้ การปฏิบัติงานสหกิจศึกษาได้สิ้นสุดลงแล้ว จึงใคร่ขอส่งรายงานการปฏิบัติงาน สหกิจศึกษาดังกล่าวมาพร้อมนี้ จำนวน 1 เล่ม เพื่อขอรับคำปรึกษาต่อไป

จึงเรียนมาเพื่อโปรดพิจารณา

ขอแสดงความนับถือ

.....
(นายศตวรรษ ธิติสุภกุล)

กิตติกรรมประกาศ

ตามที่ข้าพเจ้า นายศตวรรษ ธิติสุภกุล ได้มาปฏิบัติงานสหกิจศึกษา ณ บริษัท ไซเงิน ตั้งแต่วันที่ 1 มิถุนายน พ.ศ. 2563 ถึงวันที่ 30 กันยายน พ.ศ. 2563 ทำให้ข้าพเจ้าได้รับความรู้และประสบการณ์ต่างๆ ที่มีคุณค่ามากมาย สำหรับรายงานสหกิจศึกษานี้สำเร็จลงได้ด้วยดี จากความช่วยเหลือและความร่วมมือสนับสนุนของหลายฝ่าย ดังนี้

1. คุณ วิวัฒน์ เสงี่ยมอนต์ ตำแหน่ง Web Developer (พนักงานที่ปรึกษา)

นอกจากนี้ยังมีบุคคลท่านอื่น ๆ อีกที่ไม่ได้กล่าวไว้ ณ ที่นี้ ซึ่งให้ความกรุณาแนะนำในจัดทำรายงานสหกิจศึกษานี้ ข้าพเจ้าจึงใคร่ขอขอบพระคุณทุกท่านที่ได้มีส่วนร่วมในการให้ข้อมูลและให้ความเข้าใจเกี่ยวกับชีวิตของการปฏิบัติงาน รวมถึงเป็นที่ปรึกษาในการจัดทำรายงานฉบับนี้จนเสร็จสมบูรณ์

นายศตวรรษ ธิติสุภกุล

ผู้จัดทำรายงาน

วันที่ xx พฤศจิกายน พ.ศ. 2563

ชื่อรายงานการปฏิบัติงานสหกิจศึกษา	เว็บแอปพลิเคชันประเมินความสามารถเบื้องต้นของ ผู้สมัครงาน
ผู้รายงาน	นายศตวรรษ ธิติศุกุล
คณะ	เทคโนโลยีสารสนเทศ
สาขาวิชา	เทคโนโลยีสารสนเทศ

(ดร.สุพัฒนดา โชติพันธ์)
อาจารย์ที่ปรึกษาสหกิจศึกษา

(นายวิวัฒน์ เล็งอนันต์)
พนักงานที่ปรึกษา

คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
อนุมัติให้นับรายงานการปฏิบัติงานสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ชื่อรายงาน	เว็บแอปพลิเคชันประเมินความสามารถเบื้องต้นของผู้สมัครงาน
ชื่อนักศึกษา	นายศตวรรษ ธิติสุกุล
รหัสนักศึกษา	60070093
สาขาวิชา	เทคโนโลยีสารสนเทศ
อาจารย์ที่ปรึกษา	ดร.สุพัฒน์ดา โชติพันธ์
ปีการศึกษา	2563

บทคัดย่อ

รายงานการปฏิบัติงานสหกิจศึกษานี้กล่าวถึงวัตถุประสงค์ของการปฏิบัติงาน และรายละเอียดของการปฏิบัติงาน ซึ่งประกอบด้วย ที่มา, ความสำคัญ, รายละเอียด, การออกแบบ ของกระบวนการทำงานของเว็บแอปพลิเคชันประเมินความสามารถเบื้องต้นของผู้สมัครงาน โดยทางบริษัท ไซเงิน ได้มอบหมายในช่วงเวลาที่ปฏิบัติงานสหกิจศึกษา สร้างเว็บแอปพลิเคชันประเมินความสามารถเบื้องต้นของผู้สมัครงาน เป็นเว็บแอปพลิเคชันที่สามารถให้พนักงานมาออกแบบข้อสอบเพื่อทดสอบความสามารถของผู้สมัครงาน โดยการออกแบบข้อสอบนั้นสามารถออกแบบคำถามได้หลายรูปแบบ เช่น คำถามอัตนัย, ปรนัย หรือ คำถามที่แนบไฟล์มา เว็บแอปพลิเคชันนี้สามารถกำหนดความยากของข้อสอบ เพื่อทำการสุ่มข้อสอบจากคลังข้อสอบที่สร้างไว้ ทำให้การทดสอบแต่ละครั้งผู้ทดสอบจะได้ข้อสอบที่แตกต่างกันออกไป อ้างอิงโดยระดับความยากที่ผู้ออกข้อสอบทำการกำหนดไว้ มีการกำหนดเวลาในการทำข้อสอบเพื่อวัดความสามารถของผู้สมัครงานโดยทำข้อสอบภายในเวลาที่กำหนด และมีการกำหนดวันสิ้นสุดที่สามารถทำข้อสอบ โดยเว็บแอปพลิเคชันนี้จะทำการส่งข้อสอบให้ผู้สมัครงานทางอีเมลตามที่ผู้ออกข้อสอบกำหนด เมื่อทำข้อสอบแล้วผู้ที่ส่งข้อสอบจะได้รับผลคะแนนของผู้สมัครงานที่เป็นปรนัย ส่วนคำถามที่เป็นไฟล์แนบ และอัตนัย ผู้ที่ส่งข้อสอบให้ผู้สมัครงานต้องเป็นผู้ทำการตรวจด้วยตนเอง แล้วระบบจะทำการสรุปคะแนนทั้งหมดให้โดยอัตโนมัติ เว็บแอปพลิเคชันนี้ทำให้ทางบริษัท ไซเงิน ลดระยะเวลาในการคัดกรองผู้ที่มาสมัครงาน และได้รับผู้สมัครงานที่มีคุณภาพตรงตามความต้องการของบริษัท

Project Title	Pre-Employment Testing
Name	Satawat Thtisupakul
Student ID	60070093
Department	Information Technology
Advisor	AdvisorName AdvisorSurname
Year	2563

Abstract

Abstract

สารบัญ

	หน้า
กิตติกรรมประกาศ	vi
บทคัดย่อ	i
บทคัดย่อ ภาษาอังกฤษ	ii
สารบัญ	iii
สารบัญตาราง	iv
สารบัญภาพ	v
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์การปฏิบัติงาน	1
1.3 ประวัติ และรายละเอียดบริษัท	2
บทที่ 2 รายละเอียดการปฏิบัติงาน	4
2.1 ตำแหน่งและลักษณะงานที่นักศึกษาได้รับมอบหมาย	4
2.2 รายละเอียดของโครงการที่ได้รับผิดชอบ	4
2.3 แนวคิดและทฤษฎีที่เกี่ยวข้อง	4
2.4 เครื่องมือและเทคโนโลยีที่ใช้ในการปฏิบัติงาน	10
บทที่ 3 การออกแบบระบบและรายละเอียดการพัฒนา	16
3.1 ภาพรวมของเว็บแอปพลิเคชัน	16
3.2 วิเคราะห์ความต้องการ	17
3.3 การวิเคราะห์และออกแบบระบบ	19
บรรณานุกรม	36
บทที่ ก เรื่องที่หนึ่ง	37

สารบัญตาราง

	หน้า
ตารางที่ 3.1 รายละเอียดคุณสมบัติ ลงทะเบียน	20
ตารางที่ 3.2 รายละเอียดคุณสมบัติ เข้าสู่ระบบ	21
ตารางที่ 3.3 รายละเอียดคุณสมบัติ สร้างหมวดหมู่ของข้อสอบ	22
ตารางที่ 3.4 รายละเอียดคุณสมบัติ คู่มือหมวดหมู่ข้อสอบของผู้ใช้	23
ตารางที่ 3.5 รายละเอียดคุณสมบัติ แก้ไขหมวดหมู่ข้อสอบของผู้ใช้	24
ตารางที่ 3.6 รายละเอียดคุณสมบัติ คัดลอกหมวดหมู่ข้อสอบของผู้ใช้	25
ตารางที่ 3.7 รายละเอียดคุณสมบัติ ลบหมวดหมู่ข้อสอบของผู้ใช้	26
ตารางที่ 3.8 รายละเอียดคุณสมบัติ คู่มือหมวดหมู่ข้อสอบสาธารณะ	27
ตารางที่ 3.9 รายละเอียดคุณสมบัติ คู่มือรายละเอียดของหมวดหมู่ข้อสอบสาธารณะ	28
ตารางที่ 3.10 รายละเอียดคุณสมบัติ คัดลอกหมวดหมู่ข้อสอบสาธารณะ	29
ตารางที่ 3.11 รายละเอียดคุณสมบัติ สร้างข้อสอบ	30
ตารางที่ 3.12 รายละเอียดคุณสมบัติ คู่มือข้อสอบทั้งหมดของตนเอง	31
ตารางที่ 3.13 รายละเอียดคุณสมบัติ แก้ไขข้อสอบของตนเอง	32
ตารางที่ 3.14 รายละเอียดคุณสมบัติ แก้ไขข้อสอบของตนเอง	33
ตารางที่ 3.15 รายละเอียดคุณสมบัติ คัดลอกข้อสอบของตนเอง	34
ตารางที่ 3.16 รายละเอียดคุณสมบัติ ลบข้อสอบ	35

สารบัญภาพ

	หน้า
รูปที่ 1.1 ตราสัญลักษณ์ของ บริษัท ไซเงิน	2
รูปที่ 1.2 ตัวอย่างการให้บริการของ บริษัท ไซเงิน	3
รูปที่ 2.1 แสดง MVC architecture	8
รูปที่ 2.2 แสดงตาราง SQL เปรียบเทียบกับตาราง NoSQL	8
รูปที่ 2.3 แสดง MVVM architecture	9
รูปที่ 2.4 แสดงตัวอย่างการใช้งาน Trello	10
รูปที่ 2.5 แสดงตัวอย่างการใช้งาน Figma	10
รูปที่ 2.6 แสดงตัวอย่างการใช้งาน vscode	11
รูปที่ 2.7 แสดงตัวอย่างการทำงานของ docker	12
รูปที่ 2.8 แสดงตัวอย่างการใช้งาน postman	13
รูปที่ 2.9 แสดงตัวอย่าง jwt encoded	13
รูปที่ 2.10 แสดงตัวอย่างการใช้งาน Robo 3T	14
รูปที่ 2.11 แสดงตัวอย่างสถิติของอีเมลที่ทำการส่งด้วย sendgrid	15
รูปที่ 3.1 แสดงการทำงานของระบบประเมินความสามารถเบื้องต้นของผู้สมัครงาน	16
รูปที่ 3.2 แผนภาพยูสเคสของระบบ	19

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

บริษัท ไซเงิน เป็นบริษัทที่บริการและให้คำปรึกษาและบริการโซลูชัน SAP ในองค์กรต่างๆมากมาย และพัฒนาคิดค้นนวัตกรรมใหม่ๆ เพื่อช่วยแก้ปัญหาในองค์กร ทำให้ธุรกิจสามารถดำเนินไปได้ อย่างราบรื่นมากยิ่งขึ้น เช่น การทำ SAP BUSINESS ONE ช่วยทำให้เห็นภาพรวมของการทำธุรกิจได้อย่างชัดเจน, การทำ RPA (Robotic Process Automation) โรบอทที่จะมาช่วยทำงานออฟฟิศแทนมนุษย์ และ BUDDY RECRUIT เทคโนโลยีที่ช่วยให้บริษัทหาคนเข้ามาทำงานได้ตรงตามความต้องการของบริษัทนั้นๆ

การรับสมัครงาน เป็นกระบวนการที่ทุกๆบริษัทต้องมีเพื่อที่จะรับพนักงาน ที่มีความสามารถตรงตามที่บริษัทนั้นๆต้องการ ในปัจจุบันกระบวนการรับสมัครงานของแต่ละบริษัทส่วนใหญ่ ก่อนที่จะมีการสัมภาษณ์งานเกิดขึ้น ทางบริษัทจะดูข้อมูลของผู้สมัครผ่านทางเรซูเม่ เพื่อดูประสิทธิภาพการทำงาน และทักษะต่างๆที่ผู้สมัครมี แต่ถ้าหากมีผู้สมัครเป็นจำนวนมาก อาจทำให้ใช้ระยะเวลาในการคัดกรองผู้สมัครงานที่มากขึ้นตามไปด้วย และผู้ที่มีสำคัมนั้นอาจมีความสามารถไม่ตรงตามความต้องการของบริษัท

ในการปฏิบัติงานครั้งนี้ ผมได้เป็นในสมาชิกของทีม BUDDY RECRUIT โดยได้รับมอบหมายงานให้ทำเว็บแอปพลิเคชัน ประเมินความสามารถเบื้องต้นของผู้สมัครงาน เพื่อให้บริษัท สามารถรู้ได้เบื้องต้นว่าพนักงานที่มาสัคมมีความสามารถตรงตามความต้องการของบริษัทหรือไม่ และระบบนี้ยังมีความสามารถมากมายที่จะช่วยลดระยะเวลาในการสร้างข้อสอบเพื่อคัดกรองผู้ที่มีสำคัมนานอีกด้วย เช่น การคัดลอกข้อสอบที่มีพนักงานในบริษัทได้สร้างไว้แล้วสามารถแก้ไขข้อสอบนั้นๆได้, การส่งอีเมลให้ผู้สมัครงานทำข้อสอบโดยอัตโนมัติ หรือการสุ่มข้อสอบตามระดับความยากที่พนักงานได้กำหนดไว้

1.2 วัตถุประสงค์การปฏิบัติงาน

1. ศึกษาเทคโนโลยีใหม่ๆ ที่ใช้ในการทำงาน
2. พัฒนากระบวนการคิด และการลงมือทำในสายงาน Full Stack Developer
3. พัฒนาทักษะการทำงานภายใต้แรงกดดัน และระยะเวลาที่จำกัด
4. พัฒนาทักษะการแก้ไขปัญหาเฉพาะหน้า
5. ศึกษากระบวนการทำงานภายในทีม
6. พัฒนาทักษะการสื่อสาร
7. นำความรู้ที่ได้ ไปต่อยอดกับงานในอนาคต

1.3 ประวัติ และรายละเอียดบริษัท

1.3.1 ชื่อสถานประกอบการ

ชื่อบริษัท(ภาษาไทย): บริษัท ไชเจน จำกัด

ชื่อบริษัท(ภาษาอังกฤษ): ZyGen Company Limited

1.3.2 สถานที่ตั้ง

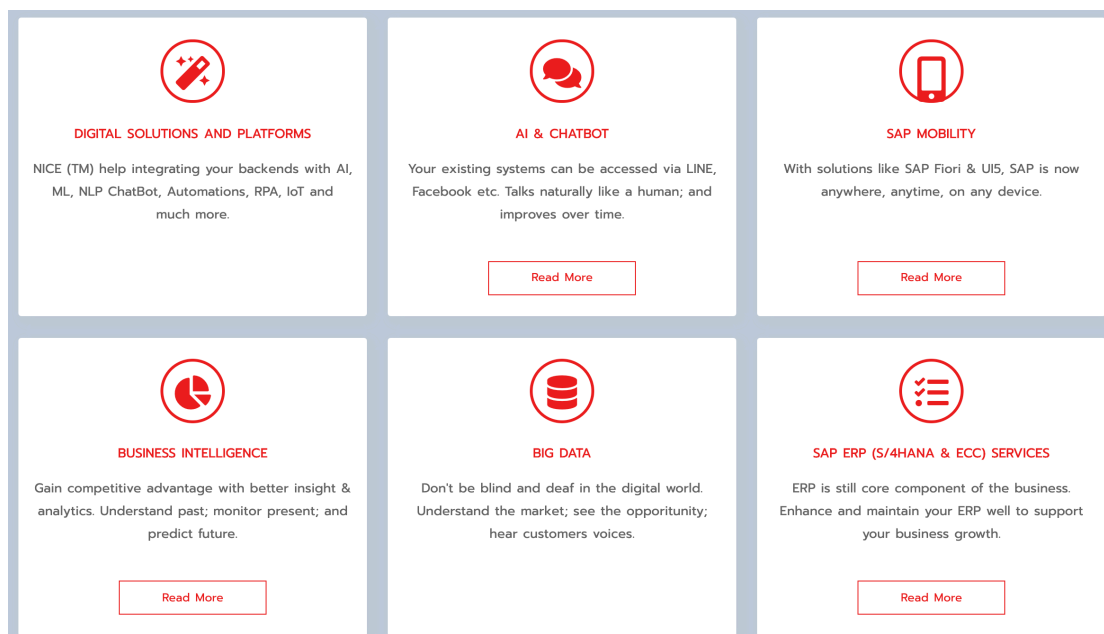
65/60 ชั้น 6 อาคาร ชำนาญเพ็ญชาติบิสเนสเซ็นเตอร์ ถนนพระราม 9 เขตห้วยขวาง, กรุงเทพฯ -
มหานคร 10310

1.3.3 ลักษณะสถานประกอบการ

บริษัท ไชเจน เป็นบริษัทที่บริการและให้คำปรึกษาและบริการโซลูชัน SAP ในองค์กรต่างๆมากมาย และพัฒนาคิดค้นนวัตกรรมใหม่ๆ เพื่อช่วยแก้ปัญหาในองค์กร ทำให้ธุรกิจสามารถดำเนินไปได้อย่างราบรื่นมากยิ่งขึ้น ไชเจน เป็นบริษัทที่มีฐานลูกค้าที่มั่นคง และยังเพิ่มฐานลูกค้าต่อไป ตัวอย่างการให้บริการของ ไชเจน เช่น การทำ SAP BUSINESS ONE ช่วยทำให้เห็นภาพรวมของการทำธุรกิจได้อย่างชัดเจน, การทำ RPA (Robotic Process Automation) โรบอทที่จะมาช่วยทำงานออฟฟิศแทนมนุษย์ และ BUDDY RECRUIT เทคโนโลยีที่ช่วยให้บริษัทหาคนเข้ามาทำงานได้ตรงตามความต้องการของบริษัทนั้นๆ



รูปที่ 1.1: ตราสัญลักษณ์ของ บริษัท ไชเจน



รูปที่ 1.2: ตัวอย่างการให้บริการของ บริษัท ไชเงิน

1.3.4 ตำแหน่งและลักษณะงานที่นักศึกษาได้รับมอบหมายให้รับผิดชอบ

ตำแหน่ง: Full Stack Developer

หน้าที่: พัฒนาเว็บแอปพลิเคชันตามความต้องการของลูกค้า

1.3.5 ชื่อและตำแหน่งของพนักงานที่ปรึกษา

ชื่อ-นามสกุล: นายวิวัฒน์ เสงี่ยมอนันต์

ตำแหน่ง: Web Developer

แผนก: Development Team

1.3.6 ระยะเวลาปฏิบัติงาน

ช่วงเวลาปฏิบัติงาน: 1 มิถุนายน 2563 - 30 กันยายน 2563

ช่วงเวลาปฏิบัติงาน: จันทร์ - ศุกร์ เวลา 09:00 น. - 18:00 น.

รวมระยะเวลา: 4 เดือน

บทที่ 2

รายละเอียดการปฏิบัติงาน

2.1 ตำแหน่งและลักษณะงานที่นักศึกษาได้รับมอบหมาย

ตำแหน่ง: Full Stack Developer

2.1.1 งานที่ได้รับผิดชอบ

1. ศึกษาเทคโนโลยีที่มีคุณภาพเพื่อประยุกต์ใช้กับการทำงาน
2. พัฒนาเว็บแอปพลิเคชันประเมินความสามารถเบื้องต้นของผู้สมัครงาน โดยเว็บแอปพลิเคชันนี้ช่วยให้คัดกรองผู้คนได้มีประสิทธิภาพมากยิ่งขึ้น ผ่านการทำข้อสอบที่สามารถกำหนดระดับความยากของข้อสอบได้ และดึงคลังคำถามมาแบบสุ่ม ส่งให้ผู้ทำข้อสอบโดยอัตโนมัติผ่านทางอีเมล เมื่อผู้สมัครงานทำข้อสอบเสร็จแล้ว ข้อสอบจะถูกส่งไปที่ผู้ออกข้อสอบ พร้อมตรวจข้อที่เป็นคำถามปรนัยให้อัตโนมัติ โดยระบบนี้แบ่งเป็น 2 ส่วนหลักๆ ได้แก่

2.2 รายละเอียดของโครงการที่ได้รับผิดชอบ

เนื่องจากนักศึกษาได้รับหน้าที่ในตำแหน่ง Developer ในทีม BUDDY RECRUIT ซึ่งทำเกี่ยวกับการช่วยให้บริษัทหาคนเข้ามาทำงาน ได้ตรงตามความต้องการของบริษัท จึงได้รับมอบหมายให้ทำเว็บแอปพลิเคชันประเมินความสามารถเบื้องต้นของผู้สมัครงาน โดยเว็บแอปพลิเคชันนี้ช่วยให้คัดกรองผู้คนได้มีประสิทธิภาพมากยิ่งขึ้น ผ่านการทำข้อสอบที่สามารถกำหนดระดับความยากของข้อสอบได้ และดึงคลังคำถามมาแบบสุ่ม ส่งให้ผู้ทำข้อสอบโดยอัตโนมัติผ่านทางอีเมล เมื่อผู้สมัครงานทำข้อสอบเสร็จแล้ว ข้อสอบจะถูกส่งไปที่ผู้ออกข้อสอบ พร้อมตรวจข้อที่เป็นคำถามปรนัยให้อัตโนมัติ โดยระบบนี้แบ่งเป็น 2 ส่วนหลักๆ ได้แก่

1. Front-end เป็นส่วนที่ผู้ใช้งานสามารถมีปฏิสัมพันธ์กับเว็บแอปพลิเคชันได้ ผ่าน UI (User Interface) ที่หน้าเว็บแอปพลิเคชัน
2. Back-end เป็นส่วนที่ผู้ใช้งานไม่สามารถมองเห็นได้จากภายนอก เป็นการทำงานหลักๆของเว็บแอปพลิเคชัน เช่น การเก็บข้อมูล การเรียกใช้ข้อมูล และการเชื่อมต่อการทำงานร่วมกับระบบอื่นๆ

อนึ่ง ข้อมูลข้างต้นเป็นเพียงภาพรวมของเว็บแอปพลิเคชันโดยย่อ ซึ่งรายละเอียดของเว็บแอปพลิเคชันนี้จะกล่าวโดยละเอียดในบทถัดไป

2.3 แนวคิดและทฤษฎีที่เกี่ยวข้อง

2.3.1 RESTful Web Services (RWS)

เป็น web service ที่ใช้ REST architectural style โดยจะอนุญาตให้ระบบ Request และเข้าถึง Resource บนเว็บโดยใช้ชุดคำสั่งที่กำหนดเอาไว้ การตอบโต้ของ REST อยู่บนพื้นฐานของ Hypertext Transfer

Protocol (HTTP) โดย Request จะส่งคำขอไปยัง URI ที่กำหนด และส่งข้อมูลกลับมาในรูปแบบ HTML, XML, JSON หรือ format อื่นๆ ทำให้สามารถบำรุงรักษาง่าย และสามารถ scale service ได้ [1]

- Client-server architecture: Client ผู้ที่เข้ามาขอ resources ไม่ต้องรู้ Business logic ภายใน ส่วน Server มีหน้าที่เก็บข้อมูล ไม่จำเป็นต้องรู้เกี่ยวกับ UI หรือสถานะของผู้เรียก
- Stateless: ส่ง request ให้เซิร์ฟเวอร์แล้วรับกลับมาเป็น response เมื่อรับ response มาแล้วจบการทำงาน
- Cache: สามารถกำหนดได้ว่าจะเก็บ cache ของ response นั้นหรือไม่
- Layered system: สามารถปรับปรุงความสามารถในการขยายระบบได้ โดยการใช้งานการ ทำ Load balance
- Interface/Uniform Contract: วิธีการที่จะคุยกับเซิร์ฟเวอร์โดยไม่คำนึงถึงประเภทของอุปกรณ์ หรือประเภทของแอปพลิเคชัน

2.3.2 HTTP Request

HTTP คือ protocol ที่อนุญาตให้ไคลเอนต์ดึงข้อมูลจากเซิร์ฟเวอร์ [1]

- Request-Line คือส่วนที่ระบุ HTTP Method, Request-URI และ version ของ protocol เช่น HTTP/1.0, HTTP/1.1, HTTP/2.0
- Headers คือส่วนที่อนุญาตให้ใส่ข้อมูลเพิ่มเติม หรือกฎเกณฑ์ต่างๆเกี่ยวกับการ request เช่น รูปแบบของข้อมูล, การเข้ารหัส
- Body คือส่วนที่ระบุข้อมูลที่ต้องการจะส่งให้ปลายทาง สามารถส่ง parameter ต่างๆ ไปใน Body เพื่อเพิ่ม, ลบ หรือแก้ไขข้อมูลบนเซิร์ฟเวอร์ได้

2.3.3 HTTP Request Methods

คือส่วนที่ใช้ในการกำหนดประเภทของคำร้องขอต่างๆ บน HTTP Request [1] โดยมี 4 Methods หลักคือ

- POST สำหรับใช้เพื่อสร้างค่าใหม่ เช่น สร้างรายชื่อพนักงานใหม่
- GET สำหรับขอข้อมูลจากเซิร์ฟเวอร์ เช่น ขอข้อมูลพนักงานทั้งหมด
- PUT สำหรับแก้ไขค่าต่างๆบนเซิร์ฟเวอร์ โดยส่งมาใน body ของ HTTP Request เช่น แก้ไขข้อมูลของพนักงาน
- DELETE สำหรับลบค่าบนเซิร์ฟเวอร์ เช่น ลบข้อมูลของพนักงาน

2.3.4 HTTP Response Status Code

คือมาตรฐานสถานะที่เซิร์ฟเวอร์ตอบสนองกับเว็บไซต์ต่างๆ [2]

- 2xx (Successful)

Request ที่ไคลเอนต์ส่งไปยังเซิร์ฟเวอร์ถูกประมวลผลเรียบร้อย และไม่มี error ใดๆ ประกอบด้วย

200 : (OK) ส่ง request สำเร็จ

201 : (Created) ผู้ใช้สร้างข้อมูลลง database สำเร็จ response นี้จะได้รับหลัง POST หรือ PUT requests

202 : (Accepted) request สำเร็จแล้วแต่ เซิร์ฟเวอร์ยังประมวลผลไม่เสร็จ

203 : (Non-Authoritative Information) เซิร์ฟเวอร์ประมวลผลสำเร็จแล้ว แต่ทำการส่งข้อมูลมาจากแหล่งอื่น

204 : (No Content) เซิร์ฟเวอร์ประมวลผลสำเร็จแล้ว แต่ไม่มีข้อมูลที่ต้องส่งคืนไป

206 : (Partial Content) เซิร์ฟเวอร์ส่งข้อมูลบางส่วน ตามที่ไคลเอนต์ต้องการ โดยกำหนดขอบเขตที่ต้องการผ่าน headers บน HTTP Request

- 3xx (Redirection)

request ที่ไคลเอนต์ส่งไปหาเซิร์ฟเวอร์ แล้วถูก redirect ส่งไปประมวลผลที่อื่น เพื่อให้กระบวนการทำงานสำเร็จ

300 : (Multiple Choices) request ที่ไคลเอนต์ส่งไปมี response มากกว่า 1 ตัว ไคลเอนต์สามารถเลือกสิ่งที่จะ redirect ไปได้

301 : (Moved Permanently) URL ที่ทำการ request ของข้อมูลถูกเปลี่ยนไปถาวร จึง response ออกมาเป็น URL ใหม่

302 : (Found) URL ที่ทำการ request มีการเปลี่ยนชั่วคราว

303 : (See Other) request ที่เรียกอยู่ภายใต้ URL อื่น

304 : (Not Modified) response นี้ยังไม่ถูกแก้ไข ดังนั้น ไคลเอนต์จะได้รับ response ที่เป็น cached version

- 4XX (Client Error)

เกิด error มาจาก request ของไคลเอนต์ที่ผิดพลาด เช่น ผิด URL หรือผิด syntax

400 : (Bad Request) ไคลเอนต์เขียน syntax ผิด หรือ ไม่ถูกรูปแบบ ทำให้เซิร์ฟเวอร์ไม่เข้าใจ

401 : (Unauthorized) ไคลเอนต์ต้องทำการยืนยันตัวตนก่อนที่จะได้รับ response

403 : (Forbidden) ไคลเอนต์ทำการยืนยันตัวตนแล้ว แต่ไม่มีสิทธิ์ในการเข้าถึงข้อมูลนี้

404 : (Not Found) ถ้าเกิดบน browser คือ URL ไม่ถูกจดจำบนเซิร์ฟเวอร์ แต่ถ้าเกิดบน API คือมีการขอข้อมูลที่ต้องการ แต่ไม่มีข้อมูลนี้อยู่

405 : (Method Not Allowed) method ที่เรียกใช้ไม่ถูกต้อง

406 : (Not Acceptable) header ที่ไคลเอนต์ request ไม่สัมพันธ์กับเซิร์ฟเวอร์

413 : (Payload Too Large) request ที่ขอใหญ่กว่า limit ที่เซิร์ฟเวอร์กำหนดไว้

414 : (URI Too Long) URL ที่ทำการ request โดยไคลเอนต์ ยาวกว่าที่เซิร์ฟเวอร์จะยอมรับได้

415 : (Unsupported Media Type) เซิร์ฟเวอร์ไม่รองรับ media (รูป หรือ สื่อต่างๆ) ดังนั้นเซิร์ฟเวอร์จึงปฏิเสธการ request

- 5XX (Server Error)

เซิร์ฟเวอร์มีปัญหา

500 : (Internal Server Error) เซิร์ฟเวอร์เจอกับสถานการณ์ที่ไม่สามารถจัดการได้

501 : (Not Implemented) ไคลเอนต์เรียก request method ที่เซิร์ฟเวอร์ไม่รองรับ และเซิร์ฟเวอร์ไม่สามารถจัดการได้

502 : (Bad Gateway) เซิร์ฟเวอร์เป็น gateway หรือ proxy ได้รับ response ที่ผิดพลาดจากเซิร์ฟเวอร์อื่น

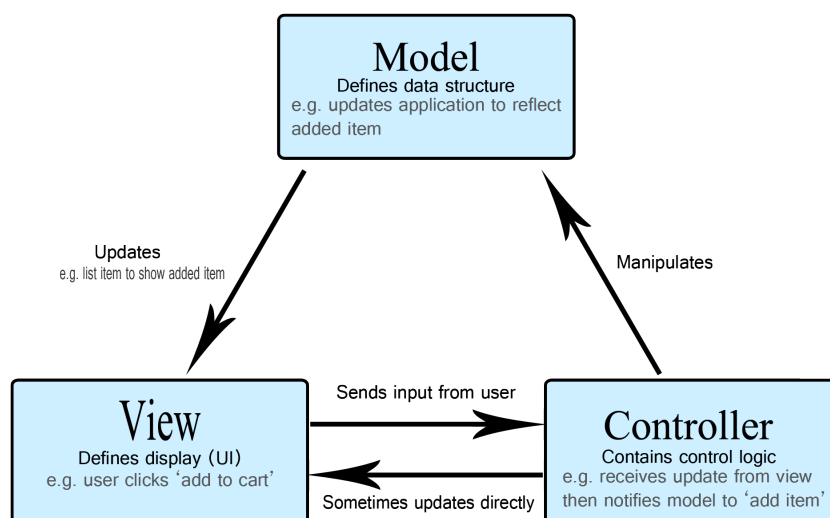
503 : (Service Unavailable) เซิร์ฟเวอร์อยู่ระหว่างการ ปรับปรุง หรือยังไม่พร้อมที่จะจัดการ request

504 : (Gateway Timeout) เซิร์ฟเวอร์เป็น gateway และไม่สามารถ response ข้อมูลในเวลาที่กำหนดได้

2.3.5 MVC (Model View Controller)

คือ software design pattern ที่แยกการทำงานของโปรแกรมออกเป็น 3 ส่วนเพื่อแยกข้อมูลภายในโปรแกรมกับข้อมูลที่แสดงให้ผู้ใช้เห็น [3]

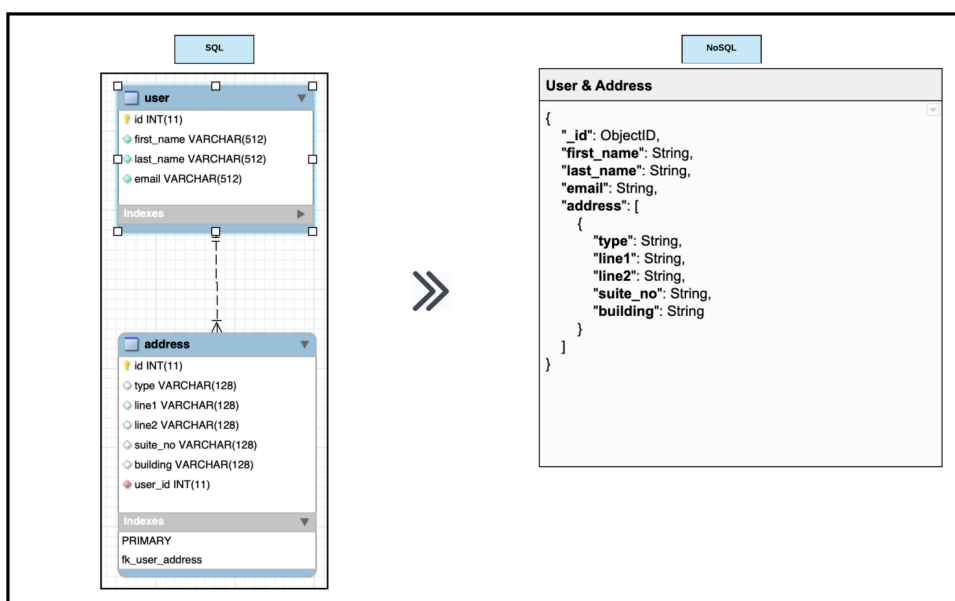
- Model คือส่วนที่เป็นโครงสร้างของข้อมูล กำหนดกฎเกณฑ์ของการเก็บข้อมูล และเป็นส่วนที่ใช้สำหรับการจัดการข้อมูลโดยตรง
- View คือส่วนที่ไว้แสดงผลตามที่ผู้ใช้ต้องการ ผู้ใช้สามารถเปลี่ยนแปลงข้อมูลที่ผู้ใช้เห็นได้
- Controller คือส่วนที่ไว้จัดการกับ Model โดยขึ้นอยู่กับการทำงานของ View ที่กำหนดโดยผู้ใช้ และสรรหาข้อมูลจาก Model เพื่อไปแสดงใน View



รูปที่ 2.1: แสดง MVC architecture

2.3.6 NoSQL Database

คือฐานข้อมูลที่สร้างมาเพื่อให้จัดการข้อมูลที่มีความซับซ้อนได้ง่ายขึ้นเมื่อเทียบกับ SQL databases ที่มีการเก็บข้อมูลในที่รูปแบบแน่นอน (structured data) โดยเพิ่มความสามารถในการจัดเก็บข้อมูลในรูปแบบที่ไม่แน่นอน (unstructured data) ทำให้สามารถเก็บข้อมูลที่ซับซ้อนได้, เพิ่มความสามารถในการขยายระบบในรูปแบบแนวนอน (Horizontal Scalability) เพื่อรองรับปริมาณข้อมูลในปัจจุบัน [4]



รูปที่ 2.2: แสดงตาราง SQL เปรียบเทียบกับตาราง NoSQL

2.3.7 Token

เป็นชุดรหัสเอาไว้ระบุตัวตนของผู้ใช้ว่าผู้ใช้นั้นเป็นใคร ไม่มีรูปแบบที่ตายตัว

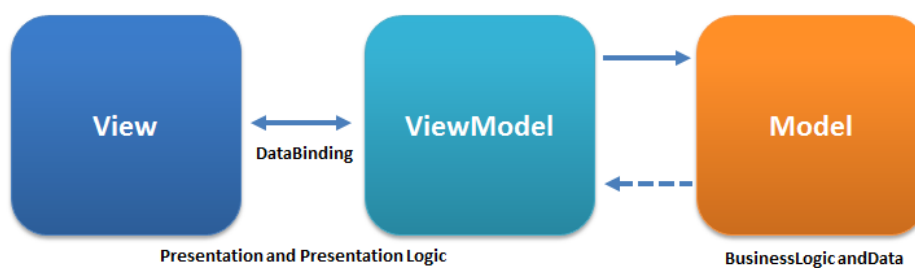
2.3.8 Container

เป็นหน่วยของซอฟต์แวร์ที่ทำการลงทรัพยากรทุกอย่างที่ต้องใช้ในแอป และตัวโค้ดของแอป ดังนั้นแอปพลิเคชันจะสามารถถูกเรียกใช้ได้อย่างรวดเร็ว และใช้ในสภาพแวดล้อมใดก็สามารถทำงานได้โดยมีความเป็นมาตรฐาน และประหยัดทรัพยากรที่ใช้ทำงานตัวแอปพลิเคชัน [5]

2.3.9 Model-view-viewmodel

เป็น software architectural pattern รูปแบบที่ช่วยแยกการพัฒนาแอปพลิเคชันออกเป็น 3 ส่วน โดยแบ่งออกจากกันอย่างชัดเจนเพื่อให้ง่ายต่อการจัดการและแสดงกระบวนการทำงานในแต่ละส่วน [6]

- Model คือส่วนที่เป็นโครงสร้างของข้อมูล กำหนดกฎเกณฑ์ของการเก็บข้อมูล และเป็นส่วนที่ไว้สำหรับการจัดการข้อมูลโดยตรง
- View คือส่วนที่ไว้แสดงผลตามที่ผู้ใช้ต้องการ ผู้ใช้สามารถเปลี่ยนแปลงข้อมูลได้
- View model มีหน้าที่เก็บข้อมูลทั้งหมดที่ View ต้องการ โดย View และ View model มีการใช้ Data-binding ซึ่งกันและกัน กล่าวคือถ้าข้อมูลของ View มีการแก้ไข ข้อมูลของ View model จะได้รับการแก้ไขไปด้วย หรือข้อมูลของ View model มีการแก้ไข ข้อมูลของ View จะได้รับการแก้ไขไปด้วย

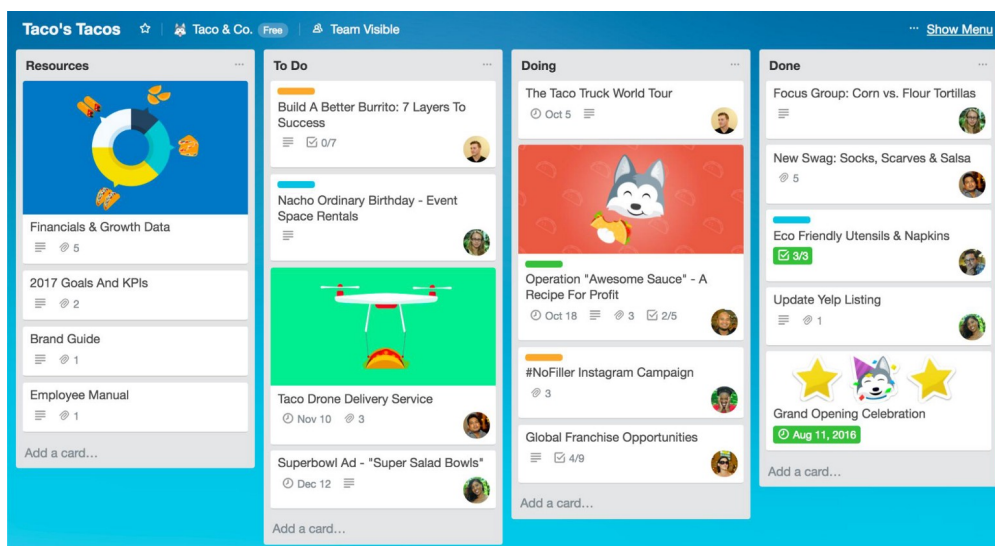


รูปที่ 2.3: แสดง MVVM architecture

2.4 เครื่องมือและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

2.4.1 Trello

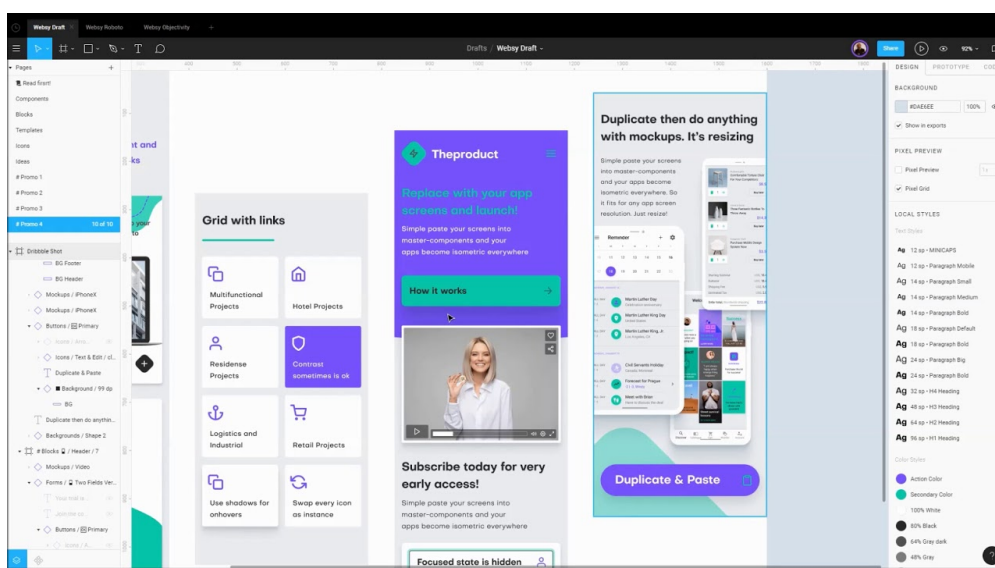
เป็นเว็บแอปพลิเคชันที่ช่วยจัดการบริหารโปรเจกต์ ให้เป็นระเบียบได้อย่างเรียบง่ายเปรียบเสมือนกระดานไว้วางแผนการทำงาน ระบุรายละเอียดของงานแต่ละงาน โดยเว็บแอปพลิเคชันนี้สามารถใช้งานร่วมกันภายในทีมได้ ทำให้การทำงานต่างๆในโปรเจกต์มีความคล่องตัวมากยิ่งขึ้น



รูปที่ 2.4: แสดงตัวอย่างการใช้งาน Trello

2.4.2 Figma

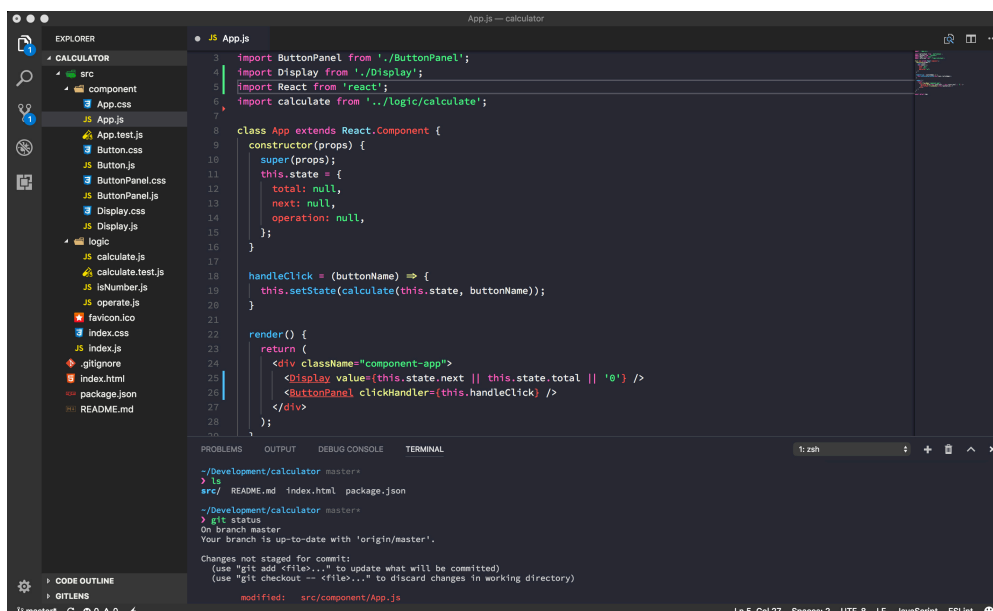
เป็นเครื่องมือที่นำมาใช้ในการออกแบบ prototype ของงานที่ทำ โดยสามารถใช้งานแอปพลิเคชัน Figma ได้ในทุกแพลตฟอร์ม



รูปที่ 2.5: แสดงตัวอย่างการใช้งาน Figma

2.4.3 Visual Studio Code

เป็นโปรแกรม code editor ที่ใช้ในการเขียน แก้ไข และปรับแต่งโค้ด โดยพัฒนาออกมาในรูปแบบ OpenSource โดยมีความสามารถในการเปลี่ยนสี Syntax ของโค้ด ตามแต่ละภาษาเพื่อให้ง่ายต่อการมอง และการเขียนโค้ด ซึ่งสนับสนุนภาษามากมาย ยกตัวอย่างเช่น ภาษา C++, Java, JavaScript, PHP, Python เป็นต้น และยังสามารถติดตั้งส่วนขยายที่เพิ่มความสะดวกสบายในการเขียนโค้ด ตามที่เราต้องการได้



รูปที่ 2.6: แสดงตัวอย่างการใช้งาน vscode

2.4.4 Vue.js

เป็น JavaScript Framework ที่สร้างส่วนต่อประสานกับผู้ใช้ (UI) และเป็น single-page applications คือ web application ที่ทำการโหลด page เพียงครั้งเดียวแต่สามารถเรียกข้อมูลอื่นๆแบบ dynamic ได้ ทำให้ประสบการณ์การใช้งานเว็บใกล้เคียง native app มากยิ่งขึ้น

2.4.5 Vuex

เป็น state management pattern ที่ผสมผสานกับ library เพื่อใช้สำหรับจัดการ state เพื่อให้ data flow ของโปรเจกต์ไปในทิศทางเดียวกัน code จึงมีความเป็นระบบมากยิ่งขึ้น และลดการเขียน code ที่ซ้ำซ้อน ทำให้การทำงานร่วมกันภายในทีมสะดวกมากยิ่งขึ้น เช่นการให้คนในทีมรับผิดชอบแต่ละ module [7]

2.4.6 Node.js

เป็น JavaScript runtime กล่าวคือเป็นตัวที่ทำให้ JavaScript สามารถใช้งานในส่วน of backend หรือเซิร์ฟเวอร์ได้

2.4.7 Express.js

เป็น Node.js web application framework ซึ่งมีฟีเจอร์ต่างๆที่ช่วยให้พัฒนาเซิร์ฟเวอร์ด้วย Node.js ได้สะดวกมากยิ่งขึ้น เช่นการจัดการ request และ response

2.4.8 Git

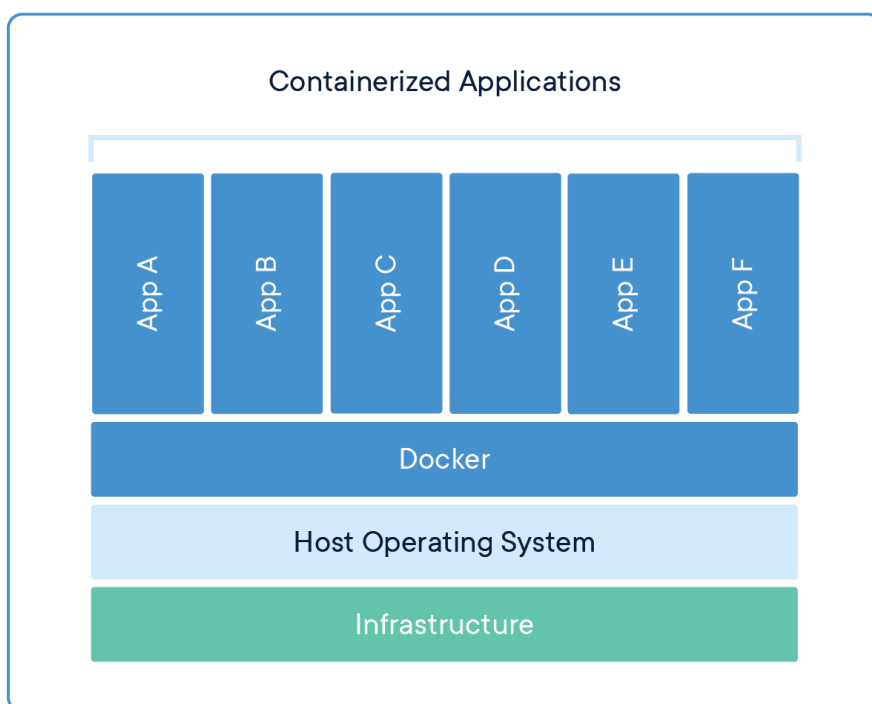
เป็น Version control ที่เอาไว้ติดตาม และควบคุมการเปลี่ยนแปลงของโค้ดเพื่อให้นักพัฒนา สามารถทำงานร่วมกันได้อย่างมีประสิทธิภาพ [8]

2.4.9 Github

เป็น website Git (version control repository) สำหรับนักพัฒนาซอฟต์แวร์ ใช้หลักการทำงานของ git แต่สามารถใช้งานร่วมกับผู้อื่นได้ผ่านอินเทอร์เน็ต

2.4.10 Docker

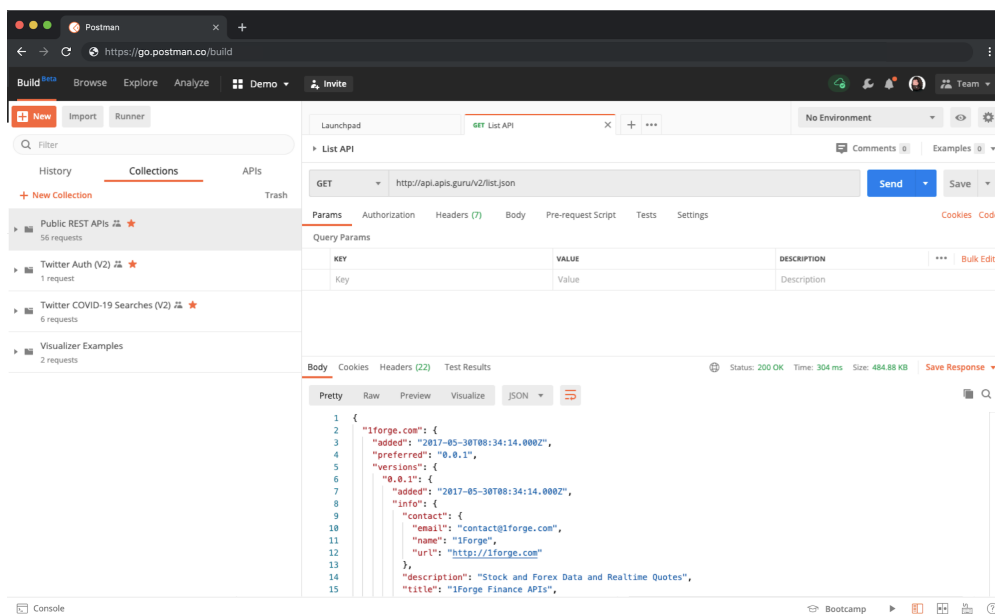
เป็น engine ที่มีการจำลองสภาพแวดล้อมขึ้นมาบนเครื่องเซิร์ฟเวอร์เพื่อทำการนำ service ที่ต้องการมาทำงานอยู่บนเซิร์ฟเวอร์ โดยใช้หลักการ container ในการจำลองสภาพแวดล้อมขึ้นมาทำงาน service ของเราโดยที่ไม่ต้องมีการนำ os เข้ามาใช้ ทำให้ service ทำงานในสภาพแวดล้อมใดก็ได้ [5]



รูปที่ 2.7: แสดงตัวอย่างการทำงานของ docker

2.4.11 Postman

เป็นเครื่องมือที่มาช่วยในการ API เพื่อทดสอบการทำงานของ Service โดยมีหน้าต่างส่วนต่อประสานกับผู้ใช้(UI) ที่สวยและใช้งานง่าย



รูปที่ 2.8: แสดงตัวอย่างการใช้งาน postman

2.4.12 JWT (Json Web Token)

เป็นมาตรฐาน RFC 7519 ในการยืนยันตัวตน (Authentication) ที่เข้ามาแก้ปัญหาการส่งข้อมูลระหว่างกันในรูปแบบดั้งเดิมคือ Server Based Authentication ที่เปลืองทรัพยากรในการเก็บ Session ID และไม่รองรับการขยายตัว (Scalability) โดย JWT นั้นสามารถเก็บข้อมูลภายในตัวได้ และมีขนาดที่กระทัดรัด เพื่อนำมาใช้กับ Single Page Web Application (SPA) [9] โดย JWT แบ่งโครงสร้างออกเป็น 3 ส่วน

1. Header เก็บประเภทของ token
2. Payload เก็บข้อมูล
3. Signature เป็นลายเซ็นที่อยู่ในรูปแบบของอิเล็กทรอนิกส์ (Digital Signed) เพื่อเช็คว่าเป็น token ที่ถูกสร้างอย่างถูกต้องหรือไม่

JWT TOKEN



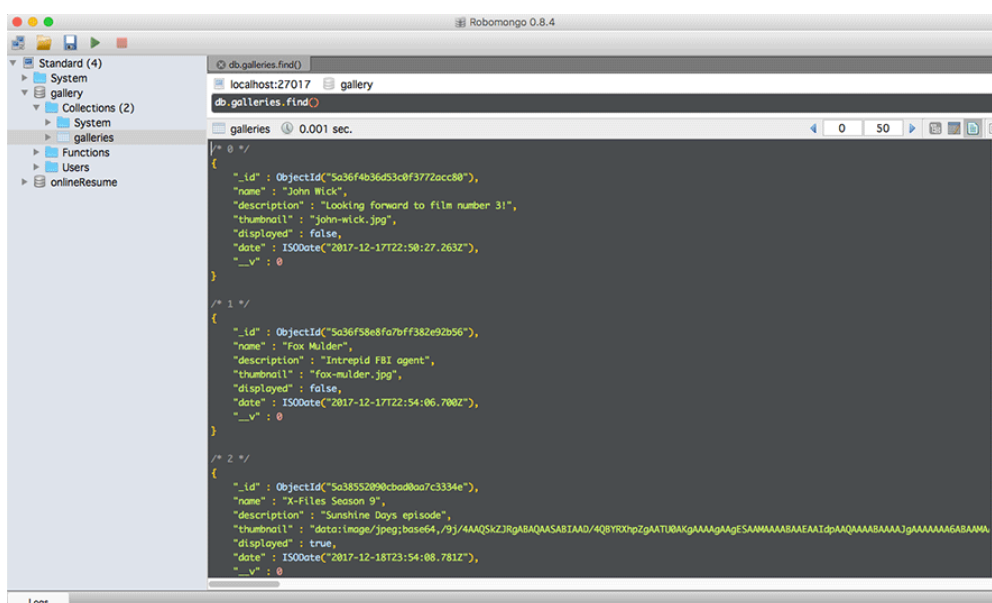
รูปที่ 2.9: แสดงตัวอย่าง jwt encoded

2.4.13 MongoDB

เป็น open-source document database ประเภทหนึ่ง โดยเก็บข้อมูลแบบ NoSQL Database และเก็บข้อมูลในรูปแบบของ JSON (JavaScript Object Notation) ซึ่งเก็บเป็น key และ value ซึ่งมีสมรรถภาพสูงกว่าการเก็บด้วยโครงสร้างแบบแถวและหลัก(row/column) แบบดั้งเดิม [10]

2.4.14 Robo 3T

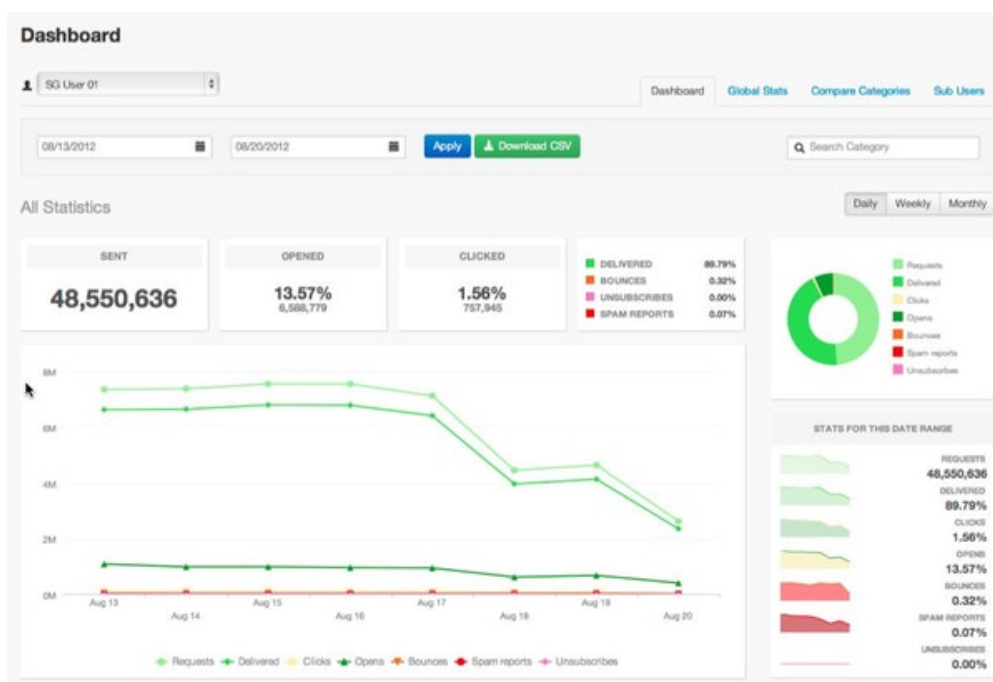
เป็นการใช้ภาพเป็นตัวประสานกับผู้ใช้ของ(GUI) ของ MongoDB ช่วยทำให้ใช้งาน MongoDB ได้สะดวกมากยิ่งขึ้น เช่น เขียน SQL เพื่อ query ข้อมูลใน MongoDB หรือ Import และ export ไฟล์ข้อมูลเป็น CSV, JSON, SQL and BSON/mongodump ได้ [?]



รูปที่ 2.10: แสดงตัวอย่างการใช้งาน Robo 3T

2.4.15 SendGrid

เป็น API ช่วยในการส่งอีเมลให้ผู้อื่น สามารถตรวจสอบอีเมลที่ส่งว่าส่งไปถึงหรือไม่ มีปัญหาอะไรเกิดขึ้นหรือเปล่า และมีการเก็บสถิติข้อมูลอีเมลที่ส่ง สรุปออกมาแสดงให้วิเคราะห์เช่น มีการเปิดเมลอ่านกี่ครั้ง มีการรายงานว่าเป็นสแปมกี่ครั้ง



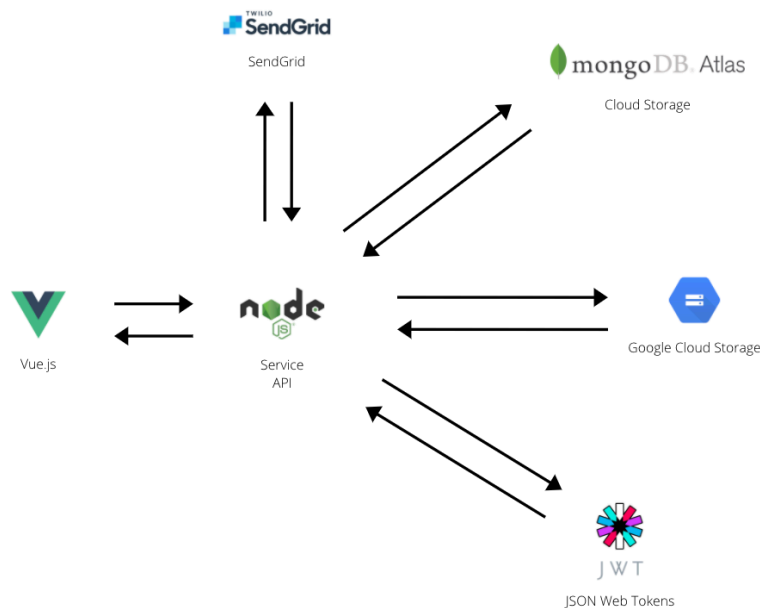
รูปที่ 2.11: แสดงตัวอย่างสถิติของอีเมลที่ทำการส่งด้วย sendgrid

บทที่ 3

การออกแบบระบบและรายละเอียดการพัฒนา

3.1 ภาพรวมของเว็บแอปพลิเคชัน

เว็บแอปพลิเคชันประเมินความสามารถเบื้องต้นของผู้สมัครงาน คือระบบที่จะช่วยให้คัดกรองผู้คนที่ได้มีประสิทธิภาพมากยิ่งขึ้น ผ่านการทำข้อสอบที่สามารถกำหนดระดับความยากของข้อสอบได้ และดึงคำถามมาแบบสุ่ม ส่งให้ผู้ทำข้อสอบโดยอัตโนมัติผ่านทางอีเมล เมื่อผู้สมัครงานทำข้อสอบเสร็จแล้ว ข้อสอบจะถูกส่งไปที่ผู้ออก พร้อมตรวจข้อที่เป็นคำถามปลายปิดให้อัตโนมัติ โดยระบบนี้แบ่งเป็น 2 ส่วนหลักๆ คือในส่วนของหน้าบ้านที่ใช้เป็น Vue.js framework และส่วนของหลังบ้านที่ใช้ Node.js framework การเก็บข้อมูลทั่วไปถูกเก็บอยู่ใน mongoDB Atlas การเก็บไฟล์ เช่น ไฟล์รูปภาพ หรือไฟล์ต่างๆ ที่ถูกใช้ในการทำข้อสอบจะถูกเก็บอยู่บน Google Cloud Storage การจัดการอีเมลใช้บริการของ SendGrid เข้ามาช่วยจัดการ และการยืนยันตัวตนจะถูกเข้ารหัสด้วย JWS Web Token ภาพรวมการทำงานของระบบจะเป็นดังรูป 3.1



รูปที่ 3.1: แสดงการทำงานของระบบประเมินความสามารถเบื้องต้นของผู้สมัครงาน

3.2 วิเคราะห์ความต้องการ

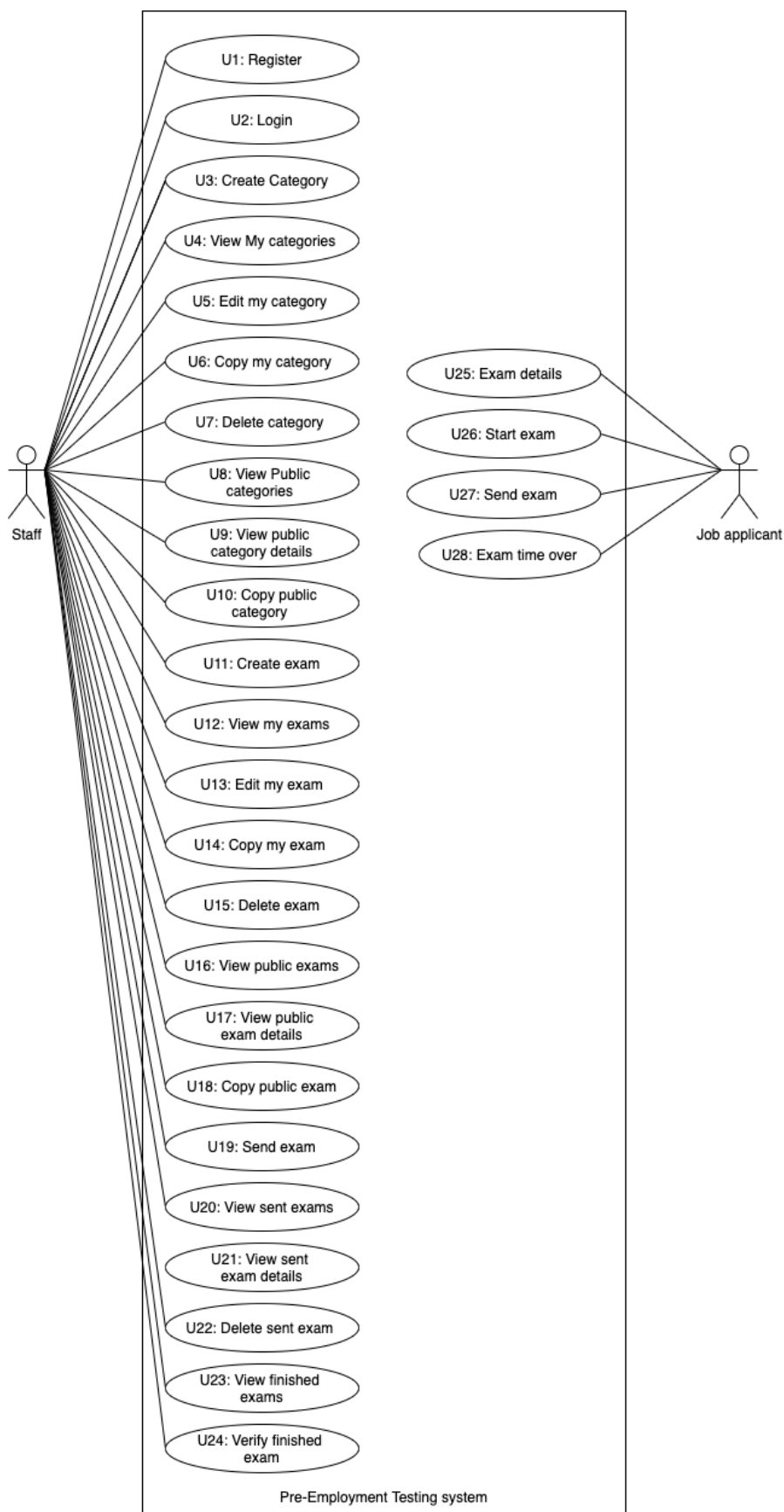
3.2.1 ความต้องการที่เป็นหน้าที่หลักของระบบ (Functional Requirement)

1. ผู้ใช้งานสามารถสามารถลงทะเบียนเป็นสมาชิกเพื่อใช้งานเว็บแอปพลิเคชันได้
2. ผู้ใช้ต้องยืนยันตัวตนผ่านอีเมลหลังจากทำการลงทะเบียน ถึงจะเข้าใช้งานระบบได้
3. ผู้ใช้งานสามารถสร้าง, แก้ไข และลบหมวดหมู่ข้อสอบได้
4. ระบบสามารถป้องกันการรีเฟรชเพจและการเปลี่ยนหน้าได้ หากผู้ใช้งานกำลังสร้างหรือแก้ไขหมวดหมู่ข้อสอบอยู่
5. ผู้ใช้งานสามารถยกเลิกการทำข้อสอบได้
6. ผู้ใช้งานไม่สามารถลบหมวดหมู่ข้อสอบได้ หากถูกใช้งานอยู่ในข้อสอบ
7. หากหมวดหมู่ข้อสอบถูกส่งไปให้ผู้สมัครงานทำแล้ว หมวดหมู่ข้อสอบนั้น สามารถแก้ไขได้ เฉพาะคำตอบเท่านั้น ไม่สามารถแก้ไขคำถามได้
8. ผู้ใช้งานสามารถสร้างคำถามประเภท ปรนัย, อัตนัย และคำถามที่มีไฟล์แนบได้
9. คำถามที่เป็นปรนัย สามารถเลือกตอบมากกว่า 1 คำถามได้
10. สามารถเพิ่มรูปภาพในคำตอบที่เป็นปรนัยได้
11. ผู้ใช้งานสามารถเปิดหมวดหมู่ข้อสอบให้เป็นสาธารณะได้
12. หมวดหมู่ข้อสอบที่เป็นสาธารณะ ผู้ที่ไม่ใช่เจ้าของไม่สามารถแก้ไขได้ แต่ระบบสามารถคัดลอกหมวดหมู่ข้อสอบสาธารณะไปเป็นเป็นของตนเองได้ หากผู้ที่ไม่ใช่เจ้าของต้องการทำการแก้ไข
13. ผู้ใช้งานสามารถค้นหาหมวดหมู่ข้อสอบได้
14. ผู้ใช้งานสามารถสร้างข้อสอบ ที่กำหนดหมวดหมู่ข้อสอบที่ต้องการเลือกคำถาม, จำนวนข้อ และความยากได้ โดยข้อสอบจะทำการสุ่มเมื่อผู้ใช้ส่งให้ผู้ที่ทำข้อสอบ
15. ผู้ใช้สามารถลบแก้ไขหมวดหมู่ข้อสอบที่ใช้ จำนวนข้อ และระดับความยากของข้อสอบได้
16. ผู้ใช้สามารถค้นหาข้อสอบได้
17. ผู้ใช้สามารถเปิดข้อสอบให้เป็นสาธารณะได้
18. ข้อสอบที่เป็นสาธารณะ ผู้ที่ไม่ใช่เจ้าของไม่สามารถแก้ไขได้ โดยระบบสามารถคัดลอกข้อสอบสาธารณะไปเป็นเป็นของตนเองได้ หากผู้ที่ไม่ใช่เจ้าของต้องการทำการแก้ไข
19. ผู้ใช้สามารถกำหนดระยะเวลาในการทำข้อสอบได้

20. ผู้ใช้สามารถกำหนดวันหมดอายุของข้อสอบได้
21. ผู้ใช้สามารถส่งข้อสอบให้ผู้สมัครงานด้วยอีเมลผ่านเว็บแอปพลิเคชันได้
22. ผู้ทำข้อสอบสามารถเข้ามาทำข้อสอบในเว็บแอปพลิเคชันได้ตลอด หากเวลาในการทำข้อสอบยังไม่หมด
23. เมื่อผู้ทำข้อสอบทำการส่งข้อสอบแล้ว ระบบสามารถตรวจข้อสอบที่เป็นปรนัยได้
24. มีการแสดงสถานะบอกว่าผู้ใช้งาน ตรวจข้อสอบนั้นหรือยัง
25. มีการสรุปคะแนนรวมข้อผู้สมัครงาน

3.3 การวิเคราะห์และออกแบบระบบ

3.3.1 แผนภาพยูสเคส (Use Case Diagram)



รูปที่ 3.2: แผนภาพยูสเคสของระบบ

3.3.2 รายละเอียดการทำงานในแต่ละยูสเคส (Use Case Description)

3.3.2.1 รายละเอียดยูสเคส ลงทะเบียน

Use Case No:	1	
Use Case Name:	ลงทะเบียน	
Use Case Scenario:	ผู้ลงทะเบียนกับระบบ	
Triggering Event:	เมื่อผู้ใช้เข้าใช้งานเป็นครั้งแรก	
Brief Description:	สำหรับให้ผู้ใช้ลงทะเบียนเข้าสู่ระบบ	
Actors:	พนักงาน	
Related Use Cases:	-	
Stakeholders:	-	
Pre - Conditions:	-	
Post - Conditions:	ผู้ใช้สามารถเข้าใช้งานระบบได้	
Flow of Events:	ผู้ใช้	ระบบ
	1.ผู้ใช้เลือกลงทะเบียน 3.ผู้ใช้ทำการกรอกชื่อ-นามสกุล,อีเมล,รหัสผ่านและยืนยันรหัสผ่านแล้วทำการคลิกลงทะเบียน 5.ผู้ใช้งานได้รับอีเมลแล้วทำการคลิกลิงค์เว็บไซต์	2.ระบบแสดงแบบฟอร์มให้กรอกข้อมูล 4.ระบบทำการส่งอีเมลให้ผู้ใช้เพื่อทำการยืนยันอีเมล 6.ระบบทำการยืนยันอีเมลแล้วแสดงหน้าจอหลักของผู้ใช้งาน
Exception Conditions:	<ul style="list-style-type: none"> - ผู้ใช้กรอกรายละเอียดไม่ครบถ้วน หรือไม่ตรงเงื่อนไข - ผู้ใช้ไม่ได้ทำการยืนยันอีเมล - ผู้ใช้กรอกอีเมลที่ไม่มีอยู่จริง - ผู้ใช้กรอกอีเมลที่ผู้ใช้งานไม่สามารถเข้าใช้อีเมลนั้นได้ 	

ตารางที่ 3.1: รายละเอียดยูสเคส ลงทะเบียน

3.3.2.2 รายละเอียดยูสเคส เข้าสู่ระบบ

Use Case No:	2	
Use Case Name:	เข้าสู่ระบบ	
Use Case Scenario:	ผู้ใช้งานต้องการเข้าสู่ระบบ	
Triggering Event:	<ul style="list-style-type: none"> - เมื่อผู้ใช้งานต้องการเข้าใช้งานระบบ - ผู้ใช้เข้าเว็บแอปพลิเคชันโดยที่ยังไม่เข้าใช้งานระบบ 	
Brief Description:	สำหรับให้ผู้ใช้งานเข้าสู่ระบบเพื่อเข้าใช้งานเว็บแอปพลิเคชัน	
Actors:	พนักงาน	
Related Use Cases:	ลงทะเบียน	
Stakeholders:	-	
Pre - Conditions:	ผู้ใช้งานต้องการลงทะเบียนและยืนยันตัวตนผ่านอีเมลเรียบร้อยแล้ว	
Post - Conditions:	ระบบแสดงหน้าจอสำหรับผู้ใช้งาน	
Flow of Events:	ผู้ใช้	ระบบ
	1. ผู้ใช้เข้าหน้าเว็บโดยที่ยังไม่ได้ทำการเข้าสู่ระบบ 3. ผู้ใช้ทำการกรอกอีเมล และรหัสผ่านแล้วทำการคลิกเข้าสู่ระบบ หรือ กด Enter บนแป้นพิมพ์	2. ระบบแสดงแบบฟอร์มให้กรอกข้อมูล 4. ระบบ แสดง หน้า จอ หลักของผู้ใช้งาน
Exception Conditions:	ผู้ใช้งานกรอกอีเมล หรือรหัสผ่านผิด	

ตารางที่ 3.2: รายละเอียดยูสเคส เข้าสู่ระบบ

3.3.2.3 รายละเอียดยูสเคส สร้างหมวดหมู่ของข้อสอบ

Use Case No:	3	
Use Case Name:	สร้างหมวดหมู่ของข้อสอบ	
Use Case Scenario:	ผู้ใช้งานต้องการสร้างหมวดหมู่ของข้อสอบ	
Triggering Event:	เมื่อผู้ใช้งานต้องการสร้างหมวดหมู่ของข้อสอบ ไว้ใช้ในข้อสอบของตนเอง หรือให้พนักงานคนอื่นใช้	
Brief Description:	สำหรับให้ผู้ใช้งาน ออกข้อสอบโดยแบ่งแยกตามแต่ละหมวดหมู่ เช่น คณิตศาสตร์, โปรแกรมมิ่ง หรือ แบบทดสอบสติปัญญา โดยมีชนิดคำถามแบ่งออกเป็น อัตนัย, ปรนัย และคำถามที่มีไฟล์แนบ	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ	
Stakeholders:	-	
Pre - Conditions:	ผู้ใช้งานต้องการเข้าสู่ระบบ	
Post - Conditions:	ผู้ใช้งานมีหมวดหมู่ของข้อสอบที่ตนเองสร้างขึ้น และแสดงหน้าหมวดหมู่ทั้งหมดของผู้ใช้	
Flow of Events:	ผู้ใช้งาน	ระบบ
	1.ไปที่หน้าสร้างหมวดหมู่ 3.ผู้ใช้งานกรอกชื่อหมวดหมู่ และกรอกหัวเรื่อง 4.สร้างคำถามภายใต้ หมวดหมู่นั้นโดยสามารถเลือกระดับความยาก และประเภทของคำถาม แล้วทำการคลิกสร้างหมวดหมู่	2.ระบบแสดงหน้าสร้างหมวดหมู่ของข้อสอบ 5.ระบบทำการสร้างหมวดหมู่ของข้อสอบ แล้วแสดงหน้าหมวดหมู่ข้อสอบทั้งหมดของผู้ใช้
Exception Conditions:	<ul style="list-style-type: none"> - ผู้ใช้งานกรอกชื่อหมวดหมู่ซ้ำกับ หมวดหมู่ที่ตนเองมีอยู่แล้ว - ผู้ใช้งานกรอกข้อมูลไม่ครบ ตามฟอร์มที่ให้กรอก - ไฟล์ที่แนบมามีขนาดใหญ่เกินที่กำหนด หรือประเภทของไฟล์ไม่ถูกต้อง 	

ตารางที่ 3.3: รายละเอียดยูสเคส สร้างหมวดหมู่ของข้อสอบ

3.3.2.4 รายละเอียดยูสเคส ดูหมวดหมู่ข้อสอบของผู้ใช้

Use Case No:	4	
Use Case Name:	ดูหมวดหมู่ข้อสอบของผู้ใช้	
Use Case Scenario:	ผู้ใช้งานต้องการดูหมวดหมู่ข้อสอบของตนเอง	
Triggering Event:	<ul style="list-style-type: none"> - เมื่อผู้ใช้งานต้องการดูหมวดหมู่ของข้อสอบของตนเอง - เมื่อผู้ใช้งานสร้างประเภทของข้อสอบสำเร็จ หน้าเว็บจะทำการไปแสดงผลที่หน้าหมวดหมู่ข้อสอบทั้งหมดของผู้ใช้ 	
Brief Description:	สำหรับให้ผู้ใช้งานดูหมวดหมู่ของข้อสอบที่ตนเองสร้างขึ้น หรือหมวดหมู่ที่ไปคัดลอกมาจากหมวดหมู่สาธารณะ	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, สร้างหมวดหมู่ของข้อสอบ, คัดลอกหมวดหมู่ข้อสอบจากหมวดหมู่ข้อสอบสาธารณะ	
Stakeholders:	-	
Pre - Conditions:	<ul style="list-style-type: none"> - ผู้ใช้ต้องทำการเข้าสู่ระบบ - สร้างหมวดหมู่ของข้อสอบ หรือคัดลอกหมวดหมู่ข้อสอบจากหมวดหมู่ข้อสอบสาธารณะ 	
Post - Conditions:	แสดงหน้าหมวดหมู่ทั้งหมดของผู้ใช้	
Flow of Events:	ผู้ใช้งาน	ระบบ
	1.ไปที่หน้าดูหมวดหมู่ข้อสอบของผู้ใช้	2.ระบบแสดงหมวดหมู่ทั้งหมดของผู้ใช้
Exception Conditions:	ผู้ใช้งานไม่มีหมวดหมู่ข้อสอบ	

ตารางที่ 3.4: รายละเอียดยูสเคส ดูหมวดหมู่ข้อสอบของผู้ใช้

3.3.2.5 รายละเอียดยูสเคส แก้ไขหมวดหมู่ข้อสอบของผู้ใช้

Use Case No:	5	
Use Case Name:	แก้ไขหมวดหมู่ข้อสอบของผู้ใช้	
Use Case Scenario:	ผู้ใช้งานต้องการแก้ไขหมวดหมู่ข้อสอบของตนเอง	
Triggering Event:	เมื่อผู้ใช้งานต้องการแก้ไขรายละเอียดในหมวดหมู่ข้อสอบ เช่น เพิ่มคำถาม, แก้ไขคำถาม หรือแก้ไขชื่อหมวดหมู่	
Brief Description:	สำหรับให้ผู้ใช้งานแก้ไขรายละเอียดต่างๆในหมวดหมู่ข้อสอบ	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, สร้างหมวดหมู่ของข้อสอบ, คัดลอกหมวดหมู่ข้อสอบจากหมวดหมู่ข้อสอบสาธารณะ	
Stakeholders:	-	
Pre - Conditions:	<ul style="list-style-type: none"> - ผู้ใช้ต้องทำการเข้าสู่ระบบ - สร้างหมวดหมู่ของข้อสอบ หรือคัดลอกหมวดหมู่ข้อสอบจากหมวดหมู่ข้อสอบสาธารณะ 	
Post - Conditions:	แสดงหน้าหมวดหมู่ทั้งหมดของผู้ใช้	
Flow of Events:	<p>ผู้ใช้งาน</p> <ol style="list-style-type: none"> 1.ไปที่หน้าคู่มือหมวดหมู่ข้อสอบของผู้ใช้ 3.คลิกที่หมวดหมู่ของข้อสอบที่ต้องการแก้ไข 5.ผู้ใช้งานทำการแก้ไขข้อมูลต่างๆในประเภทที่ต้องการแล้วทำการคลิกยืนยันการแก้ไข 	<p>ระบบ</p> <ol style="list-style-type: none"> 2.ระบบแสดงหมวดหมู่ทั้งหมดของผู้ใช้ 4.ระบบแสดงหน้าแก้ไขหมวดหมู่นั้นๆ 6.ระบบทำการแก้ไขข้อมูล 7.ระบบแสดงหน้าหมวดหมู่ทั้งหมดของผู้ใช้
Exception Conditions:	<ul style="list-style-type: none"> - ผู้ใช้กรอกชื่อหมวดหมู่ซ้ำกับ หมวดหมู่ที่ตนเองมีอยู่แล้ว - ผู้ใช้กรอกรายละเอียดไม่ครบถ้วน หรือไม่ตรงเงื่อนไข - ไฟล์ที่แนบมามีขนาดใหญ่เกินที่กำหนด หรือประเภทของไฟล์ไม่ถูกต้อง - สามารถแก้ไขได้แค่คำตอบ หากข้อสอบเคยถูกส่งไปแล้ว 	

ตารางที่ 3.5: รายละเอียดยูสเคส แก้ไขหมวดหมู่ข้อสอบของผู้ใช้

3.3.2.6 รายละเอียดยูสเคส คัดลอกหมวดหมู่ข้อสอบของผู้ใช้

Use Case No:	6	
Use Case Name:	คัดลอกหมวดหมู่ข้อสอบของผู้ใช้	
Use Case Scenario:	ผู้ใช้งานต้องการคัดลอกหมวดหมู่ข้อสอบของตนเอง	
Triggering Event:	เมื่อผู้ใช้งานต้องการแก้ไขหมวดหมู่ของข้อสอบที่ถูกส่งให้ผู้สมัครงานแล้ว ทำให้ไม่สามารถแก้ไขหมวดหมู่เดิมได้ หรือต้องการคัดลอกหมวดหมู่มาแก้ไข โดยที่ไม่ต้องการให้กระทบหมวดหมู่เดิม	
Brief Description:	สำหรับให้ผู้ใช้งานคัดลอกหมวดหมู่ข้อสอบของตนเอง	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, สร้างหมวดหมู่ของข้อสอบ, ส่งข้อสอบให้ผู้สมัคร	
Stakeholders:	-	
Pre - Conditions:	<ul style="list-style-type: none"> - ผู้ใช้ต้องทำการเข้าสู่ระบบ - สร้างหมวดหมู่ของข้อสอบ หรือคัดลอกหมวดหมู่ข้อสอบจากหมวดหมู่ข้อสอบสาธารณะ 	
Post - Conditions:	ระบบทำการคัดลอกหมวดหมู่ของข้อสอบที่ผู้ใช้งานต้องการ	
Flow of Events:	ผู้ใช้งาน	ระบบ
	1.ไปที่หน้าคู่มือหมวดหมู่ข้อสอบของผู้ใช้ 3.คลิกปุ่มคัดลอกที่หมวดหมู่ที่ต้องการคัดลอก 5.ผู้ใช้งานกดยืนยันการคัดลอก	2.ระบบแสดงหมวดหมู่ทั้งหมดของผู้ใช้ 4.ระบบแสดงหน้ายืนยันการคัดลอก 6.ระบบทำการคัดลอกหมวดหมู่ข้อสอบนั้น
Exception Conditions:	ผู้ใช้งานไม่มีหมวดหมู่ข้อสอบ	

ตารางที่ 3.6: รายละเอียดยูสเคส คัดลอกหมวดหมู่ข้อสอบของผู้ใช้

3.3.2.7 รายละเอียดยูสเคส ลบหมวดหมู่ข้อสอบของผู้ใช้

Use Case No:	7	
Use Case Name:	ลบหมวดหมู่ข้อสอบของผู้ใช้	
Use Case Scenario:	ผู้ใช้งานต้องการลบหมวดหมู่ข้อสอบของตนเอง	
Triggering Event:	เมื่อผู้ใช้งานไม่ต้องการที่จะใช้หมวดหมู่ข้อสอบนั้นๆแล้ว	
Brief Description:	สำหรับให้ผู้ใช้งานลบหมวดหมู่ข้อสอบออกจากหมวดหมู่ข้อสอบของตนเอง	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, สร้างหมวดหมู่ของข้อสอบ, คัดลอกหมวดหมู่ข้อสอบจากหมวดหมู่ข้อสอบสาธารณะ, ข้อสอบของผู้ใช้	
Stakeholders:	-	
Pre - Conditions:	<ul style="list-style-type: none"> - ผู้ใช้ต้องทำการเข้าสู่ระบบ - สร้างหมวดหมู่ของข้อสอบ หรือคัดลอกหมวดหมู่ข้อสอบจากหมวดหมู่ข้อสอบสาธารณะ 	
Post - Conditions:	ระบบลบหมวดหมู่ข้อสอบของผู้ใช้	
Flow of Events:	ผู้ใช้งาน	ระบบ
	1.ไปที่หน้าดูหมวดหมู่ข้อสอบของผู้ใช้ 3.คลิกที่รูปถึงขยะของหมวดหมู่ข้อสอบที่ต้องการลบ 5.ผู้ใช้งานทำการยืนยันการลบ	2.ระบบแสดงหมวดหมู่ทั้งหมดของผู้ใช้ 4.ระบบทำการแสดงหน้าต่างข้อความยืนยันการลบ 6.ระบบทำการลบหมวดหมู่ข้อสอบที่ผู้ใช้งานต้องการลบ
Exception Conditions:	<ul style="list-style-type: none"> - ผู้ใช้ไม่ทำการยืนยันการลบ - หมวดหมู่ข้อสอบที่ต้องการลบ ถูกใช้งานอยู่ในข้อสอบ ทำให้ไม่สามารถลบได้ 	

ตารางที่ 3.7: รายละเอียดยูสเคส ลบหมวดหมู่ข้อสอบของผู้ใช้

3.3.2.8 รายละเอียดยูสเคส ดูหมวดหมู่ข้อสอบสาธารณะ

Use Case No:	8	
Use Case Name:	ดูหมวดหมู่ข้อสอบสาธารณะ	
Use Case Scenario:	ผู้ใช้งานต้องการดูหมวดหมู่ข้อสอบสาธารณะ	
Triggering Event:	<ul style="list-style-type: none"> - เมื่อผู้ใช้งานต้องการดูหมวดหมู่ข้อสอบสาธารณะ - เมื่อผู้ใช้งานต้องการนำหมวดหมู่ข้อสอบสาธารณะไปใช้ในข้อสอบของตนเอง 	
Brief Description:	สำหรับให้ผู้ใช้งานดูว่ามีหมวดหมู่ข้อสอบสาธารณะใดบ้างที่อยู่ในระบบและสามารถคัดลอกไปใช้ได้	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ	
Stakeholders:	-	
Pre - Conditions:	ผู้ใช้งานต้องการเข้าสู่ระบบ	
Post - Conditions:	ระบบแสดงหมวดหมู่ข้อสอบทั้งหมด ที่เป็นสาธารณะ	
Flow of Events:	ผู้ใช้งาน	ระบบ
	1.ไปที่หน้าดูหมวดหมู่ข้อสอบสาธารณะ	2.ระบบแสดงหมวดหมู่ทั้งหมดที่เป็นสาธารณะ
Exception Conditions:	ไม่มีหมวดหมู่ข้อสอบที่เป็นสาธารณะอยู่ในระบบ	

ตารางที่ 3.8: รายละเอียดยูสเคส ดูหมวดหมู่ข้อสอบสาธารณะ

3.3.2.9 รายละเอียดยูสเคส ดูรายละเอียดของหมวดหมู่ข้อสอบสาธารณะ

Use Case No:	9	
Use Case Name:	ดูรายละเอียดของหมวดหมู่ข้อสอบสาธารณะ	
Use Case Scenario:	ผู้ใช้งานต้องการดูรายละเอียดในหมวดหมู่ข้อสอบสาธารณะที่ต้องการ	
Triggering Event:	<ul style="list-style-type: none"> - เมื่อผู้ใช้งานต้องการดูรายละเอียดของหมวดหมู่ข้อสอบสาธารณะ - เมื่อผู้ใช้งานต้องการนำหมวดหมู่ข้อสอบสาธารณะนั้นๆไปใช้ในข้อสอบของตนเอง 	
Brief Description:	สำหรับให้ผู้ใช้งานดูรายละเอียดของหมวดหมู่ข้อสอบสาธารณะ ว่าในหมวดหมู่นั้นมีคำถามอะไรบ้าง	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, ดูหมวดหมู่ข้อสอบสาธารณะ	
Stakeholders:	-	
Pre - Conditions:	ผู้ใช้งานต้องการเข้าสู่ระบบ	
Post - Conditions:	ระบบรายละเอียดของหมวดหมู่ข้อสอบนั้นๆ	
Flow of Events:	ผู้ใช้งาน	ระบบ
	1.ไปที่หน้าดูหมวดหมู่ข้อสอบสาธารณะ 3.คลิกหมวดหมู่ที่ต้องการดูรายละเอียด	2.ระบบแสดงหมวดหมู่ทั้งหมดที่เป็นสาธารณะ 4.ระบบแสดงรายละเอียดของหมวดหมู่ข้อสอบนั้น
Exception Conditions:	ไม่มีหมวดหมู่ข้อสอบที่เป็นสาธารณะอยู่ในระบบ	

ตารางที่ 3.9: รายละเอียดยูสเคส ดูรายละเอียดของหมวดหมู่ข้อสอบสาธารณะ

3.3.2.10 รายละเอียดยูสเคส คัดลอกหมวดหมู่ข้อสอบสาธารณะ

Use Case No:	10	
Use Case Name:	คัดลอกหมวดหมู่ข้อสอบสาธารณะ	
Use Case Scenario:	ผู้ใช้งานต้องการนำหมวดหมู่ข้อสอบสาธารณะ	
Triggering Event:	เมื่อผู้ใช้งานต้องการนำหมวดหมู่ข้อสอบสาธารณะนั้นๆ ไปใช้ในข้อสอบของตนเอง	
Brief Description:	สำหรับให้ผู้ใช้งานทำการคัดลอกหมวดหมู่ข้อสอบสาธารณะ ไปใช้ในข้อสอบของตนเอง	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, คู่มือหมวดหมู่ข้อสอบสาธารณะ	
Stakeholders:	-	
Pre - Conditions:	ผู้ใช้งานต้องการเข้าสู่ระบบ	
Post - Conditions:	ระบบทำการคัดลอกหมวดหมู่สาธารณะที่ผู้ใช้งานต้องการ ไปเก็บในหมวดหมู่ข้อสอบของตนเอง	
Flow of Events:	ผู้ใช้งาน	ระบบ
	1.ไปที่หน้าคู่มือหมวดหมู่ข้อสอบสาธารณะ 3.คลิกปุ่มคัดลอกที่หมวดหมู่ที่ต้องการ	2.ระบบแสดงหมวดหมู่ทั้งหมดที่เป็นสาธารณะ 4.ระบบทำการคัดลอกข้อมูลสาธารณะ ที่ต้องการ ไปเก็บในหมวดหมู่ข้อสอบของตนเอง
Exception Conditions:	ไม่มีหมวดหมู่ข้อสอบที่เป็นสาธารณะอยู่ในระบบ	

ตารางที่ 3.10: รายละเอียดยูสเคส คัดลอกหมวดหมู่ข้อสอบสาธารณะ

3.3.2.11 รายละเอียดยูสเคส สร้างข้อสอบ

Use Case No:	11						
Use Case Name:	สร้างข้อสอบ						
Use Case Scenario:	ผู้ใช้งานต้องการสร้างข้อสอบ						
Triggering Event:	เมื่อผู้ใช้งานต้องการสร้างข้อสอบไปของตนเอง เพื่อนำไปทดสอบกับผู้สมัครงาน						
Brief Description:	สำหรับผู้ใช้งานที่ต้องการทำการออกข้อสอบ โดยการดึงคำถามจากประเภทของข้อสอบ ตามจำนวนข้อและความยากที่ต้องการ เช่น ต้องการคำถามหมวดหมู่ คณิตศาสตร์ ข้อระดับง่าย 2 ข้อ ข้อระดับยาก 1 ข้อ และคำถามหมวดหมู่ ภาษาอังกฤษ ข้อระดับปานกลาง 3 ข้อ สามารถกำหนดระยะเวลาในการทำข้อสอบ และวันหมดอายุของข้อสอบได้						
Actors:	พนักงาน						
Related Use Cases:	เข้าสู่ระบบ, สร้างหมวดหมู่ของข้อสอบ, คัดลอกหมวดหมู่ข้อสอบสาธารณะ						
Stakeholders:	-						
Pre - Conditions:	<ul style="list-style-type: none"> - ผู้ใช้ต้องทำการเข้าสู่ระบบ - ผู้ใช้งานต้องสร้างหมวดหมู่ข้อสอบของตนเอง หรือคัดลอกหมวดหมู่ข้อสอบสาธารณะ 						
Post - Conditions:	ระบบทำการสร้างข้อสอบตามที่ผู้ใช้งานกำหนด						
Flow of Events:	<table border="1"> <thead> <tr> <th>ผู้ใช้</th><th>ระบบ</th></tr> </thead> <tbody> <tr> <td>1.ไปที่หน้าสร้างข้อสอบ</td><td>2.ระบบแสดงหมวดหมู่ของตนเองสำหรับเลือกใช้ในข้อสอบ และแสดงรายละเอียดให้กรอก</td></tr> <tr> <td>3.เลือกหมวดหมู่ข้อสอบที่ต้องการ แล้วเลือกระดับความยากและจำนวนข้อที่ต้องการในหมวดหมู่นั้นๆ แล้วคลิกสร้างข้อสอบ</td><td>4.ระบบทำการสร้างข้อสอบตามที่ผู้ใช้งานกำหนด</td></tr> </tbody> </table>	ผู้ใช้	ระบบ	1.ไปที่หน้าสร้างข้อสอบ	2.ระบบแสดงหมวดหมู่ของตนเองสำหรับเลือกใช้ในข้อสอบ และแสดงรายละเอียดให้กรอก	3.เลือกหมวดหมู่ข้อสอบที่ต้องการ แล้วเลือกระดับความยากและจำนวนข้อที่ต้องการในหมวดหมู่นั้นๆ แล้วคลิกสร้างข้อสอบ	4.ระบบทำการสร้างข้อสอบตามที่ผู้ใช้งานกำหนด
ผู้ใช้	ระบบ						
1.ไปที่หน้าสร้างข้อสอบ	2.ระบบแสดงหมวดหมู่ของตนเองสำหรับเลือกใช้ในข้อสอบ และแสดงรายละเอียดให้กรอก						
3.เลือกหมวดหมู่ข้อสอบที่ต้องการ แล้วเลือกระดับความยากและจำนวนข้อที่ต้องการในหมวดหมู่นั้นๆ แล้วคลิกสร้างข้อสอบ	4.ระบบทำการสร้างข้อสอบตามที่ผู้ใช้งานกำหนด						
Exception Conditions:	<ul style="list-style-type: none"> - ไม่มีหมวดหมู่ข้อสอบของตนเอง - ผู้ใช้กรอกรายละเอียดไม่ครบถ้วน หรือไม่ตรงเงื่อนไข 						

ตารางที่ 3.11: รายละเอียดยูสเคส สร้างข้อสอบ

3.3.2.12 รายละเอียดยูสเคส ดูข้อสอบทั้งหมดของตนเอง

Use Case No:	12	
Use Case Name:	ดูข้อสอบทั้งหมดของตนเอง	
Use Case Scenario:	ผู้ใช้ต้องการดูข้อสอบทั้งหมดของตนเอง	
Triggering Event:	เมื่อผู้ใช้ต้องการดูว่าตนเองมีข้อสอบอะไรบ้าง	
Brief Description:	สำหรับผู้ใช้ที่ดูข้อสอบที่ตนเองสร้างขึ้น หรือดูข้อสอบที่ทำการคัดลอกมาจากข้อสอบสาธารณะ	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, สร้างข้อสอบ, คัดลอกข้อสอบจากสาธารณะ	
Stakeholders:	-	
Pre - Conditions:	ผู้ใช้ต้องทำการเข้าสู่ระบบ	
Post - Conditions:	ระบบทำการแสดงข้อสอบทั้งหมดของตนเอง	
Flow of Events:	ผู้ใช้	ระบบ
	1.ไปที่หน้าดูข้อสอบทั้งหมดของตนเอง	2.ระบบแสดงข้อสอบทั้งหมดของผู้ใช้
Exception Conditions:	ไม่มีข้อสอบของตนเอง	

ตารางที่ 3.12: รายละเอียดยูสเคส ดูข้อสอบทั้งหมดของตนเอง

3.3.2.13 รายละเอียดยูสเคส แก้ไขข้อสอบของตนเอง

Use Case No:	13	
Use Case Name:	แก้ไขข้อสอบของตนเอง	
Use Case Scenario:	ผู้ใช้งานต้องการแก้ไขข้อสอบของตนเอง	
Triggering Event:	เมื่อผู้ใช้งานต้องการแก้ไขข้อสอบที่ตนเองสร้างขึ้น หรือข้อสอบที่ทำการคัดลอกมาจากข้อสอบสาธารณะ	
Brief Description:	สำหรับผู้ใช้งานที่ต้องการแก้ไขข้อสอบ เช่น เพิ่มหรือลดจำนวนข้อและระดับความยากในหมวดหมู่ของข้อสอบนั้นๆ	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, สร้างข้อสอบ, คัดลอกข้อสอบจากสาธารณะ	
Stakeholders:	-	
Pre - Conditions:	<ul style="list-style-type: none"> - ผู้ใช้ต้องทำการเข้าสู่ระบบ - ผู้ใช้ต้องสร้างข้อสอบ หรือคัดลอกข้อสอบจากข้อสอบสาธารณะ 	
Post - Conditions:	ระบบทำการแก้ไขข้อสอบตามที่ผู้ใช้งานกำหนด	
Flow of Events:	ผู้ใช้งาน	ระบบ
	1.ไปที่หน้าดูข้อสอบทั้งหมดของตนเอง 3.เลือกข้อสอบที่ต้องการแก้ไข 5.ทำการแก้ไขข้อสอบตามที่ต้องการ แล้วคลิกยืนยัน	2.ระบบแสดงข้อสอบทั้งหมดของผู้ใช้ 4.ระบบทำการแสดงรายละเอียดของข้อสอบนั้น 6.ระบบทำการแก้ไขตามที่ผู้ใช้งานกำหนด
Exception Conditions:	<ul style="list-style-type: none"> - ไม่มีข้อสอบของตนเอง - ผู้ใช้มีการใช้ข้อซ้ำกับข้อสอบของตนเองที่มีอยู่ - ผู้ใช้กรอกรายละเอียดไม่ครบถ้วน หรือไม่ตรงเงื่อนไข 	

ตารางที่ 3.13: รายละเอียดยูสเคส แก้ไขข้อสอบของตนเอง

3.3.2.14 รายละเอียดยูสเคส แก้ไขข้อสอบของตนเอง

Use Case No:	13	
Use Case Name:	แก้ไขข้อสอบของตนเอง	
Use Case Scenario:	ผู้ใช้งานต้องการแก้ไขรายละเอียดข้อสอบของตนเอง	
Triggering Event:	เมื่อผู้ใช้งานต้องการแก้ไขรายละเอียดข้อสอบที่ตนเองสร้างขึ้น หรือ ข้อสอบที่ทำการคัดลอกมาจากข้อสอบสาธารณะ	
Brief Description:	สำหรับผู้ใช้งานที่ต้องการแก้ไขรายละเอียดข้อสอบ เช่น เพิ่มหรือลด จำนวนข้อและระดับความยากในหมวดหมู่ของข้อสอบนั้นๆ	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, สร้างข้อสอบ, คัดลอกข้อสอบจากสาธารณะ	
Stakeholders:	-	
Pre - Conditions:	<ul style="list-style-type: none"> - ผู้ใช้ต้องทำการเข้าสู่ระบบ - ผู้ใช้ต้องสร้างข้อสอบ หรือคัดลอกข้อสอบจากข้อสอบสาธารณะ 	
Post - Conditions:	ระบบทำการแก้ไขข้อสอบตามที่ผู้ใช้งานกำหนด	
Flow of Events:	ผู้ใช้	ระบบ
	1.ไปที่หน้าดูข้อสอบทั้งหมด ของตนเอง 3.เลือกข้อสอบที่ต้องการ แก้ไข 5.ทำการแก้ไขข้อสอบตามที่ ต้องการ แล้วคลิกยืนยัน	2.ระบบแสดงข้อสอบ ทั้งหมดของผู้ใช้ 4.ระบบทำการแสดงราย ละเอียดของข้อสอบนั้น 6.ระบบทำการแก้ไขตามที่ผู้ ใช้กำหนด
Exception Conditions:	<ul style="list-style-type: none"> - ไม่มีข้อสอบของตนเอง - ผู้ใช้มีการใช้ข้อจำกัดข้อสอบของตนเองที่มีอยู่ - ผู้ใช้กรอกรายละเอียดไม่ครบถ้วน หรือไม่ตรงเงื่อนไข 	

ตารางที่ 3.14: รายละเอียดยูสเคส แก้ไขข้อสอบของตนเอง

3.3.2.15 รายละเอียดยูสเคส คัดลอกข้อสอบของตนเอง

Use Case No:	14	
Use Case Name:	คัดลอกข้อสอบของตนเอง	
Use Case Scenario:	ผู้ใช้งานต้องการคัดลอกข้อสอบของตนเอง	
Triggering Event:	เมื่อผู้ใช้งานต้องการแก้ไขรายละเอียดข้อสอบที่ถูกส่งให้ผู้สมัครงานแล้ว ทำให้ไม่สามารถแก้ไขรายละเอียดของข้อสอบเดิมได้ หรือต้องการคัดลอกข้อสอบมาแก้ไข โดยที่ไม่ต้องการให้กระทบข้อสอบเดิม	
Brief Description:	สำหรับให้ผู้ใช้งานคัดลอกข้อสอบของตนเอง	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, สร้างข้อสอบ, คัดลอกข้อสอบจากสาธารณะ	
Stakeholders:	-	
Pre - Conditions:	<ul style="list-style-type: none"> - ผู้ใช้ต้องทำการเข้าสู่ระบบ - ผู้ใช้ต้องสร้างข้อสอบ หรือคัดลอกข้อสอบจากข้อสอบสาธารณะ 	
Post - Conditions:	ระบบทำการคัดลอกข้อสอบที่ผู้ใช้งานต้องการ	
Flow of Events:	ผู้ใช้งาน	ระบบ
	1.ไปที่หน้าดูข้อสอบทั้งหมดของตนเอง 3.คลิกปุ่มคัดลอกที่ข้อสอบที่ต้องการคัดลอก 5.ผู้ใช้งานกดยืนยันการคัดลอก	2.ระบบแสดงข้อสอบทั้งหมดของผู้ใช้งาน 4.ระบบแสดงหน้าต่างยืนยันการคัดลอก 6.ระบบทำการคัดลอกหมวดหมู่ข้อสอบนั้น
Exception Conditions:	ผู้ใช้งานไม่มีข้อสอบ	

ตารางที่ 3.15: รายละเอียดยูสเคส คัดลอกข้อสอบของตนเอง

3.3.2.16 รายละเอียดยูสเคส ลบข้อสอบ

Use Case No:	15	
Use Case Name:	ลบข้อสอบ	
Use Case Scenario:	ผู้ใช้งานต้องการลบข้อสอบ	
Triggering Event:	เมื่อผู้ใช้งานไม่ต้องการข้อสอบนั้นแล้ว	
Brief Description:	สำหรับผู้ใช้งานที่ต้องการลบข้อสอบที่ตนเองสร้างขึ้น หรือลบข้อสอบที่ทำการคัดลอกมาจากข้อสอบสาธารณะ	
Actors:	พนักงาน	
Related Use Cases:	เข้าสู่ระบบ, สร้างข้อสอบ, คัดลอกข้อสอบจากสาธารณะ	
Stakeholders:	-	
Pre - Conditions:	<ul style="list-style-type: none"> - ผู้ใช้ต้องทำการเข้าสู่ระบบ - ผู้ใช้ต้องสร้างข้อสอบ หรือคัดลอกข้อสอบจากข้อสอบสาธารณะ 	
Post - Conditions:	ระบบทำการคัดลอกข้อสอบที่ผู้ใช้งานต้องการ	
Flow of Events:	ผู้ใช้	ระบบ
	1.ไปที่หน้าดูข้อสอบทั้งหมดของตนเอง 3.เลือกข้อสอบที่ต้องการลบ 5.ทำการยืนยันการลบข้อสอบ	2.ระบบแสดงข้อสอบทั้งหมดของผู้ใช้ 4.ระบบทำการแสดงหน้าต่างข้อความยืนยันการลบ 6.ระบบทำการคัดลอกหมวดหมู่ข้อสอบนั้น
Exception Conditions:	<ul style="list-style-type: none"> - ไม่มีข้อสอบของตนเอง - ผู้ใช้ไม่ทำการยืนยันการลบข้อสอบ 	

ตารางที่ 3.16: รายละเอียดยูสเคส ลบข้อสอบ

บรรณานุกรม

- [1] Guru99, “Restful web services tutorial with example,” 2020. [Online]. Available: <https://www.guru99.com/restful-web-services.html>
- [2] M. contributors, “Http response status codes.” [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Status>
- [3] W. contributors, “Model–view–controller.” [Online]. Available: <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>
- [4] M. Inc., “What is nosql?” [Online]. Available: <https://www.mongodb.com/nosql-explained>
- [5] D. Inc., “What is a container?” [Online]. Available: <https://www.docker.com/resources/what-container>
- [6] W. contributors, “Model–view–viewmodel.” [Online]. Available: <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93viewmodel>
- [7] Vue.js, “What is vuex?” [Online]. Available: <https://vuex.vuejs.org>
- [8] L. Torvalds, “Git.” [Online]. Available: <https://git-scm.com>
- [9] Auth0, “Introduction to json web tokens.” [Online]. Available: <https://jwt.io/introduction>
- [10] I. MongoDB, “mongodb.” [Online]. Available: <https://www.mongodb.com>

ภาคผนวก ก

เรื่องที่หนึ่ง