



ระบบบริหารจัดการห้องเช่ารายเดือน ผ่านเว็บแอปพลิเคชันและโมบายแอปพลิเคชัน

นายสุภกฤต สอาด

รหัสประจำตัว 6003051617287

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีวิศวกรรมอิเล็กทรอนิกส์ (โทรคมนาคม)

ภาควิชาเทคโนโลยีวิศวกรรมอิเล็กทรอนิกส์

วิทยาลัยเทคโนโลยีอุตสาหกรรม

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

พ.ศ. 2567

วิทยาลัยเทคโนโลยีอุตสาหกรรม

แบบเสนอขออนุมัติหัวข้อปริญญานิพนธ์

ภาควิชา เทคโนโลยีวิศวกรรมอิเล็กทรอนิกส์ ภาควิชาปีที่ 2/2567

ชื่อปริญญานิพนธ์ (ภาษาไทย) ระบบบริหารจัดการห้องเช่ารายเดือน

ชื่อปริญญานิพนธ์ (ภาษาอังกฤษ) Monthly Room Rental Management Platform

ชื่อผู้เสนอปริญญานิพนธ์ 1. นายศุภกฤต สอาด (หัวหน้าปริญญานิพนธ์)

นักศึกษาสาขาวิชาเทคโนโลยีวิศวกรรมอิเล็กทรอนิกส์ (โทรคมนาคม) รอบเช้า (สมทบ)

ชื่ออาจารย์ที่ปรึกษาปริญญานิพนธ์ ผศ. ดร. ณัฐพล ประสงค์พันธุ์

มีความประสงค์ขออนุมัติหัวข้อปริญญานิพนธ์ เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตรบัณฑิต สาขาวิชาเทคโนโลยีวิศวกรรมอิเล็กทรอนิกส์ (โทรคมนาคม) จึงเรียนมาเพื่อโปรดพิจารณา ดังรายละเอียดปริญญานิพนธ์ที่แนบมาด้วย

ลงชื่อ นายศุภกฤต สอาด (หัวหน้าปริญญานิพนธ์)

30 / พฤศจิกายน / 2567

ความเห็นอาจารย์ที่ปรึกษาปริญญานิพนธ์	ความเห็นหัวหน้าภาค / โครงการภาควิชา
.....
.....
.....
ลงนาม	ลงนาม
...../...../...../...../.....

1. ชื่อหัวข้อโครงการงานปริญญานิพนธ์

ระบบบริหารจัดการห้องเช่ารายเดือน

2. ความเป็นมาและความสำคัญของโครงการงานปริญญานิพนธ์

ในปัจจุบันธุรกิจห้องเช่าแบบรายเดือนยังคงมีบทบาทสำคัญและความต้องการเป็นอย่างมาก โดยเฉพาะในเมืองที่เป็นภาคอุตสาหกรรมของประเทศไทยที่มีผู้เช่าส่วนมากมีภูมิลำเนาอยู่ต่างจังหวัดและย้ายมาทำงานในเมืองนั้นแบบชั่วคราว จึงไม่นิยมซื้อที่อยู่อาศัยแบบถาวรแต่จะนิยมการเช่าห้องพักแบบรายเดือนเสียมากกว่า และด้วยเนื่องจากห้องเช่าแห่งนี้ให้บริการห้องเช่ารายเดือนตั้งอยู่ในพื้นที่ ตำบลบ้านฉาง อำเภอบ้านฉาง จังหวัดระยองนั้น ซึ่งเป็นจังหวัดที่มีนิคมอุตสาหกรรมหลักจำนวนมากอยู่ในพื้นที่ การบริหารจัดการห้องเช่าแบบรายเดือนจึงเป็นงานที่ซับซ้อนขึ้น เนื่องจากการที่มีผู้เช่าส่วนมากเป็นคนต่างจังหวัด จึงมีการโยกย้ายเข้าออกห้องพักเกิดขึ้นบ่อยครั้ง จึงทำให้เกิดปัญหาการซ้ำซ้อนของข้อมูลผู้เช่า การจัดเก็บข้อมูลที่ไม่เป็นระเบียบ การสูญหายของข้อมูลและเอกสาร การค้นหาข้อมูลย้อนหลังเพื่อนำมาใช้ประโยชน์ในด้านต่างๆจึงทำได้ยากและเสียเวลา และโดยส่วนมากแล้วผู้เช่าของห้องเช่าแห่งนี้จะมีอาชีพเป็นพนักงานโรงงานหรือพนักงานบริษัทที่มีการเข้างานล่วงเวลาบ่อยครั้งทำให้ไม่ค่อยได้พบเจอกับเจ้าของห้องเช่าโดยตรง จึงทำให้เกิดปัญหาการติดต่อสื่อสารที่ล่าช้าทั้งจากผู้เช่าไปยังเจ้าของห้องเช่าและจากทางเจ้าของห้องเช่าไปยังผู้เช่า ไม่ว่าจะเป็นการแจ้งปัญหาต่างๆหรือคำร้องซ่อมแซมตกหล่น และยังพบปัญหาในการที่ผู้เช่าชำระเงินค่าเช่ารายเดือนให้แก่เจ้าของห้องเช่าล่าช้า โดยทางผู้เช่ามีการอ้างเหตุผลเรื่องเวลาการเข้างาน หรือในช่วงวันที่เจ้าของห้องเช่าไม่อยู่ เช่น เดินทางไปต่างจังหวัดหรือต่างประเทศก็จะไม่สามารถชำระเงินค่าเช่าได้ เพราะต้องพบเจอกันโดยตรงเพื่อจ่ายเงินค่าเช่าเป็นเงินสดเท่านั้น

จากปัญหาดังกล่าว จึงเป็นที่มาของหัวข้อปริญญานิพนธ์ชิ้นนี้ หัวข้อปริญญานิพนธ์ระบบบริหารจัดการห้องเช่ารายเดือน ผู้จัดทำเห็นความสำคัญที่จะพัฒนาแพลตฟอร์มระบบบริหารจัดการห้องเช่าแบบรายเดือน ให้ทำงานอยู่บนระบบอินเทอร์เน็ต ด้วยแอปพลิเคชันบนสมาร์ตโฟนสำหรับผู้เช่าที่จะมีฟีเจอร์ในการจัดการกับข้อมูลส่วนตัวและสัญญาเช่า ฟีเจอร์ในการได้รับแจ้งชะร่าบิลที่รวดเร็วและถูกต้อง ฟังก์ชันอัปโหลดสลิปการชำระเงินตามบิลต่างๆ ฟีเจอร์ในการแจ้งปัญหาการอยู่อาศัย รวมถึงการแจ้งซ่อมของใช้ภายในห้องเช่าเป็นต้น และสำหรับเจ้าของห้องเช่าก็จะมีระบบเว็บแอปพลิเคชันหลังบ้านที่ใช้งานผ่านบราวเซอร์ ที่จะมีฟีเจอร์หลักๆในการจัดการข้อมูลผู้เช่าและระบบ, จัดการข้อมูลชุมชนและอาคาร, จัดการข้อมูลห้องเช่า, จัดการข้อมูลผู้เช่า, จัดการบิลและค่าเช่า, การแจ้งเตือนและประกาศ, การรับเรื่องแจ้งจากผู้เช่าต่างๆ โดยระบบนี้จะเข้ามาช่วยให้กระบวนการบริหารธุรกิจสำหรับเจ้าของกิจการดำเนินการได้สะดวกและรวดเร็ว ลดปัญหาความล่าช้าหรือข้อมูลตกหล่น และยังช่วยอำนวยความสะดวกให้ผู้เช่าได้เข้าถึงบริการที่สะดวก รวดเร็วและทันถ่วงที

3. วัตถุประสงค์ของโครงการปริญญานิพนธ์

- 3.1 เพื่อศึกษาการพัฒนาเว็บแอปพลิเคชันด้วยภาษา Javascript (Angular และ Node.js)
- 3.2 เพื่อศึกษาการเขียนแอปพลิเคชันด้วยภาษา Dart (Flutter)
- 3.3 เพื่อศึกษาการเขียนเว็บแอปพลิเคชันที่มีการเชื่อมต่อกับฐานข้อมูลด้วย API
- 3.4 เพื่อศึกษาการใช้งาน MySQL Database ในการเก็บฐานข้อมูล
- 3.5 เพื่อศึกษาการออกแบบระบบฐานข้อมูล
- 3.6 เพื่อศึกษาการใช้งาน HTML Bootstrap CSS JavaScript ในการตกแต่งหน้าเว็บแอปพลิเคชัน
- 3.7 เพื่อศึกษาการใช้งานระบบบริการรับเช็คสลิปโอนเงินผ่าน SlipOK API บนโมบายแอปพลิเคชัน

4. ขอบเขตของโครงการปริญญานิพนธ์

ส่วนการทำงานของระบบบริหารจัดการห้องเช่ารายเดือนผ่านเว็บและแอปพลิเคชันมือถือมีรายละเอียดดังนี้ พัฒนาส่วนเว็บแอปพลิเคชันหลังบ้านสำหรับผู้ดูแลระบบโดยใช้ภาษา HTML CSS Bootstrap Angular TypeScript พัฒนาในส่วนของ Server และ API ด้วย ภาษา Javascript Node.js และพัฒนาในส่วนแอปพลิเคชันโดยใช้ภาษา Dart (Flutter) โดยมีตัวกลางสื่อสารระหว่างเซิร์ฟเวอร์และไคลเอนต์ด้วยสถาปัตยกรรม RESTful API ผ่านโปรโตคอล HTTP โดยขอบเขตส่วนหน้ามี 2 ระบบคือ โมบายแอปพลิเคชันและเว็บแอปพลิเคชันหลังบ้านสำหรับผู้ดูแลระบบ

- 4.1 ระบบโมบายแอปพลิเคชันมีไว้เพื่อให้ผู้เช่าสามารถใช้งานระบบเพื่อรับการแจ้งเตือนข่าวสารต่างๆจากแอดมินรวมถึงรายการค่าเช่ารายเดือนและสามารถอัปโหลดสลิปการโอนเงินจ่ายค่าเช่าเพื่อเป็นหลักฐานให้กับทางแอดมินได้ และสามารถร้องเรียน แจ้งซ่อม หรือสื่อสารกับแอดมิน และดูข้อมูลส่วนตัว หรือสัญญาเช่าของตัวเองได้

4.1.1 รายละเอียดของโมบายแอปพลิเคชัน

4.1.1.1 ฟังก์ชันลงชื่อเข้าใช้งาน

1. ถ้าเป็นการเข้าใช้งานระบบครั้งแรกผู้เช่าจะได้รับ Username จากทางแอดมิน ให้กดที่ปุ่ม ตั้งค่ารหัสผ่าน จากนั้นกรอก Username ลงไป ระบบจะทำการเช็คในฐานข้อมูล ว่ามีชื่อผู้ใช้งานนี้ภายในระบบหรือไม่ ถ้ามีจะให้การตั้งรหัสผ่านสำหรับการเข้าใช้งานในครั้งต่อไป

2. ถ้าตั้งรหัสผ่านเรียบร้อยแล้ว ผู้ใช้สามารถใช้ Username จากทางแอดมิน และ รหัสผ่านที่ตนเองตั้งไว้เข้าใช้งานในระบบได้

4.1.1.2 เมนูหลักของระบบ

1. หลังจากเข้าสู่ระบบเรียบร้อยแล้ว จะเข้าสู่หน้าหลักของระบบ โดยอยู่ที่แถบ หน้าหลัก มีฟังก์ชันการใช้งานดังนี้

1.1 การแจ้งเตือนค่าใช้จ่าย สามารถได้รับแจ้งเตือนค่าใช้จ่ายและอัปโหลดสลิปการโอนเงินเป็นหลักฐานในการจ่ายค่าเช่าได้

1.2 ห้องของฉัน สามารถดูข้อมูลที่ตั้งของสถานที่ รวมถึงสัญญาเช่าได้

1.3 สมุดโทรศัพท์ สามารถดูข้อมูลเบอร์ติดต่อเจ้าของห้องเช่าหรือ แอดมินได้

1.4 กฎระเบียบ สามารถดูกฎระเบียบต่าง ๆ ของห้องเช่าที่กำหนดไว้ได้

2. แถบประกาศ มีฟังก์ชันการใช้งาน

2.1 สามารถเข้าถึงประกาศข้อมูลข่าวสารจากทางผู้ดูแลระบบได้ โดยแบ่งแยกประเภทประกาศเป็น 2 ประเภท 1. ประกาศสำคัญ 2. ข่าวสารทั่วไป

3. แถบแจ้งเรื่อง มีฟังก์ชันการใช้งาน

3.1 สามารถแจ้งเรื่องให้กับทางผู้ดูแลระบบได้รับทราบได้ เช่น แจ้งซ่อม แจ้งปัญหาสำหรับการอยู่อาศัยต่างๆ และสามารถดูประวัติที่เราเคยแจ้งผู้ดูแลระบบไปแล้วย้อนหลังได้ โดยจะมีสถานะบอกว่าผู้ดูแลระบบรับทราบการแจ้งเรื่องหรือยัง ด้วยสถานะ อ่านแล้ว และ ยังไม่อ่าน

4. แถบตั้งค่า มีฟังก์ชันการใช้งาน

4.1 สามารถดูข้อมูลส่วนตัวของตนเองได้ และสามารถอัปโหลดเปลี่ยนรูปภาพโปรไฟล์ของตนเองได้

4.2 สามารถกดออกจากระบบได้

4.2 ระบบเว็บแอปพลิเคชันหลังบ้านสำหรับผู้ดูแลระบบมีไว้เพื่อให้ผู้ดูแลระบบและผู้ดูแลระบบสูงสุดสามารถจัดการกับข้อมูลของผู้เช่าให้เป็นระเบียบ จัดการเรื่องแจ้งเตือนค่าเช่ารายเดือน รวมถึงการสื่อสารแจ้งเตือนประกาศข่าวสารต่าง ๆ ไปยังผู้เช่าได้

4.2.1 รายละเอียดของเว็บแอปพลิเคชันหลังบ้านสำหรับผู้ดูแลระบบ ได้มีการกำหนดสิทธิ์ผู้ใช้งานสามารถแบ่งได้ 2 ประเภท ได้แก่ ผู้ดูแลระบบและผู้ดูแลระบบสูงสุด โดยฟังก์ชันการใช้งานของระบบของผู้ใช้แต่ละประเภทมีรายละเอียดดังต่อไปนี้

4.2.1.1 ผู้ดูแลระบบสูงสุด

4.2.1.1.1 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลผู้ดูแลระบบภายในระบบเว็บแอปพลิเคชันได้

4.2.1.1.2 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลผู้เช่าภายในระบบเว็บแอปพลิเคชันได้

4.2.1.1.3 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลของชุมชนและอาคารภายในระบบเว็บแอปพลิเคชันได้

4.2.1.1.4 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลของห้องเช่าภายในระบบเว็บแอปพลิเคชันได้

4.2.1.1.5 สามารถดูรายละเอียดคำร้องเรียนหรือแจ้งเรื่องของผู้เช่าและยืนยันรับทราบคำร้องนั้นภายในระบบเว็บแอปพลิเคชันได้

4.2.1.1.6 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลของการแจ้งเตือนและประกาศที่ส่งไปให้ผู้เช่าบนแอปพลิเคชันภายในระบบเว็บแอปพลิเคชันได้

4.2.1.1.7 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลของบิลแจ้งเตือนค่าเช่ารายเดือนและบิลค่าใช้จ่ายต่าง ๆ ที่ส่งไปให้ผู้เช่าบนแอปพลิเคชันภายในระบบเว็บแอปพลิเคชันได้

4.2.1.1.8 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลของบิลค่าเช่ารายเดือนและบิลค่าใช้จ่ายต่าง ๆ ที่ผู้เช่าส่งหลักฐานเป็นสลิปยืนยันการโอนเงินเข้ามาภายในระบบเว็บแอปพลิเคชันได้

4.2.1.2 ผู้ดูแลระบบ

4.2.1.2.1 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลผู้เช่าภายในระบบเว็บแอปพลิเคชันได้

4.2.1.2.2 สามารถดูรายละเอียดคำร้องเรียนหรือแจ้งเรื่องของผู้เช่าและยืนยันรับทราบคำร้องนั้นภายในระบบเว็บแอปพลิเคชันได้

4.2.1.2.3 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลของการแจ้งเตือนและประกาศที่ส่งไปให้ผู้เช่าบนแอปพลิเคชันภายในระบบเว็บแอปพลิเคชันได้

4.2.1.2.4 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลของบิลแจ้งเดือนค่าเช่ารายเดือนและบิลค่าใช้จ่ายต่าง ๆ ที่ส่งไปให้ผู้เช่าบนแอปพลิเคชันภายในระบบเว็บแอปพลิเคชันได้

4.2.1.2.5 สามารถดูรายละเอียด เพิ่ม ลบ แก้ไข ข้อมูลของบิลค่าเช่ารายเดือนและบิลค่าใช้จ่ายต่าง ๆ ที่ผู้เช่าส่งหลักฐานเป็นสลิปยืนยันการโอนเงินเข้ามาภายในระบบเว็บแอปพลิเคชันได้

4.3 ระบบหลังบ้านหรือในส่วนเซิร์ฟเวอร์ที่ติดต่อกับฐานข้อมูลจะเป็นหัวใจหลักในการประมวลผลและจัดการข้อมูลทั้งหมดในระบบบริหารจัดการห้องเช่ารายเดือนนี้ ซึ่งการพัฒนาเว็บด้วย Node.js และ RESTful API จะทำให้การสื่อสารระหว่างเซิร์ฟเวอร์และไคลเอนต์ให้เป็นไปอย่างมีประสิทธิภาพและตอบสนองได้รวดเร็ว ระบบมีการจัดการข้อมูลต่างๆ ที่เกี่ยวข้องกับผู้เช่า ผู้ดูแลระบบ การแจ้งเตือน รายการค่าใช้จ่าย และประกาศต่างๆ โดยระบบนี้จะใช้ RESTful API ในการเชื่อมต่อระหว่างโมบายแอปพลิเคชันและเว็บแอปพลิเคชันหลังบ้านสำหรับผู้ดูแลระบบกับฐานข้อมูล ข้อมูลที่ถูกส่งผ่าน API จะอยู่ในรูปแบบ JSON ซึ่งจะมีการใช้ภาษา Javascript Node.js สำหรับการพัฒนาเซิร์ฟเวอร์ และ API และมี MySQL สำหรับเป็นฐานข้อมูล เพื่อจัดเก็บข้อมูลและดำเนินการต่างๆ ดังนี้

4.3.1 การจัดการผู้ใช้งาน ระบบสามารถบันทึกและอัปเดตข้อมูลของผู้เช่าและผู้ดูแลระบบ เช่น ชื่อผู้ใช้งาน รหัสผ่าน ข้อมูลส่วนตัว ข้อมูลการเช่า สถานะการชำระค่าเช่า และอื่นๆ

4.3.2 การจัดการห้องเช่า ชุมชนและอาคาร ระบบจะมีการจัดการข้อมูลของห้องเช่า ชุมชน และอาคารภายในระบบ เช่น จำนวนห้อง ข้อมูลเกี่ยวกับการเช่า ราคาเช่า และการบันทึกการเปลี่ยนแปลงต่างๆ ที่เกี่ยวข้อง

4.3.3 การจัดการการแจ้งเตือน ระบบสามารถจัดการข้อมูลการแจ้งเตือนต่างๆ เช่น ประกาศ ข่าวสาร, การแจ้งเตือนค่าใช้จ่าย, และคำร้องของผู้เช่า โดยจะมีฟังก์ชันในการส่งข้อมูลไปยังแอปพลิเคชันมือถือเพื่อให้ผู้เช่าได้รับการแจ้งเตือนแบบเรียลไทม์ หรือฟังก์ชันการส่งข้อมูลจากผู้เช่ามายังผู้ดูแลระบบ

4.3.4 การจัดการบิลค่าเช่าและติดตามสถานะการจ่ายค่าเช่ารายเดือน ระบบจะทำการส่งแจ้งเตือนบิลค่าเช่าตามรอบบิลของแต่ละเดือนตามที่ระบุในสัญญาเช่า โดยที่จะส่งออกไปก่อนถึงรอบเดือนนั้น 5 วัน เพื่อให้ผู้เช่าเตรียมตัวจ่ายค่าเช่าได้ทันเวลาโดยไม่มีการล่าช้า

4.3.5 ระบบได้มีการติดต่อกับผู้ให้บริการตรวจสอบเช็คสลิปการโอนเงินของผู้เช่า เพื่อเป็นการยืนยันว่าการโอนเงินของผู้เช่านั้นโปร่งใส โดยที่ไม่ได้ใช้สลิปการโอนเงินปลอมหรือเป็นสลิปการโอนเงินเก่า

4.3.6 การเชื่อมต่อกับฐานข้อมูล ระบบจะทำการติดต่อกับฐานข้อมูลเพื่อดึงข้อมูลที่เกี่ยวข้องกับผู้เช่าและห้องเช่า รวมถึงการบันทึกข้อมูลการอัปโหลดสลิปการโอนเงิน ประวัติการแจ้งซ่อม หรือข้อมูลต่างๆ ที่ผู้เช่าได้ทำการอัปเดต โดยใช้คำสั่ง SQL หรือ ORM สำหรับการจัดการข้อมูล เช่น การ INSERT UPDATE SELECT และ DELETE

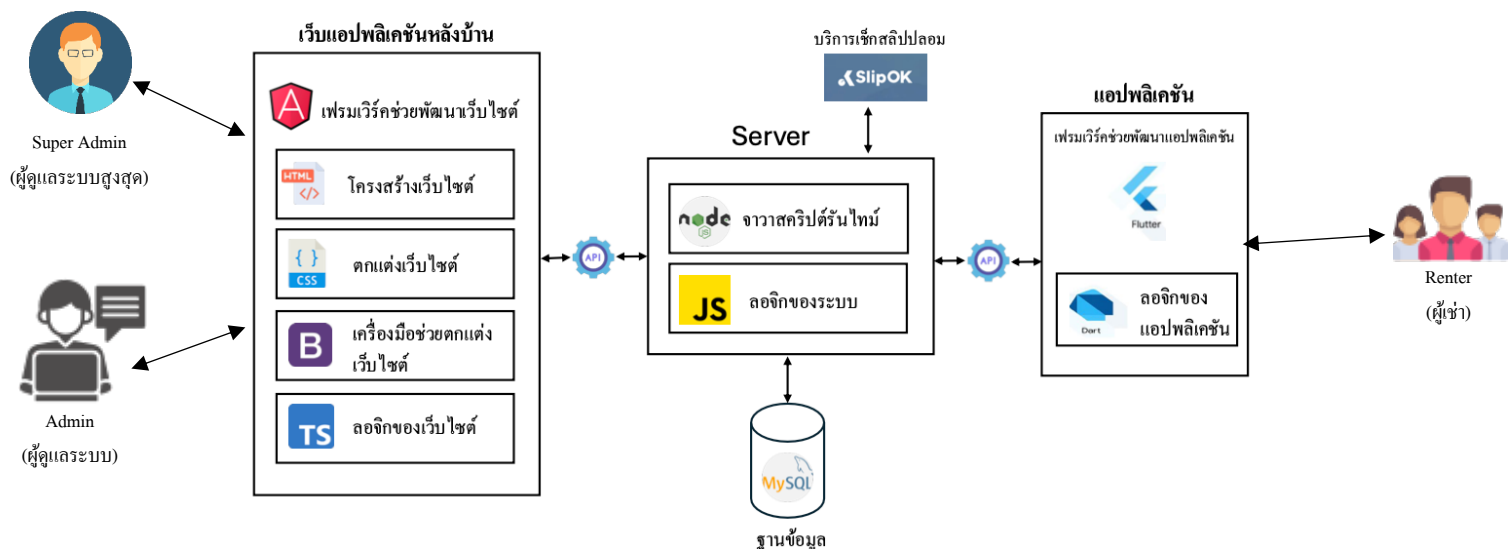
4.3.7 การตรวจสอบสิทธิ์ผู้ใช้งาน ระบบจะมีการตรวจสอบสิทธิ์ของผู้ใช้งานในการเข้าถึงข้อมูลต่างๆ เช่น การตรวจสอบว่าเป็นผู้ดูแลระบบหรือผู้ดูแลระบบสูงสุดเพื่อกำหนดสิทธิ์ในการเพิ่ม, แก้ไข, ลบ ข้อมูล

4.3.8 การรักษาความปลอดภัย การใช้ JWT (JSON Web Token) ในการยืนยันตัวตนและการเข้าถึง API ของผู้ใช้งาน เพื่อให้แน่ใจว่าเฉพาะผู้ที่ได้รับอนุญาตเท่านั้นที่จะสามารถเข้าถึงข้อมูลที่จำเป็นได้

5. ประโยชน์ที่คาดว่าจะได้รับจากการทำโครงการปริญญานิพนธ์

- 5.1 เพื่อช่วยให้กระบวนการในการบริหารจัดการธุรกิจสำหรับเจ้าของห้องเช่ามีความสะดวก และลดระยะเวลาการดำเนินการ
- 5.2 เพื่ออำนวยความสะดวกให้กับผู้เช่าในการใช้บริการห้องเช่า
- 5.3 สามารถนำความรู้จากสิ่งที่ศึกษามาใช้ต่อยอดได้
- 5.4 เรียนรู้การเขียนโปรแกรมเพื่อพัฒนาเว็บแอปพลิเคชันและโมบายแอปพลิเคชัน

6. รายละเอียดเกี่ยวกับโครงการปริญญานิพนธ์



รูปที่ 1 การทำงานโดยรวมของโครงการปริญญานิพนธ์

ระบบบริหารจัดการห้องเช่ารายเดือนผ่านเว็บแอปพลิเคชันและโมบายแอปพลิเคชันจะพัฒนาโดยใช้ Angular HTML CSS Bootstrap Typescript ในการจัดทำส่วนหน้าของเว็บแอปพลิเคชันสำหรับผู้ดูแลระบบ และใช้ Flutter Dart ในการพัฒนาโมบายแอปพลิเคชัน ใช้บริการระบบของ SlipOK ในการทำในส่วนของการเช็คสลิปหลักฐานการโอนเงินค่าเช่า โดยส่วนของ Server จะใช้ Javascript Node.js ในการพัฒนาระบบหลังบ้าน และใช้ MySQL เป็นฐานข้อมูลในการจัดเก็บข้อมูลของระบบ

7. ทฤษฎีที่สำคัญและงานวิจัยที่เกี่ยวข้อง

7.1 Angular

Angular คือเฟรมเวิร์กที่ใช้สำหรับการพัฒนาเว็บแอปพลิเคชันแบบ single-page applications (SPA) โดยพัฒนาโดย Google ด้วยการใช้ TypeScript เป็นภาษาหลักในการเขียนโค้ด ซึ่งช่วยเพิ่มความสะดวกในการพัฒนาแอปพลิเคชันที่มีความซับซ้อน ด้วยโครงสร้างที่เน้นการใช้งาน component-based architecture ซึ่งทำให้สามารถแยกแอปพลิเคชันออกเป็นส่วนย่อยๆ ที่มีความสามารถในการใช้งานซ้ำและง่ายต่อการบำรุงรักษา Angular ยังรองรับ two-way data binding ซึ่งทำให้ข้อมูลระหว่าง model และ view ถูกเชื่อมโยงกันโดยอัตโนมัติ นอกจากนี้ Angular ยังมาพร้อมกับระบบ dependency injection ที่ช่วยให้การจัดการกับการสร้างและจัดการอ็อบเจกต์ต่างๆ ในแอปพลิเคชันทำได้ง่ายมีประสิทธิภาพ นอกจากนี้ยังมีระบบ routing สำหรับการจัดการการนำทางในแอปพลิเคชันที่ทำงานแบบ SPA ทำให้การเปลี่ยนแปลงหน้าจอในแอปพลิเคชันไม่ต้องโหลดหน้าใหม่ทั้งหมด ซึ่งทำให้การใช้งานแอปพลิเคชันเป็นไปได้อย่างราบรื่น Angular ยังรองรับการทำงานกับ RxJS ที่ช่วยในการจัดการกับการทำงานที่เป็น asynchronous ได้อย่างมีประสิทธิภาพ การใช้ Angular CLI ก็ช่วยให้นักพัฒนาสามารถสร้าง, บิวด์, และทดสอบแอปพลิเคชันได้อย่างสะดวกและรวดเร็ว

7.2 HTML

ความหมายของ HTML ย่อมาจากภาษาอังกฤษในคำว่า Hypertext Markup Language ซึ่งถ้าแปลความหมายให้เข้าใจได้ง่ายมากขึ้นนั้นก็คือ เป็นภาษาที่ไว้สำหรับการสร้างเพจหรือเว็บไซต์ ภาษาตรงนี้จะไว้สำหรับการกำหนดรูปแบบของเว็บไซต์ว่าควรมีหน้าตาหรือควรมีลักษณะจุดเด่นอย่างไร ยกตัวอย่างเช่น บางครั้งเราอาจจะเห็นในเรื่องของหน้าเว็บไซต์บางเว็บไซต์ที่มีตัวหนังสือที่สวยงามหรือมีลูกเล่น และสีสันทันต่างๆ หรือมีการใช้ในส่วนของการทำภาพเคลื่อนไหวต่างๆ ซึ่งภาษาในการเขียนเว็บอย่าง HTML เหล่านี้แหละที่จะเป็นตัวบ่งบอกถึงคำสั่งในการจัดรูปแบบหน้าเพจหรือข้อความต่างๆ หน้าเว็บไซต์ได้เป็นอย่างดี

7.3 CSS

CSS ย่อมาจาก Cascading Style Sheet มักเรียกโดยย่อว่า "สไตลชีต" คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (หรือ "Style") ของเนื้อหาในเอกสาร อันได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ หรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสาร ไม่ขึ้นอยู่กับเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลผลลัพธ์ของเอกสาร HTML โดยเฉพาะ

ในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสารบ่อยครั้ง หรือต้องการควบคุมให้รูปแบบการแสดงผลเอกสาร HTML มีลักษณะของความสม่ำเสมอทั่วกันทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน โดยกฎเกณฑ์ในการกำหนดรูปแบบ (Style) เอกสาร HTML ถูกเพิ่มเข้ามาครั้งแรกใน HTML 4.0 เมื่อปี พ.ศ. 2539 ในรูปแบบของ CSS level 1 Recommendations ที่กำหนดโดย องค์กร World Wide Web Consortium หรือ W3C

7.4 Bootstrap

Bootstrap คือเฟรมเวิร์กสำหรับการพัฒนาเว็บไซต์ที่ช่วยให้การออกแบบและพัฒนาเว็บแอปพลิเคชันเป็นไปอย่างรวดเร็วและมีประสิทธิภาพ โดยเน้นการใช้งานที่ง่ายและตอบสนองได้ดีบนอุปกรณ์ทุกขนาด (responsive design) โดยมาพร้อมกับเครื่องมือและคอมโพเนนต์ที่ใช้บ่อยในการพัฒนาเว็บไซต์ เช่น ปุ่ม (buttons), เมนูนำทาง (navigation bars), ฟอรม (forms), การ์ด (cards), ป๊อปอัพ (modals) และอีกมากมาย ซึ่งทำให้การออกแบบเว็บไซต์ดูดีและใช้งานได้อย่างราบรื่นบนทุกแพลตฟอร์ม Bootstrap ใช้ HTML, CSS, และ JavaScript ในการสร้างโครงสร้างของหน้าเว็บและเพิ่มฟังก์ชันการทำงานที่สำคัญ โดยช่วยให้การจัดการกับรูปแบบต่างๆ เช่น การจัดวางเลย์เอาต์ของเว็บ หรือการทำให้เว็บไซต์รองรับการแสดงผลในหลายขนาดหน้าจอได้ง่ายและเร็วขึ้น นอกจากนี้ Bootstrap ยังรองรับการปรับแต่งได้ตามความต้องการของผู้ใช้ผ่านตัวแปรและการตั้งค่าต่างๆ ที่สามารถปรับแต่งได้ ทำให้การพัฒนาเว็บไซต์มีความยืดหยุ่นและง่ายดายมากยิ่งขึ้น

7.5 MySQL

MySQL คือระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS) ที่ได้รับความนิยมอย่างมากและใช้กันอย่างแพร่หลายในการพัฒนาแอปพลิเคชันและเว็บไซต์ต่างๆ โดยเฉพาะในด้านการจัดเก็บและจัดการข้อมูลในรูปแบบตาราง ซึ่งข้อมูลจะถูกจัดระเบียบในแถวและคอลัมน์ และสามารถเชื่อมโยงข้อมูลระหว่างตารางได้อย่างมีระเบียบผ่านการใช้คีย์ต่างๆ เช่น primary key และ foreign key MySQL ใช้ SQL (Structured Query Language) ในการเข้าถึงและจัดการข้อมูล เช่น การเพิ่ม, ลบ, แก้ไข, และดึงข้อมูลจากฐานข้อมูล นอกจากนี้ MySQL ยังรองรับการทำธุรกรรม (transaction) ที่ช่วยให้การดำเนินการหลายๆ อย่างสามารถทำในหน่วยงานเดียวได้อย่างปลอดภัยและคงความถูกต้องของข้อมูล ระบบนี้เหมาะสำหรับการจัดการฐานข้อมูลขนาดใหญ่และรองรับการใช้งานจากแอปพลิเคชันต่างๆ โดยที่มีประสิทธิภาพสูงและการขยายระบบที่ง่าย

7.6 JavaScript

JavaScript คือภาษาการเขียนโปรแกรมที่ใช้ในการพัฒนาเว็บไซต์และแอปพลิเคชันบนเว็บ โดยช่วยเพิ่มความสามารถในการโต้ตอบและความน่าสนใจให้กับเว็บไซต์ JavaScript ทำงานฝั่ง client (ใน

เบราว์เซอร์) ซึ่งช่วยให้เว็บไซต์มีการตอบสนองที่รวดเร็วและไม่ต้องโหลดข้อมูลจากเซิร์ฟเวอร์ทุกครั้งที่มีการเปลี่ยนแปลงบนหน้าเว็บ ด้วยความสามารถในการจัดการกับเหตุการณ์ต่างๆ (Event-driven) เช่น การคลิกปุ่ม, การเลื่อนหน้าจอ, หรือการกดแป้นพิมพ์ JavaScript ช่วยให้เว็บไซต์มีการตอบสนองที่มีชีวิตชีวาและสามารถโต้ตอบกับผู้ใช้ได้อย่างราบรื่น อีกทั้งยังรองรับการเขียนโค้ดแบบ asynchronous ซึ่งช่วยให้สามารถทำงานหลายๆ อย่างพร้อมกันได้ เช่น การดึงข้อมูลจากเซิร์ฟเวอร์โดยไม่ต้องรอให้การทำงานอื่นๆ เสร็จสิ้น ฟังก์ชันการทำงานต่างๆ ใน JavaScript เช่น การจัดการกับอาร์เรย์, วัตถุ, และการใช้ฟังก์ชันทำให้การพัฒนาเว็บไซต์มีความยืดหยุ่นและง่ายดายเมื่อผสมผสานกับเทคโนโลยีอื่นๆ เช่น API, JavaScript ยังช่วยให้เว็บไซต์สามารถดึงข้อมูลจากแหล่งภายนอกและแสดงผลข้อมูลได้โดยไม่ต้องโหลดหน้าใหม่ทั้งหมด

7.7 Typescript

TypeScript คือภาษาโปรแกรมที่ถูกพัฒนาขึ้นโดย Microsoft ซึ่งเป็นซูเปอร์เซตของ JavaScript โดยที่ TypeScript เพิ่มฟีเจอร์ใหม่ๆ เช่น ระบบประเภทข้อมูล (Static Typing) และการตรวจสอบชนิดข้อมูล (Type Checking) ก่อนที่โค้ดจะถูกคอมไพล์เป็น JavaScript ทำให้สามารถจับข้อผิดพลาดได้ในช่วงพัฒนา และทำให้โค้ดมีความปลอดภัยและมั่นคงมากขึ้น TypeScript ช่วยให้โปรแกรมเมอร์สามารถกำหนดประเภทข้อมูลของตัวแปร ฟังก์ชัน หรือพารามิเตอร์ได้ ซึ่งช่วยลดข้อผิดพลาดที่อาจเกิดขึ้นในระหว่างการทำงานจริง โดยเฉพาะในโปรเจกต์ที่มีขนาดใหญ่หรือทีมพัฒนาหลายคน นอกจากนี้ TypeScript ยังรองรับฟีเจอร์ของ JavaScript เวอร์ชันใหม่ๆ รวมถึงการใช้งาน คลาส (Classes), อินเตอร์เฟซ (Interfaces) และการขยายฟังก์ชันการทำงาน (Decorators) ซึ่งทำให้สามารถพัฒนาแอปพลิเคชันที่มีโครงสร้างและการจัดการที่ดีกว่า JavaScript ธรรมดา อีกทั้ง TypeScript สามารถทำงานร่วมกับเครื่องมือและไลบรารีที่พัฒนาจาก JavaScript ได้อย่างราบรื่น

7.8 NodeJS

Node.js คือแพลตฟอร์มที่ใช้สำหรับการรัน JavaScript บนเซิร์ฟเวอร์ ซึ่งช่วยให้สามารถพัฒนาแอปพลิเคชันฝั่งเซิร์ฟเวอร์ได้โดยใช้ภาษา JavaScript ที่เป็นที่ยอมรับในฝั่งไคลเอนต์ (client-side) โดย Node.js ใช้ Google V8 JavaScript engine ซึ่งทำให้มีประสิทธิภาพในการประมวลผลที่สูงและรองรับการทำงานแบบ non-blocking I/O ที่ช่วยให้ระบบสามารถจัดการคำขอหลายๆ คำขอพร้อมกันได้อย่างมีประสิทธิภาพ เหมาะสำหรับการพัฒนาแอปพลิเคชันที่ต้องการความเร็วและสามารถรองรับการเชื่อมต่อจำนวนมากได้ เช่น เว็บแอปพลิเคชัน, API server หรือแอปพลิเคชันที่ใช้ real-time การพัฒนาใน Node.js ช่วยให้การพัฒนาแอปพลิเคชันทั้งฝั่งไคลเอนต์และเซิร์ฟเวอร์สามารถทำได้ในภาษาเดียวกัน ทำให้กระบวนการพัฒนาและบำรุงรักษาแอปพลิเคชันมีความง่ายขึ้น

นอกจากนี้ Node.js ยังมีชุดเครื่องมือและไลบรารีที่ช่วยในการพัฒนาแอปพลิเคชันอย่างรวดเร็ว เช่น การจัดการกับไฟล์, การเชื่อมต่อฐานข้อมูล และการสร้าง API

7.9 Dart

Dart คือภาษาโปรแกรมที่พัฒนาโดย Google ซึ่งออกแบบมาเพื่อใช้ในการพัฒนาแอปพลิเคชันทั้งในฝั่งเว็บและมือถือ โดยเฉพาะในการพัฒนาแอปพลิเคชันที่ใช้ Flutter ซึ่งเป็นเฟรมเวิร์กของ Google สำหรับการพัฒนาแอปพลิเคชันมือถือและเว็บที่สามารถทำงานได้ทั้งบน iOS, Android, และ Web ด้วยโค้ดเดียว Dart ถูกออกแบบมาให้มีความเร็วในการประมวลผลสูงและมีความสามารถในการทำงานแบบ asynchronous โดยใช้ Future และ Stream เพื่อจัดการกับการทำงานที่ต้องรอผลลัพธ์ เช่น การดึงข้อมูลจาก API หรือการทำงานกับไฟล์ใหญ่ ๆ นอกจากนี้ Dart ยังรองรับการเขียนโปรแกรมในลักษณะของ object-oriented programming (OOP) ซึ่งทำให้การพัฒนาแอปพลิเคชันมีความยืดหยุ่นและสามารถบำรุงรักษาได้ง่าย ภาษานี้มีเครื่องมือที่ช่วยในการดีบั๊กและทดสอบแอปพลิเคชันที่พัฒนาได้อย่างมีประสิทธิภาพ รวมถึงสามารถคอมไพล์เป็น native code เพื่อให้แอปพลิเคชันทำงานได้อย่างรวดเร็วและมีประสิทธิภาพบนอุปกรณ์ต่างๆ

7.10 Flutter

Flutter คือเฟรมเวิร์กที่พัฒนาโดย Google สำหรับการสร้างแอปพลิเคชันมือถือ, เว็บ และเดสก์ท็อปจากโค้ดเดียวกัน โดยใช้ภาษา Dart เป็นภาษาหลักในการเขียนโค้ด Flutter ถูกออกแบบมาเพื่อให้สามารถพัฒนาแอปพลิเคชันที่มีประสิทธิภาพสูงและมี UI ที่สวยงามโดยไม่ต้องเขียนโค้ดหลายชุดสำหรับแต่ละแพลตฟอร์ม ทำให้การพัฒนาแอปพลิเคชันเป็นไปอย่างรวดเร็วและสะดวก ด้วยการใช้ widget-based architecture ซึ่งช่วยให้ผู้พัฒนาสามารถสร้างส่วนประกอบของ UI ได้อย่างยืดหยุ่นและมีความสามารถในการปรับแต่งได้ตามต้องการ Flutter ยังรองรับการเขียนโปรแกรมแบบ reactive programming ที่ช่วยให้สามารถจัดการกับสถานะต่างๆ ของแอปพลิเคชันได้อย่างง่ายดายและมีประสิทธิภาพ อีกทั้งยังมีเครื่องมือที่ช่วยในการทดสอบและดีบั๊กแอปพลิเคชันได้อย่างรวดเร็ว โดยสามารถพัฒนาแอปพลิเคชันที่ทำงานได้ทั้งบน iOS, Android, และ Web ด้วยโค้ดเดียว ทำให้ Flutter เป็นทางเลือกที่นิยมในวงการพัฒนาแอปพลิเคชันที่ต้องการรองรับหลายแพลตฟอร์มพร้อมกัน

7.11 API

API (Application Programming Interface) คือชุดของกฎและโปรโตคอลที่ช่วยให้แอปพลิเคชันต่างๆ สามารถสื่อสารและแลกเปลี่ยนข้อมูลกันได้ โดยไม่ต้องรู้รายละเอียดของการทำงานภายในของแต่ละแอปพลิเคชันหรือระบบ API จะเป็นตัวกลางที่ให้แอปพลิเคชันหนึ่งสามารถร้องขอข้อมูล

หรือบริการจากแอปพลิเคชันหรือเซิร์ฟเวอร์อื่น ๆ และรับผลลัพธ์ที่ต้องการกลับมาในรูปแบบที่เข้าใจได้ เช่น JSON หรือ XML ตัวอย่างเช่น เมื่อแอปพลิเคชันบนมือถือต้องการดึงข้อมูลจากเซิร์ฟเวอร์หรือฐานข้อมูล API จะทำหน้าที่จัดการคำขอเหล่านั้นแล้วส่งกลับข้อมูลที่ร้องขอ API ช่วยให้การพัฒนาแอปพลิเคชันมีความยืดหยุ่นและสะดวกมากขึ้น เพราะสามารถใช้บริการจากระบบอื่นที่มีอยู่แล้วโดยไม่ต้องพัฒนาขึ้นใหม่ทั้งหมด นอกจากนี้ API ยังช่วยให้การทำงานร่วมกันระหว่างระบบต่างๆ เป็นไปได้อย่างมีประสิทธิภาพและสามารถขยายฟังก์ชันการทำงานของแอปพลิเคชันได้อย่างง่ายดาย

7.12 ORM

ORM (Object-Relational Mapping) คือเทคนิคการพัฒนาโปรแกรมที่ช่วยในการเชื่อมโยงข้อมูลจากฐานข้อมูลเชิงสัมพันธ์ (RDBMS) กับอ็อบเจกต์ในภาษาโปรแกรม ซึ่งทำให้สามารถเข้าถึงและจัดการข้อมูลในฐานข้อมูลได้ในรูปแบบของอ็อบเจกต์แทนการใช้คำสั่ง SQL โดยตรง ORM ช่วยให้การทำงานกับฐานข้อมูลสะดวกขึ้นโดยการแปลงข้อมูลระหว่างรูปแบบที่ฐานข้อมูลต้องการและรูปแบบที่โปรแกรมต้องการใช้ เช่น การใช้คลาสในภาษาโปรแกรมเพื่อแทนที่ตารางในฐานข้อมูลและการใช้แอตทริบิวต์ของคลาสเพื่อแทนที่คอลัมน์ในตาราง โดยที่นักพัฒนาสามารถดำเนินการเพิ่ม, ลบ, แก้ไข หรือดึงข้อมูลจากฐานข้อมูลได้โดยการใช้อ็อบเจกต์และเมธอดในโปรแกรมโดยไม่ต้องเขียน SQL ด้วยตนเอง ซึ่งช่วยลดข้อผิดพลาดที่อาจเกิดจากการเขียนคำสั่ง SQL และทำให้โค้ดมีความยืดหยุ่นและบำรุงรักษาง่ายขึ้น

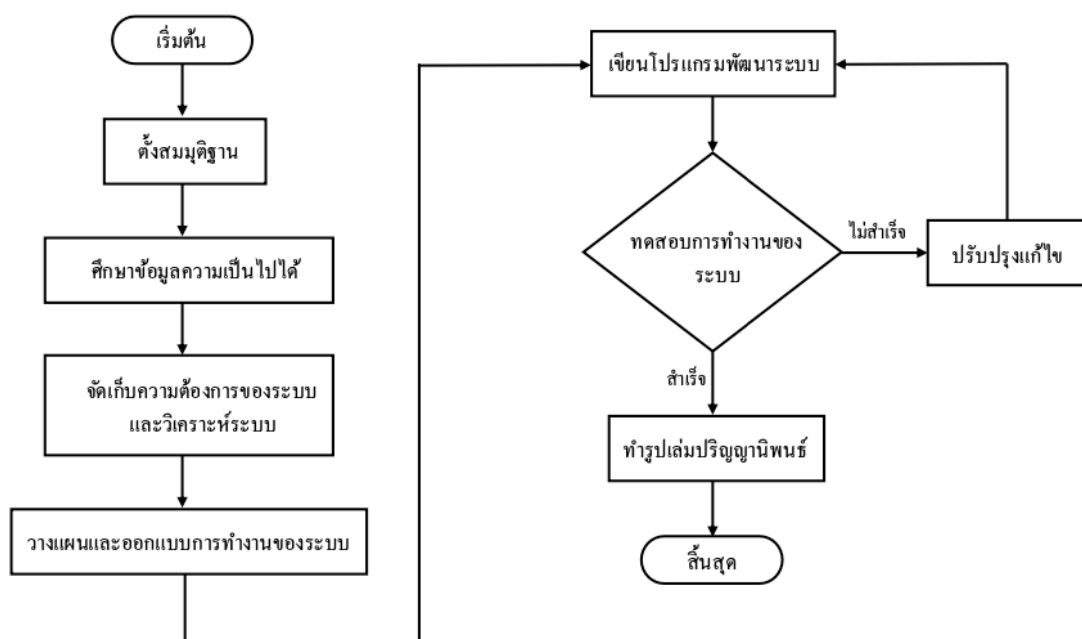
7.13 RESTful API

RESTful คือแนวทางการออกแบบสถาปัตยกรรมสำหรับการพัฒนาเว็บเซอร์วิสที่ใช้หลักการของ REST (Representational State Transfer) ซึ่งเป็นวิธีการที่กำหนดให้ระบบสามารถสื่อสารกันได้ผ่านโปรโตคอล HTTP โดยใช้คำสั่งพื้นฐานของ HTTP เช่น GET, POST, PUT, DELETE เพื่อดำเนินการกับข้อมูลที่อยู่ในรูปแบบของทรัพยากร (Resources) ที่ถูกระบุด้วย Uniform Resource Identifier (URI) โดยไม่จำเป็นต้องใช้สถานะในการเก็บข้อมูลระหว่างการเชื่อมต่อแต่ละครั้ง ซึ่งทำให้การสื่อสารในระบบ RESTful เป็นแบบ stateless หรือไม่เก็บสถานะใดๆ การใช้ RESTful API ช่วยให้การพัฒนาเว็บแอปพลิเคชันมีความยืดหยุ่นและสามารถปรับขยายได้ง่าย เนื่องจากสามารถแยกส่วนการทำงานระหว่างฝั่งไคลเอนต์และเซิร์ฟเวอร์ได้ชัดเจน ทำให้การบำรุงรักษาและขยายระบบทำได้สะดวกยิ่งขึ้น RESTful API จึงเป็นมาตรฐานที่นิยมใช้ในการพัฒนาแอปพลิเคชันที่ต้องการการสื่อสารข้อมูลระหว่างเซิร์ฟเวอร์และไคลเอนต์อย่างมีประสิทธิภาพ

7.14 HTTP Protocol

HTTP (Hypertext Transfer Protocol) คือโปรโตคอลที่ใช้ในการสื่อสารข้อมูลระหว่างเว็บเซิร์ฟเวอร์และเว็บเบราว์เซอร์หรือไคลเอนต์อื่นๆ บนอินเทอร์เน็ต โดย HTTP จะกำหนดวิธีการแลกเปลี่ยนข้อมูลระหว่างเครื่องคอมพิวเตอร์ในลักษณะที่ง่ายและรวดเร็ว เช่น การดึงหน้าเว็บ, การส่งข้อมูลจากฟอร์ม หรือการดาวน์โหลดไฟล์ HTTP ใช้รูปแบบข้อความที่ง่ายต่อการทำความเข้าใจ และใช้โปรโตคอลแบบ request-response ซึ่งหมายความว่าเมื่อไคลเอนต์ส่งคำขอ (request) ไปยังเซิร์ฟเวอร์ เซิร์ฟเวอร์จะตอบกลับ (response) ด้วยข้อมูลที่ร้องขอ เช่น หน้าสำหรับแสดงผล หรือข้อมูลที่ถูกประมวลผล HTTP เป็นโปรโตคอลที่ไม่คงสถานะ (stateless) ซึ่งหมายความว่าแต่ละคำขอจะถูกจัดการแยกจากกันโดยไม่ขึ้นกับคำขอก่อนหน้านี้ ทำให้สามารถรองรับการทำงานจำนวนมากได้ในเวลาเดียวกัน ซึ่งช่วยให้การทำงานบนเว็บมีประสิทธิภาพสูงและยืดหยุ่น

8.แผนภูมิการทำปริญญานิพนธ์



รูปที่ 2 แผนภูมิการดำเนินงาน

9. แผนตารางเวลาในการทำปริญญานิพนธ์

กิจกรรม	ระยะเวลาการปฏิบัติงาน				
	พ.ย. 2567	ธ.ค. 2567	ม.ค. 2568	ก.พ. 2568	มี.ค. 2568
1. ศึกษาข้อมูลและวางแผนการทำงาน					
2. ออกแบบหน้าเว็บแอปพลิเคชันและแอปพลิเคชัน					
3. ออกแบบและจัดทำฐานข้อมูล					
4. เริ่มพัฒนาระบบ					
5. ทดสอบระบบ					
6. จัดทำรูปเล่มปริญญานิพนธ์					

ตารางที่ 1 ตารางเวลาการดำเนินงาน

10. เอกสารอ้างอิง

1. <https://developer.mozilla.org/en-US/docs/Web/HTML>
2. <https://developer.mozilla.org/en-US/docs/Web/CSS>
3. <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
4. <https://developer.mozilla.org/en-US/docs/Glossary/TypeScript>
5. <https://v17.angular.io/guide/what-is-angular>
6. <https://www.w3schools.com/bootstrap5/>
7. <https://www.quest.com/learn/what-is-my-sql.aspx>
8. <https://kongruksiam.medium.com/%E0%B8%A3%E0%B8%B9%E0%B9%89%E0%B8%88%E0%B8%B1%E0%B8%81%E0%B8%81%E0%B8%B1%E0%B8%9A-javascript-%E0%B9%81%E0%B8%A5%E0%B8%B0-nodejs-8b5041853eae>
9. <https://aws.amazon.com/th/what-is/flutter/>
10. <https://aws.amazon.com/th/what-is/api/>

11. <https://aws.amazon.com/th/compare/the-difference-between-https-and-http/>

12. [https://medium.com/@amanee.ha/orm-](https://medium.com/@amanee.ha/orm-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3-9c8c6ffc13cb)

[orm-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%](https://medium.com/@amanee.ha/orm-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3-9c8c6ffc13cb)

[B8%A3-9c8c6ffc13cb](https://medium.com/@amanee.ha/orm-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3-9c8c6ffc13cb)