



CPE327

SOFTWARE ENGINEERING

TOAA NIUNG

จัดทำโดย

นายเดชาวัฒน์ ร้อยด้า 64070501021

นายจิณณะ วรรณโสภา 64070501060

นายทวัต สุริยะ 64070501065

นางสาวน้ำทิพย์ พันสาย 64070501071

นายไพบูลย์ อาจทวีคุณ 64070501079

อาจารย์ผู้สอน

ผศ.ดร.พร พันธุ์จงหาญ

ผศ.ดร.ชจรพงษ์ อัคร吉ตสกุล

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ภาคเรียนที่ 1 ปีการศึกษา 2565

## บทคัดย่อ

TOAA-NIUNG เป็นเว็บแอปพลิเคชันเพื่อการพัฒนาระบบจดจำการคิวสำหรับร้านอาหารสำหรับผู้ประกอบการ ซึ่งแก้ไขปัญหาที่ลูกค้าพบในการจองและรอคิวในร้านอาหารขนาดใหญ่ โดยใช้แพลตฟอร์มเว็บแอปที่จะทำให้การจองเป็นไปอย่างราบรื่นและสะดวกสบาย ระบบนี้สามารถให้ลูกค้าทำการจองโต๊ะ เลือกเวลา และระบุความต้องการเพิ่มเติม ระบบการยกเลิกและแจ้งเตือนเวลาเข้าร้าน ลดปัญหาเรื่องเวลาของผู้ประกอบการตามสถานการณ์ นอกจากตอบสนองความต้องการของลูกค้า และผู้ประกอบการแล้ว สามารถที่เข้ามีส่วนช่วยเรื่องข้อมูลที่แสดงรูปแบบ DashBoard เพื่อวิเคราะห์ข้อมูลลูกค้า

ผู้จัดทำหวังว่าเว็บแอปพลิเคชันถูกสร้างสำหรับผู้ประกอบการร้านอาหารที่ต้องการรองรับจำนวนลูกค้าและสามารถนำข้อมูลปรับแก้ส่วนที่มีผลกระทบต่อร้านอาหารลดลง โดยผู้จัดทำเลือกวิธีการดำเนินงานด้วยกระบวนการ Scum และ Waterfall เพื่อฝึกฝนการทำงานของผู้จัดทำ

វគ្គិទ្ទេ

អនុវត្ត

Problem

To-Do List

Functional Requirement

Data Design

Development Activities

- Software Process Paradigms

- Process Activities

- Project Planning

Coding Standard

Configuration Documentation

- Source code management

- Development process tools

- Development Outcomes

- Use Case Diagram

- Sequence Diagram

- Navigation Map

Testing

Component-level Design

Self-Evaluation and Summary

**Problem**

ปัจจุบันมีร้านค้าจำนวนมาก ในประเทศไทย แต่บางร้านนั้นมีลูกค้าจำนวนมากที่อยากรองเข้าไปใช้ผลิตภัณฑ์ของร้าน แต่ก็ต้องเดินทางไปเพื่อไปต่อคิว คณะผู้จัดทำโดยมีจุดประสงค์ในการสร้าง Simulation จัดการจองร้าน และจองที่นั่งได้ เพื่อที่จะสะดวกต่อร้านค้าขนาดใหญ่ที่รับจำนวนลูกค้าจำนวนนวนมาก ลูกค้าไม่ต้องมาจองหน้าร้าน แจ้งเตือนทั้งเจ้าของร้านและลูกค้าในระบบ และจัดเป็นคิวเรียบร้อย

เว็บไซต์นี้เป็นแพลตฟอร์มที่ช่วยในการจัดการระบบจองและคิวออนไลน์ให้กับร้านค้าและลูกค้า โดยช่วยให้ลูกค้าสามารถทำการจองและคิวออนไลน์ได้ง่ายเสริมสร้างประสบการณ์การเข้าร้านค้าที่สะดวกและเรียบง่ายสำหรับลูกค้า ช่วยให้ร้านค้าสามารถจัดโต๊ะหรือที่นั่งตามแผนการของร้านได้อย่างมีประสิทธิภาพมากขึ้นในการจัดการและปรับปรุงการดำเนินงานของร้านค้าด้วยการใช้เทคโนโลยีออนไลน์และการแจ้งเตือนในเวลาที่เหมาะสม

## To-Do List

### 1. Project Overview:

- เขียนบทนำเพื่ออธิบายวัตถุประสงค์และเป้าหมายของโปรเจค
- กำหนดขอบเขตของโครงการรวมถึงคุณสมบัติและฟังก์ชันการทำงาน

### 2. Project Planning:

- สร้างปฏิทินการทำงาน

### 3. Requirements Analysis:

- รวบรวม requirement ต่างๆจากร้านอาหารจริงๆ
- สร้าง use cases diagram

### 4. System Design:

- สร้าง diagram ต่างๆที่แสดงถึงส่วนต่างๆในระบบว่ามัน interact กันยังไง
- ออกแบบและร่างโครงระบบ

### 5. User Interface Design:

- ออกแบบเว็บไซต์ที่ออกแบบมาหน้าตาดียังไงดอย่างไร (UX และ UI)

### 6. Database Design:

- กำหนด schema database รวมถึงตาราง, ความสัมพันธ์ของตาราง และชนิดของ data
- พิจารณาความปลอดภัยของข้อมูลและความคุ้มครองเข้าถึงการใช้

### 7. Development:

- เริ่มน้ำสีงที่ออกแบบไว้ไป Implement ใน code จริงๆ
- ค่อยอัพเดต source code ที่ใช้ต่างๆใน github

### 8. Testing and Quality Assurance:

- ลองใช้งานเว็บจริงๆตามสถานการณ์ต่างๆที่กำหนดไว้เพื่อดู bugs หรือ errors ต่างๆ และแก้ไข
- จดบันทึก bugs กับ errors ต่างๆที่เจอเพื่อได้เจออีก

### 9. Deployment:

- วางแผนการ deployment รวมถึง server setup และ configuration และลงมือทำตามแผน
- ทดสอบเว็บที่ถูก deploy เพื่อเช็คเรื่องความปลอดภัยและ errors ต่างๆที่อาจจะเกิดขึ้นตอน deploy

## Functional Requirements

(ผู้ด้านลูกค้า)

### ระบบจอง

- เลือกโต๊ะในร้านค้าที่จะจอง
- เลือกวันที่ เวลา จำนวนคนในโต๊ะ และคำขอเพิ่มเติมเพิ่มเติมได้ ( เช่น อยากรีหัจด์ตั๊ะแบบไหน )
- ลูกค้าจะต้องกดยกเลิก แล้วจองใหม่เท่านั้น ถ้าจะแก้ไข
- จองและไม่มาเกิน 3 ครั้ง ไอเดียจะกระงับการใช้งาน (ล็อคอินไม่สำเร็จ)
- ลูกค้า 1 user สามารถจอง 1 ครั้ง ต่อ 1 โต๊ะ ถ้าต้องจองเพิ่มให้จองใหม่

### ระบบบริการลูกค้า

- ระบบการยกเลิกโต๊ะที่จอง

### แดชบอร์ดลูกค้า

- หน้าโปรไฟล์
- ประวัติการจอง
- การตั้งค่าโปรไฟล์ แก้ไขข้อมูลส่วนตัว
- แสดงสถานะการจองต่างๆที่จองไป

(ผู้ด้าน admin)

### ระบบผังร้าน(Create)

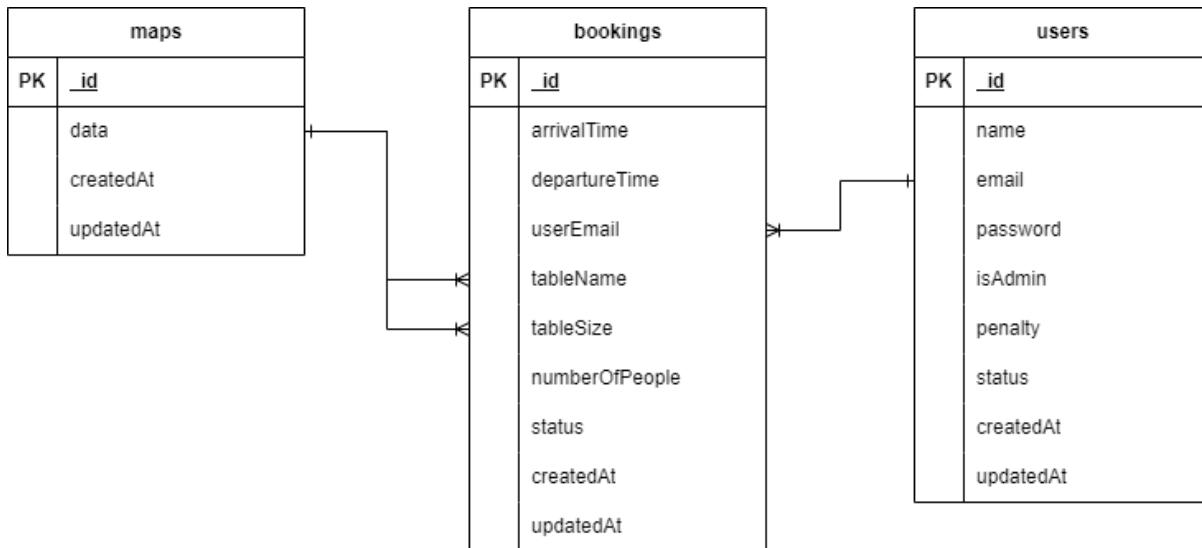
- สามารถสร้าง Sim จัดการที่นั่งของร้านค้าตัวเองได้ (edit) -Mqr โต๊ะในผังร้านใส่จำนวนคนต่อโต๊ะ
- เปลี่ยนผังร้านในวันนั้นตามที่ลูกค้าอวยากรีด (ข้อเสนอจากลูกค้าที่ต้องการเปลี่ยนผังการวาง)
- สามารถเปลี่ยนผังร้านแล้วจะอัพเดตทันที

### แดชบอร์ด admin

- ดูประวัติการจองโต๊ะทั้งหมดของลูกค้า
- Blacklist ลูกค้าที่จองแล้วไม่มา
- ดูข้อมูลของลูกค้าทั้งหมดได้

## Data Design

- ER Diagram



### 1) ตาราง bookings

ตาราง bookings เป็นตารางสำหรับเก็บ bookings ในการจองโต๊ะของ user

bookings			
Attribute	Data Type	Detail	Example
_id	ObjectId	เก็บ ID ของผู้ใช้	6552787421acf999e16b0ea5
arrivalTime	Date	เก็บเวลาที่ผู้ใช้จะมาถึง	2023-11-14T02:26:00.000+00:00
departureTime	Date	เก็บเวลาที่ผู้ใช้จะกลับ (arrivalTime + 2 ชม.)	2023-11-14T04:26:00.000+00:00
userEmail	String	เก็บ Email ของผู้ใช้	jin@gmail.com
tableName	String	เก็บชื่อโต๊ะที่ผู้ใช้นั่ง	S2
tableSize	String	เก็บขนาดของโต๊ะที่ผู้ใช้เลือก	TableS
numberOfPeople	Integer(32)	เก็บจำนวนคนที่มา	1
status	String	เก็บสถานะของ booking	Cancelled
createdAt	Date	เวลาที่ข้อมูลถูกสร้างขึ้นใน database	2023-11-13T19:26:44.740+00:00
updatedAt	Date	เวลาที่ข้อมูลล่าสุด update ใน database	2023-11-14T16:43:29.330+00:00

2) ตาราง maps

ตาราง maps เป็นตารางสำหรับเก็บค่า ตำแหน่ง ขนาด ชื่อ จำนวนสูงสุดที่สามารถนั่งได้ เพื่อสร้างโต๊ะใน maps ของร้าน

maps			
Attribute	Data Type	Detail	Example
_id	ObjectId	เก็บ ID ของ maps	656dc4ecc76887b07060d4c9
data	Object	เก็บตำแหน่ง,ขนาด,ชื่อโต๊ะ,จำนวนสูงสุดของคนนั่ง	<pre>"data": [   {     "x": 74,     "y": 68,     "shape": "TableL",     "tableName": "B1",     "maxPeople": 6   } ]</pre>
createdAt	Date	เวลาที่ข้อมูลนี้ถูกสร้างขึ้นใน database	2023-12-04T12:24:12.934+00:00
updatedAt	Date	เวลาที่ข้อมูลนี้ถูก update ใน database	2023-12-04T12:24:12.934+00:00

### 3) ตาราง users

ตาราง users เป็นตารางสำหรับเก็บข้อมูลส่วนตัวของ user รวมไปถึง Role ของ user และสถานะของบัญชีที่ยังสามารถใช้ได้หรือไม่ได้

users			
Attribute	Data Type	Detail	Example
_id	ObjectId	เก็บ ID ของผู้ใช้	6552367589b2e78cb097c269
name	String	เก็บ ชื่อของผู้ใช้	Jinna
email	String	เก็บ Email ของผู้ใช้	jin@gmail.com
password	String	เก็บ Hashed Password ของผู้ใช้	\$2a\$10\$oVH1npO9DCLSnzK9A033g.Xje0V6Z0N.rv05ZC3IJYI UJDQ7b5cAu
isAdmin	Boolean	เก็บค่าสถานะว่าเป็น Admin	true
penalty	Integer(32)	ใช้ในการนับจำนวน booking ที่ผู้ใช้จ่อแล้ว ไม่มา (ถ้ามากกว่า 3 จะถูกแบน)	1
status	String	เก็บค่าสถานะ ที่บัญชีนี้ยังไม่ถูกระงับ	Active(ใช้ได้อยู่), Banned(ไม่สามารถใช้ได้)
createdAt	Date	เวลาที่ข้อมูลนี้ถูกสร้างขึ้นใน database	2023-11-13T14:45:09.448+00:00
updatedAt	Date	เวลาที่ข้อมูลนี้ถูก update ใน database	2023-11-19T06:36:30.384+00:00

## Development Activities

### Software Process Paradigms

ทางผู้จัดทำเลือก Scrum เป็น Process หลัก เนื่องจากเลือกใช้ข้อด้อยเกิดจากระยะเวลาสั่งผลกระทบต่อการศึกษา ดำเนินการเปลี่ยน Process เป็นนำบางส่วนของ Scrum มาปรับใช้ร่วมกับ Waterfall แบ่งเป็น 2 ส่วน

- Waterfall Model เราได้ตั้ง Process ของ Waterfall มาใช้ช่วยในการทำงาน Sprint

- a. กำหนดหน้าที่ให้ผู้จัดทำ
- b. การย้อนขันตอนกลับไปดำเนินการขั้นตอนก่อนหน้า

ผู้จัดทำ	หน้าที่
64070501021 นายเดชาวัฒน์ ร้อยดา	Front End
64070501060 นายจิณณะ วรรණโสภาค	Full Stack / Report
64070501065 นายทรัพ สริยะ	Document / Planning / Report
64070501071 นางสาวน้ำทิพย์ พันสาย	Design / Report
64070501079 นายไพบูลย์ อาจทวีคุณ	Front End / Design

- Scrum เรานำ Scrum บางส่วนมาใช้เป็น Process หลักในการดำเนินงาน

- a. Sprint แบ่งเป็น 7 Sprint
- b. Sprint Planning วางแผนแต่ละ Sprint เพื่อเร้นจากช่วงสอบ
- c. Sprint Review ผู้จัดทำ Review เมื่อจบแต่ละ Sprint
- d. Product Backlog แบ่ง Backlog ตาม Requirement
- e. Sprint Backlog
- f. Daily Scrum สำรวจเกิดขึ้นวันแสาร์หรือวันอาทิตย์

### Process Activities

- Requirement Gathering and Analysis

เริ่มจากเก็บ Requirement จากผู้ต้องการใช้งานจากพนักงานร้าน Indy Bar ผู้ใช้เสนอ จากการที่เราที่เก็บ Requirement จากพนักงานร้านมี Feature ดังนี้

- เว็บไซต์ที่สามารถให้ลูกค้าจองโต๊ะ
- กำหนดลิมิตคนที่นั่ง / โต๊ะ
- กำหนดเวลาการจองโต๊ะ

- กำหนดสีของโต๊ะแสดงสถานะของ / ไม่จอง
- สามารถจัดการโต๊ะตามรูปแบบร้าน

## 2. Design

ผู้จัดทำวางแผนพัฒนาออกแบบโดยใช้เครื่องมือ เช่น      Use case diagram Sequence diagram Website Development Tools (MERN Stack)

## 3. Development

เมื่อได้รับ Requiement เราทำการพัฒนา Web application ตามที่มีการออกแบบ

- Design บางส่วนไม่สามารถทำได้ตามกำหนดจึงลดบางส่วนเพื่อตอบโจทย์กับการใช้งาน เช่น ลดขนาด Header ให้ขนาดเหมาะสมสำหรับแพทฟอร์ม PC และ Phone
- มีการสร้าง Git hub ใช้สำหรับผู้จัดทำ เพื่อแบ่งงานตามหน้าที่
- การทำงานแบ่งเป็นแต่ละ Sprint เพื่อกำหนดขอบเขต
- เปลี่ยน Daily Scum เป็น 1 - 2 ครั้ง ต่ออาทิตย์ เพื่อรวมเข้ากับช่วงสอบที่เปลี่ยนไป
- เนื่องจากแบ่งหน้าที่ตั้งแต่เริ่มต้น ในแต่ละ Backlog แต่จะมีการ Review กับผู้จัดทำและบันทึกทุกครั้ง ลดปัญหาการเกิด Error human

## 4. Testing

มีการวางแผน Test cases และ Test ตามความต้องการให้สามารถตอบโจทย์

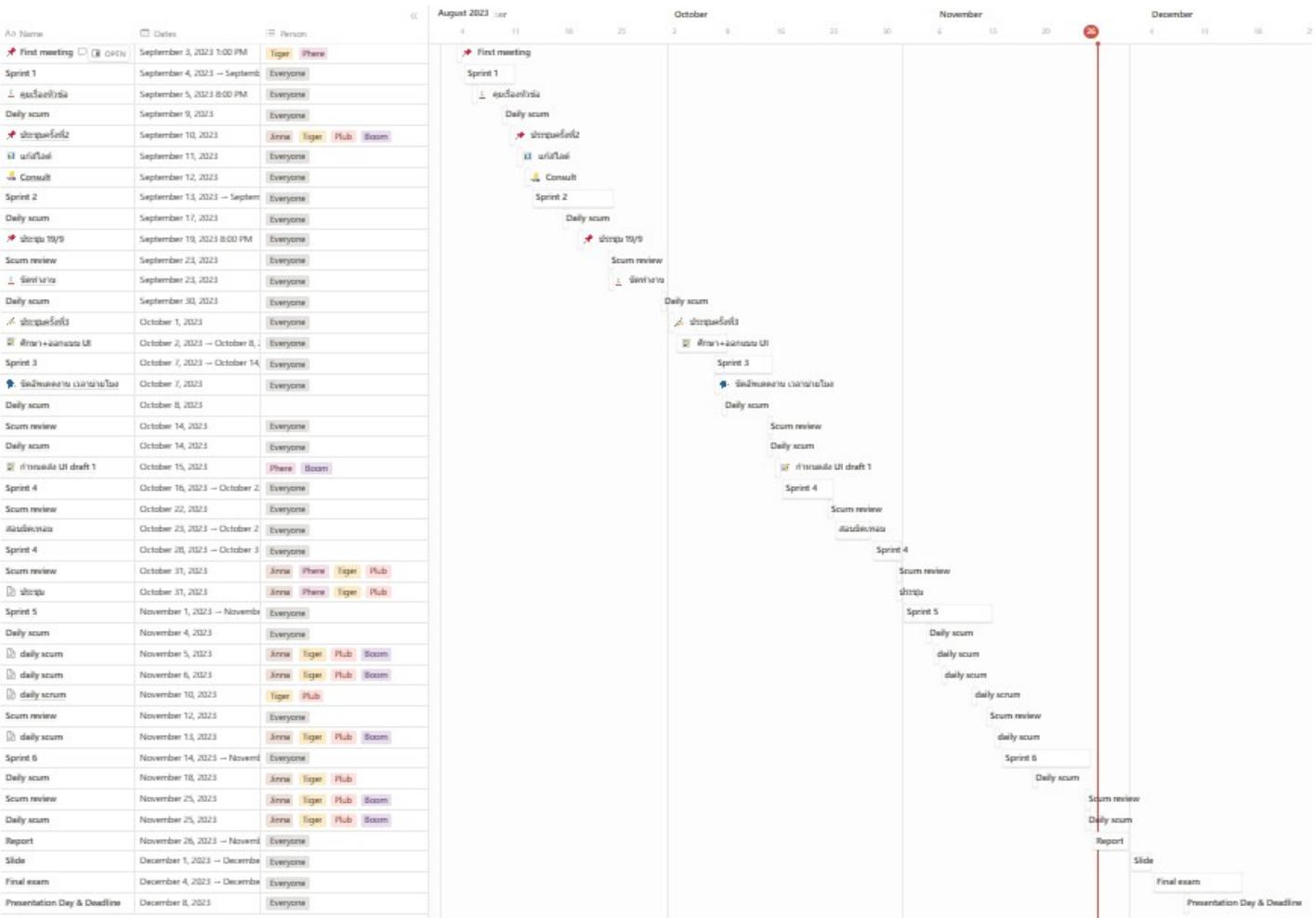
## 5. Deployment

เราทำการ Deploy web โดยใช้ Vercel

## Project Planing

เรามีการวางแผน Project เริ่มต้นตั้งแต่กระบวนการแรก โดยนำ Scum , Waterfall Model ปรับใช้ดังนี้

- 1 Sprint ถูกย่อที่มีความยาวประมาณ 1 - 2 สัปดาห์
- Daily scum ระยะเวลาประมาณ 30 นาที ทุกวันเสาร์หรือวันอาทิตย์ โดยอัพเดตของแต่ละคนว่าทำงานอะไรบ้าง ทำอะไรต่อไปและเกิดปัญหาอะไรบ้าง สรุปของแต่ละคนจะให้สมาชิกในกลุ่มเสนอว่าผ่านการพิจารณาหรือไม่ตามความเห็นของผู้จัดทำ ถ้าเกิดปัญหาจะต้องทำการแก้ไขปัญหานั้นโดยทันที
- Sprint Review หลังจบเราจะเปลี่ยนวัน Daily scum เป็นวัน review สำหรับเมื่อจบ Sprint การประชุมมีหัวข้อเกี่ยวกับงานของแต่ละคนทำว่าเสร็จสิ้นในงานที่เสร็จสิ้นเมื่อไรบ้าง โดยให้สถานะของคนอื่นประมาณความถูกต้องจาก Requirement ว่าตรงหรือไม่ ถ้าไม่จะมีการ Comment เพื่อกลับไปทำการปรับปรุงแก้ไขและกำหนด Deadline ส่งให้ทำการประเมินอีกรอบ
- หลัง Sprint review ผู้จัดทำจะทำการวางแผนสำหรับ Sprint ถัดไปเมื่อ Sprint ก่อนหน้าถูกแก้ไขปัญหาเสร็จสิ้นเท่านั้น
- การวางแผน Sprint จะถูกวางแผนไว้วันเพื่อไม่ให้เข้าใกล้วันสอบและเหลือเวลาเพื่อแก้ไขปัญหาที่อาจจะเกิดขึ้น



รูปภาพที่ 1 Project Scheduling

Sprint	Date duration
1	4/09/2566 - 10/09/2566
2	13/09/2566 - 23/09/2566
3	7/10/2566 - 14/10/2566
4	16/10/2566 - 22/10/2566
5	1/11/2566 - 12/11/2566
6	14/11/2566 - 25/11/2566

## ตารางที่ 1 Sprint Planning

### Coding Standard

หัวข้อ	รายละเอียด
Variable and Function Names	ใช้ camelCase ในการตั้งชื่อตัวแปรและฟังก์ชันต่างๆ
Comment	Double-slash "//" สำหรับ comment บรรทัดเดียว, /* ... */ สำหรับ comment หลายบรรทัด
Quotations	ใช้ “(Double Quotes) คุณ string ทุกตัว
Path ไปหน้าต่างๆ (Folder: (site))	ใช้ตัวอักษรภาษาอังกฤษตัวพิมพ์เล็กทั้งหมด
Imported component names	ใช้ PascalCase ในการตั้งชื่อ component ต่างๆ ที่ import เข้ามา

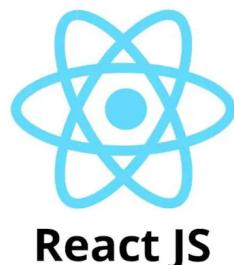
### Configuration Documentation

#### Source code management

มี Github , Git รวบรวม Project แต่ละไฟล์จะถูกรวมไว้ใน ToaaNiung การเพิ่มหน้า ไฟล์ API หรือ API จะต้องสร้างไฟล์ใหม่เพื่อเชื่อมไปยังหน้าที่ต้องการ เมื่อแก้ไขเสร็จสิ้นจะทำการ Branch ขึ้นบน Github ที่ Master เนื่องจากไฟล์ใหม่ที่ถูกแก้ไขจะถูกแก้ไขจากไฟล์หลักแยกตามหน้าที่และนำเข้ามาร่วมกันทีหลัง ปัญหาที่เกิดขึ้นมี Source code บางส่วน Erorr แก้ไขโดยการสื่อสารการนำเข้าร่วมกับไฟล์หลัก

#### Development process tools

- Website Development Tools





# NextAuth.js



- Deployment Tool



- API Testing Tool



# POSTMAN

- Design and Prototyping Tool



## Development Outcomes

1. Report
2. Source code
3. Sprint Backlog

Sprint 1 : 04/09 - 10/09

เป้าหมาย Sprint 1 คือเลือกหัวข้อโปรเจค เก็บ Requirement กำหนดขอบเขตการทำ

Story point : 28

งานที่ทำภายใน Sprint เสร็จสิ้น

งาน	Story point
กำหนดหัวข้อโปรเจค	2
เก็บ Requirement	7
กำหนดฟีเจอร์	7
เลือกรอบบการทำงาน	3
รวม	19

งานที่ทำภายใน Sprint ไม่เสร็จสิ้น

งาน	Story point
เลือก Tools	9
รวม	9

Sprint 2 : 13/09 - 23/09

เป้าหมาย Sprint 2 คือออกแบบหน้าตาและ functionality ของเว็บไซต์ และหาวิธีการทำ feature หลักของเว็บไซต์

Story point : 111

งานที่ทำเสร็จก่อนเริ่ม Sprint

งาน	Story point
เลือก Tools	9

รวม	9
-----	---

งานที่ทำภายใน Sprint เสร็จสิ้น

งาน	Story point
พาข้อมูลเกี่ยวกับการใช้ canvas เพื่อใช้ทำ feature หลัก(สร้าง/จอง เตี๊ย) ของเว็บไซต์	21
ทำความสะอาดให้กับ NextAuth.js	34
รวม	55

งานที่ทำภายใน Sprint ไม่เสร็จสิ้น

งาน	Story point
ออกแบบ UI ใน figma	34
เขียน Use Case กรณีต่างๆ	13
รวม	47

Sprint 3 : 07/10 - 14/10

เป้าหมาย Sprint 3 คือลงมือ implement feature หลักของเว็บไซต์และออกแบบ UI ที่ยังไม่เสร็จต่อ

Story point : 157

งานที่ทำภายใน Sprint เสร็จสิ้น

งาน	Story point
ออกแบบ UI ใน figma	34
เขียน Use Case กรณีต่างๆ	13
ทำหน้า admin ส่วนที่สร้างແ榜ผั้งของร้าน	55
ทำหน้า register และ log in	34
รวม	136

งานที่ทำภายใน Sprint ไม่เสร็จสิ้น

งาน	Story point
ทำหน้า customer ส่วนที่เลือกตัวของ (หน้า findtable)	21
รวม	21

Sprint 4 : 16/10 - 22/10

เป้าหมาย Sprint 4 คือทำ feature หลักที่เหลือต่อและทำเริ่ม feature รองๆ และเริ่มต้นแต่งเว็บไซต์ให้เป็นไปตามที่ design ไว้

Story point : 110

งานที่ทำภายใน Sprint เสร็จสิ้น

งาน	Story point
ทำหน้า customer ส่วนที่เลือกตัวของ (หน้า findtable)	21
ทำหน้า confirmbooking เพิ่ม	13
ทำหน้า myprofile ฝั่งลูกค้า (component ย่อย: bookinghistory และ editprofile)	34
เพิ่ม component ย่อยหน้า admin (allbookings)	21
รวม	89

งานที่ทำภายใน Sprint ไม่เสร็จสิ้น

งาน	Story point
ตกแต่งเว็บไซต์หน้าที่ทำเสร็จ (เสร็จแค่ functionality) แล้วให้ตรงตามที่ออกแบบ UI	21
รวม	21

Sprint 5 : 01/11 - 12/11

เป้าหมาย Sprint 5 คือทำระบบเช็ค booking ที่หมดอายุโดยใช้ timeinterval() และ Vercel Cron Jobs ,ตกลแต่งเว็บไซต์ต่อให้ออกมาตามที่ design ไว้ และทดลอง deploy เว็บไซต์

Story point : 47

งานที่ทำภายใน Sprint เสร็จสิ้น

งาน	Story point
ตกลแต่งเว็บไซต์หน้าที่ทำservice (service แค่ functionality) และวิธีตrang ตามที่ออกแบบ UI	21
ทดลอง deploy เว็บไซต์ลงบน vercel	13
รวม	34

งานที่ทำภายใน Sprint ไม่เสร็จสิ้น

งาน	Story point
ทำระบบเช็ค booking ที่หมดอายุ	13
รวม	13

Sprint 6 : 14/11 - 25/11

เป้าหมาย Sprint 6 คือ test ระบบและ feature ต่างๆ และแก้ไขบัคต่างๆ

Story point : 165

งานที่ทำภายใน Sprint เสร็จสิ้น

งาน	Story point

ทำระบบเช็ค booking ที่หมดอายุ	13
Test feature ต่างๆบนเว็บที่ deploy ไป	8
Testing ระบบ	55
รวม	76

งานที่ทำภายใน Sprint ไม่เสร็จสิ้น

งาน	Story point
รายงาน	89
รวม	89

Sprint 7 : 26/11 - 07/12

เป้าหมาย Sprint 7 คือการทำรายงานโครงการให้เสร็จและ สไลด์พรีเซ็นต์

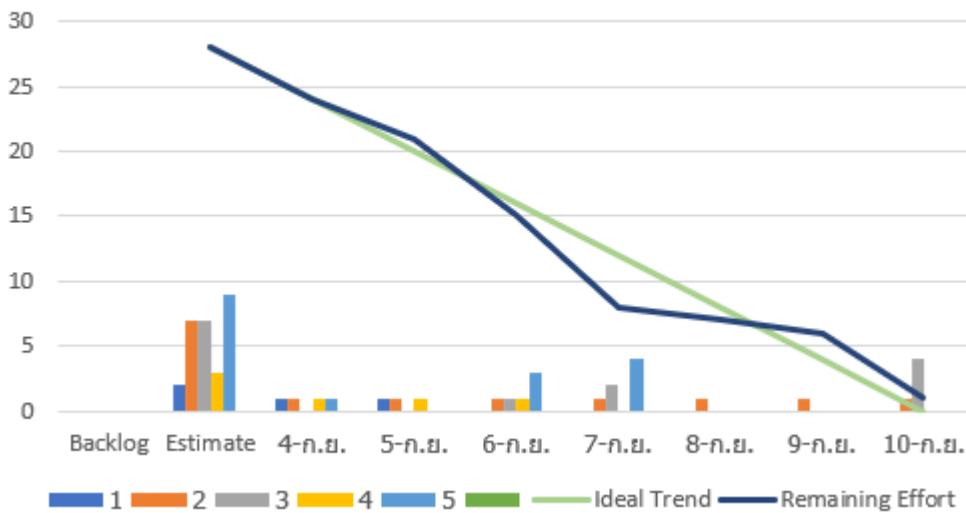
Story point : 110

งานที่ทำภายใน Sprint เสร็จสิ้น

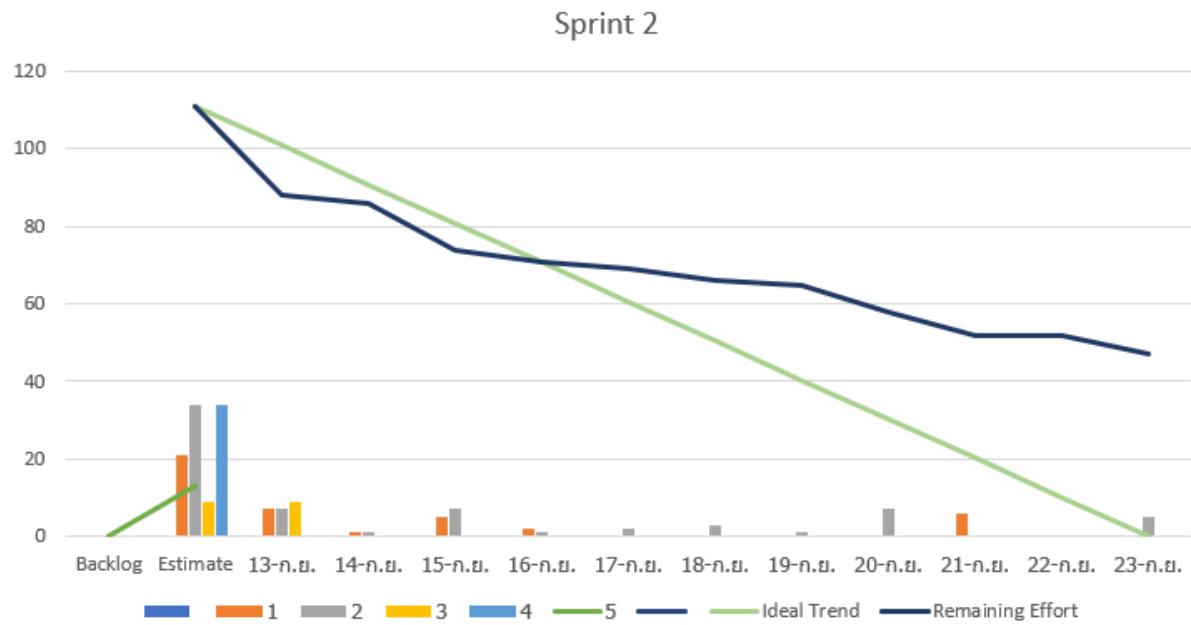
งาน	Story point
รายงาน	89
สไลด์นำเสนอ	21
รวม	110

Sprint 1

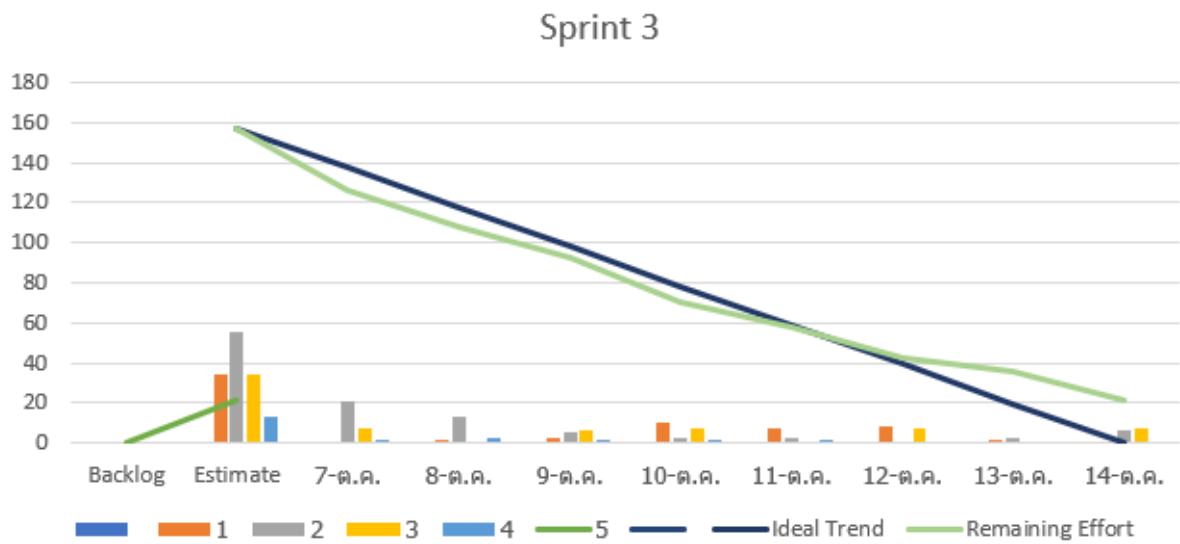
### Sprint1



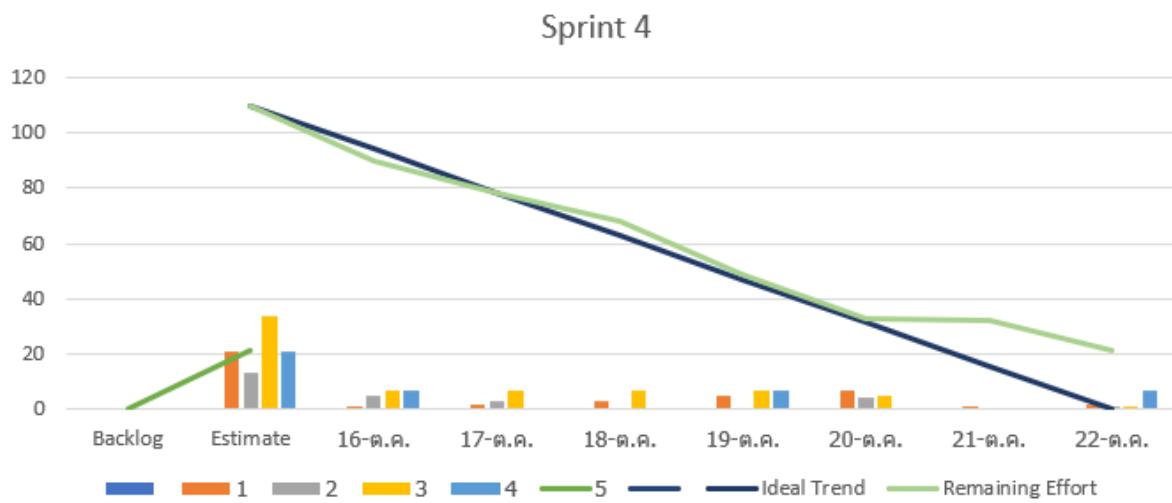
### Sprint 2



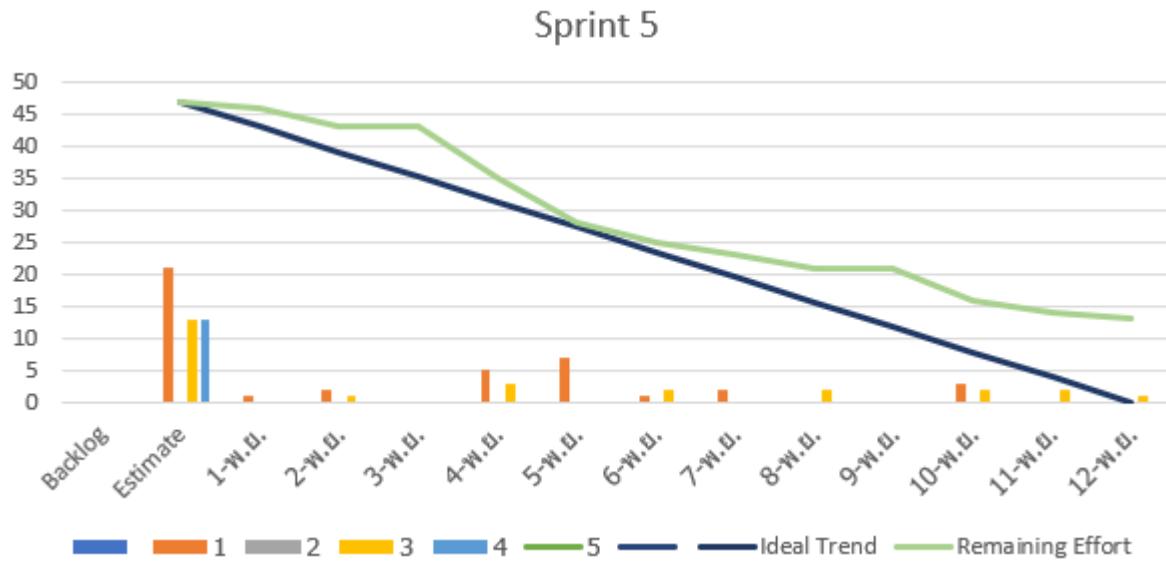
### Sprint 3



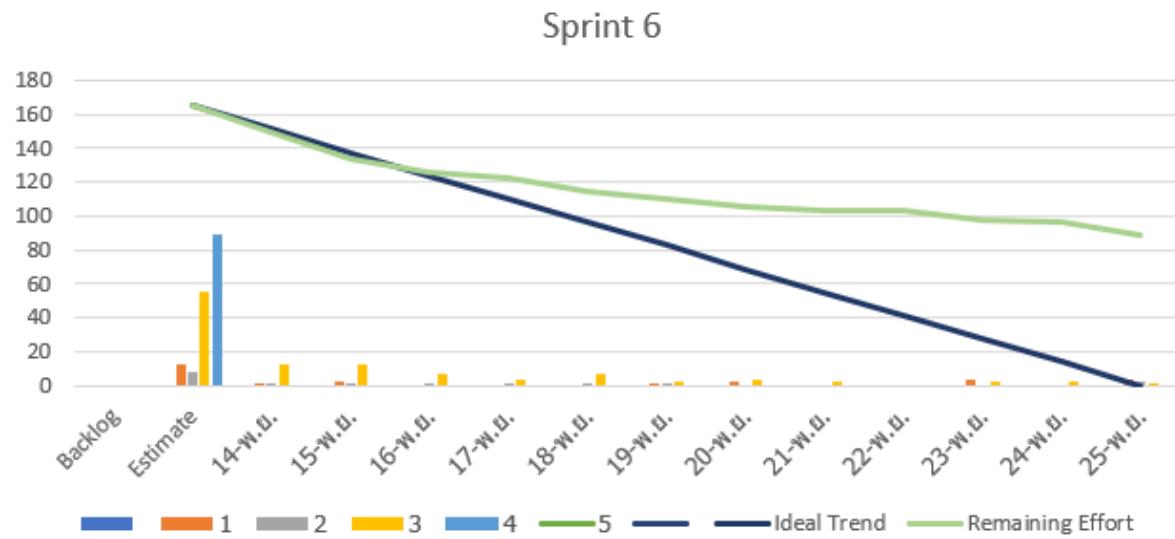
#### Sprint 4



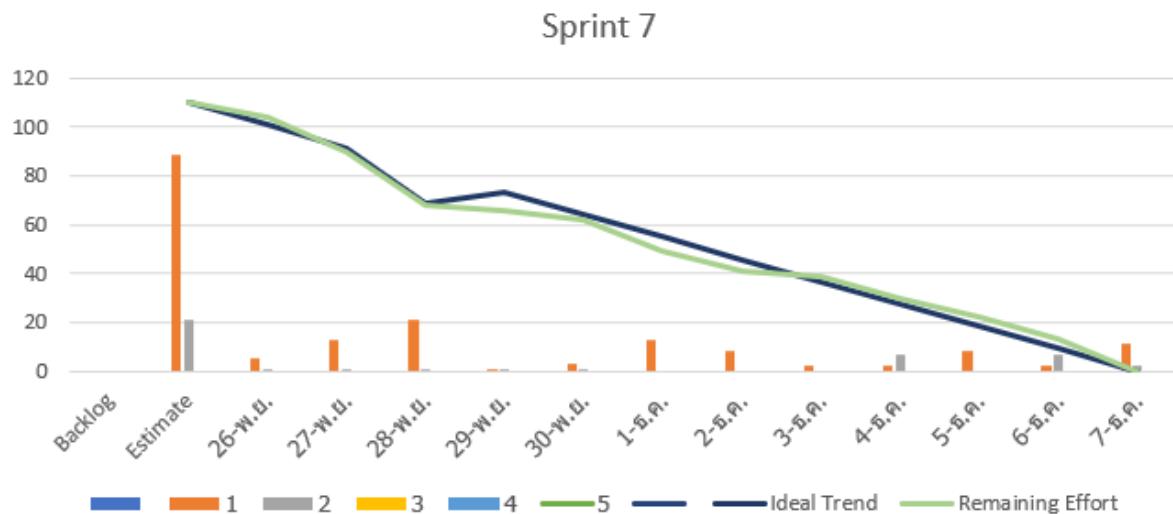
## Sprint 5



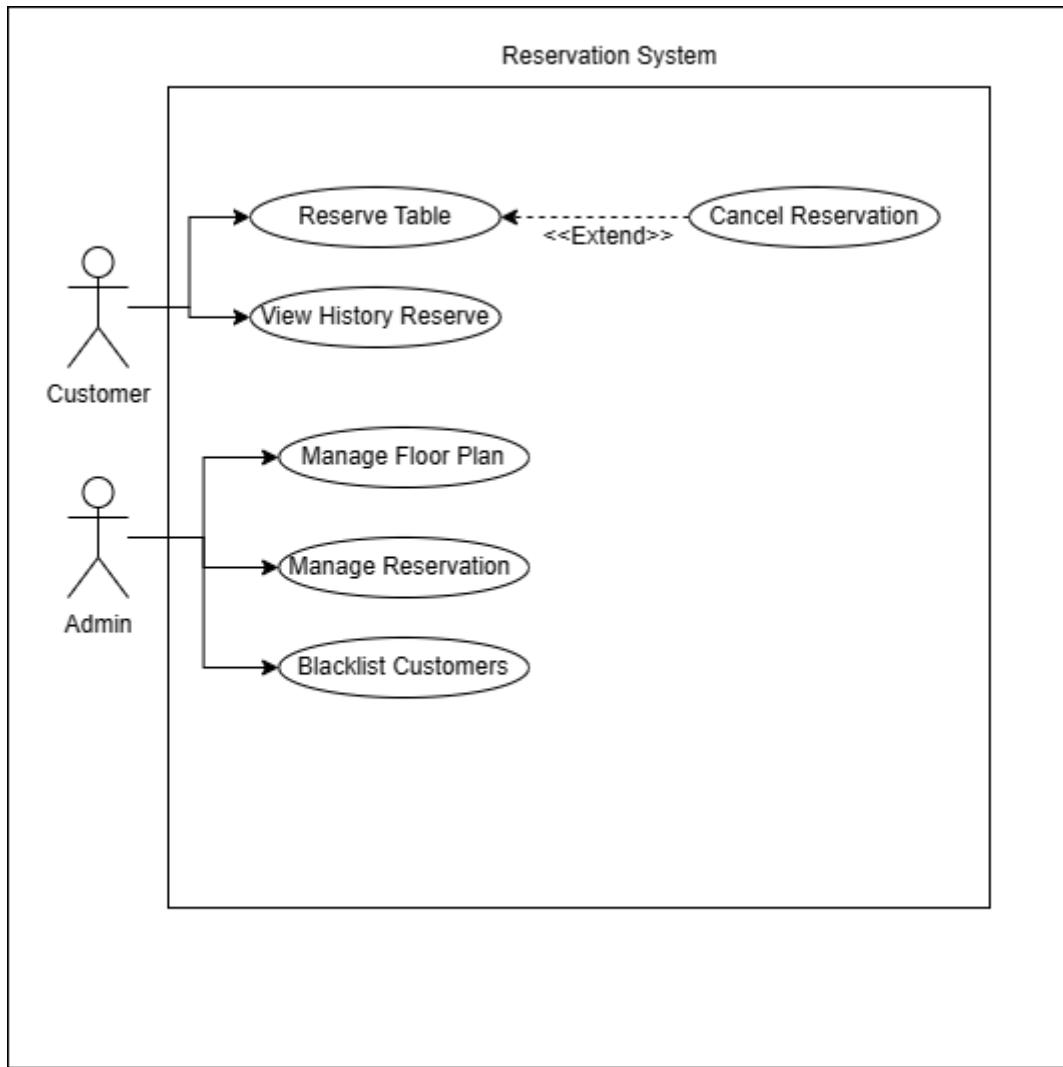
## Sprint 6



## Sprint 7



Use Case Diagram



Use Case Title: Reserve a Table

Use Case Id: 1

Actor: Customer

Goal: จองโต๊ะที่ร้านอาหาร

Preconditions: ลูกค้าเข้าสู่ระบบเรียบร้อยแล้ว

Main Success:

1. ระบบแสดงโต๊ะว่าง ณ เวลาตอนนี้
2. ลูกค้าเลือกวันที่และเวลา
3. ลูกค้าเลือกจำนวนคน

4. ระบบแสดงตัวว่าง กรองตามวันที่, เวลาและจำนวนคนที่เลือก
5. ลูกค้าเลือกตัว
6. ลูกค้าไปที่หน้ายืนยันการจอง
7. ลูกค้ายืนยันการจอง

Extensions (A) :

6A. ลูกค้ายกเลิกการจอง

7A. ลูกค้ากดปุ่มไปที่หน้าจองตัว

Postconditions: การจองของลูกค้าขึ้นไปอยู่ในหน้าประวัติการจอง

Use Case Title: View Reservation History

Use Case ID: 2

Actor: Customer

Goal: ตรวจสอบประวัติการจอง

Preconditions: ลูกค้าเข้าสู่ระบบแล้วกดไปที่หน้า MyProfile

Main Success:

1. ลูกค้าไปกดไปที่ส่วนประวัติการจอง (History Booking)
2. ระบบแสดงรายการการจองที่ผ่านมา

Extensions (A) :

2A. ลูกค้ากดไปที่ปุ่ม Cancel Reservation (ยกเลิกการจอง)

3A. ลูกค้าเลือกการจองทั้งหมดที่ต้องการยกเลิก (เลือกได้เฉพาะการจองที่มีสถานะเป็น In Progress)

4A. ลูกค้ากดไปที่ปุ่ม Confirm Cancel (ยืนยันการยกเลิก)

Postconditions: ลูกค้าดูประวัติการจองทั้งหมดของตัวเอง หากทำ Extention A สถานะการจองที่ถูกยกเลือกจะเปลี่ยนเป็น Cancelled

Use Case Title: Edit User Information

Use Case Id: 3

Actor: Customer

Goal: แก้ไขข้อมูล, อีเมลและรหัสผ่านของผู้ใช้

Preconditions: แอดมินเข้าสู่ระบบแล้วกดไปที่หน้า Admin

Main Success:

1. ลูกค้าไปกดไปที่ส่วนแก้ไขข้อมูลตนเอง (Setting Profile)
2. ลูกค้ากรอกข้อมูลอีเมลใหม่ที่ต้องการลงในฟอร์ม
3. ลูกค้ากดที่ปุ่ม Continue

Extensions (A) :

3A. ลูกค้าใช้อีเมลใหม่ที่ซ้ำกับคนอื่นที่อยู่ในระบบ

4A. ขึ้นข้อความเตือนว่าอีเมลนั้นซ้ำกับของคนอื่นและให้กรอกใหม่

Extensions (B) :

2B. ลูกค้ากดไปที่ Change my password (เปลี่ยนรหัสผ่านใหม่)

3B. ลูกค้ากรอกรหัสผ่านใหม่ที่ต้องการลงในฟอร์ม

4B. ลูกค้ากดที่ปุ่ม Continue

Postconditions: ข้อมูลและรหัสผ่านของผู้ใช้ถูกอัปเดตเป็นอันใหม่

Use Case Title: Change Reservation Status

Use Case ID: 4

Actor: Admin

Goal: แก้ไขสถานะของการจอง

Preconditions: แอดมินเข้าสู่ระบบแล้วกดไปที่หน้า Admin

Main Success:

1. แอดมินกดไปที่สถานะ(status)ของการจองที่ต้องการจะเปลี่ยน
2. ระบบแสดง dropdown ของสถานะต่างๆที่เปลี่ยนได้ (In Progress, Completed, Cancelled)
3. แอดมินกดเลือกสถานะใหม่
4. ระบบอัพเดตสถานะการจอง

Postconditions: สถานะการจองที่ถูกเลือกถูกอัปเดต

Use Case Title: Delete Reservation

Use Case ID: 5

Actor: Admin

Goal: ลบการจองออกจากระบบ

Preconditions: แอดมินเข้าสู่ระบบแล้วกดไปที่หน้า Admin

Main Success:

1. แอดมินกดไปที่ส่วนประวัติการจองทั้งหมด (All Reservations)
2. แอดมินกดไปที่ปุ่ม Delete Reservation (ลบการจอง)
3. แอดมินเลือกรายการจองทั้งหมดที่ต้องการยกเลิก
4. แอดมินกดไปที่ปุ่ม Confirm Delete (ยืนยันการลบ)

Postconditions: การจองห้องน้ำที่ถูกเลือกจะถูกลบ

Use Case Title: Login

Use Case ID: 6

Actors: Customer, Admin

Goal: สำหรับการเข้าสู่ระบบ

Preconditions: ผู้ใช้อยู่ที่หน้า Log In

Main Success:

1. ผู้ใช้กรอกข้อมูลอีเมลและรหัสผ่านลงในฟอร์ม
2. ผู้ใช้กดปุ่ม Continue

Extensions (A) :

2A. ลูกค้ากรอกอีเมลหรือรหัสผ่านผิดหรือผู้ใช้ถูกแบน

3A. ข้อความเตือนว่าเกิดข้อผิดพลาดในการล็อกอินและให้กรอกข้อมูลใหม่

Postconditions: ผู้ใช้เข้าสู่ระบบสำเร็จและเข้าสู่หน้าจอໂตัว

Use Case Title: Register

Use Case ID: 7

Actors: Customer

Goal: สำหรับการลงทะเบียนเข้าสู่ระบบ

Preconditions: ผู้ใช้อยู่ที่หน้า Register

Main Success:

1. ผู้ใช้กรอกข้อมูลชื่อ, อีเมลและรหัสผ่านลงในฟอร์ม
2. ผู้ใช้กดปุ่ม Continue

Extensions (A) :

2A. ลูกค้ากรอกอีเมลเข้ากับคนอื่นที่อยู่ในระบบอยู่แล้ว

3A. ขึ้นข้อความเมื่อตนว่าเกิดข้อผิดพลาดในการลงทะเบียนและให้กรอกข้อมูลใหม่

Postconditions: ผู้ใช้ลงทะเบียนเข้าสู่ระบบสำเร็จและเข้าสู่หน้า Log In

Use Case Title: Admin Edit Table Map

Use Case ID: 8

Actor: Admin

Goal: แอดมินแก้ไขแผนผังการจัดวางโต๊ะของร้าน

Preconditions: แอดมินเข้าสู่ระบบแล้วกดไปที่หน้า Admin

Main Success:

1. แอดมินกดไปที่ส่วนการจัดการแผนผังร้าน (Edit Table Map)
2. แอดมินเลือกขนาดโต๊ะที่ต้องการจะให้มีในผังร้าน
3. แอดมินกดไปที่ตำแหน่งของแผนที่ที่ต้องการให้โต๊ะโผล่มา ณ จุดจุดนั้น
4. แอดมินกดไปที่เพื่อสร้างมาและกรอกข้อมูลต่างๆของโต๊ะและกด Save
5. ทำขั้นตอนที่ 2 ถึง 4 ไปเรื่อยๆจนพอใจกันผังร้านที่ต้องการ
6. แอดมินกดไปที่ปุ่ม Save Objects เพื่อบันทึกแผนผังร้านใหม่ลงบน database

Extensions (A) :

6A. แอดมินกรอกข้อมูลของโต๊ะไม่ครบถ้วนโดย

7A. ขั้นตอนความเดือนວ่าข้อมูลของตัวนั้นยังคงกรอกไม่ครบทุกตัว

Extensions (B) :

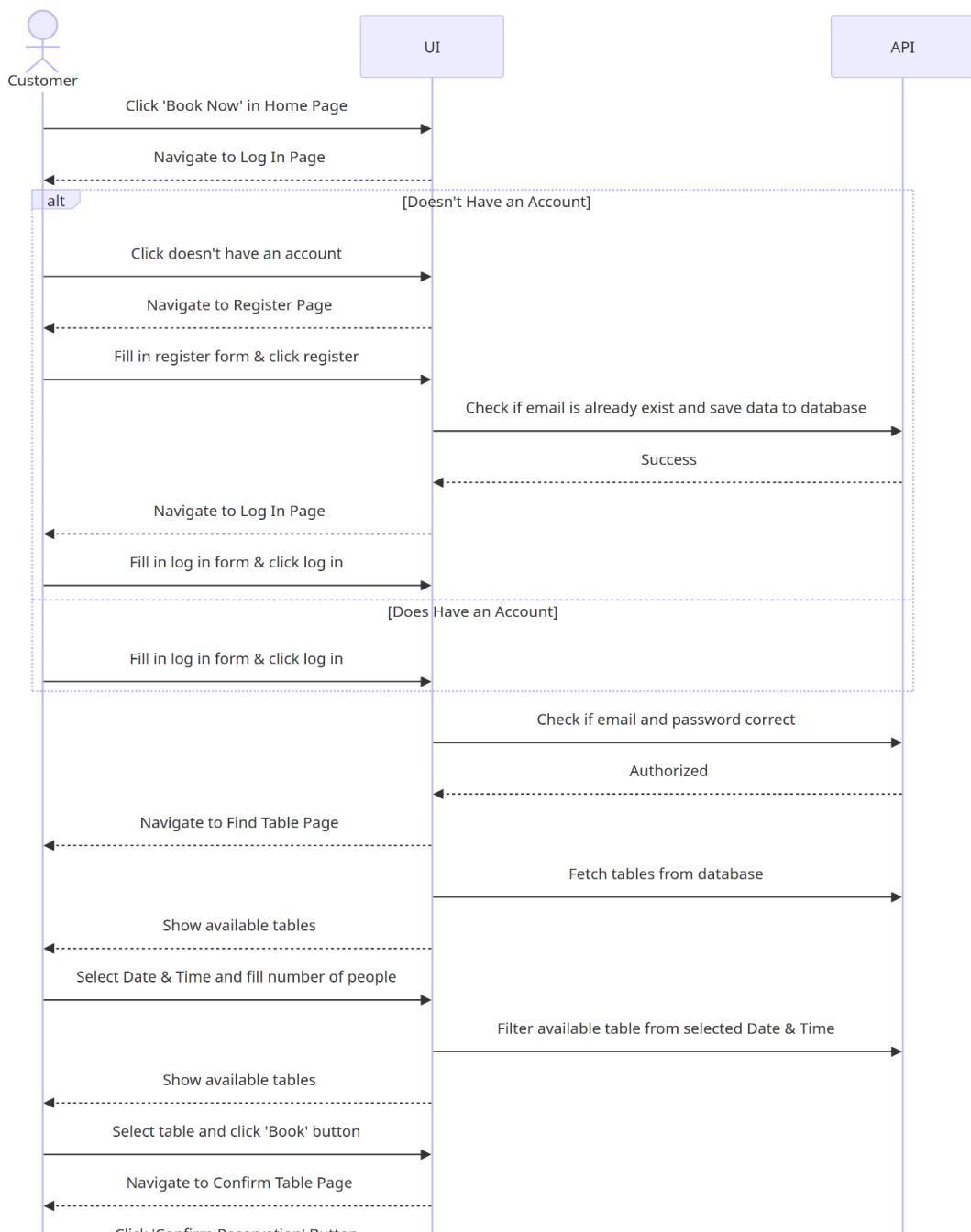
3B. ตำแหน่งของตัวที่ต้องการสร้างนั้นทับกับตัวที่มีอยู่แล้ว

4B. ขั้นตอนความเดือนว่าไม่สามารถสร้างตัวที่ทับกันได้

Postconditions: แผนผังร้านใหม่ถูกอัปเดตลงบนระบบ

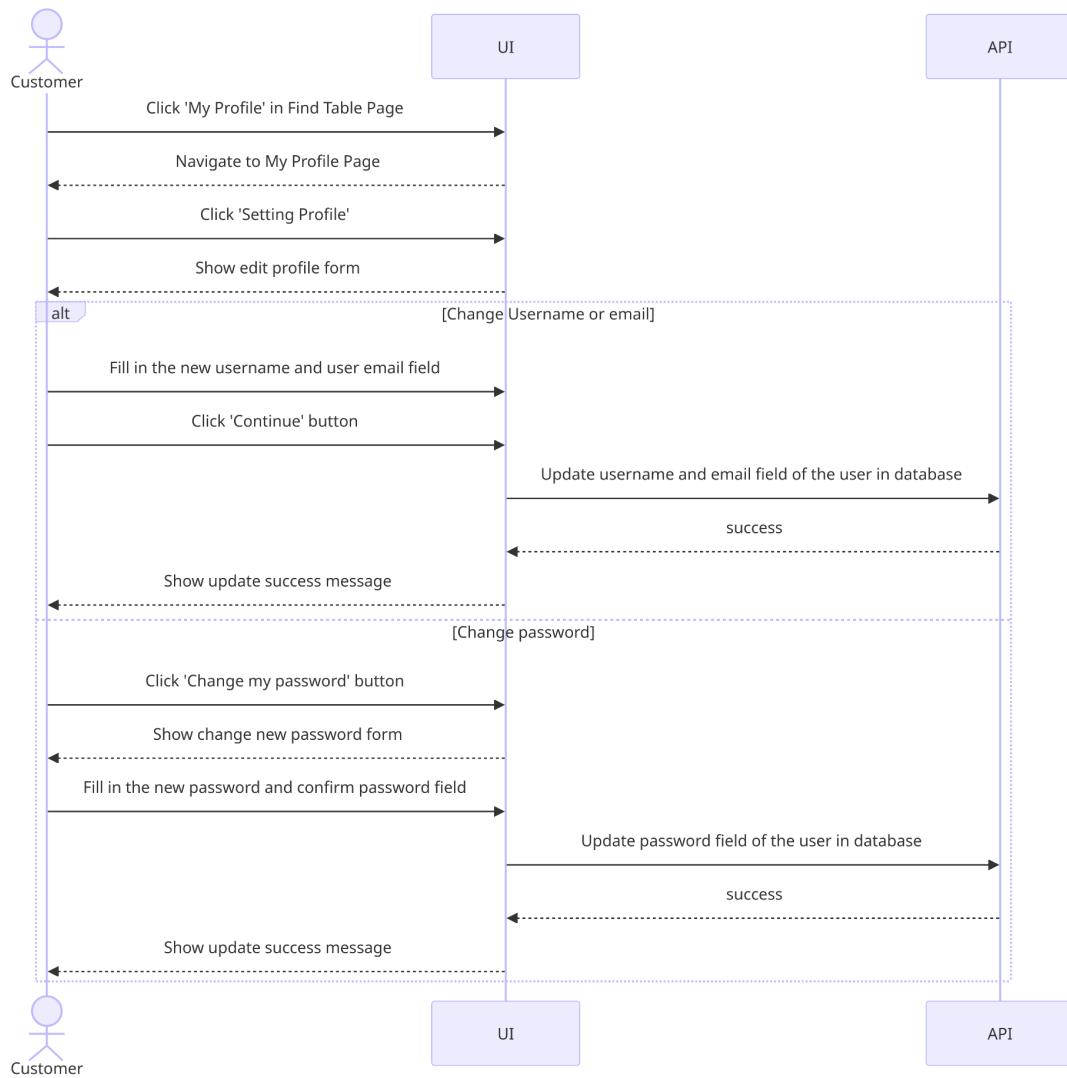
### Sequence Diagram

#### - Customer (Book a table)



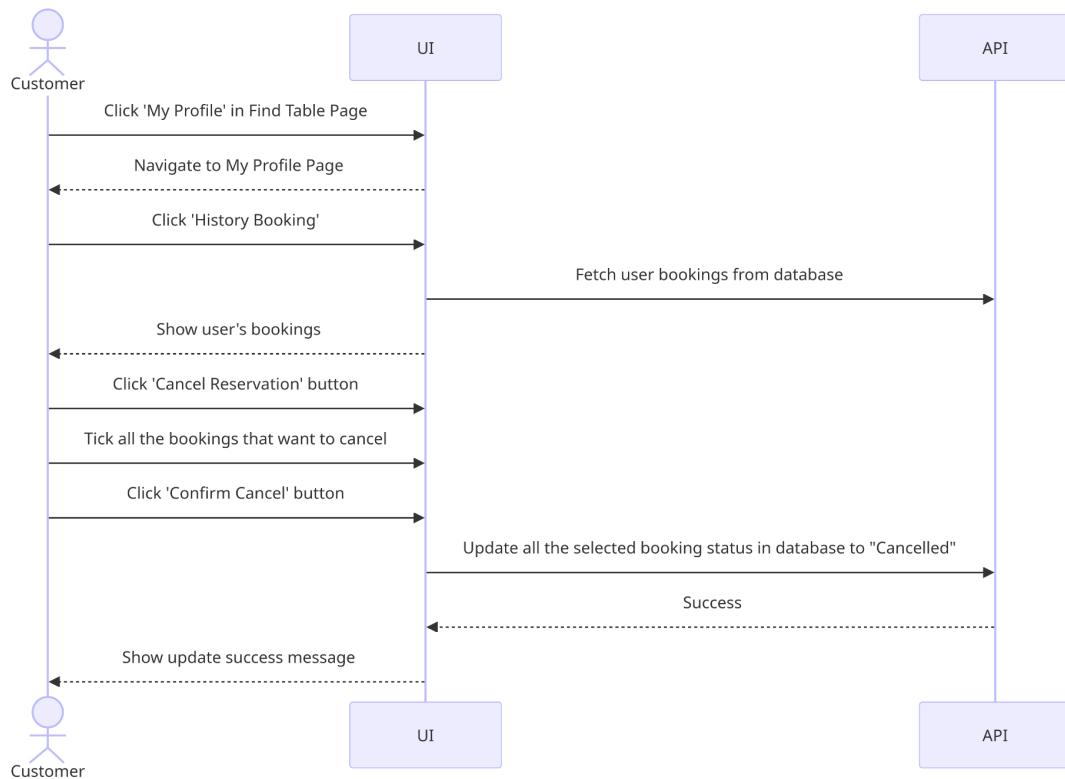
- Customer (Edit Profile)

\*\*Assume that user is already logged in and is in Find Table Page



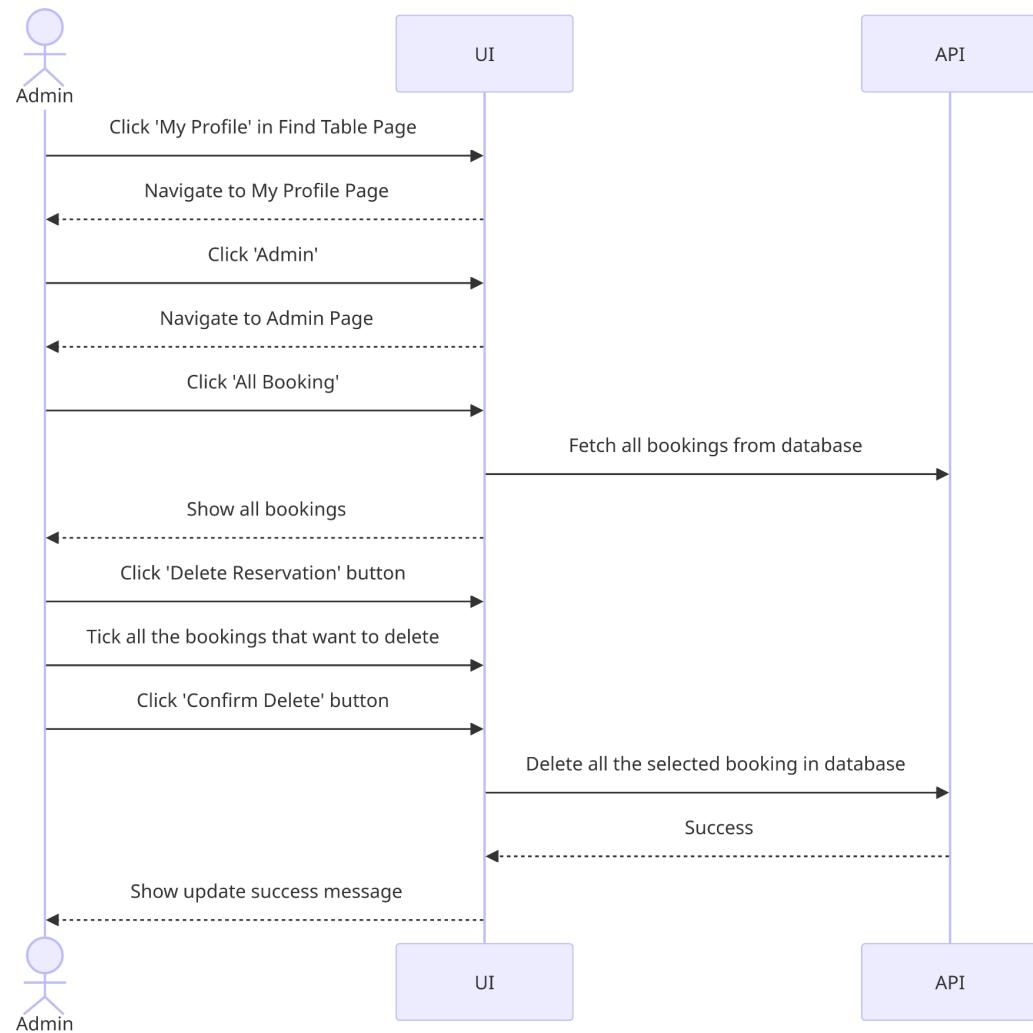
- Customer (Cancel reservation)

**\*\*Assume that user is already logged in and is in Find Table Page**



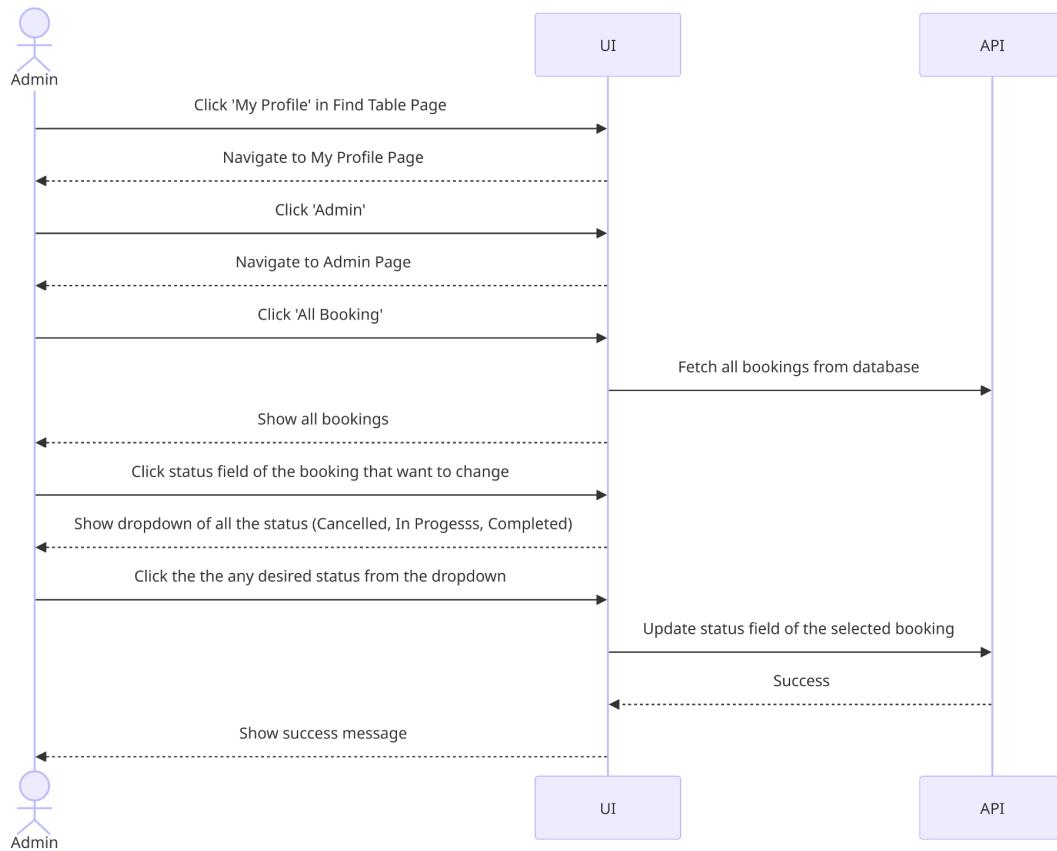
- Admin (Delete Bookings)

\*\*Assume that admin is already logged in and is in Find Table Page



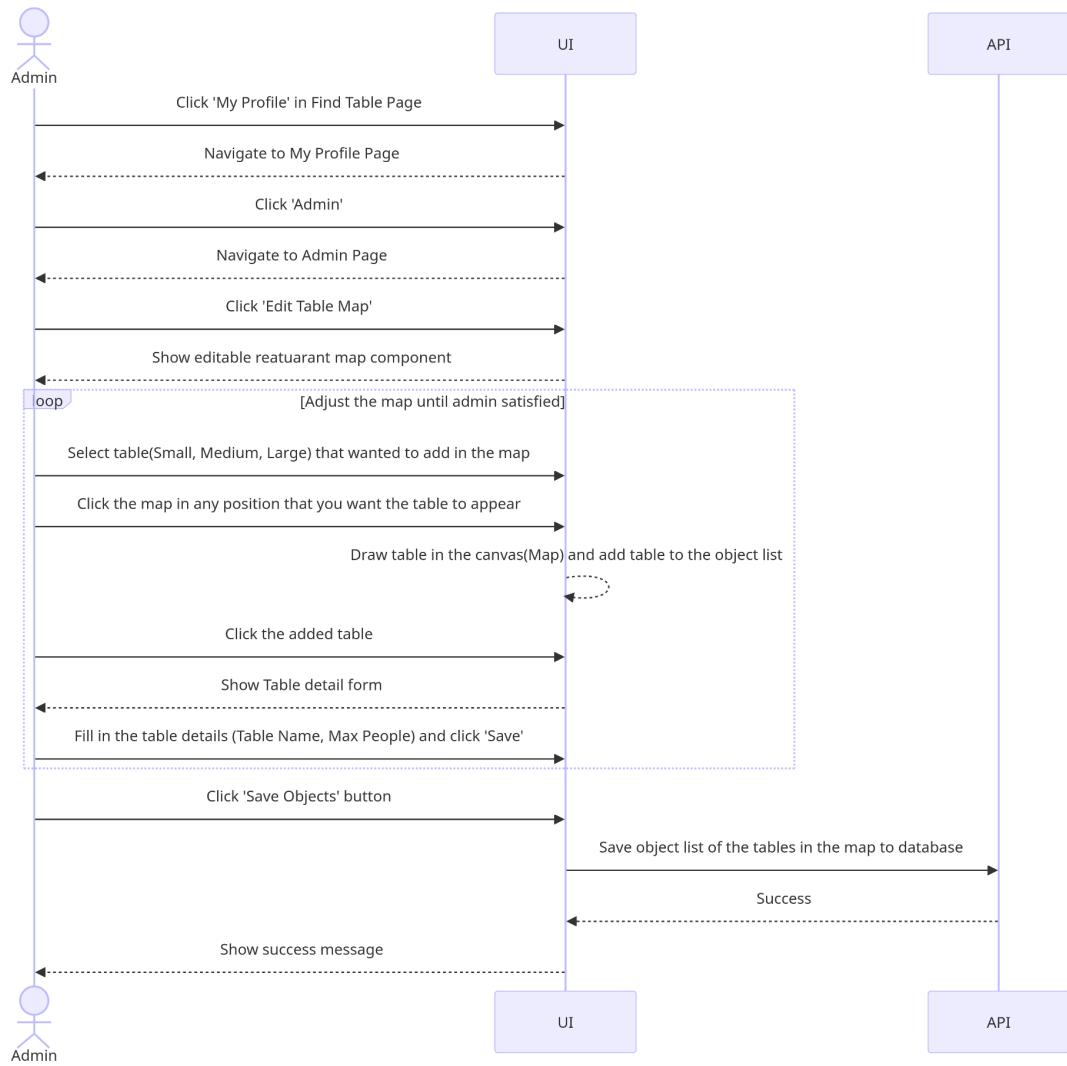
- Admin (Change Booking Status)

**\*\*Assume that admin is already logged in and is in Find Table Page**

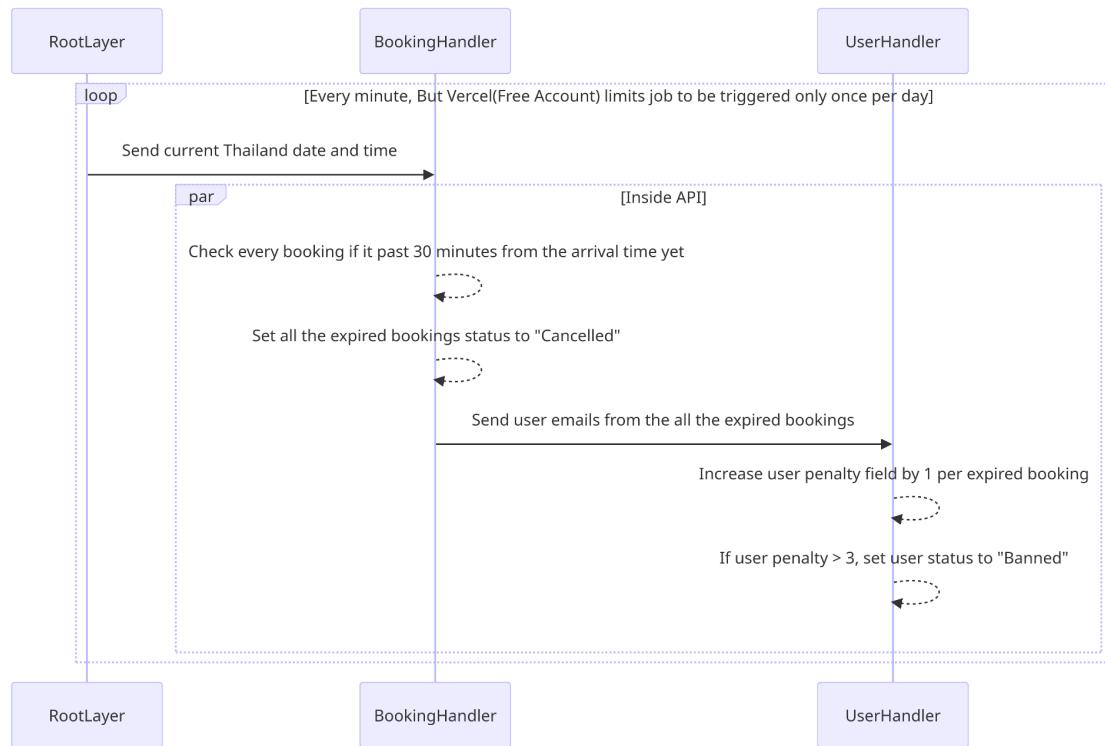


- Admin (Edit Table Map)

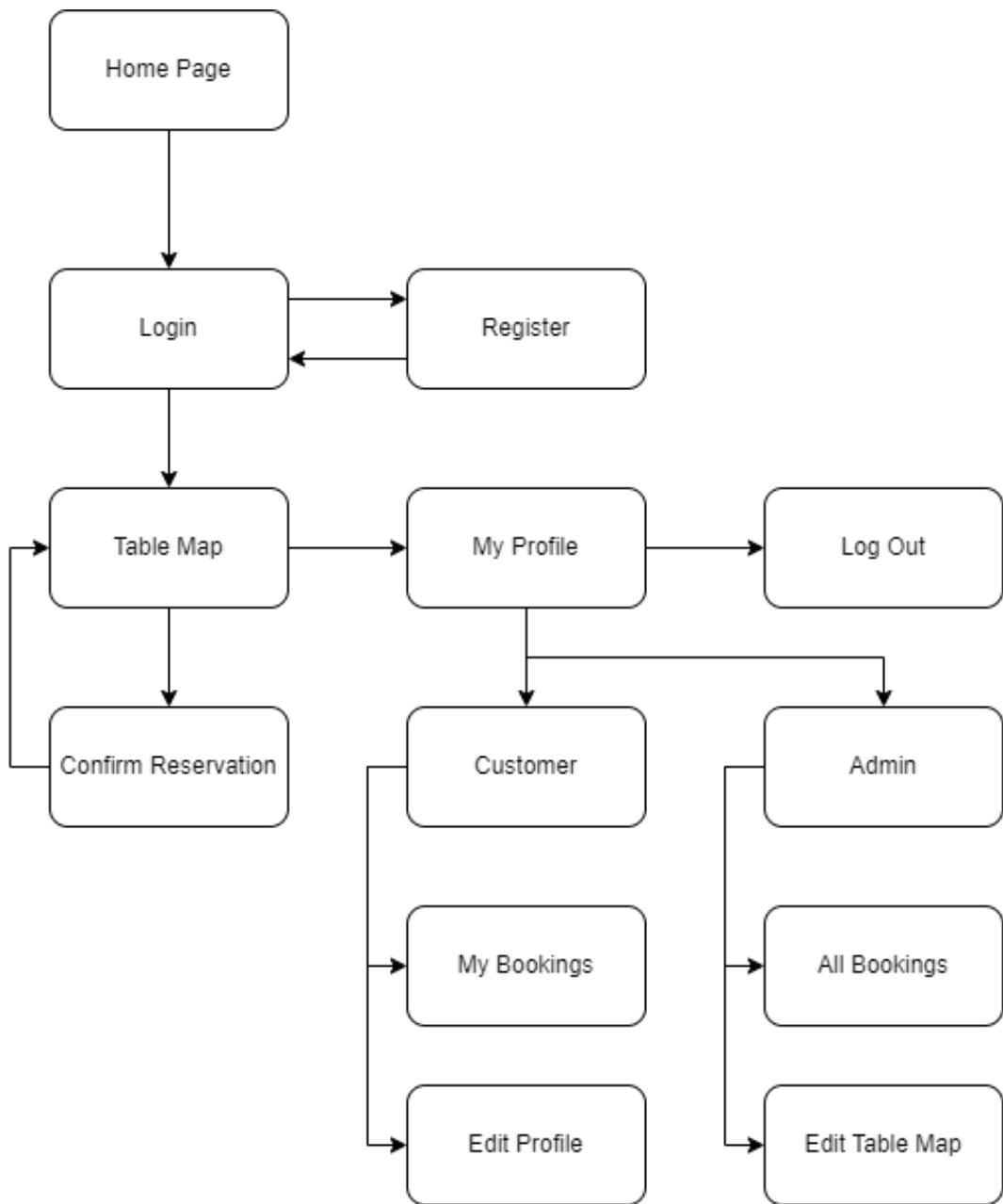
\*\*Assume that admin is already logged in and is in Find Table Page



- Cron Job (Check Expire Booking)



Navigation Map



## Testing

### Application Programming Interface

Priority	Inputs	Method	Expected	Note	Result
1	ทดสอบเรียก API เพื่อตีงโต๊ะออก มาแสดง	GET	แสดงแผนผังการจัดวาง โต๊ะของร้านมาใน canvas		Pass
1	ทดสอบส่ง API เพื่ออัปเดต แผนผังของร้าน (Admin)	POST	แผนผังของร้านจะถูก อัปเดตใน database		Pass
2	ทดสอบเรียก API เพื่อกรองโต๊ะที่ อยู่ใกล้ไปแล้ว	GET	โต๊ะในผังร้านที่ถูกจ่อไป แล้วจะถูกเปลี่ยนเป็นสีเทา		Pass
2	ทดสอบ API เพื่อใช้เช็คเวลา ของลูกค้าที่เข้าร้าน ภายในเวลาไป 30 นาทีหรือยัง	POST	การจองที่เลยเวลาจะถูก เปลี่ยนสถานะเป็น Cancelled และผู้จองจะ โดนเพิ่ม Penalty ไป 1 และถ้า Penalty มากกว่า 3 จะถูกแบน	ปกติแล้ว API นี้จะถูก เรียกทุก 1 นาทีแต่ เมื่อจากการ Deployment นั้น Vercel จำกัดการส่ง API ถ้าทำให้สามารถส่ง ได้แค่ 1 ครั้งต่อวัน เนื่องจาก API ตัวนี้ (Hobby Plan)	Pass
3	ทดสอบเรียก API เพื่อดึงรายการ ห้องหมุดของผู้ใช้ออกมานะ	GET	แสดงรายการห้องทั้งหมดของ ผู้ใช้นั้นๆและรายละเอียด ต่างๆของการจองนั้นๆ		Pass
3	ทดสอบส่ง API เพื่อยกเลิกการ จอง	PUT	สถานะการจองจะถูก เปลี่ยนเป็น Cancelled		Pass
3	ทดสอบส่ง API เพื่อเปลี่ยนชื่อ และอีเมลของผู้ใช้	PUT	ชื่อและอีเมลของผู้ใช้จะ ถูกเปลี่ยน		Pass
3	ทดสอบส่ง API เพื่อเปลี่ยนรหัส ผ่านของผู้ใช้	PUT	รหัสของผู้ใช้จะถูกเปลี่ยน		Pass
3	ทดสอบเรียก API เพื่อดึงรายการ ห้องหมุดออกมานะ (Admin)	GET	แสดงรายการห้องทั้งหมดและ รายละเอียดต่างๆของการ จองนั้นๆ		Pass
3	ทดสอบส่ง API เพื่อเปลี่ยน สถานะการจอง (Admin)	PUT	สถานะการจองจะถูก เปลี่ยนเป็น สถานะที่ เลือก (In Progress, Completed, Cancelled)		Pass

3	ทดสอบส่ง API เพื่อลบการจอง (Admin)	POST	การจองที่ถูกเลือกจะถูกลบออกจาก database		Pass
4	ทดสอบส่ง API เพื่อ register ผู้ใช้เข้าสู่ระบบ	POST	ผู้ใช้ที่ register จะสามารถ log in เข้าสู่ระบบเพื่อไปจองโต๊ะได้		Pass
4	ทดสอบการเรียกใช้ฟังก์ชัน signIn() ของ NextAuthJs	-	ผู้ใช้เข้าไปสู่หน้า Find Table และสามารถจองโต๊ะได้		Pass
4	ทดสอบการเรียกใช้ฟังก์ชัน signOut() ของ NextAuthJs	-	ผู้ใช้กลับไปสู่หน้า log in และไม่สามารถจองโต๊ะได้		Pass
4	ทดสอบการเรียกใช้ฟังก์ชัน useSession() ของ NextAuthJs	-	หาก session ไม่เท่ากับ NULL จะสามารถจองโต๊ะได้		Pass

#### User Interface (Customer)

Priority	Inputs	Expected	Note	Result
1	ทดสอบเปิดหน้า WebApplication	เข้าหน้า Homepage		Pass
1	ทดสอบกดไอคอน Book now	เข้าหน้า Login		Pass
1	ทดสอบกด Click here	เข้าหน้า Registration		Pass
1	ทดสอบกด Reservation , Promotion , Contact	เข้าหน้า Booking Table		Pass
2	ทดสอบระบบ Login กรอก Email Address , Password	เข้าหน้า Booking Table		Pass
2	ทดสอบระบบ Registration กรอก Name , Email Address , Password	เข้าหน้า Login		Pass
2	ทดสอบระบบ Booking กรอก วันที่ - เวลา , จำนวนคน , เลือกโต๊ะ 1 โต๊ะ	เข้าหน้า Confirm Table		Pass
2	ทดสอบเข้าหน้า Confirm Table	แสดงรายละเอียด Booking Table , ชื่อโต๊ะ		Pass
2	ทดสอบกด Confirm booking	เข้าหน้า Booking Table ขึ้นสถานะบนเว็บไซต์ Confirm booking		Pass

2	ทดสอบระบบ Booking เพิ่มจำนวนคน เกินจำนวนที่นั่งได้	ไม่สามารถเลือกโต๊ะได้ ชั้น สถานะบนเว็บไซต์ The selected table is not suitable for the provided number of people		Pass
2	ทดสอบเปลี่ยนจำนวนคน	โต๊ะแสดงสถานะเป็นสี สีฟ้า สามารถจองได้และสีเทาไม่สามารถจองได้		Pass
2	ทดสอบเปลี่ยนวันที่ - เวลา	Reset วันที่ด้วยการกด Today จะแสดงวันที่ปัจจุบัน		Pass
2	ทดสอบระบบ Booking	ไม่สามารถจองโต๊ะชั้นลูกค้าคนอื่นได้ภายในวันนั้น ถ้า Admin ไม่เปลี่ยนสถานะของโต๊ะ		Pass
3	ทดสอบกด User name ขวางเว็บไซต์	แสดง My Profile , Logout		Pass
3	ทดสอบกด Logout	เข้าหน้า Login		Pass
3	ทดสอบกด My Profile	เข้าหน้า User Dashbord History		Pass
4	ทดสอบเข้าหน้า User Dashbord History	แสดงรายละเอียด Booking history		Pass
4	ทดสอบระบบ Cancel reservation	กดเลือกการจองที่ต้องการยกเลิกเฉพาะ Satatus inprogress การจองหายไป จาก Booking history		Pass
4	ทดสอบเข้าหน้า Setting Progile	แสดงรายละเอียด Edit you profile		Pass
4	ทดสอบระบบ Edit you profile	เปลี่ยน Name , Email บนหน้า Profile		Pass
4	ทดสอบกด change my password	เข้าหน้า Change Password		Pass
4	ทดสอบระบบ Change Password กรอก New Password, Confirm Password	เว็บไซต์แสดงสถานะ Password changed successfully		Pass

4	ทดสอบระบบ Change Password เข้าหน้า Login	เข้าหน้า Homepage		Pass
4	ทดสอบกดไอคอน TOAA NIUNG	เข้าหน้า Booking Table (Logged in)		Pass

User Interface (admin)

Admin สามารถเข้าใช้งานได้แบบ Customer เพิ่มเติมดังนี้

Priority	Inputs	Expected	Note	Result
5	ทดสอบเข้าหน้า Admin	แสดงรายละเอียดหน้า Admin		Pass
5	ทดสอบกด All Booking	แสดงรายละเอียดแต่ละ Booking ทั้งหมด		Pass
5	ทดสอบระบบ Delete reservation	กดเลือก Booking ที่ต้องการลบ		Pass
5	ทดสอบระบบ Delete reservation กด Confirm	Booking ที่เลือกถูกลบ		Pass
5	ทดสอบระบบ Search	ค้นหาด้วย Keywords userEmail , tableName , tableSize , numberOfRows , status , arrivialTime		Pass
5	ทดสอบระบบ Search ค้นหาด้วย Keywords	แสดง Booking ตาม Keywords		Pass
5	ทดสอบระบบ status	เปลี่ยน status ตามสถานะ inprogress , Completed , Cancelled		Pass
5	ทดสอบระบบเปลี่ยน status	สถานะ status เป็นยังไง ผู้ Customer , Admin		Pass
5	ทดสอบกด Edit Table Map	เข้าหน้า Edit Table Map		Pass
5	ทดสอบกด Edit Table Map	นำตัวแอล์ฟขนาดของบนพื้นที่ Save Object และ สถานะ Objects saved		Pass

		successfully		
5	ทดสอบกด Edit Table Map นำตัว แต่ละขนาดวางบนพื้นที่	โดยแต่ละขนาดสามารถวาง ทีละ 1 โถะ เดือดเปลี่ยน ขนาดได้ที่ไอคอนเพื่อวาง โดยขนาดอื่น		Pass
5	ทดสอบกด Edit Table Map กด Save Object	ระบบ saved และแสดงผู้ Customer		Pass
5	ทดสอบกด Clear All	โดยทั้งหมดถูกลบ		Pass
5	ทดสอบกด Edit Table Map ไม่กด Save Objects	โดยที่ถูกจัดไม่เปลี่ยนแปลง และไม่แสดงผู้ Customer		Pass
5	ทดสอบกด Edit Table Map กดที่ตัว Clear All ไม่กด Save Objects	โดยที่ถูกจัดไม่เปลี่ยนแปลง และไม่แสดงผู้ Customer		Pass
5	ทดสอบกด Edit Table Map กดที่ตัว	แสดงให้กำหนดรายละเอียด Table Name , Max People กด Save เพื่อ Save		Pass
5	ทดสอบกด Edit Table Map กดที่ตัว	กด Remove Object เพื่อ ลบตัว		Pass
5	ทดสอบกด Edit Table Map กดที่ตัว เพื่อเคลื่อนย้ายตำแหน่ง	โดยสามารถเคลื่อนย้าย		Pass
5	ทดสอบว่างานตัว กด Save Objects	โดยใหม่ไม่ถูก Save		Pass
5	ทดสอบว่างานตัว กด Save Objects	โดยใหม่ถูก Save		Pass
5	ทดสอบกด Back To Profile	เข้าหน้า Booking history (Customer)		Pass
6	ทดสอบกดไอคอน TOAA NIUNG	เข้าหน้า Booking Table (Logged in)		Pass

## Component-level Design

**A**

TOAA-NIUNG

Home | Promotion | Contact | Login

หน้าหลัก (Homepage) :

Recommended promotions for you!!

Recommended products for today

Kurobuta Crispy Pork

Great value promotion!  
Get 10% off din sum for when you collect  
expenses of 200 baht or more at the end of your  
receipt.

When ordering more than 4 food items  
10 Jun 2024

Book now

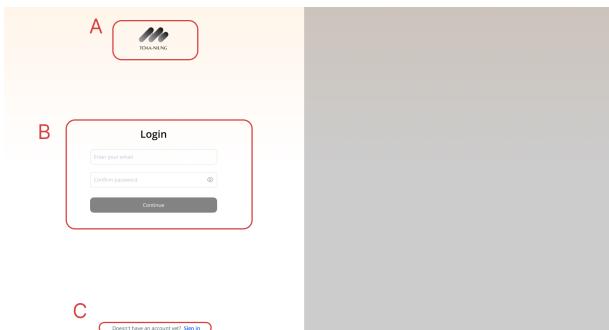
Jay Fai's Dim Sum

Great value promotion!  
Get 10% off din sum for when you collect  
expenses of 200 baht or more at the end of your  
receipt.

When ordering more than 4 food items  
10 Sep 2024

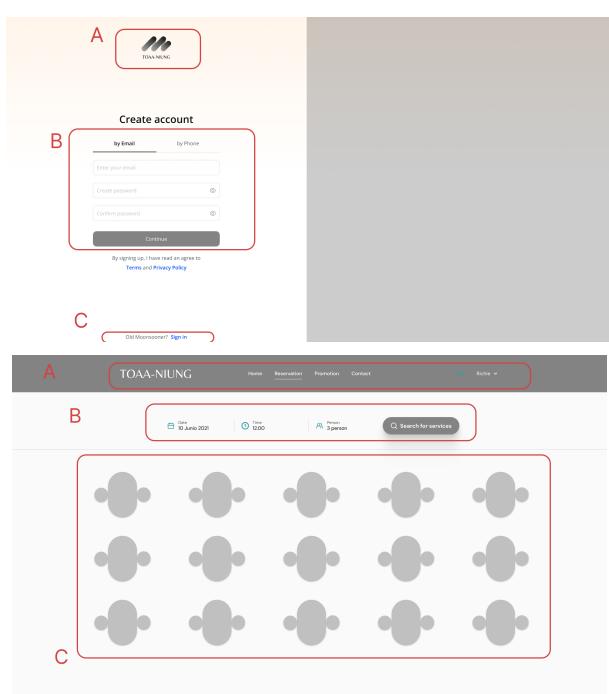
Book now

- A. Header: ประกอบด้วยโลโก้ของร้านอาหาร เมนูหลัก และปุ่มสำหรับเข้าสู่หน้าต่างๆ เช่น หน้าเมนู หน้าโปรโมชั่น หน้าติดต่อ เข้าสู่ระบบ
- B. Main content: เป็นส่วนหลักของหน้าเว็บไซต์ ประกอบด้วยเนื้อหาเกี่ยวกับร้านอาหาร เช่น ข้อมูลเกี่ยวกับร้านอาหาร เมนูอาหาร โปรโมชั่น
- C. Footer: ประกอบด้วยข้อมูลติดต่อของร้านอาหาร เช่น ที่อยู่ เบอร์โทรศัพท์ และอีเมล



### หน้าเข้าสู่ระบบ (Login page) :

- A. Header: ประกอบด้วยโลโก้ของเว็บไซต์และปุ่มสำหรับเข้าสู่หน้าหลัก
- B. 表单สำหรับกรอกข้อมูลเข้าสู่ระบบ
- C. ลิงค์สำหรับไปยังหน้าสร้างบัญชีใหม่



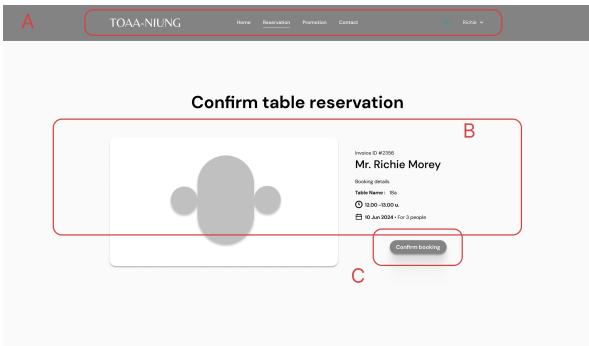
### สร้างบัญชี (Create account)

- A. Header: ประกอบด้วยโลโก้ของเว็บไซต์และปุ่มสำหรับกลับสู่หน้าหลัก
- B. Main content: ประกอบด้วยฟอร์มสำหรับกรอกข้อมูลสร้างบัญชี
- C. Footer: ประกอบด้วยลิงค์สำหรับไปยังหน้าเข้าสู่ระบบ

### หน้าค้นหาและจองตั๋ว (Search page) :

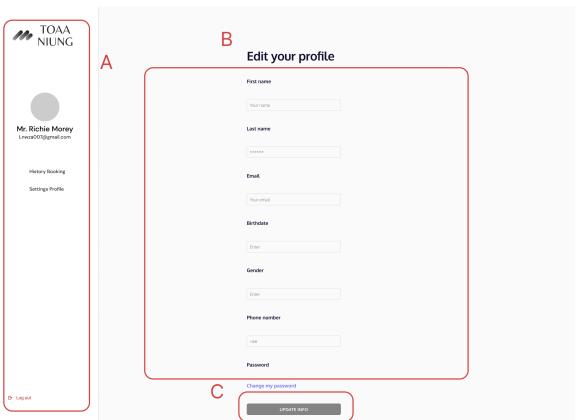
- A. Navbar : ประกอบด้วยโลโก้ของเว็บไซต์ ปุ่มสำหรับกลับสู่หน้าหลักและปุ่มสำหรับเข้าชมหน้าแสดงผลของลูกค้า
- B. ส่วนกล่องค้นหา: สำหรับทำการค้นหาตั๋วที่ว่าง โดย มีช่องสำหรับให้ระบุ วันและเวลาที่ต้องการ และจำนวนผู้ที่จะมาใช้บริการ
- C. ส่วนแสดงผลการค้นหา : สามารถกรุ๊ปตั๋วเลือกที่นั่งที่ต้องการ

จองได้เลย



#### หน้ายืนยันการจองโต๊ะ (Table reservation) :

- A. Header: ประกอบด้วยโลโก้อร่อยวันอาหาร เมนูหลัก และปุ่มสำหรับเข้าสู่หน้าต่างๆ เช่น หน้าเมนู หน้าโปรโมชั่น หน้าติดต่อ เข้าสู่ระบบ
- B. แสดงตัวอย่างข้อมูล ชื่อ วันที่ และเวลาที่ต้องการจองก่อนยืนยัน
- C. ปุ่มยืนยันการจอง



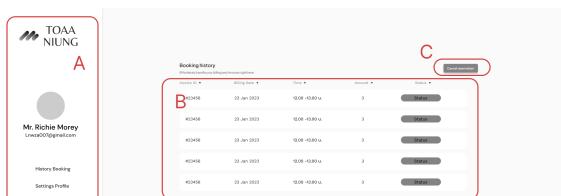
#### หน้าแก้ไขโปรไฟล์ (Edit profile) :

- A. Side Bar : มีโลโก้อร่อยเว็บไซต์เป็น ปุ่มสำหรับกลับสู่หน้าหลัก
  - ส่วนแสดงชื่อและอีเมลของลูกค้า
  - ແນບสำหรับเข้าสู่หน้าอื่นๆ ในแดรชบอร์ดของลูกค้า
  - ปุ่มออกจากระบบ
- B. 表单สำหรับแก้ไขข้อมูลโปรไฟล์ ปุ่มสำหรับเข้าสู่หน้าเปลี่ยนรหัสผ่าน
- C. ปุ่มสำหรับบันทึกการเปลี่ยนแปลง



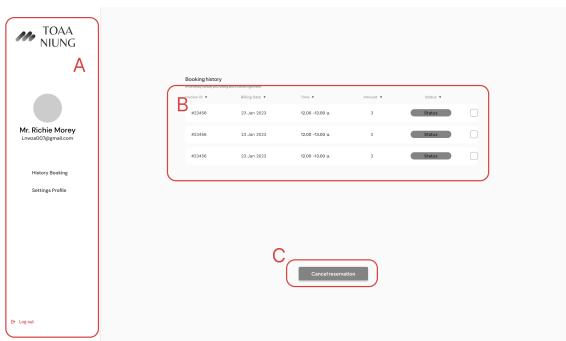
#### หน้าเปลี่ยนรหัสผ่าน (Change password) :

- A. Side Bar : มีโลโก้อร่อยเว็บไซต์เป็น ปุ่มสำหรับกลับสู่หน้าหลัก
  - ส่วนแสดงชื่อและอีเมลของลูกค้า
  - ແນບสำหรับเข้าสู่หน้าอื่นๆ ในแดรชบอร์ดของลูกค้า
  - ปุ่มออกจากระบบ
- B. 表单สำหรับเปลี่ยนรหัสผ่าน
- C. ปุ่มสำหรับบันทึกการเปลี่ยนแปลง



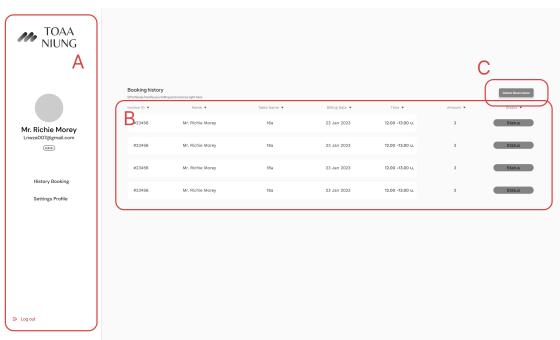
### หน้าประวัติการจอง (Booking History):

- A. Side Bar : มีโลโก้ของเว็บไซต์เป็น ปุ่มสำหรับกลับสู่หน้าหลัก
  - ส่วนแสดงชื่อและอีเมลของลูกค้า
  - แบบสำหรับเข้าสู่หน้าอื่นๆในแดรชบอร์ดของลูกค้า
  - ปุ่มออกจากระบบ
- B. ดูรายละเอียดการจองและ มีແຄบสถานะแสดงประวัติ
- C. ปุ่มสำหรับกดยกเลิกการจอง



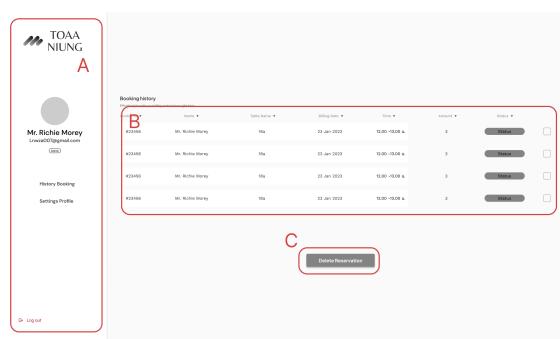
### หน้ายกเลิกประวัติการจอง (Cancel Booking History) :

- A. Side Bar : มีโลโก้ของเว็บไซต์เป็น ปุ่มสำหรับกลับสู่หน้าหลัก
  - ส่วนแสดงชื่อและอีเมลของลูกค้า
  - แบบสำหรับเข้าสู่หน้าอื่นๆในแดรชบอร์ดของลูกค้า
  - ปุ่มออกจากระบบ
- B. ดูรายละเอียดการจองและสามารถเลือกรายการที่ต้องการจะยกเลิกได้
- C. ปุ่มสำหรับกดยกเลิกการจอง



### หน้าประวัติการจอง-แอดมิน (Booking History) :

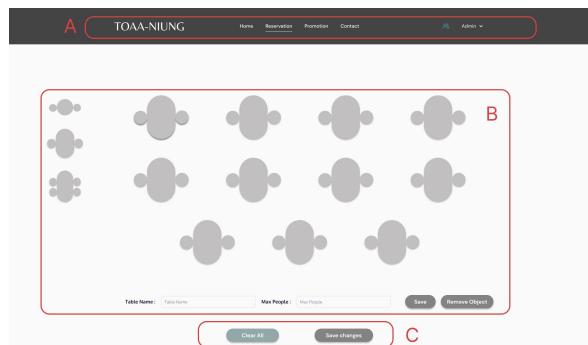
- A. Side Bar : มีโลโก้ของเว็บไซต์เป็น ปุ่มสำหรับกลับสู่หน้าหลัก
  - ส่วนแสดงชื่อและอีเมลของลูกค้า
  - แบบสำหรับเข้าสู่หน้าอื่นๆในแดรชบอร์ดของลูกค้า
  - ปุ่มออกจากระบบ
- B. ดูรายละเอียดการจองและ มีແຄบสถานะแสดงประวัติ
- C. ปุ่มสำหรับกดยกเลิกการจอง



### หน้ายกเลิกประวัติการจอง-แอดมิน (Cancel Booking History) :

- A. Side Bar : มีโลโก้ของเว็บไซต์เป็น ปุ่มสำหรับกลับสู่หน้าหลัก
  - ส่วนแสดงชื่อและอีเมลของแอดมิน

- ແກບສໍາຫຼັບເຂົ້າສູ່ໜ້າອື່ນໃນແດຮທບອົດຂອງແຄມິນ
- B. ປຸ່ມອອກຈາກຮບບ ດູຮາຍລະເຂີຍດາຈອງແລະສາມາດເລືອກຮາຍກາຣທີ່ຕ້ອງກາຈະຍົກເລີກໄດ້
- C. ປຸ່ມສໍາຫຼັບກົດຍົກເລີກດາຈອງ



#### หน้าจัดรูปแบบໂຕະ-ແອດມິນ():

- A. Header: ປະກອບດ້ວຍໂລກີຂອງຮ້ານອາຫາຣ ເມຸນຫລັກ ແລະປຸ່ມສໍາຫຼັບເຂົ້າ

ສູ່ໜ້າຕ່າງໆ ເປັນ ມ້ານາມ້າ ພ້າໂປຣໂນໜັ້ນ ພ້າຕິດຕ່ອ ເຂົ້າສູ່ຮະບບ

- B. ສ່ວນໜ້າຈໍາລອງກາຈັດໂຕະ

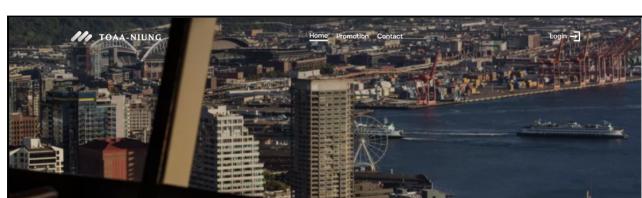
- ເລືອກຮູບແບບໜາດຂອງໂຕະທີ່ຕ້ອງກາຈັດ

- ສາມາດຄລາກໂຕະເພື່ອຈັດວາງຮູບແບບຕາມສຕານທີ່ຈະຈິງໄດ້

- ດັ່ງຊື່ໂຕະແລະກຳທັນດຳຈຳນວນຄົນຕາມໝາດຈະຈິງໄດ້

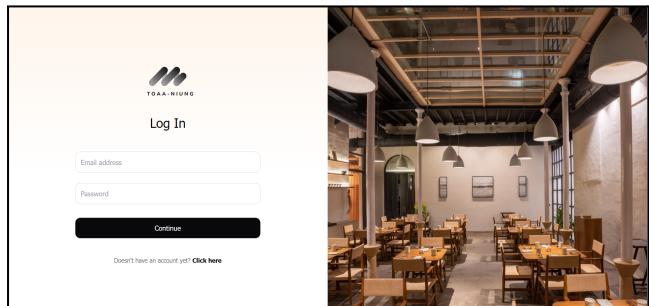
- C. ປຸ່ມບັນທຶກກາຮປັບປຸງປະລົງ

#### Final Component-level Design



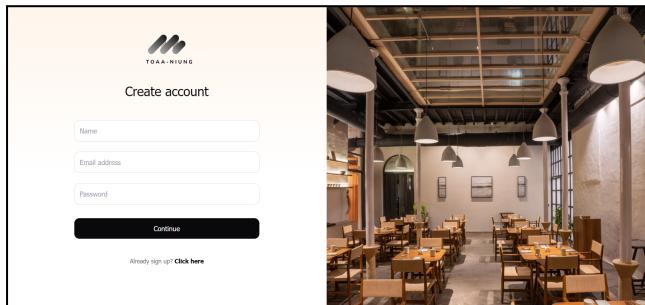
### หน้าหลัก (Homepage) :

- เป็นหน้าเริ่มต้นสำหรับไปยังหน้าต่างๆ และมี ส่วนโปรโมชั่นที่แสดงโปรโมชั่นต่างๆ ของร้านอาหาร และ ผลิตภัณฑ์ที่แนะนำ ส่วนสุดท้าย คือข้อมูลสำหรับติดต่อที่แสดงข้อมูลการติดต่อของร้านอาหาร ส่วนนี้ประกอบด้วยที่อยู่ หมายเลขโทรศัพท์ และอีเมลของร้านอาหาร



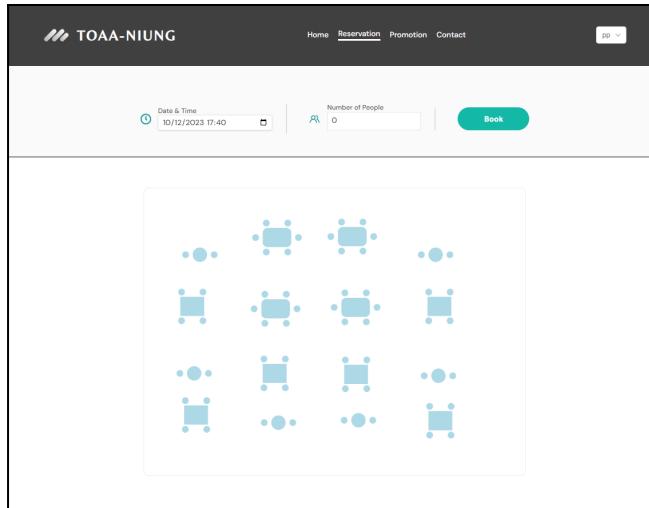
### หน้าเข้าสู่ระบบ (Login page) :

หน้า login เป็นหน้าเข้าใช้ website จะต้อง Register(ลงทะเบียน) ก่อนเข้าใช้



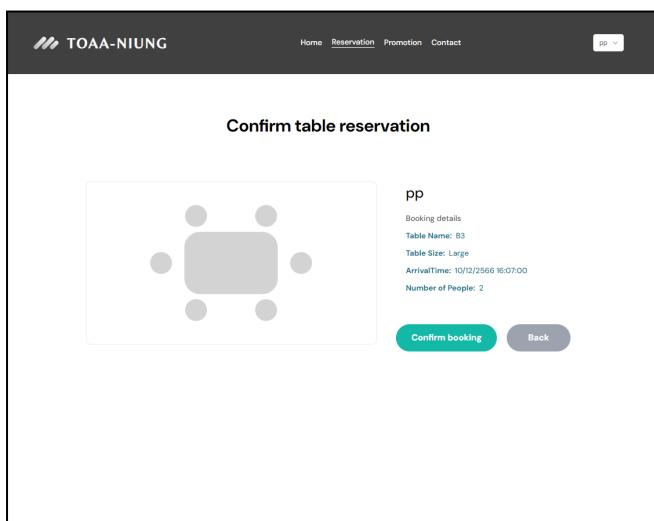
### สร้างบัญชี (Create account)

หน้า Register เป็นหน้าเข้าใช้ website สำหรับ User ที่ยังไม่มีบัญชี ก่อนเข้าใช้



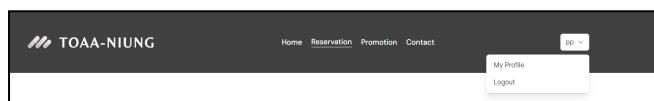
หน้าค้นหาและจองโต๊ะ (Search page) :

เมื่อทำการ login เข้ามา จะแสดงที่หน้า จอง โต๊ะ ทันที โดยต้องกรอก วัน ที่ เวลาที่จะจอง จำนวนคน คลิกโดยที่จะจอง แต่ละโต๊ะจำกัดจำนวนคน เมื่อใส่จำนวนคน เกินจำนวนสูงสุดของโต๊ะ โต๊ะจะเปลี่ยนสีเป็น "สีเทา" หลังจากกรอกข้อมูลแล้วเลือกโต๊ะแล้ว กด ปุ่ม Next เพื่อไปที่หน้ายืนยัน ได้ที่ที่จอง

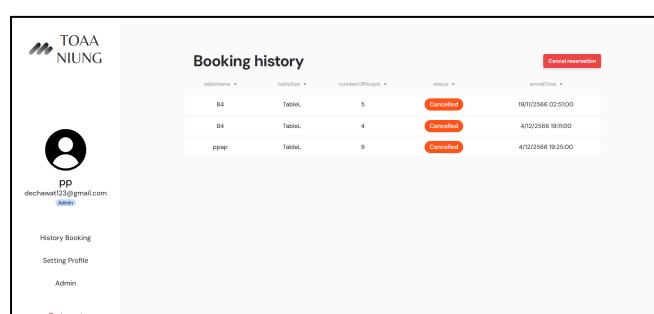


หน้ายืนยันการจองโต๊ะ (Table reservation) :

เป็นหน้าสำหรับให้ User กดเพื่อยืนยันการทำกิจกรรม หากต้องการยกเลิกสามารถกดปุ่ม Back เพื่อทำการจองใหม่ได้



Dropdown เพื่อ logout หรือ ไปยังหน้า profile



หน้าประวัติการจอง (Booking History):

หลังจากคลิกเข้า profile จะแสดง เป็นส่วน History booking ให้เห็นก่อน

Table name	Table ID	Number of People	Status	arrivalTime
B4	Table5	5	Cancelled	18/7/2566 02:00:00
B4	Table5	4	Cancelled	18/7/2566 10:00:00
pp00	Table5	9	Cancelled	18/7/2566 19:25:00

หน้ายกเลิกประวัติการจอง (Cancel Booking History) :

สามารถ cancel booking ที่เราอยาก ยกเลิกได้

ที่ setting profile เราสามารถแก้ไข name ของเรา และ Email รวมไปถึงการเปลี่ยนรหัสผ่าน

หน้าเปลี่ยนรหัสผ่าน (Change password) :

เมื่อเรา คลิกที่ change my password จะแสดงตั้งรูป ให้เราสามารถแก้ไข password ของเรา

ถ้าได้ role admin จะขึ้น admin ตรง list ดังรูป

หน้าประวัติการจอง-แอดมิน (Booking History) :

หลังจากที่คลิกเข้ามา admin และ list จะเปลี่ยนไปดังรูป จะแสดงส่วน All Booking ก่อน

เป็น booking ของทุก user ที่ทำการ booking เข้ามา สามารถแก้ status ของ booking ได้

สามารถแก้ status เป็นไปตามรูป

The screenshots show the 'Edit Table Map' feature in the TOAA NIUNG application. The interface includes a sidebar with user information (pp, dechawat23@gmail.com) and navigation links (All Booking, Edit Table Map, Back To Profile, Log out). The main area displays a grid of tables with their coordinates and status (Available or Booked). In the bottom screenshot, one specific table is highlighted in red, indicating it has been selected for reservation.

#### หน้าจัดรูปแบบโต๊ะ-แอดมิน():

เป็นส่วนของ Edit Table Map ทำเพื่อจัดผังร้านในแบบจำลอง สามารถเพิ่มเติมได้หลายโหมดได้

แล้วต้องตั้งชื่อและใส่จำนวนคนที่นั่งต่อหนึ่งให้ครบก่อนถึงจะ อัพเดตเป็นโหมดสำหรับหน้า Reservation ให้ user จองได้

## Self-Evaluation and Summary

### Self-Evaluation

ทีมผู้จัดทำมีความตั้งใจการทำโปรเจค TOAA NIUNG เมื่อเสร็จสิ้นโปรเจคตามที่ดำเนินการตาม Requirement และ Scrum process มีรายการที่มีปัญหาดังนี้

1. การวางแผนดำเนินงานแต่ละอาทิตย์ ไม่สามารถทำได้ตรง 100 เปอร์เซนต์ เนื่องจากช่วงการสอบที่เปลี่ยนเป็น 3 ช่วง ทำให้ 1 เดือน จะมีเวลาทำได้อย่างเต็มที่ประมาณ 3 อาทิตย์ ซึ่งแก้ไขโดยการนำ Backlog ของ Sprint ถัดไปเริ่มทำงานช่วงหลังของการสอบ

2.ปัญหาการกำหนดขนาดของงาน Design ไวนี้ขนาดที่ใหญ่เกินความจำเป็นและไม่สามารถทำออกมาได้ตรงตามทั้งหมด จึงต้องตัดบางส่วนออก เพื่อลดขอบเขตและสนใจกับฟีเจอร์หลักของเว็บไซต์

3.สมาชิกในทีมมีเรื่องที่ติดคล้ายกัน ทำให้ติดปัญหาเรื่องความรู้หรือความคุ้นเคยในการทำเว็บไซต์ ฟีเจอร์ ที่ต้องการ เริ่มต้นจากการเรียนรู้ใหม่ เกือบทุกรสเรื่อง จึงเสียเวลาการทำงานที่ควรทำงานที่วางแผน แก้ไขโดยการนำ Scum บางส่วนรวมกับ Waterfall เพื่อลดความตึงเครียดในการทำงานแต่ละ Backlog

4.การ Testing เว็บไซต์คิดว่าการทดสอบควรที่จะใช้ Framework ในการทดสอบจำลองการเข้าใช้งาน User ทำให้พบว่าบางขั้นตอนไม่ถูกต้องตาม Design แก้ไขโดยการเข้าไปแก้ Api ระหว่างหน้าเว็บไซต์ของ BackEnd

5.การสื่อสาร จาก Scum เลือกโดยทีมผู้จัดทำยังไม่สามารถสื่อสารเพื่อดำเนินงานตามแผนได้อย่าง 100 เปอร์เซนต์ทำให้เกิดงานล่าช้า แก้ไขโดยการ มี Dashborad สำหรับรายงาน ระยะเวลา ผู้จัดทำ แสดงเพื่อให้เห็นชัดเจน

6.ปัญหาหลักของโปรเจคคือฟีเจอร์ **Reservation** ที่ทีมผู้จัดทำต้องการมีการเปลี่ยนแปลงโครงสร้างโปรแกรมเป็นส่วน และมีการใช้ข้อมูล แบบ Dynamic , state management library น้อย ทำให้โค้ดมีความซับซ้อน

#### Summary

โปรเจค TOAA NIUNG สำเร็จตามเป้าหมายที่กำหนดไว้ โดยสามารถพัฒนาระบบได้ ตามความต้องการใช้งานของร้านอาหาร การเลือกใช้กระบวนการพัฒนาแบบ Waterfall ผสม ตอนเชปต์การทำงานแบบ Scrum ชึ่งช่วยให้การทำงานมีความยืดหยุ่น สามารถปรับเปลี่ยน แผนงานได้ตามสถานการณ์ และช่วยให้การทำงานเป็นทีมเวิร์คและสามารถแก้ไขปัญหาร่วมกัน ยังทำให้คนในทีมรู้สึกขึ้นใจมาก ความสามารถต่องานที่ได้รับว่างานที่คิดกับการลงมือทำจริงต่าง กันและจะจัดการกับงานที่เหลืออย่างไร การประเมินชึ่งเป็นสำคัญ โดยสรุปแล้ว โปรเจคนี้เป็น ประสบการณ์ที่ยอดเยี่ยมจากการทำงานร่วมกันและการแก้ปัญหาที่เกิดขึ้น ทีมผู้พัฒนาเชื่อ ว่าโปรเจคนี้จะเป็นประโยชน์ต่อผู้ใช้ และจะช่วยให้ผู้คนสะดวกขึ้นและสามารถใช้ชีวิตได้ง่ายขึ้น