

ระบบจองร้านอาหาร
RESTAURANT RESERVATION SYSTEM

โดย
นางสาวหทัยทิพย์ บุคดี
นางสาวอังษณา ชลศักดิ์พิพัฒน์

ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

ระบบจองร้านอาหาร
RESTAURANT RESERVATION SYSTEM

โดย

นางสาวหทัยทิพย์	บุคดี	57011444
นางสาวอังษณา	ชลศักดิ์พิพัฒน์	57011513

อาจารย์ที่ปรึกษา

ผศ.สุรพล บุญจันทร์

อาจารย์ที่ปรึกษาร่วม

ผศ.ดร.กฤษณ์ วงจรูจิระ

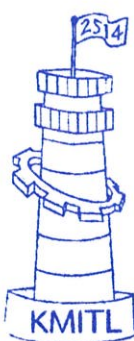
ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

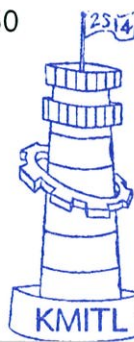
ปีการศึกษา 2560



ผ่านการตรวจรูปเล่มแล้ว

()
อาจารย์ที่ปรึกษา
...16/05/61

วิศวกรรมโทรคมนาคม
Telecommunications Engineering



ผ่านการตรวจชิ้นงานแล้ว

()
กรรมการผู้ตรวจชิ้นงาน
...16/05/61

วิศวกรรมโทรคมนาคม
Telecommunications Engineering

ปริญญาานิพนธ์ปีการศึกษา 2560

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบจองร้านอาหาร

RESTAURANT RESERVATION SYSTEM

ผู้จัดทำ

1. นางสาวหทัยทิพย์ บุตดี 57011444
2. นางสาวอังษณา ชลศักดิ์พิพัฒน์ 57011513

..... อาจารย์ที่ปรึกษา
(ผศ.สุรพล บุญจันทร์)

..... อาจารย์ที่ปรึกษาร่วม
(ผศ.ดร.กฤษณ์ วงจริระ)

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จส่งไปได้ด้วยดีด้วยความช่วยเหลืออย่างดียิ่งจากอาจารย์ที่ปรึกษา คือ ผศ.สุรพล บุญจันทร์ และ ผศ.ดร.กฤษณ์ วรจิริระ ที่ให้คำแนะนำ คำสั่งสอน ให้ความรู้ คอยดูแลเอาใจใส่ รวมถึงตรวจสอบ แก้ไข ข้อบกพร่อง และข้อเสนอแนะเกี่ยวกับแนวทางในการทำ ปริญญานิพนธ์นี้ ขอขอบพระคุณเป็นอย่างสูง

ขอขอบคุณท่านอาจารย์ ประจำภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่านที่ได้อบรมสั่งสอนแก่ผู้จัดทำ

สุดท้ายนี้ขอกราบขอบคุณบิดามารดา ที่ให้ความสนับสนุนทางด้านทุนทรัพย์ และเป็นกำลังใจในการเรียนรวมถึงการทำปริญญานิพนธ์นี้จนประสบความสำเร็จ

นางสาวหทัยทิพย์ บุคดี
นางสาวอังษณา ชลศักดิ์พิพัฒน์
ผู้จัดทำ

ระบบจองร้านอาหาร
RESTAURANT RESERVATION SYSTEM

โดย นางสาวหทัยทิพย์ บุคดี 57011444
นางสาวอังษณา ชลศักดิ์พิพัฒน์ 57011513

อาจารย์ที่ปรึกษา ผศ.สุรพล บุญจันทร์
ผศ.ดร.กฤษณ์ วงจรจิระ

บทคัดย่อ

โครงการนี้มีวัตถุประสงค์เพื่อศึกษาและสร้างแอปพลิเคชันสำหรับระบบปฏิบัติการแอนดรอยด์ (Android) เพื่อทำการสร้างระบบจองที่นั่งในร้านอาหาร และส่งข้อมูลไปยังฐานข้อมูลเพื่อประมวลผล ซึ่งจะแสดงผลข้อมูลแก่ผู้ใช้บริการ รวมทั้งออกแบบและสร้างอัลกอริทึมสำหรับการประมวลผลเพื่อสร้างระบบจองคิวร้านอาหารผ่านทางแอปพลิเคชันของสมาร์ตโฟนบนระบบปฏิบัติการแอนดรอยด์ (Android) และจำลองร้านอาหารหนึ่งร้านที่แบ่งคิวตามประเภทของโต๊ะซึ่งใช้อุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi) สำหรับเป็นเซิร์ฟเวอร์ ซึ่งช่วยให้ผู้ใช้บริการสามารถจองคิวร้านอาหารผ่านแอปพลิเคชันผ่านสมาร์ตโฟนระบบปฏิบัติการแอนดรอยด์ (Android) โดยไม่ต้องรออยู่หน้าร้านอาหาร และสะดวกรวดเร็วมากขึ้น

ABSTRACT

The objective of this project is making an application for queuing reservation based on Android operating system. This queuing application will be used for reservation a seat in restaurant. Most restaurant will composed of many types of table size. Some tables can be varied from one seat to many seats. The application that we designed will cope with user to making a reservation on different size of table. The queue number will show on user smart phone screen and also on the screen at the front of restaurant. At the end of day, the owner of restaurant will know the number of customer and congestion time of a day. This can alleviate service of restaurant and user waiting in a queue.

สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	III
สารบัญรูป	V
บทที่ 1	
บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงการ	1
บทที่ 2	
ทฤษฎีและหลักการที่เกี่ยวข้อง	2
2.1 ภาษาจาวา (JAVA)	2
2.2 ระบบปฏิบัติการแอนดรอยด์ (Android)	2
2.3 โปรแกรม android studio สำหรับแอปพลิเคชันแอนดรอยด์	6
2.4 อุปกรณ์รึาสเบอร์รี่พายสำหรับการสร้างเซิร์ฟเวอร์	25
2.5 การติดตั้ง Apache บนอุปกรณ์รึาสเบอร์รี่พายสำหรับสร้างเซิร์ฟเวอร์	37
2.6 ภาษาพีเอชพี (PHP)	42
2.7 HTML	42
2.8 CSS	43
2.9 จาวาสคริปต์ (Java Script)	43
บทที่ 3	
การออกแบบและการจัดทำปฏิญญานิพนธ์	46
3.1 การออกแบบ	46
3.2 เครื่องมือที่ใช้ในการทดลอง	72
3.3 การจัดเก็บผลการทดลอง	74

สารบัญ (ต่อ)

	หน้า
บทที่ 4	76
ผลการทดลอง	
4.1 การทดสอบแอปพลิเคชัน	76
4.2 การทดสอบการทำงานของเว็บแสดงคิว	88
4.3 การทดสอบการทำงานของเว็บแอดมิน	89
4.4 การทดสอบอุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi) สำหรับการสร้างเซิร์ฟเวอร์ (Server)	98
บทที่ 5	100
สรุปผลและข้อเสนอแนะ	
5.1 สรุปผล	100
5.2 ข้อเสนอแนะ	100
บรรณานุกรม	101
ภาคผนวก ก	102
โค้ดสำหรับแอปพลิเคชันแอนดรอยด์ในการจองร้านอาหาร	
ภาคผนวก ข	118
โค้ดสำหรับการประมวลคิว	

สารบัญรูป

รูปที่		หน้า
2.1	สถาปัตยกรรมของแอนดรอยด์	4
2.2	ดาวน์โหลด JDK	7
2.3	กดดาวน์โหลดตัวติดตั้ง JDK ที่ตรงกับ Environment ของเครื่องพัฒนา	8
2.4	Double click ที่ไฟล์ติดตั้ง เพื่อเริ่มทำการติดตั้ง JDK	8
2.5	เลือก Option ในการติดตั้ง JDK	9
2.6	เลือก Folder ที่ต้องการติดตั้ง	9
2.7	หน้าจอเมื่อการติดตั้ง JKD เรียบร้อยแล้ว	10
2.8	Download ตัวติดตั้งโปรแกรม	11
2.9	Download ตัวติดตั้ง โปรแกรม Android Studio	11
2.10	กดปุ่ม Next เพื่อเข้าสู่หน้าจอสำหรับเลือก Component	12
2.11	กดปุ่ม Next เพื่อยืนยัน License Agreement	12
2.12	กดปุ่ม “I Agree” เพื่อเข้าสู่หน้าจอสำหรับเลือกโฟลเดอร์ที่ต้องการติดตั้ง	13
2.13	หน้าต่างแรกสำหรับสร้างโปรเจค	13
2.14	แถบเมนูเริ่มต้นสำหรับ Android Studio	14
2.15	โค้ดตัวอย่างของแอนดรอยด์	14
2.16	Version Control	15
2.17	เมนู Configure	15
2.18	Android SDK Manager	16
2.19	หน้าต่างตั้งค่า Android Studio	16
2.20	หน้าต่าง Plugins	17
2.21	หน้าต่าง Import File Location	17
2.22	หน้าต่าง Export Settings	18
2.23	เมนูย่อยของหน้าต่าง Project Defaults	18
2.24	Version Control	19
2.25	หน้าต่าง Project Structure	19
2.26	หน้าต่าง Run/Debug	20
2.27	เมนูย่อยของ Docs na How-Tos	20

สารบัญรูป(ต่อ)

รูปที่		หน้า
2.28	หน้าต่าง Help บนเว็บของ IntelliJ IDEA	21
2.29	หน้าต่าง Tip of the day	21
2.30	หน้าเว็บ IntelliJ IDEA	22
2.31	กรอกข้อมูลของแอปพลิเคชัน	23
2.32	เลือก version ของ Android	23
2.33	เลือก Activity	24
2.34	ตั้งชื่อ Activity และ Layout	24
2.35	MAINACTIVITY JAVA สำหรับสร้างการตอบสนองการใช้งานองค์ประกอบต่าง ๆ บนหน้า Application Studio	25
2.36	ราสเบอร์รี่พาย 2 (Raspberry Pi 2)	26
2.37	ส่วนประกอบของ Raspberry Pi	27
2.38	ดาวนโหลดไฟล์อิมเมจ	28
2.39	ไฟล์อิมเมจหลังจากแตกไฟล์แล้ว	29
2.40	ดาวนโหลดโปรแกรม Win32 Disk Imager	29
2.41	เลือกไฟล์เพื่อติดตั้ง Image File ใน Micro SD	30
2.42	Write Image File ลง Micro SD Card	30
2.43	การเปิดใช้งาน Raspberry Pi	31
2.44	การตั้งค่าและใช้งานเบื้องต้นกับ Raspberry Pi	32
2.45	การตั้งค่าและใช้งานเบื้องต้นกับ Raspberry Pi	32
2.46	หน้า Desktop ของ Raspbian	33
2.47	เปลี่ยน Keyboard Layout	33
2.48	เปลี่ยน Keyboard Layout	34
2.49	ตั้งเวลาให้กับ Raspberry Pi	34
2.50	ตั้งเวลาให้กับ Raspberry Pi	35
2.51	การปิด (Shut Down) Raspberry Pi กรณีอยู่ที่หน้า Desktop	36
2.52	การติดตั้ง Apache ด้วยคำสั่ง sudo apt-get install apache2	37

สารบัญรูป(ต่อ)

รูปที่		หน้า
2.53	Web Browser บน Raspberry Pi apacheE2	37
2.54	ติดตั้ง PHP	38
2.55	พิมพ์คำสั่ง cd /var/www เพื่อไปยัง Directory	38
2.56	ติดตั้ง MYSQL ได้ด้วยคำสั่ง Sudo APT-GET	38
2.57	กำหนดรหัสผ่านสำหรับ MySQL	39
2.58	ติดตั้ง PHPMyAdmin	39
2.59	กำหนด Web Server เป็น Apache2	40
2.60	กำหนดรหัสผ่านสำหรับ Administrator	40
2.61	เปิด Web Browser ไปที่ PHPMyAdmin	41
3.1	บล็อกไดอะแกรมของโครงการ	46
3.2	รูปแบบฐานข้อมูล	49
3.3	บล็อกไดอะแกรมการทำงานของแอปพลิเคชัน (Application)	50
3.4	แผนผัง User Interface ของแอปพลิเคชัน	51
3.5	โปรแกรมแอนดรอยด์สตูดิโอ (Android Studio)	52
3.6	เลือก Activity	52
3.7	หน้าต่างที่ใช้สำหรับเขียนโค้ดคำสั่ง	53
3.8	หน้าต่างที่สำหรับออกแบบ	53
3.9	การส่งข้อมูลเข้าสู่ฐานข้อมูล (data base)	54
3.10	การรับข้อมูลจากแอปพลิเคชันในฝั่งเซิร์ฟเวอร์ (server)	54
3.11	แผนผังงาน (flowchart) แสดงการทำงานของระบบล็อกอิน (Log in)	55
3.12	หน้าล็อกอิน (Log In) และหน้าการจองคิว ตามลำดับ	56
3.13	Flowchart แสดงการทำงานของอัลกอริทึมการจองคิว	57
3.14	แผนผังงาน (FLOWCHART) การยกเลิกคิว	58
3.15	หน้าแสดงผลคิว	59
3.16	แผนผังงาน (FLOWCHART) การประมาณเวลารอคิว	60
3.17	หน้าแสดงผลคิว	61

สารบัญรูป(ต่อ)

รูปที่	หน้า	
3.18	หน้าเกมส์	62
3.19	ออกแบบเว็บไซต์สำหรับแสดงผลคิว	63
3.20	Flowchart แสดงการทำงานของเว็บไซต์แสดงผลคิว	64
3.21	หน้าล็อกอินของเว็บแอดมิน	65
3.22	Flowchart แสดงการทำงานของระบบล็อกอิน (Log in)	66
3.23	ออกแบบหน้าเว็บแอดมิน (admin web)	67
3.24	หน้าเว็บแอดมิน สำหรับเลือกช่วงเวลาพล็อตข้อมูล	68
3.25	ข้อมูลจากฐานข้อมูล (DATABASE) ที่นำมาพล็อต	69
3.26	โค้ดการดึงข้อมูลจากฐานข้อมูล (DATABASE) ที่นำมาพล็อต	69
3.27	โค้ดส่วนรูปแบบกราฟ	70
3.28	รูปแบบปฏิทินกิจกรรม	70
3.29	Bootstrap Modal สำหรับการเพิ่มข้อมูล	71
3.30	Bootstrap Modal สำหรับการลบและแก้ไขข้อมูล	71
3.31	ราสเบอร์รี่พาย 2 (Raspberry Pi 2)	72
3.32	ส่วนประกอบของ Raspberry Pi	73
3.33	ส่วนประกอบของ Raspberry Pi	73
4.1	แสดงผลคิวของโต๊ะประเภท 1-2 ท่าน	76
4.2	แสดงผลคิวของโต๊ะประเภท 3-4 ท่าน	77
4.3	แสดงผลคิวของโต๊ะประเภท 5-6 ท่าน	77
4.4	แสดงผลคิวของโต๊ะประเภท 7 ท่านขึ้นไป	78
4.5	ไอคอนของแอปพลิเคชัน (Application)	78
4.6	การแจ้งเตือนการเชื่อมต่อ อินเทอร์เน็ต (Internet)	79
4.7	หน้าเมนูหลัก	79
4.8	หน้าแสดงผลคิวที่ได้	80
4.9	หน้าแสดงผลระยะเวลารอคิว (จองครั้งที่ 1)	81
4.10	หน้าแสดงผลระยะเวลารอคิว (จองครั้งที่ 2)	81

สารบัญรูป(ต่อ)

รูปที่	หน้า	
4.11	หน้าแสดงผลระยะเวลาารอคิว (จองครั้งที่ 6)	82
4.12	หน้าแสดงผลระยะเวลาารอคิว (จองครั้งที่ 11)	82
4.13	หน้าแสดงผลระยะเวลาารอคิว (จองครั้งที่ 16)	83
4.14	หน้าแสดงผลระยะเวลาารอคิว (จองครั้งที่ 21)	83
4.15	หน้าแสดงผลระยะเวลาารอคิว (จองครั้งที่ 22)	84
4.16	หน้าแสดงผลคิวและยกเล็กคิว	85
4.17	คิว A3 เปลี่ยนค่าสแตตัส (STATUS) เป็นยกเล็กคิว	85
4.18	ชิ้นส่วนรูปภาพขนาด 3 X 3	86
4.19	เกมส์ PUZZLE	87
4.20	เมื่อชนะเกมส์	87
4.21	ข้อมูลในฐานข้อมูล (DATABASE) สำหรับทดสอบเว็บแสดงคิว	88
4.22	เว็บแสดงคิว	88
4.23	ข้อมูลในฐานข้อมูล (database) สำหรับทดสอบเว็บแอดมิน (admin web)	89
4.24	การทำงานของเว็บแอดมิน (admin web)	89
4.25	แสดงคิวสุดท้ายที่ทำการจองของโต๊ะแต่ละประเภท	90
4.26	status ของคิว A1	91
4.27	การทำงานของเว็บแอดมินหลังคลิกปุ่มรันคิวประเภทโต๊ะ 1-2 ท่าน	91
4.28	เว็บแสดงผลคิวหลังกดปุ่มรันคิวโต๊ะประเภท 1-2 ท่าน	92
4.29	ประกาศข้อความผ่านระบบแอดมิน	92
4.30	เว็บแสดงผลคิวหลังจากประกาศข้อความผ่านเว็บแอดมิน	93
4.31	ข้อมูลจากฐานข้อมูล (database) ที่นำมาพล็อต	93
4.32	เลือกช่วงเวลาที่ต้องการพล็อตผ่านหน้าเว็บแอดมิน (admin web)	94
4.33	กราฟประเภทที่ 1 แสดงความสัมพันธ์จำนวนคนแต่ละประเภทในแต่ละเดือน	94

สารบัญรูป(ต่อ)

รูปที่		หน้า
4.34	กราฟประเภทที่ 2 แสดงความสัมพันธ์จำนวนโตะแต่ละประเภทในแต่ละเดือน	95
4.35	เพิ่มข้อมูลผ่าน Bootstrap Modal	95
4.36	ปฏิทินกิจกรรมหลังจากเพิ่มข้อมูลแล้ว	96
4.37	แก้ไขข้อมูลในปฏิทินกิจกรรม	96
4.38	ปฏิทินกิจกรรมหลังจากแก้ไขข้อมูล	97
4.39	ปฏิทินกิจกรรมหลังจากลบข้อมูล	97
4.40	ราสเบอร์รี่พาย (Raspberry Pi) เสียบอุปกรณ์ต่าง ๆ	98
4.41	หน้าจอของอุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi) เมื่อเปิดใช้งาน	99
4.42	เซิร์ฟเวอร์บนอุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi)	99

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากในปัจจุบัน ระบบจองคิวร้านอาหารตามห้างสรรพสินค้ายังมีประสิทธิภาพไม่มากพอต่อความต้องการของผู้บริโภค อาทิ เช่น ลูกค้าต้องยืนรอคิวหน้าร้านเป็นระยะเวลาานาน , พนักงานหน้าร้านอาจทำงานผิดพลาดได้เมื่อมีลูกค้าจำนวนมาก เป็นต้น ผู้วิจัยจึงได้มีการศึกษาและวิจัย “ระบบจองร้านอาหาร” โดยลูกค้าสามารถจองคิวร้านอาหารผ่านแอปพลิเคชันบนสมาร์ทโฟนระบบปฏิบัติการแอนดรอยด์ และติดตามการแสดงผลคิวผ่านเว็บไซต์และหน้าร้าน ในส่วนสุดท้าย ผู้วิจัยได้ทำการสร้างเว็บไซต์สำหรับผู้ดูแลระบบ (Admin) เพื่อที่จะสามารถบริหารจัดการคิวและดูแลความผิดพลาดที่เกิดขึ้นต่างๆ ได้ง่ายขึ้น

จากความสำคัญข้างต้นจึงเป็นที่มาของปริญญานิพนธ์ “ระบบจองร้านอาหาร” ซึ่งจะพัฒนาระบบให้มีประสิทธิภาพ มีความสมบูรณ์ และเหมาะสมแก่การใช้งานมากยิ่งขึ้น ทั้งนี้เนื้อหาในส่วนต่างๆ จะมีการกล่าวถึงอย่างละเอียดในบทถัดๆ ไป

1.2 วัตถุประสงค์

- 1) ศึกษาและสร้างแอปพลิเคชันสำหรับระบบปฏิบัติการแอนดรอยด์ (Android)
- 2) ศึกษาและสร้างอัลกอริทึมสำหรับระบบจองคิวร้านอาหาร
- 3) ศึกษาและสร้างหน้าเว็บไซต์สำหรับแสดงผลคิว
- 4) ศึกษาและสร้างหน้าเว็บไซต์สำหรับผู้ดูแลระบบ
- 5) ศึกษาอุปกรณ์ราสเบอร์รี่พาย (Raspberry pi) สำหรับสร้างเซิร์ฟเวอร์ (Server)

1.3 ขอบเขตของปริญญานิพนธ์

ออกแบบและสร้างแอปพลิเคชันสำหรับระบบปฏิบัติการแอนดรอยด์ (Android) สำหรับจองคิวร้านอาหาร โดยเริ่มจากจำลองร้านอาหารจำนวนหนึ่งร้าน เมื่อผู้ใช้งานทำการจองคิวร้านอาหารผ่านแอปพลิเคชันแอนดรอยด์ (Application Android) ข้อมูลจากแอปพลิเคชัน (Application) จะถูกส่งไปที่อุปกรณ์ราสเบอร์รี่พาย (Raspberry pi) สำหรับใช้เป็นเซิร์ฟเวอร์ (Server) เพื่อทำการประมวลผลคิวผ่านอัลกอริทึม (Algorithm) การจองคิว จากนั้น เซิร์ฟเวอร์ (Server) จะส่งข้อมูลไปที่แอปพลิเคชัน (Application) อีกครั้งเพื่อแสดงผลคิวที่ได้ ทั้งนี้ผู้ใช้งานสามารถทราบการแสดงผลคิว

ปัจจุบันผ่านหน้าเว็บไซต์และหน้าร้านได้ นอกจากนี้ ยังมีการสร้างหน้าเว็บไซต์สำหรับผู้ดูแลระบบ สำหรับบริหารจัดการคิว และแก้ไขข้อผิดพลาดที่เกิดขึ้นต่างๆ

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 ภาษา Java

ภาษา Java เป็นภาษาโปรแกรมที่พัฒนาขึ้นโดยทีมวิจัยโครงการชื่อ Green ของบริษัท Sun Microsystems ซึ่งมีวัตถุประสงค์เพื่อพัฒนาภาษาที่ใช้สร้างโปรแกรม หรือระบบงานที่สนับสนุนการทำงานของแต่ละระบบย่อยๆ ซึ่งภาษา Java มีคุณสมบัติดังนี้

- 1) ง่ายต่อการเข้าใจ เนื่องจากมีกลไกของภาษาไม่ซับซ้อน
- 2) มีความคงทน เนื่องจากมีการตรวจจับข้อผิดพลาด
- 3) มีความปลอดภัย
- 4) ทำงานได้ทุกระบบ คือ มีคุณลักษณะของ Java Platform
- 5) มีคลาสและอินเทอร์เฟซจำนวนมาก

จากคุณสมบัติดังกล่าวทำให้ Java สนับสนุนการเขียนโปรแกรมเชิงวัตถุที่เน้นการแบ่งงานให้เป็นส่วนย่อยๆ ที่มีความสัมพันธ์กัน และเป็นอิสระจากส่วนย่อยของงานอื่นๆ ซึ่งจะช่วยลดความซ้ำซ้อนและเวลาของการพัฒนางาน

2.2 ระบบปฏิบัติการแอนดรอยด์ (Android)

แอนดรอยด์ คือ ระบบปฏิบัติการสำหรับอุปกรณ์พกพา เช่น โทรศัพท์มือถือและคอมพิวเตอร์แบบแท็บเล็ต เป็นต้น ซึ่งพัฒนามาจากระบบปฏิบัติการลินุกซ์

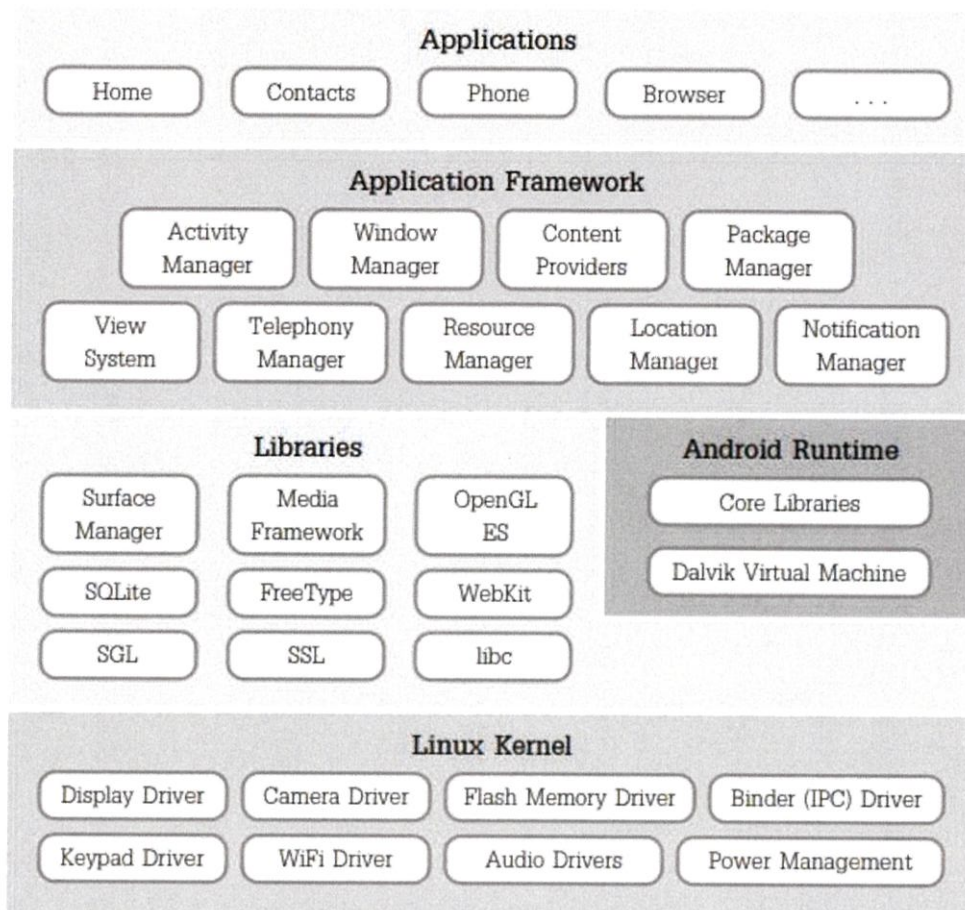
แรกเริ่มนั้นแอนดรอยด์ถูกพัฒนาโดยบริษัท Android Inc. ซึ่งก่อตั้งในปี ค.ศ. 2003 โดย Andy Rubin และ Rich Miner ต่อมาในปี ค.ศ. 2005 กูเกิลได้เข้าซื้อบริษัทดังกล่าว หลังจากนั้นกูเกิลได้ร่วมมือกับกลุ่มบริษัททางด้านฮาร์ดแวร์ ซอฟต์แวร์ และการสื่อสาร เช่น Intel, HTC, LG เพื่อจัดตั้งองค์กรความร่วมมือที่มีชื่อว่า Open Handset Alliance ขึ้นในปี ค.ศ. 2007 โดยมีจุดประสงค์ในการสร้างแพลตฟอร์มสำหรับอุปกรณ์พกพาที่มีพื้นฐานอยู่บนมาตรฐานเปิด (Open Standard) ซึ่งโปรเจ็คแรกที่กลุ่ม Open Handset Alliance เปิดตัวออกมา คือ The Android Open Source Project

2.2.1 คุณสมบัติและความสามารถของแอนดรอยด์

- 1) การเชื่อมต่อ เทคโนโลยีการเชื่อมต่อที่แอนดรอยด์สนับสนุนประกอบด้วย GSM/EDGE, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE และ NFC
- 2) Messaging สนับสนุน SMS, MMS, Threaded Text Messaging และ Cloud To Device Messaging Framework (C2DM3)
- 3) การจัดเก็บข้อมูล แอนดรอยด์มี SQLite ซึ่งเป็นฐานข้อมูลเชิงสัมพันธ์ขนาดเล็กที่มีประสิทธิภาพสูง สำหรับใช้จัดเก็บข้อมูล
- 4) เว็บเบราว์เซอร์ แอนดรอยด์ติดตั้งมาพร้อมกับโปรแกรมเว็บเบราว์เซอร์ที่พัฒนาบนเอ็นจิน WebKit และใช้จาวาสคริปต์เอ็นจิน V8 ของเว็บเบราว์เซอร์ Google Chrome
- 5) มีเดีย สนับสนุนเสียง วิดีโอ และรูปภาพในฟอร์แมตยอดนิยมเช่น MPEG4, H.264, MP3, AAC, AMR, JPG และ PNG
- 6) สตรีมมิง สนับสนุน RTP/RTSP streaming และ HTML progressive download
- 7) สนับสนุนจาวา การพัฒนาแอปพลิเคชันบนแอนดรอยด์จะใช้ภาษาจาวา โดยโค้ดจาวาที่คอมไพล์แล้วจะไม่ได้รันใน Java Virtual Machine เหมือนจาวาแอปพลิเคชันทั่วไป
- 8) มัลติทัช รองรับการใช้นิ้วมือแตะหน้าจอเพื่อสั่งงานได้มากกว่า 1 จุดพร้อมกัน
- 9) Tethering (หรือ Mobile Hotspot) คือความสามารถในการแชร์อินเทอร์เน็ตผ่านมือถือหรืออุปกรณ์แอนดรอยด์
- 10) สนับสนุนฮาร์ดแวร์เสริมอื่นๆ เช่น กล้องถ่ายรูป, GPS, Accelerometer และ เทอร์โมมิเตอร์ เป็นต้น
- 11) สนับสนุนหลายภาษา

2.2.2 สถาปัตยกรรมของแอนดรอยด์

รูปต่อไปนี้จะแสดงส่วนประกอบหลักๆของระบบปฏิบัติการแอนดรอยด์ โดยส่วนประกอบในแต่ละชั้นจะเรียกใช้บริการของชั้นที่อยู่ถัดลงไป



รูปที่ 2.1 สถาปัตยกรรมของแอนดรอยด์

1) Linux Kernel

ส่วนที่เป็นแกนหลักหรือเคอร์เนล (Kernel) ของแอนดรอยด์นั้น ความจริงก็คือ เคอร์เนลของลินุกซ์ ซึ่งเป็นระบบปฏิบัติการที่สร้างโดย Linus Torvalds ในปี ค.ศ. 1991 ปัจจุบันเราสามารถพบลินุกซ์ได้ในทุกสิ่งทุกอย่าง ตั้งแต่นาฬิกาข้อมือไปจนถึงซูเปอร์คอมพิวเตอร์

ส่วนของ Linux Kernel นี้จะทำหน้าที่เป็น Hardware Abstraction Layer กล่าวคือเป็นตัวกลางระหว่างฮาร์ดแวร์กับส่วนของซอฟต์แวร์ที่อยู่ถัดไป และทำหน้าที่บริหารจัดการทรัพยากรต่างๆ ของเครื่อง เช่น การจัดการหน่วยความจำ การจัดการโปรเซส ฯลฯ ผู้ผลิตอุปกรณ์สามารถ “พอร์ต” แอนดรอยด์ให้ไปรันบนฮาร์ดแวร์แบบต่างๆ ได้โดยเปลี่ยนแปลงในส่วนของ Linux Kernel นี้

ผู้ใช้มือถือหรืออุปกรณ์แอนดรอยด์จะไม่เห็นว่ามีลินุกซ์อยู่ในเครื่อง และแอปพลิเคชันที่เราพัฒนายังจะไม่ได้เรียกไปยังลินุกซ์โดยตรง แต่ในฐานะนักพัฒนาเราจำเป็นต้องรับรู้ว่ามีลินุกซ์อยู่เนื่องจากโปรแกรมยูทิลิตี้บางตัวที่ Android SDK เตรียมมาให้จะติดต่อกับลินุกซ์ในอุปกรณ์แอนดรอยด์ เช่น โปรแกรม adb ที่ช่วยให้เราเรียกใช้คำสั่งเพื่อสำรวจระบบไฟล์ของเครื่อง ดูโปรเซสที่รันอยู่ในขณะนั้นและอื่นๆได้

2) ไบเบรารี (Native Libraries)

ถัดขึ้นมาจาก Linux Kernel ก็คือส่วนที่เป็นไลบรารีของแอนดรอยด์ ซึ่งทั้งหมดเขียนด้วยภาษา C หรือ C++ และถูกคอมไพล์มาสำหรับฮาร์ดแวร์ของอุปกรณ์แต่ละรุ่น ไลบรารีที่น่าสนใจมีดังนี้

2.1) Surface Manager คือไลบรารีจัดการส่วนแสดงผลที่มีความสามารถในการผสมกราฟฟิกทั้ง 2 มิติและ 3 มิติจากแอปพลิเคชันต่างๆ เข้าด้วยกัน ทำให้สามารถสร้างเอฟเฟ็ค เช่น วินโดว์ ที่มองทะลุไปข้างหลังได้ และ Transition ในรูปแบบต่างๆ

2.2) Media Libraries คือไลบรารีที่จัดเตรียมบริการในการเล่นและบันทึกเสียง วิดีโอ และรูปภาพ ในฟอร์แมตต่างๆ

2.3) SQLite คือ Database Engine ที่มีประสิทธิภาพและมีขนาดเล็ก เพื่อให้เราจัดเก็บข้อมูลของแอปพลิเคชันไว้ในรูปของฐานข้อมูลเชิงสัมพันธ์

2.4) WebKit คือไลบรารีที่ใช้แสดงเนื้อหาเว็บเพจ ซึ่งเป็นตัวเดียวกับที่ใช้ใน Google Chrome และ Apple safari รวมถึงเว็บเบราว์เซอร์ในมือถือ iPhone และมือถือตระกูล S60 ของโนเกียด้วย

3) Android Runtime

อีกส่วนหนึ่งที่ทำงานอยู่บนเคอร์เนลของลินุกซ์คือ Android Runtime ซึ่งประกอบด้วย Core Library สำหรับภาษาจาวา และ Dalvik Virtual Machine อธิบายง่ายๆคือ Java Virtual Machine ในแบบของแอนดรอยด์เอง ซึ่งถูกออกแบบมาสำหรับอุปกรณ์ที่ใช้ไฟฟ้าจากแบตเตอรี่และมีหน่วยความจำจำกัด

4) Application Framework

เป็นส่วนที่ใช้พัฒนาแอปพลิเคชัน ซึ่งประกอบด้วยคอมโพเนนต์พื้นฐานต่างๆ ที่ใช้ในการสร้างแอปพลิเคชันของเรา คอมโพเนนต์เหล่านี้จะติดตั้งมากับแอนดรอยด์อยู่แล้ว และเราสามารถแทนที่ด้วยคอมโพเนนต์ที่เราสร้างขึ้นเองได้

ส่วนสำคัญใน Application Framework มีดังนี้

4.1) Activity Manager คือคอมโพเนนต์ที่ควบคุม Lifecycle ของแอปพลิเคชัน

4.2) Content Providers คือคอมโพเนนต์ที่ทำให้แอปพลิเคชันต่างๆ สามารถแชร์ข้อมูลกันได้

4.3) View System ประกอบด้วยคอมโพเนนต์ที่ใช้สร้างส่วนติดต่อผู้ใช้ เช่น ปุ่ม เท็กซ์บ็อกซ์ ลิส กริด

4.4) Resource Manager คือตัวจัดการรีซอร์ส ซึ่งรีซอร์สหมายถึงข้อมูลใดๆ ในแอปพลิเคชันที่ไม่ใช่โค้ด เช่น ค่าสตริง และรูปภาพ เป็นต้น

4.5) Notification Manager คือคอมโพเนนต์ที่ทำให้แอปพลิเคชันสามารถแสดงข้อความแจ้งเตือนผู้ใช้ออกมาที่แถบสถานะได้

5) Application

ส่วนบนสุดของสถาปัตยกรรมแอนดรอยด์ ก็คือแอปพลิเคชันต่างๆ ทั้งที่ติดตั้งมากับเครื่องอยู่แล้ว (Core Application) เช่น Phone dialer, Email, Contacts, Web browser และ Google Play (Android Market เดิม) เป็นต้น รวมถึงแอปพลิเคชันที่เราสร้างขึ้นด้วย ซึ่งแอปพลิเคชันทั้งหมดในส่วนนี้จะเขียนด้วยภาษาจาวา

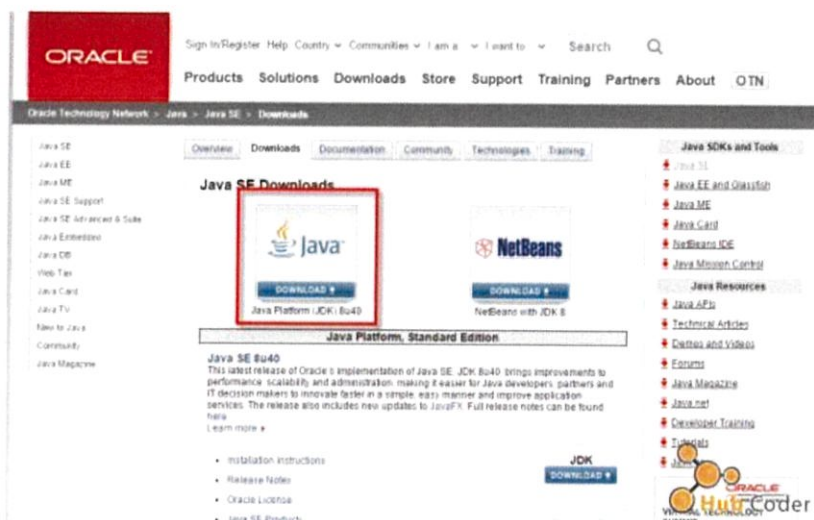
2.3 โปรแกรม android studio สำหรับแอปพลิเคชันแอนดรอยด์

Android Studio เป็นเครื่องมือพัฒนา (IDE : Integrated Development Environment) ที่ถูกสร้าง ขึ้นมาเพื่อการพัฒนาแอนดรอยด์แอปพลิเคชัน บนพื้นฐานของแนวคิด IntelliJ

2.3.1 การติดตั้ง Android Studio

การติดตั้ง Android Studio (IDE สำหรับพัฒนา Application) จะต้องตรวจสอบก่อนว่าใน เครื่องพัฒนานั้นได้มีการติดตั้ง JDK ไว้แล้วหรือไม่ โดยหากติดตั้งไว้แล้ว ก็สามารถข้ามขั้นตอนไปยังการติดตั้ง Android Studio ได้

1) การติดตั้ง JDK (Java Development Kit) ให้เริ่มดาวน์โหลดตัวติดตั้ง JDK ได้ที่เว็บไซต์ <http://www.oracle.com/technetwork/java/javase/downloads/index.html> และเลือกที่ปุ่มนี้ Java- Platform (JDK) ดังแสดงในรูปที่ 2.2



รูปที่ 2.2 ดาวน์โหลด JDK

2) อ่าน License Agreement ให้เรียบร้อย แล้วเลือกที่ "Accept License Agreement" จากนั้นให้กดดาวน์โหลดตัวติดตั้ง JDK ที่ตรงกับ Environment ของเครื่องพัฒนา ดังแสดงในรูปที่ 2.3

Oracle Technology Network | Java | Java SE | Downloads

Overview Downloads Documentation Community Technologies Training

Java SE Development Kit 8 Downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language. The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- Java Developer Newsletter (tick the checkbox under Subscription Center in Oracle Technology News)
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK: MD5 Checksum

Looking for JDK 8 on ARM? JDK 8 for ARM downloads have moved to the JDK 8 for ARM download page

Java SE Development Kit 8u40

You must accept the Oracle Binary Code License Agreement for Java SE to download this software.

Accept License Agreement Decline License Agreement

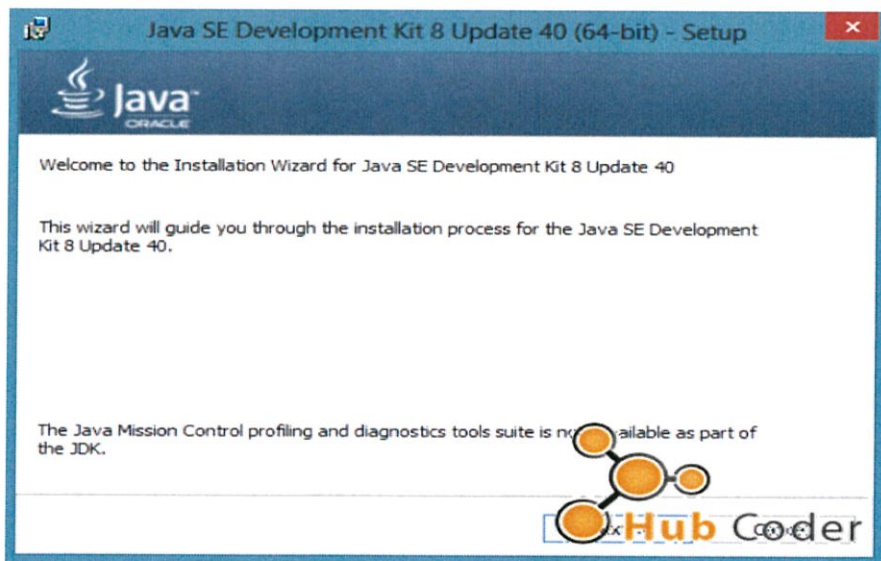
Platform for Architecture	File Size	Download
Linux x85	145.84 MB	jdk-8u40-linux-x85-gpr
Linux x85	155.85 MB	jdk-8u40-linux-x85-tar.gz
Linux x84	145.14 MB	jdk-8u40-linux-x84-gpr
Linux x84	155.17 MB	jdk-8u40-linux-x84-tar.gz
Mac OS X x84	221.91 MB	jdk-8u40-macosx-x84-dmg
Solaris SPARC 64-bit (SVR4 package)	132.05 MB	jdk-8u40-solaris-sparcv9-tar.gz
Solaris SPARC 64-bit	99.58 MB	jdk-8u40-solaris-x84-tar.gz
Solaris x84 (SVR4 package)	130.57 MB	jdk-8u40-solaris-x84-tar.gz
Windows x85	175.71 MB	jdk-8u40-windows-x85.exe
Windows x84	182.19 MB	jdk-8u40-windows-x84.exe

Java SE Development Kit 8u40 Demos and Samples Downloads

Hub Coder

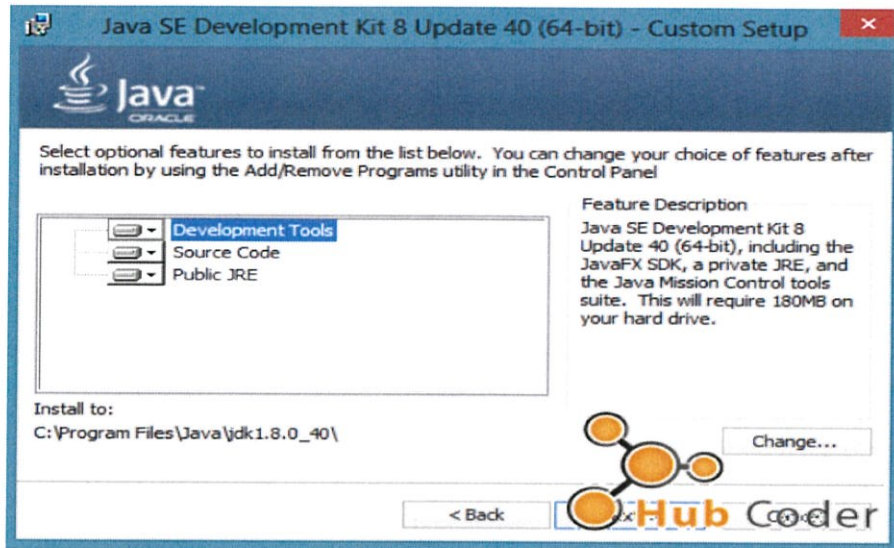
รูปที่ 2.3 กดดาวน์โหลดตัวติดตั้ง JDK ที่ตรงกับ Environment ของเครื่องพัฒนา

3) เมื่อดาวน์โหลด JDK installation เรียบร้อยแล้วให้เลือกที่ไฟล์ติดตั้ง เพื่อเริ่มทำการติดตั้ง โดยจะปรากฏหน้าจอแสดงในรูปที่ 2.4



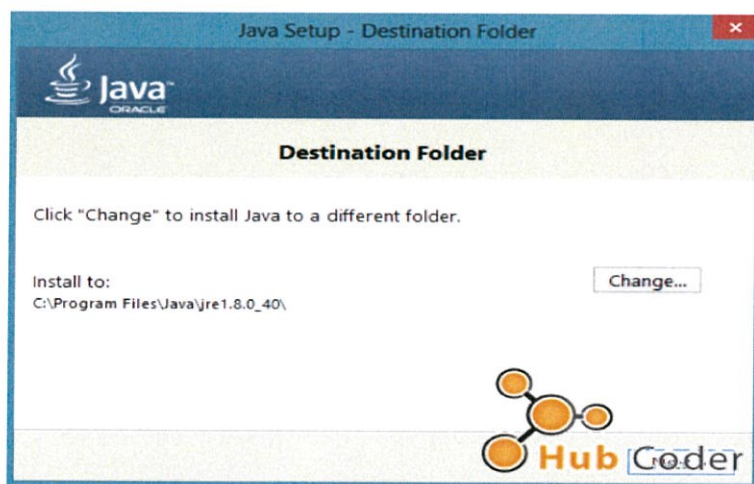
รูปที่ 2.4 Double Click ที่ไฟล์ติดตั้ง เพื่อเริ่มทำการติดตั้ง JDK

4) กดปุ่ม Next เพื่อเลือก Option ในการติดตั้ง JDK ให้ใช้ค่า Default ของการติดตั้ง ที่กำหนดมาไม่ต้องเลือก จากนั้นให้กดปุ่ม Next เพื่อเตรียมการติดตั้ง ดังแสดงในรูปที่ 2.5



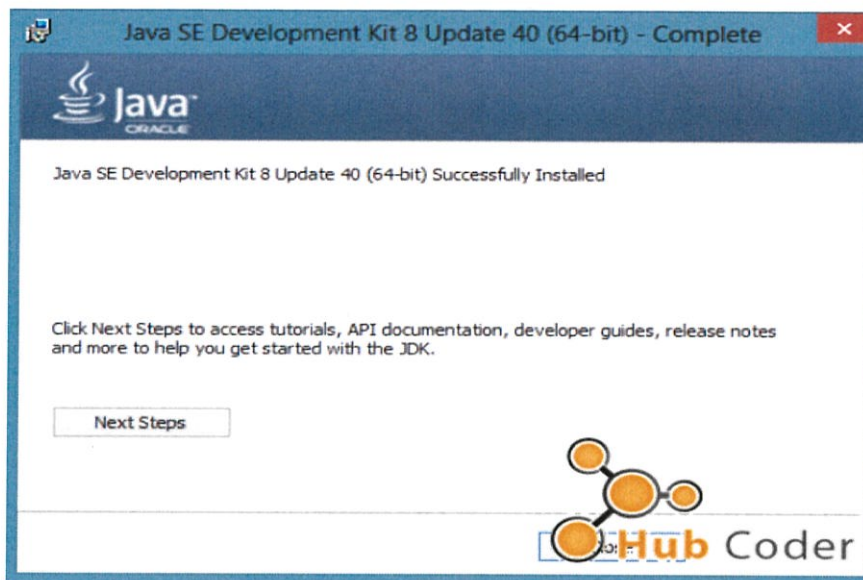
รูปที่ 2.5 เลือก Option ในการติดตั้ง JDK

5) เมื่อเตรียมการติดตั้งเสร็จเรียบร้อยแล้ว จะปรากฏหน้าจอให้เลือกโฟลเดอร์ที่ต้องการ ติดตั้ง JDK จากนั้นให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้ง JDK ทันที ดังแสดงในรูปที่ 2.6



รูปที่ 2.6 เลือก Folder ที่ต้องการติดตั้ง JDK

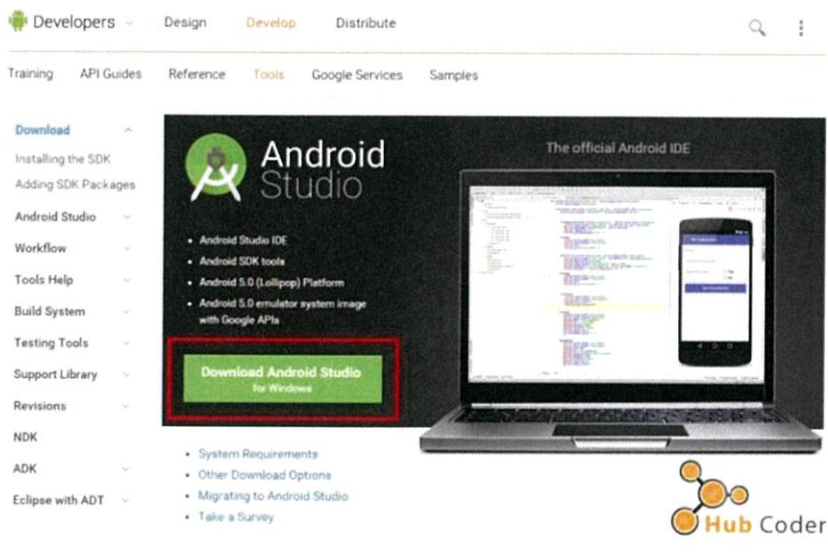
6) เมื่อการติดตั้ง JDK เรียบร้อยแล้ว ก็จะปรากฏหน้าจอแจ้งเตือนการติดตั้ง ดังแสดงในรูปที่ 2.7



รูปที่ 2.7 หน้าจอเมื่อการติดตั้ง JDK เรียบร้อยแล้ว

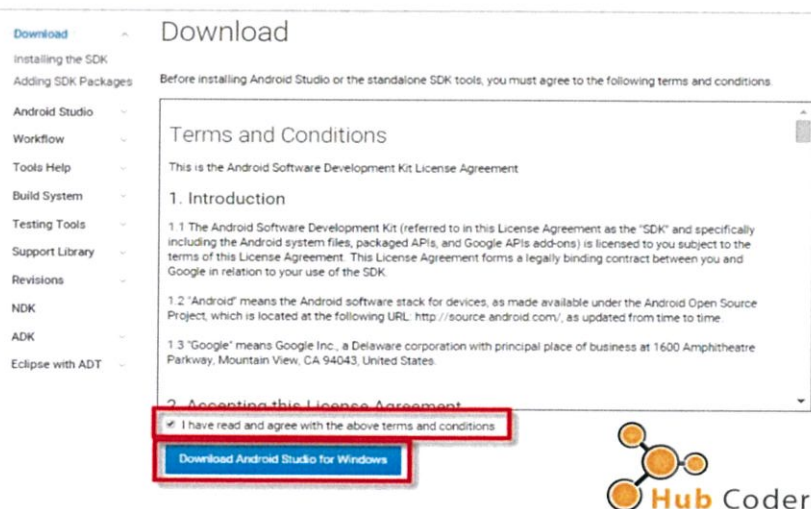
2.3.3 การติดตั้ง Android Studio

1) การติดตั้งโปรแกรม Android Studio นั้นให้เริ่มดาวน์โหลดตัวติดตั้งจากเว็บไซต์ดังนี้ <http://developer.android.com/sdk/index.html> หลังจากนั้น ให้กดปุ่ม Download Android Studio (กรณีที่ใช้ Windows มันจะแสดงปุ่ม ให้ดาวน์โหลดตัวติดตั้งสำหรับ Windows แต่กรณีที่ใช้ OSX หรือ MAC จะแสดงปุ่ม ให้ดาวน์โหลดตัวติดตั้งสำหรับ OS ดังแสดงในรูปที่ 2.8



รูปที่ 2.8 Download ตัวติดตั้ง โปรแกรม Android

2) อ่าน Terms and Conditions ให้เรียบร้อย และเลือก "I have read and agree with the above terms and conditions" จากนั้น ให้กดปุ่ม "Download Android Studio" เพิ่มเรืมาดาวน์โหลด ตัวติดตั้งโปรแกรม Android Studio เมื่อดาวน์โหลดตัวติดตั้งเรียบร้อย แล้ว ให้เลือกที่ตัวติดตั้งเพื่อเริ่มทำ การติดตั้งโปรแกรม ดังแสดงในรูปที่ 2.9



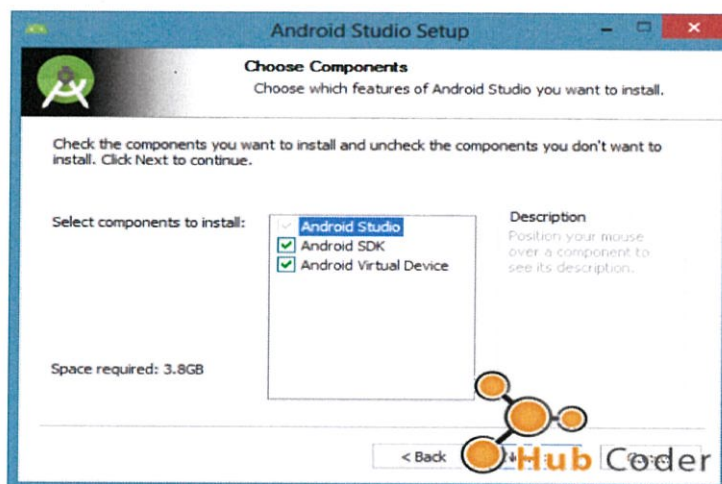
รูปที่ 2.9 Download ตัวติดตั้ง โปรแกรม Android Studio

3) กดปุ่ม Next เพื่อเข้าสู่หน้าจอสำหรับเลือก Component ที่ต้องการติดตั้ง ดังแสดงในรูปที่ 2.10



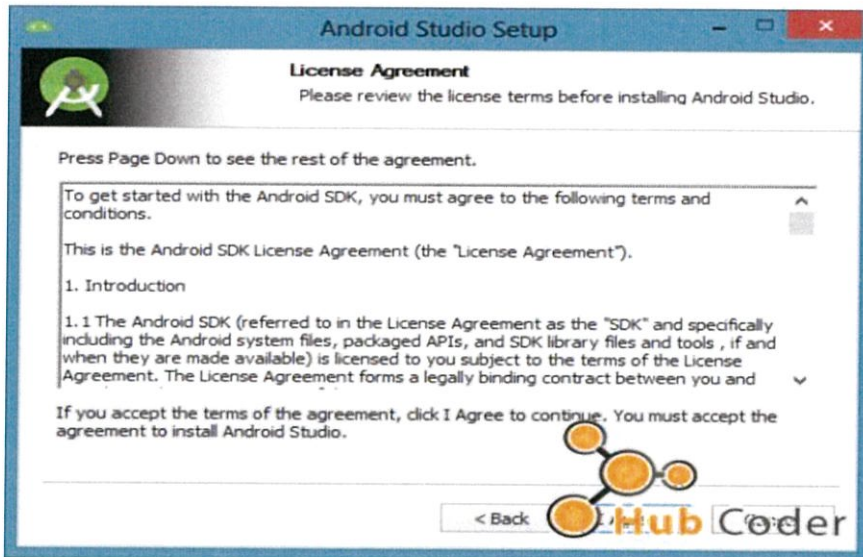
รูปที่ 2.10 กดปุ่ม Next เพื่อเข้าสู่หน้าจอสำหรับเลือก Component

4) ให้เลือกตามค่า Default (เลือกทั้งหมด) หลังจากนั้น กดปุ่ม Next เพื่อยืนยัน License Agreement อีกครั้งหนึ่ง ดังแสดงในรูปที่ 2.11

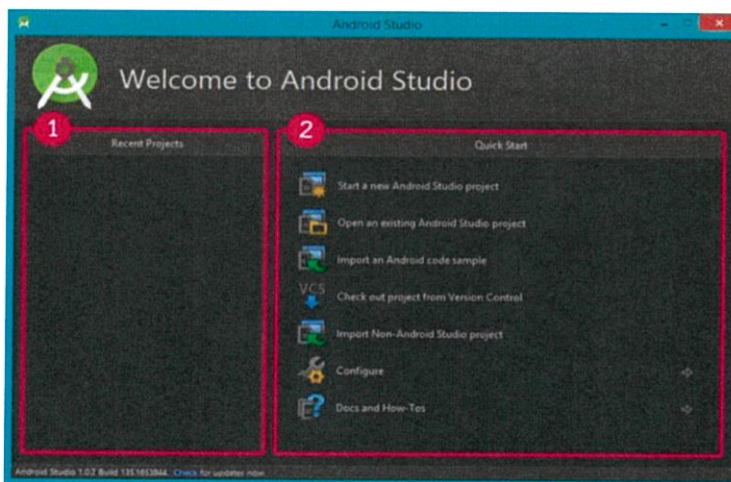


รูปที่ 2.11 กดปุ่ม Next เพื่อยืนยัน License Agreement

5) กดปุ่ม "I Agree" เพื่อเข้าสู่หน้าจอสำหรับเลือกไฟล์เดออร์ที่ต้องการติดตั้ง Android Studio และ SDK ดังแสดงในรูปที่ 2.12



รูปที่ 2.12 กดปุ่ม "I Agree" เพื่อเข้าสู่หน้าจอสำหรับเลือกไฟล์เดออร์ที่ต้องการติดตั้ง
เปิดใช้งาน Android Studio ครั้งแรกก็จะเจอกับหน้าต่างแบบนี้ ดังแสดงในรูปที่ 2.13

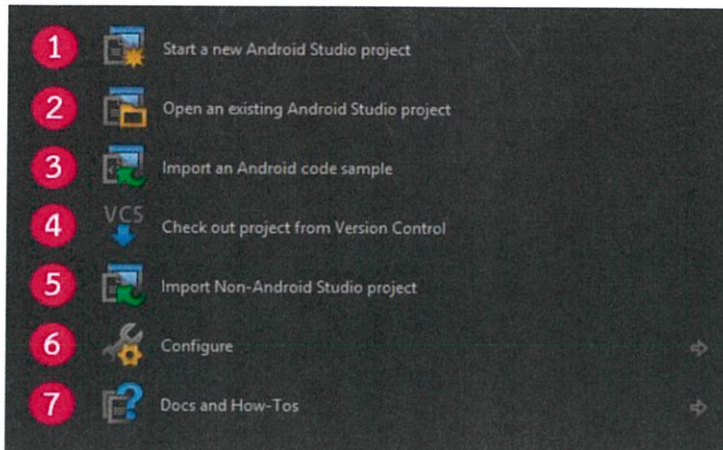


รูปที่ 2.13 หน้าต่างแรกสำหรับสร้างโปรเจค

1) Recent Projects โปรเจคที่เคยเปิดบน Android Studio จะถูกแสดงบนนี้

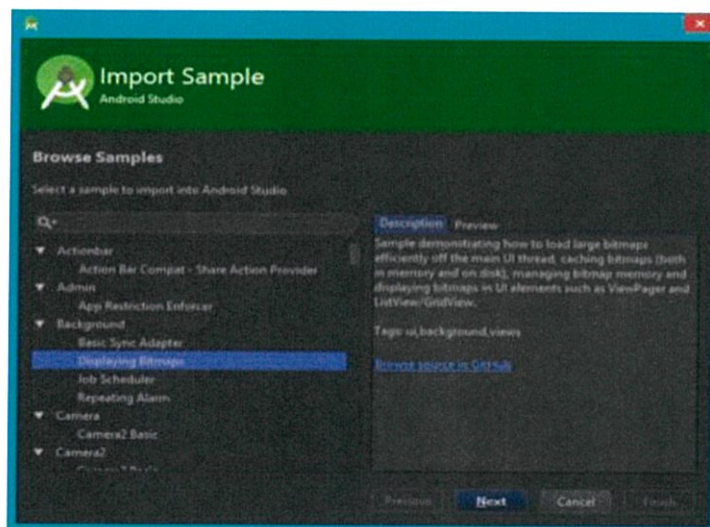
เพื่อให้สามารถ เลือกเปิดได้ทันที

2) Quick Start แถบเมนูเริ่มต้นสำหรับ Android Studio ที่จะให้เลือกว่าจะทำอะไรบน โปรแกรม Android Studio ดังแสดงในรูปที่ 2.14



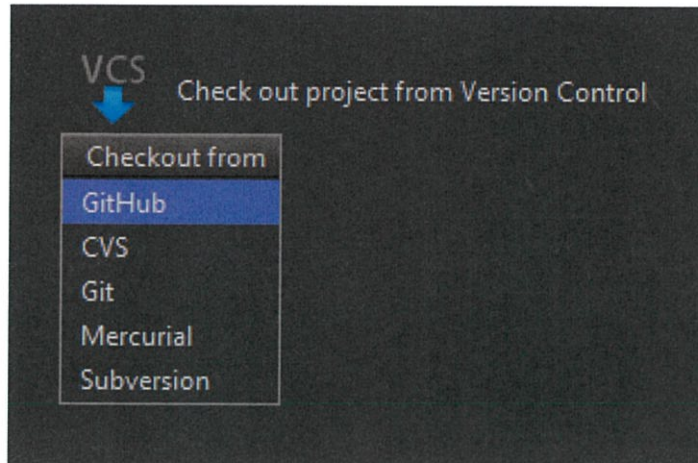
รูปที่ 2.14 แถบเมนูเริ่มต้นสำหรับ Android Studio

3) Start a new Android Studio Project สร้างโปรเจกต์ขึ้นมาใหม่ Open an existing Android Studio project เปิดโปรแกรมที่มีอยู่แล้ว Import an Android code sample ดาวน์โหลดโค้ด ตัวอย่างของแอนดรอยด์จาก GitHub ดังแสดงในรูปที่ 2.15



รูปที่ 2.15 โค้ดตัวอย่างของแอนดรอยด์

4) Check out project from Version Control ดึงโปรเจกต์มาจาก Version Control โดยจะมีให้เลือกว่าจะดึงจาก Version Control แบบไหน ดังแสดงในรูปที่ 2.16



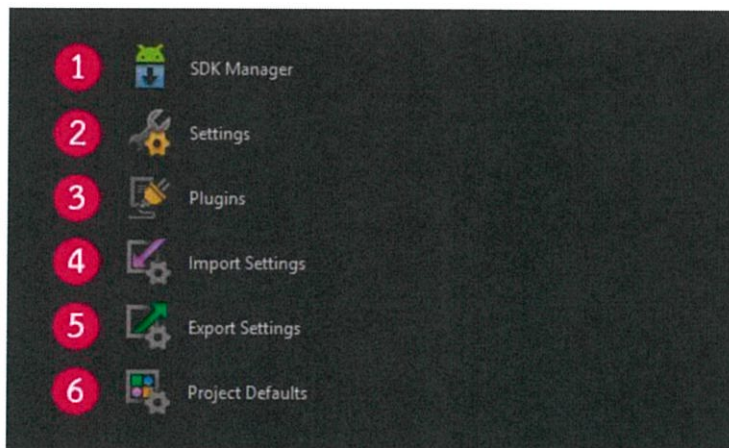
รูปที่ 2.16 Version Control

5) Import Non-Android Studio project นำโปรเจกต์จาก Eclipse ADT เข้ามาใน Android Studio

6) Configure การตั้งค่าต่าง ๆ

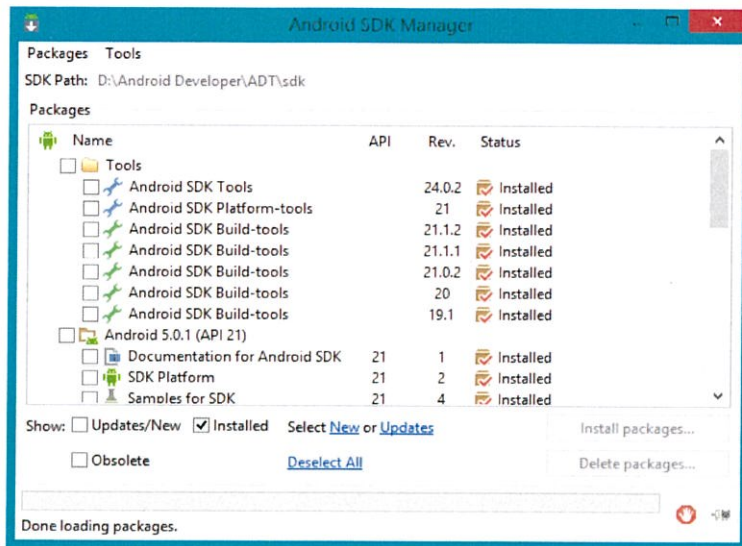
7) Docs and How-Tos เอกสารข้อมูลต่าง ๆ สำหรับการใช้งาน ดังแสดงในรูปที่

2.17



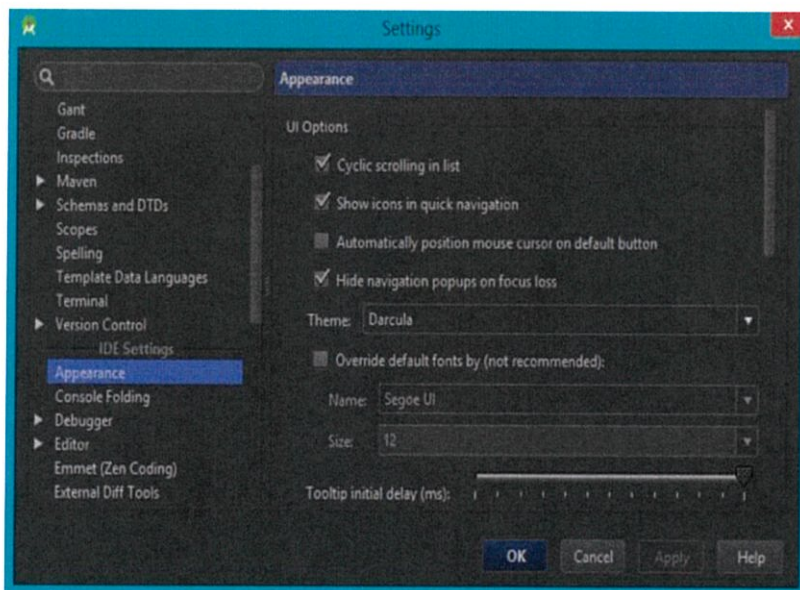
รูปที่ 2.17 เมนู Configure

8) แถบเมนู SDK Manager เปิด Android SDK Manager ดังแสดงในรูปที่ 2.18



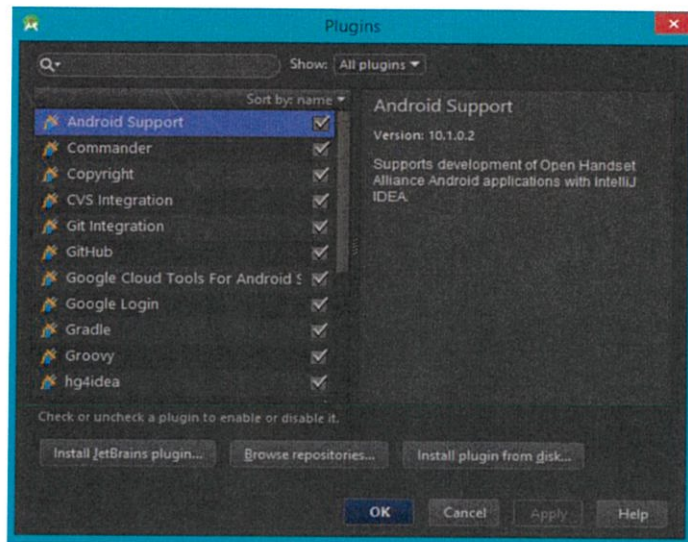
รูปที่ 2.18 Android SDK Manager

9) แถบเมนู Settings เปิดหน้าต่างตั้งค่า Android Studio ดังแสดงในรูปที่ 2.19



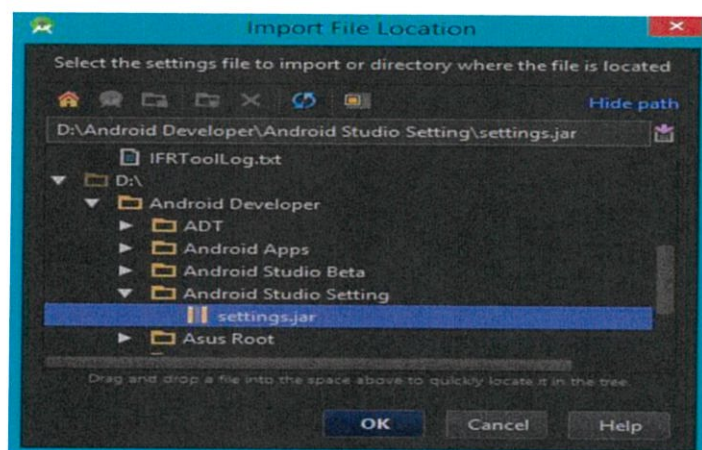
รูปที่ 2.19 หน้าต่างตั้งค่า Android Studio

10) แถบเมนู Plugins เปิดหน้าต่างจัดการกับ Plugins ที่ติดตั้งไว้ใน IntelliJ IDEA ดังแสดงในรูปที่ 2.20



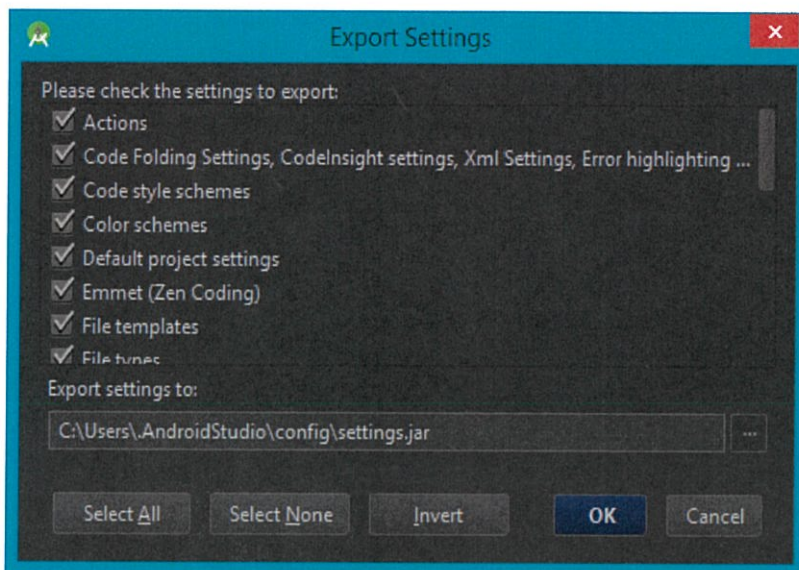
รูปที่ 2.20 หน้าต่าง Plugins

11) Import Settings เนื่องจาก Android Studio สามารถบันทึกการตั้งค่าต่างๆ ในโปรแกรมได้เยอะมาก จึงสามารถเก็บเป็นไฟล์เพื่อไปกำหนดค่าบนเครื่องอื่นๆ ได้ ดังแสดงในรูปที่ 2.21



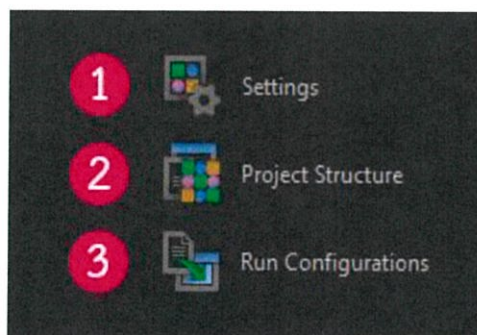
รูปที่ 2.21 หน้าต่าง Import File Location

12) Export Settings ทำการ Export การตั้งค่าต่าง ๆ ใน Android Studio เป็นไฟล์ .jar ดังแสดงในรูปที่ 2.22



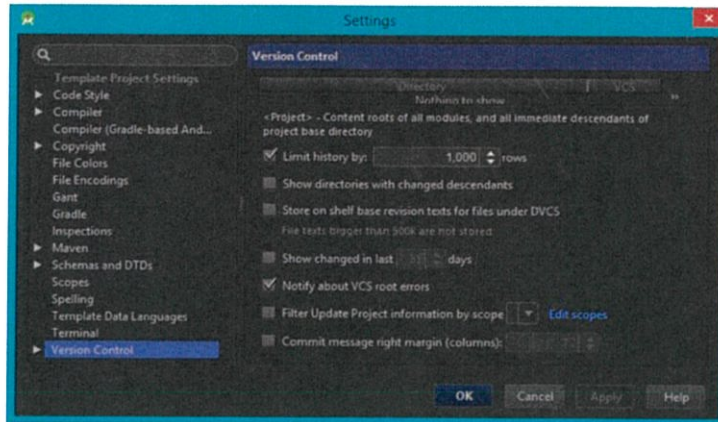
รูปที่ 2.22 หน้าต่าง Export Settings

13) Project Defaults การตั้งค่าเกี่ยวกับโปรเจกต์โดยที่ Project Defaults จะมีการเมนูย่อยอีกได้แก่ Settings, Project Structure และ Run Configuration ดังแสดงในรูปที่ 2.23



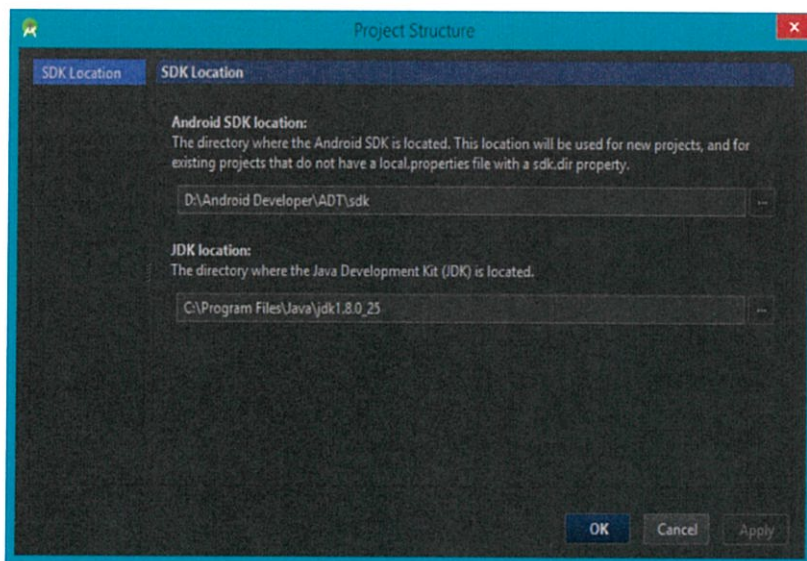
รูปที่ 2.23 เมนูย่อยของหน้าต่าง Project Defaults

Settings เป็นการเปิดหน้าต่าง Settings ที่จะเลือกไปที่ Version Control ให้โดยอัตโนมัติ ดังแสดงในรูปที่ 2.24



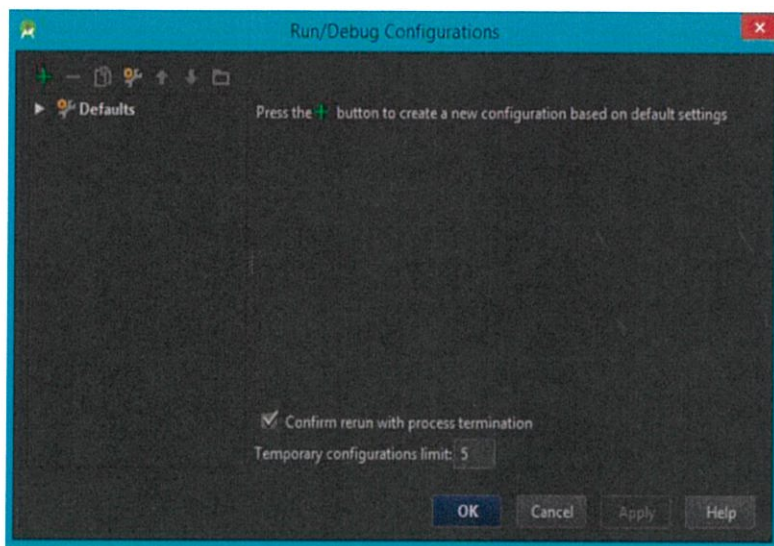
รูปที่ 2.24 Version Control

Project Structure เป็นการเปิดหน้าต่างตั้งค่าสำหรับโปรเจกต์นั้น ๆ เช่น Path ของ Android SDK หรือ Path ของ JDK ดังแสดงในรูปที่ 2.25



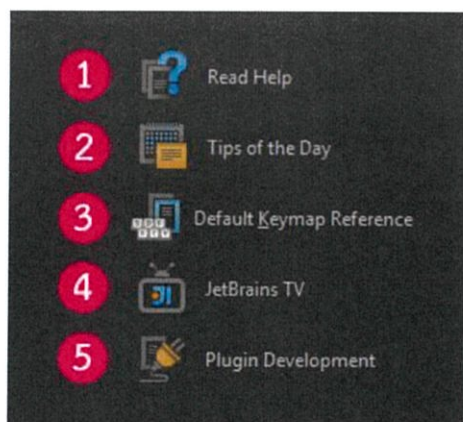
รูปที่ 2.25 หน้าต่าง Project Structure

Run Configuration ตั้งค่าการ Run หรือ Debug โปรเจกต์ ดังแสดงในรูปที่ 2.26



รูปที่ 2.26 หน้าต่าง Run/Debug

14) เมื่อย่อยของ Docs na How-Tos มีดังนี้ Read Help, Tips of the Day, Default Keymap Reference, JetBrains TV และ Plugin Development ดังแสดงในรูปที่ 2.27



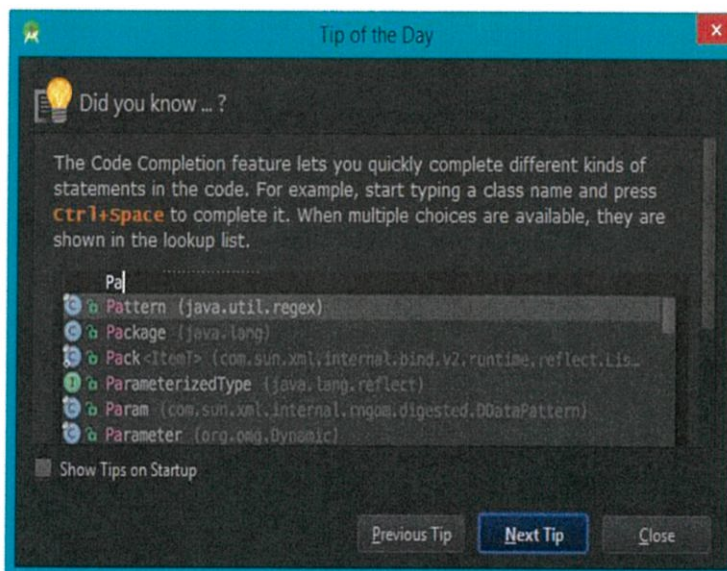
รูปที่ 2.27 เมื่อย่อยของ Docs na How-Tos

Read Help เปิดหน้า Help บนเว็บของ IntelliJ IDEA ดังแสดงในรูปที่ 2.28



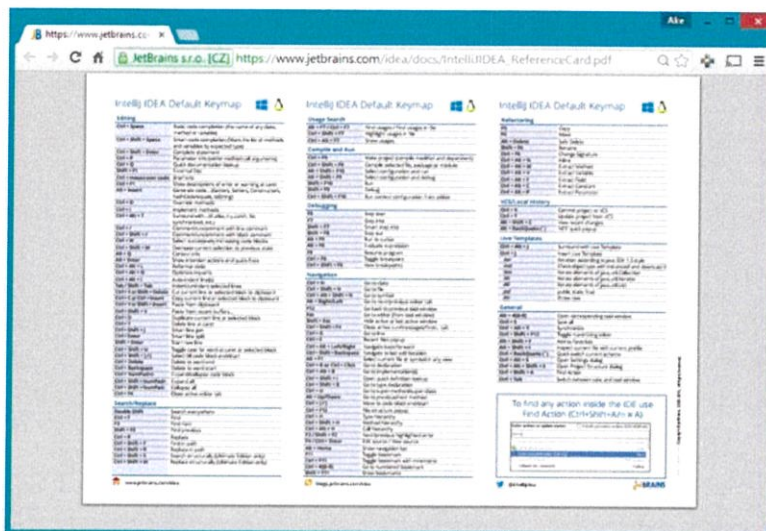
รูปที่ 2.28 หน้าต่าง Help บนเว็บของ IntelliJ IDEA

Tips of the Day เคล็ดลับและเทคนิครายวันเกี่ยวกับการใช้งาน IntelliJ IDEA ดังแสดงในรูปที่ 2.29



รูปที่ 2.29 หน้าต่าง Tip of the Day

Default Keymap Reference เปิดหน้าเว็บ IntelliJ IDEA เพื่อแสดงรายชื่อคีย์ลัดทั้งหมด ของ IntelliJ IDEA ดังแสดงในรูปที่ 2.30



รูปที่ 2.30 หน้าเว็บ IntelliJ IDEA

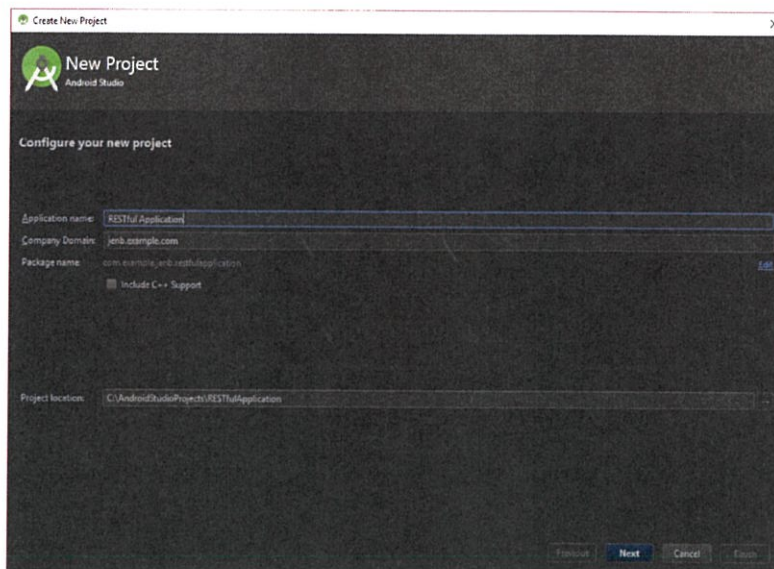
2.3.5 การใช้งาน Android Studio แบบพื้นฐาน

Activity คือการแสดงผลแต่ละหน้าของ Application โดยในแต่ละ Activity สามารถจัด องค์ประกอบต่างๆ ได้ตามต้องการ เพื่อทำการสร้าง UI ตามที่ได้ออกแบบไว้ ดังนั้น การจัดองค์ประกอบในแต่ละ หน้าให้กับ Application จึงต้องทำผ่าน class ที่เป็น Activity

1) เมื่อเปิดโปรแกรมแล้ว ให้คลิกที่ Stat a new Android Studio project จากนั้น จะมีข้อมูลของ Application ให้กรอกดังนี้

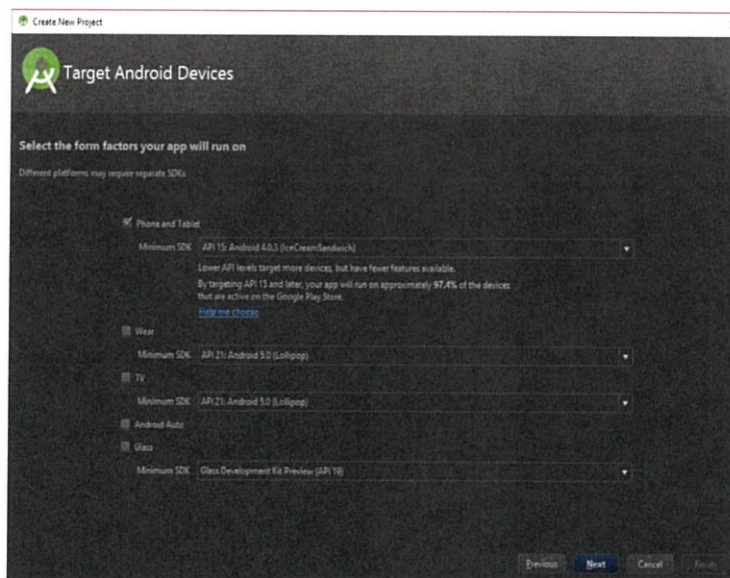
- Application Name จากตัวอย่างในรูปจะใช้ชื่อว่า RESTful Application
- Company Domain คือ โดเมนเนมของหน่วยงานเรา ถ้าไม่มี ให้ใช้ค่า default ซึ่งโดเมนนี้ จะเป็นชื่อของ package ของโปรเจคเรา
- Project Location คือ ตำแหน่งที่ใช้จัดเก็บโปรเจคต์ของเราดังแสดงในรูปที่

2.31



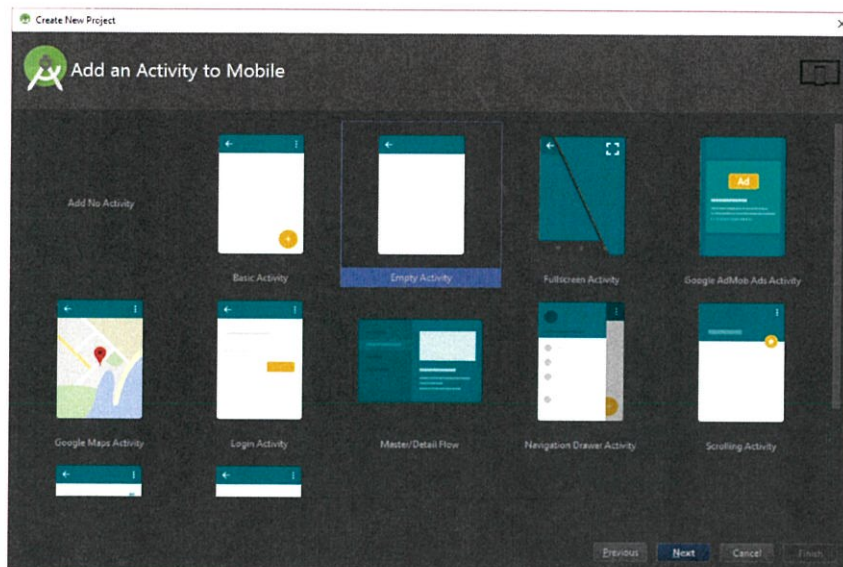
รูปที่ 2.31 กรอกข้อมูลของแอปพลิเคชัน

2) ต่อไปจะเป็นการเลือก version ของ Android ต่ำสุดที่จะรองรับ Application อาจตั้งค่าเป็นค่า default ก่อนแล้วค่อยตั้งค่าเปลี่ยนแปลงทีหลังได้ ดังแสดงในรูปที่ 2.32



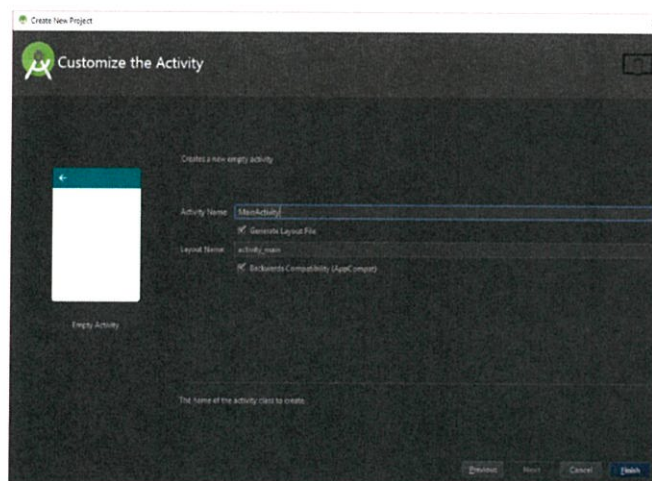
รูปที่ 2.32 เลือก version ของ Android

3) สำหรับ Android Studio จะมี Activity หลายรูปแบบให้เลือกใช้ จากรูป เรา จะทำการเลือก Empty Activity ดังแสดงในรูปที่ 2.33



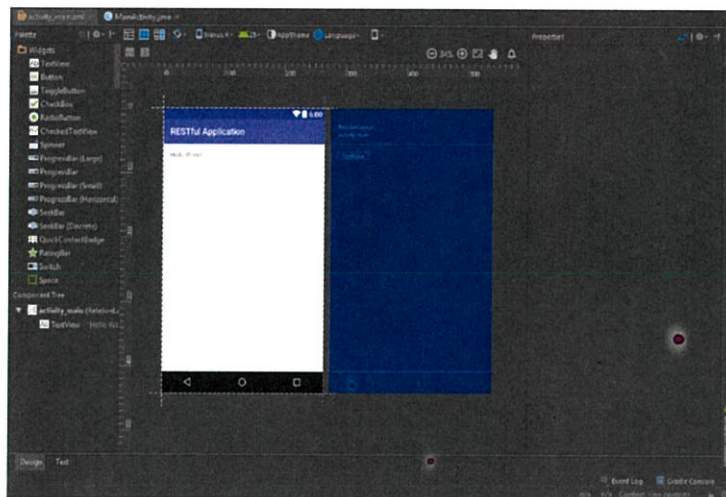
รูปที่ 2.33 เลือก Activity

4) ตั้งชื่อ Activity และ Layout จากรูป เราจะตั้งชื่อ Activity ว่า MainActivity และตั้งชื่อ Layout ว่า activity_main จากนั้นกด Finish ดังแสดงในรูปที่ 2.34



รูปที่ 2.34 ตั้งชื่อ Activity และ Layout

5) เมื่อทำการสร้างโปรเจคเสร็จเรียบร้อยแล้ว จะพบว่า มี class ที่ชื่อ MainActivity.java อยู่ใน โฟลเดอร์ชื่อ java และ Layout ที่ชื่อ activity_main.xml อยู่ใน folder ชื่อ res>layout ตามที่เราได้ตั้งค่าไว้ ซึ่งเราจะใช้ MainActivity.java สำหรับสร้างการตอบสนองการใช้งานองค์ประกอบต่างๆบนหน้า application ดังแสดงในรูปที่ 2.35



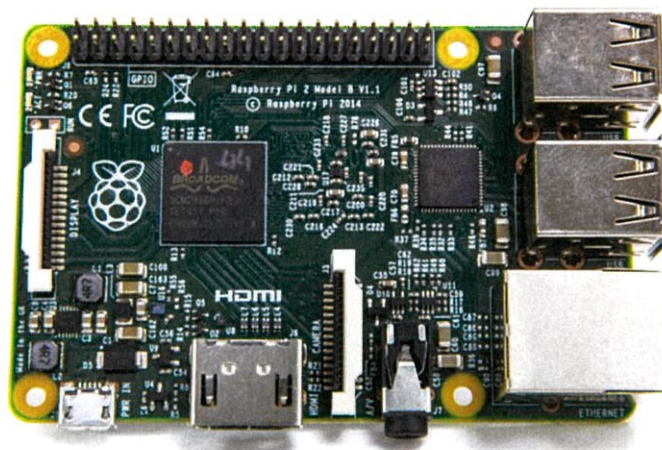
รูปที่ 2.35 MainActivity.java สำหรับสร้างการตอบสนองการใช้งาน
องค์ประกอบต่างๆบนหน้า application

6) ส่วนของ Layout เราจะใช้เพื่อจัดองค์ประกอบบนหน้า application ซึ่งสามารถมองในมุมมองได้ 2 รูปแบบ ได้แก่ มุมมอง Design ที่เราสามารถลากองค์ประกอบต่างๆจาก Palette แล้วลาก มาไว้บนหน้า application ได้

2.4 อุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi) สำหรับการสร้างเซิร์ฟเวอร์ (Server)

Raspberry Pi เป็นเครื่องคอมพิวเตอร์ขนาดเล็ก ประมาณบัตรเครดิต ออกแบบโดย Raspberry Pi Foundation สำหรับใช้ในการทดลองและการศึกษา มีพอร์ตหรือช่องต่ออุปกรณ์ เช่น HDMI, USB, Ethernet เป็นต้น สามารถนำไปใช้งานได้หลากหลาย เช่น ใช้เป็นศูนย์รวมเพลงและวิดีโอ ทั้งแบบออนไลน์และออฟไลน์ ต่อเข้ากับทีวีทางพอร์ต HDMI ก็สามารถใช้งานได้คล้ายๆกับกล่องแอนดรอยด์ หรือใช้เป็น Microcontroller เพื่อควบคุมวงจรอิเล็กทรอนิกส์ เช่น ปิดหรือเปิดไฟผ่านอินเทอร์เน็ต เป็นต้น รวมถึงใช้เป็นเครื่องเซิร์ฟเวอร์ เป็นได้ทั้ง Storage Server, FTP Server, Web Server เป็นต้น โดยที่เราสามารถเชื่อมต่อผ่านอินเทอร์เน็ตที่ใดก็ได้

อุปกรณ์ Raspberry Pi ถือได้ว่าเป็นอุปกรณ์ที่ใช้งานง่าย ประหยัด ทั้งมีขนาดเล็กและราคาถูก ซึ่งทำงานได้หลากหลาย จึงเหมาะที่จะใช้เป็นเครื่องมือในการทดลอง สำหรับ Raspberry Pi ที่ผู้วิจัยใช้สร้าง Web Server คือ Raspberry Pi 2 Model B ดังแสดงในรูปที่ 2.36

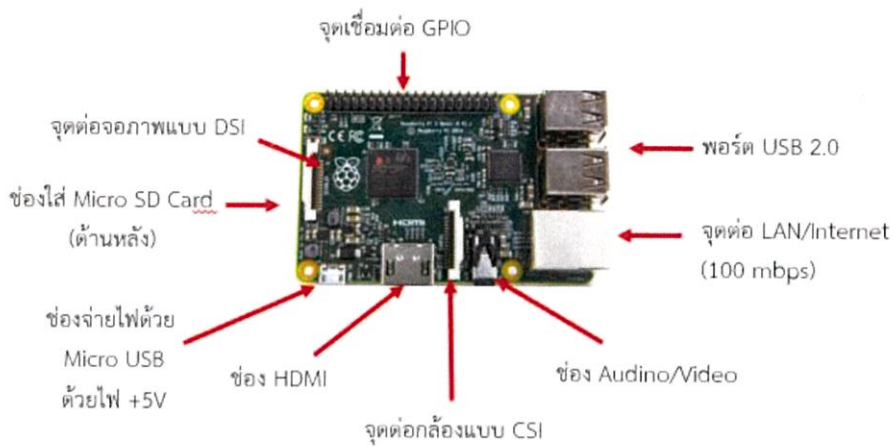


รูปที่ 2.36 รัสเบอร์รี่พาย 2 (Raspberry Pi 2)

2.4.1 สเปกของ Raspberry Pi 2 Model B

- 1) หน่วยประมวลผล (CPU) : Broadcom BCM2836 900 MHz, Quad-core
- 2) กราฟฟิก (GPU) : Broadcom VideoCore IV
- 3) แรม (RAM) : 1 GB
- 4) หน่วยความจำ (Storage) : 64 GB SD Card (ขึ้นอยู่กับ SD card ที่ใช้)
- 5) Video Output : HDMI, Composite RCA
- 6) Audio Output : YES
- 7) Ethernet : YES
- 8) USB 2.0 : YES, 4 Ports
- 9) ระบบปฏิบัติการ (OS) : Windows/Linux
- 10) ขนาด : 8.6×5.4×1.7 ซม.

2.4.2 ส่วนประกอบของ Raspberry Pi



Pi Model B+		Pi Model B+	
3V3 Power	1	2	5V Power
GPIO2 (M2_1)	3	4	5V Power
GPIO3 (M2_0)	5	6	Ground
GPIO4	7	8	GPIO14 (UART_TXD)
Ground	9	10	GPIO15 (UART_RXD)
GPIO17	11	12	GPIO18 (I2C_1)
GPIO27	13	14	Ground
GPIO22	15	16	GPIO23
3V3 Power	17	18	GPIO24
GPIO10 (SPI_MISO)	19	20	Ground
GPIO9 (SPI_MOSI)	21	22	GPIO25
GPIO11 (SPI_CS)	23	24	GPIO8 (SPI_CS_0)
Ground	25	26	GPIO7 (SPI_CS_1)
ID_SD (IO_1 EXPAN)	27	28	ID_SC (IO_5 EXPAN)
GPIO5	29	30	Ground
GPIO6	31	32	GPIO12
GPIO13	33	34	Ground
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
Ground	39	40	GPIO21

รูปที่ 2.37 ส่วนประกอบของ Raspberry Pi

2.4.3 อุปกรณ์ที่ต้องใช้ในการเริ่มต้นกับ Raspberry Pi

อุปกรณ์ที่จำเป็น

- 1) USB Power Adapter (5V, 1A)
- 2) สาย Micro USB
- 3) Micro SD Card (อย่างน้อย 4 GB, Class 4)

- 4) สาย HDMI
- 5) สาย Ethernet หรือ WIFI USB dongle
- 6) หน้าจอที่พอร์ต HDMI
- 7) USB Mouse, USB Keyboard
- 8) คอมพิวเตอร์
- 9) Card Reader หรือ SD card Adapter

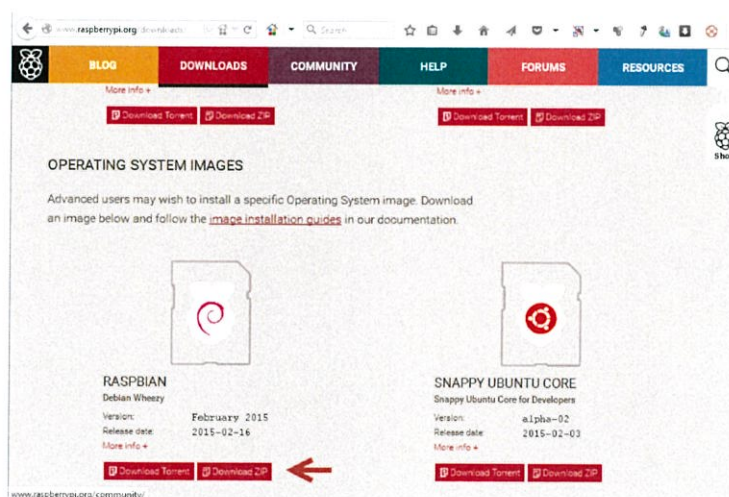
สำหรับผู้ที่ต้องการต่อยอด หรือทำให้ Raspberry Pi มีความสามารถเพิ่มขึ้นไปอีก ก็สามารถหาอุปกรณ์เสริมเหล่านี้มาใช้ได้

อุปกรณ์เสริม

- 1) Case
- 2) Heatsink
- 3) Camera Module
- 4) LCD Display
- 5) Real Time Clock

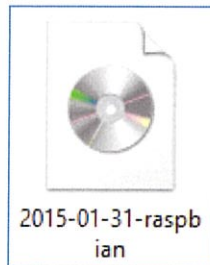
2.4.4 ติดตั้งระบบปฏิบัติการลง Micro SD card

1) ดาวน์โหลดไฟล์อิมเมจจาก <http://www.raspberrypi.org/downloads/> แล้วเลือก Raspbian ดังแสดงในรูปที่ 2.38



รูปที่ 2.38 ดาวน์โหลดไฟล์อิมเมจ

2) เมื่อดาวโหลดเสร็จเรียบร้อยแล้ว ให้ทำการแตกไฟล์ จะได้ไฟล์อิมเมจ (.img) ดังแสดงในรูปที่ 2.39



รูปที่ 2.39 ไฟล์อิมเมจหลังจากแตกไฟล์แล้ว

3) ดาวโหลดโปรแกรม Win32 DiskImager ดังแสดงในรูปที่ 2.40



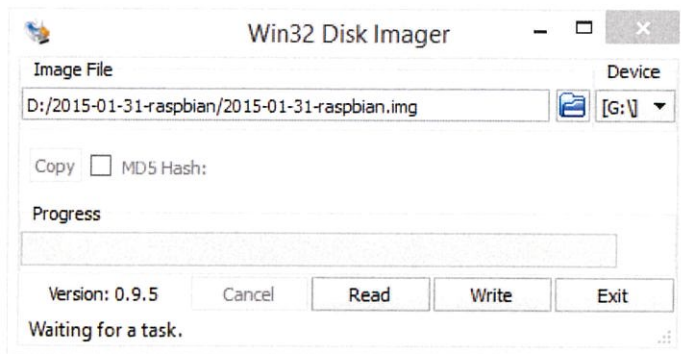
รูปที่ 2.40 ดาวโหลดโปรแกรม Win32 DiskImager

4) แตกซิปไฟล์ และติดตั้งโปรแกรม Win32DiskImager (คลิกขวา Run as administrator)

5) ใส่ SD card เข้ากับเครื่องคอมพิวเตอร์หรือ card reader

6) เปิดโปรแกรม Win32 Disk Imager เลือก Drive เป็นไดรฟ์ของ Micro SD card

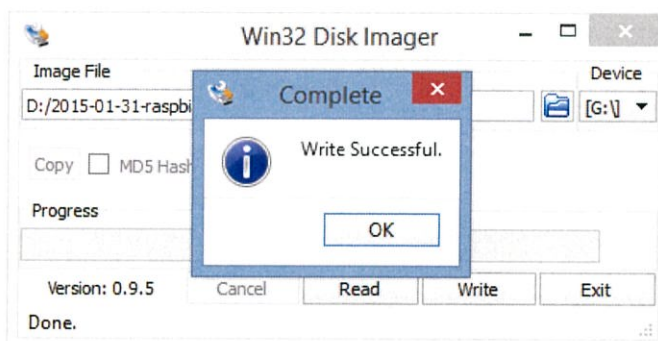
7) ที่ช่อง Image File ให้เลือก OS ที่เราจะทำการติดตั้งใน Micro SD card โดยคลิกที่รูปแผ่นสีน้ำเงิน ดังแสดงในรูปที่ 2.41



รูปที่ 2.41 เลือกไฟล์เพื่อติดตั้ง Image File ใน Micro SD card

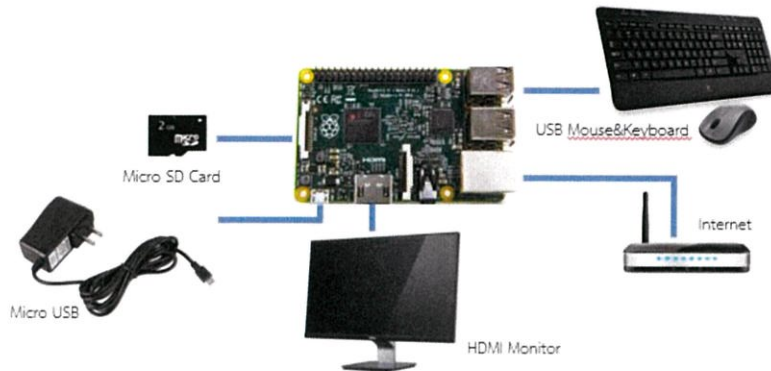
8) ให้เลือกไฟล์อิมเมจของ Raspbian ที่เราดาวน์โหลดไว้

9) กดปุ่ม Write เพื่อติดตั้ง OS ลง Micro SD card แล้วกด YES เพื่อยืนยัน ดังแสดงในรูปที่ 2.42



รูปที่ 2.42 Write Image File ลง Micro SD card

2.4.5 การเปิดใช้งาน Raspberry Pi



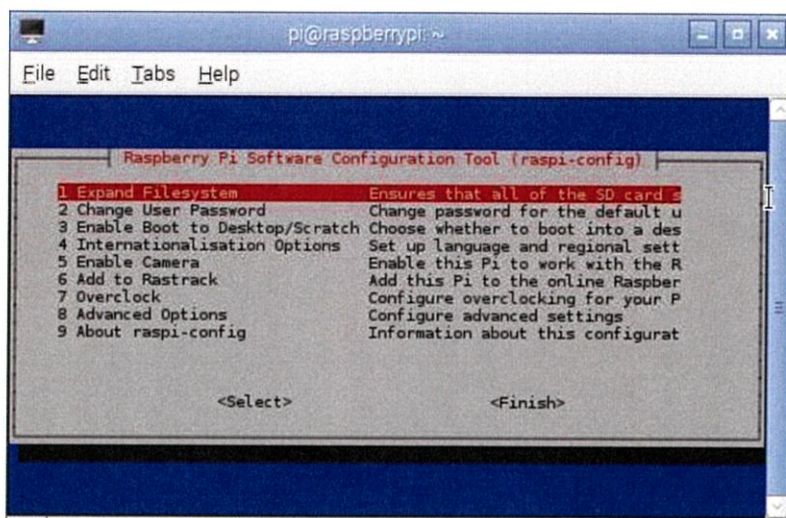
รูปที่ 2.43 การเปิดใช้งาน Raspberry Pi

หลังจากที่เราได้ทำการลงระบบปฏิบัติการลง Micro SD card แล้ว เราก็มาทำขั้นตอนในการเปิดใช้งาน Raspberry Pi กัน ด้วยขั้นตอนดังต่อไปนี้

- 1) ใส่ Micro SD card เข้าไปที่ช่องใส่ ที่ Raspberry Pi (ด้านหลัง)
- 2) เสียบ USB Mouse, USB Keyboard
- 3) เสียบสาย Ethernet หรือ WIFI USB dongle
- 4) เสียบสาย HDMI ให้ Raspberry Pi กับหน้าจอ
- 5) เสียบสาย Micro USB เข้ากับ USB Power Adapter และทำการจ่ายไฟ

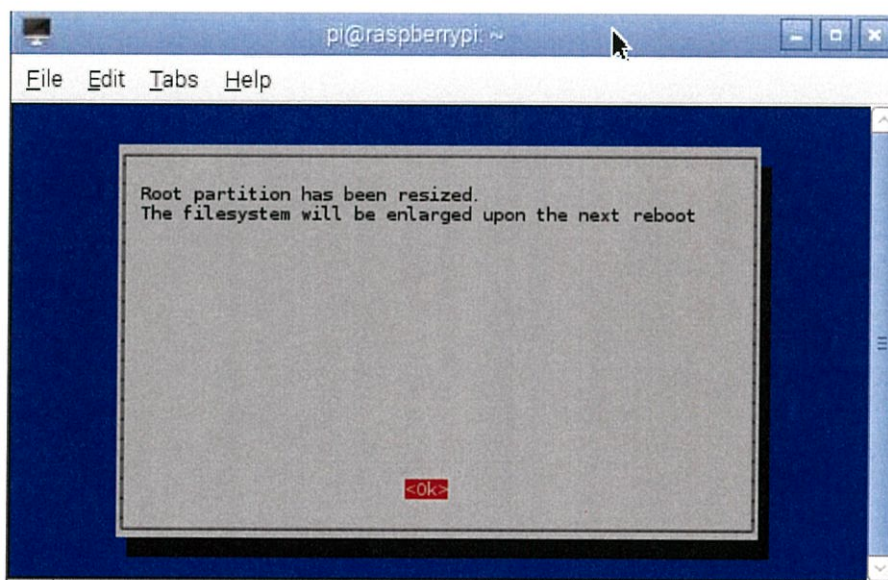
2.4.6 การตั้งค่าและใช้งานเบื้องต้นกับ Raspberry Pi

- 1) เลือก Expand_FileSystem ดังแสดงในรูปที่ 2.44



รูปที่ 2.44 การตั้งค่าและใช้งานเบื้องต้นกับ Raspberry Pi

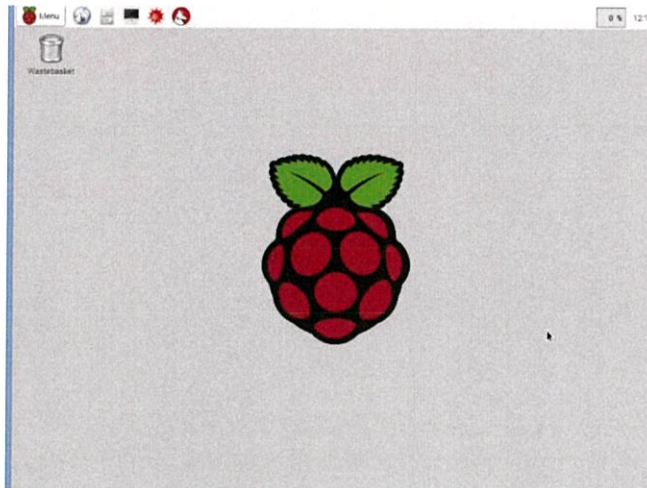
2) เลือก Ok และ ระบบจะทำการ Reboot ดังแสดงในรูปที่ 2.45



รูปที่ 2.45 การตั้งค่าและใช้งานเบื้องต้นกับ Raspberry Pi

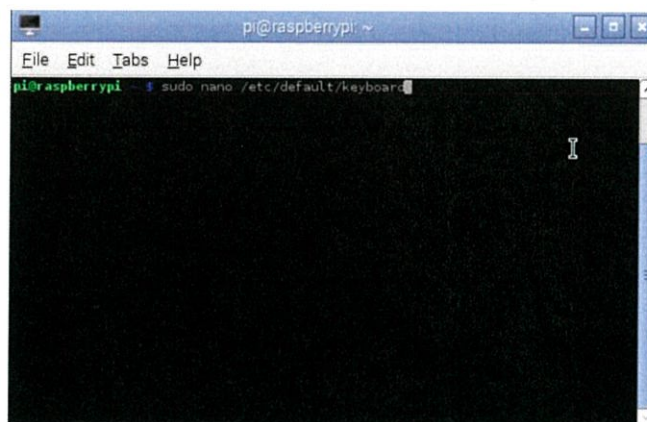
3) ปกติหลังจากที่เราเข้าใช้งาน Raspberry Pi นั้น เราจะต้องเข้ามาที่หน้า Command Line ก่อนทุกครั้ง โดย Raspberry Pi จะถามชื่อผู้ใช้งาน ให้พิมพ์คำว่า pi และพาส

เวิร์ด ให้เราพิมพ์ raspberry โดยทุกครั้งที่เราพิมพ์พาสเวิร์ด ใน Command Line จะไม่แสดงอักขระอะไรขึ้นเลย เมื่อพิมพ์เสร็จแล้วให้กด Enter เพื่อเข้าหน้า Desktop ให้เราพิมพ์คำสั่งว่า startx แล้วกด Enter ดังแสดงในรูปที่ 2.46



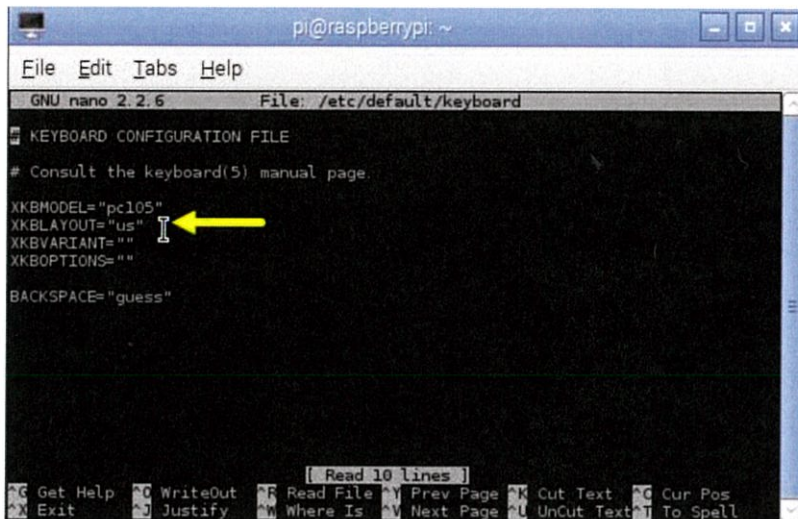
รูปที่ 2.46 หน้า Desktop ของ Raspbian

4) เปลี่ยน keyboard layout เปิด Terminal พิมพ์คำสั่ง `sudo nano /etc/default/keyboard` ดังแสดงในรูปที่ 2.47



รูปที่ 2.47 เปลี่ยน keyboard layout

5) เปลี่ยน XKBLAYOUT="gb" เป็น XKBLAYOUT="us" กด Ctrl-x แล้วกด y เพื่อยืนยัน แล้วกด Enter พิมพ์คำสั่ง Reboot ดังแสดงในรูปที่ 2.48



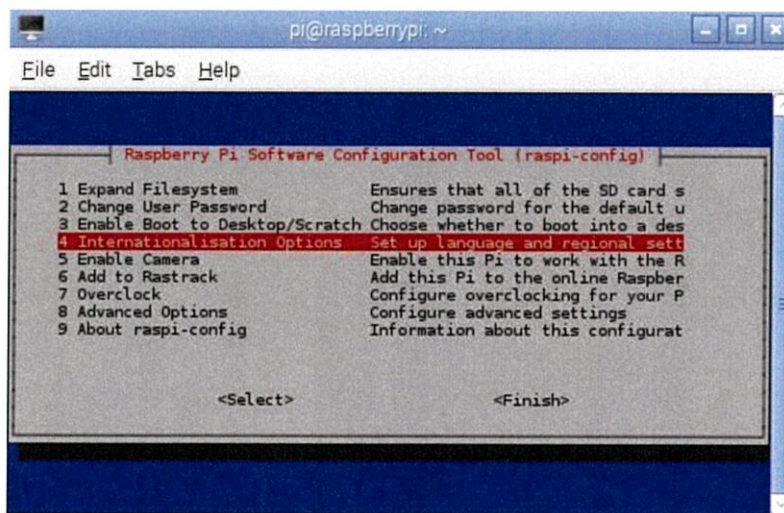
```

pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 2.2.6 File: /etc/default/keyboard
KEYBOARD CONFIGURATION FILE
# Consult the keyboard(5) manual page.
XKBMODEL="pc105"
XKBLAYOUT="us"
XKBVARIANT=""
XKBOPTIONS=""
BACKSPACE="guess"
[ Read 10 lines ]
Get Help WriteOut Read File Prev Page Cut Text Cur Pos
Exit Justify Where Is Next Page UnCut Text To Spell

```

รูปที่ 2.48 เปลี่ยน keyboard layout

6) ตั้งเวลาให้กับ Raspberry Pi เปิด Terminal พิมพ์คำสั่ง raspi-config เลือก 4. Internationalization Option ดังแสดงในรูปที่ 2.49



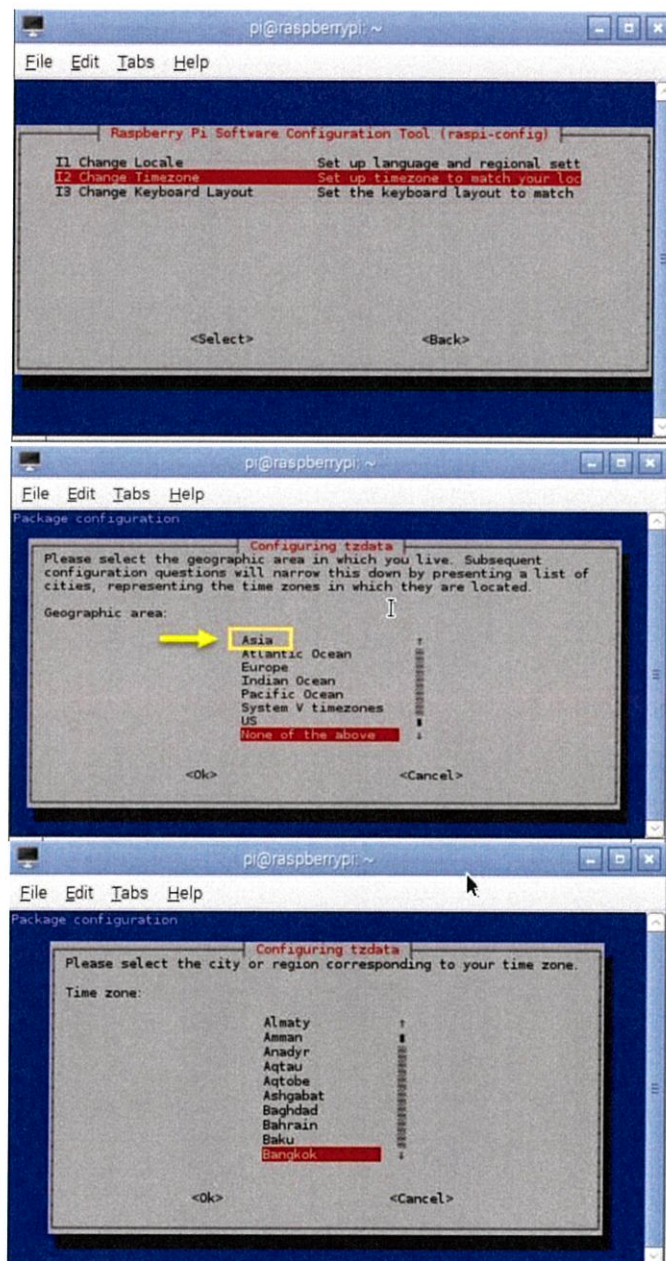
```

pi@raspberrypi: ~
File Edit Tabs Help
Raspberry Pi Software Configuration Tool (raspi-config)
1 Expand Filesystem Ensures that all of the SD card s
2 Change User Password Change password for the default u
3 Enable Boot to Desktop/Scratch Choose whether to boot into a des
4 Internationalisation Options Set up language and regional sett
5 Enable Camera Enable this Pi to work with the R
6 Add to Rastrack Add this Pi to the online Raspber
7 Overclock Configure overclocking for your P
8 Advanced Options Configure advanced settings
9 About raspi-config Information about this configurat
<Select> <Finish>

```

รูปที่ 2.49 ตั้งเวลาให้กับ Raspberry Pi

7) เลือก Chang Timezone เลือก Asia เลือก Bangkok และทำการ reboot ดังแสดงในรูปที่ 2.50



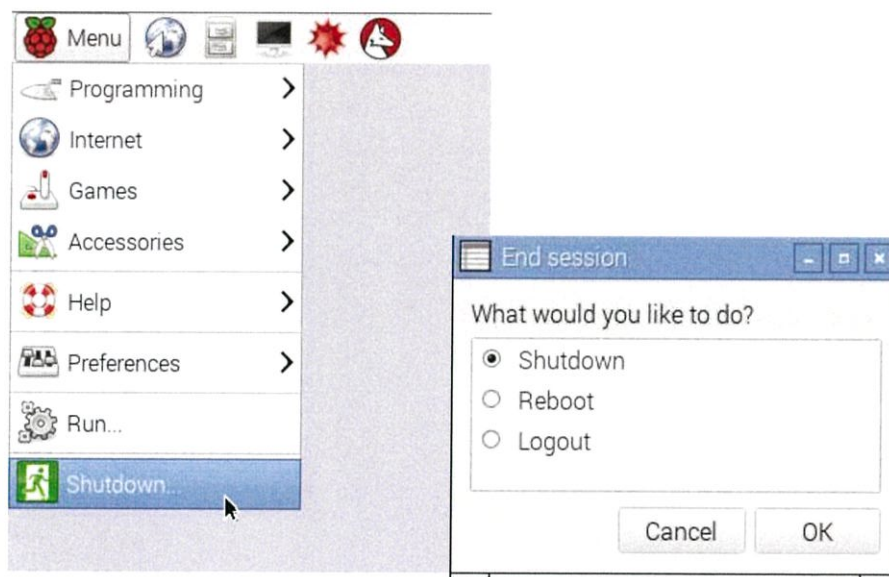
รูปที่ 2.50 ตั้งเวลาให้กับ Raspberry Pi

8) Update ระบบปฏิบัติการ

สำหรับสิ่งที่ควรทำหลังจากที่เราติดตั้งระบบปฏิบัติการให้ทุกครั้ง คือการอัปเดตระบบปฏิบัติการของเรา โดยให้เราไปที่ Terminal พิมพ์คำสั่ง `sudo apt-get update` เพื่อเป็นการสั่งให้ Raspberry Pi ไปทำการเปรียบเทียบไฟล์ของระบบว่ามีไฟล์ไหนที่มีการปรับปรุงล่าสุดหรือไม่ หากมีการก็จะทำการลิสไฟล์เหล่านั้นไว้ เพื่อให้ทราบว่าจะต้องปรับปรุงไฟล์เหล่านั้น เมื่อทำการ Update แล้ว เราต้องทำการ Upgrade ต่อ ซึ่งก็คือการดาวน์โหลดไฟล์เมื่อสักครู่นี้ที่เราเช็คแล้วว่า จะมีการปรับปรุง และเมื่อดาวน์โหลดเสร็จ ก็ทำการติดตั้ง (ข้อควรระวัง ! ขั้นตอนนี้อาจจะใช้เวลานาน) ด้วยคำสั่ง `sudo apt-get upgrade`

9) การปิด (Shut Down) Raspberry Pi อย่างถูกต้อง

Raspberry Pi นั้น ไม่มีปุ่มเปิด/ปิด ดังนั้นเราจึงควรมาเรียนรู้วิธีการปิดหรือ Shutdown อย่างถูกต้องกัน เพื่อไม่ให้ระบบปฏิบัติการของเรามีปัญหาภายหลัง กรณีอยู่ที่หน้า Desktop ไปที่ Menu > Shutdown > Shutdown ดังแสดงในรูปที่ 2.51



รูปที่ 2.51 การปิด (Shut Down) Raspberry Pi กรณีอยู่ที่หน้า Desktop

กรณีอยู่ที่หน้า Terminal พิมพ์คำสั่ง `sudo apt-get -h now`

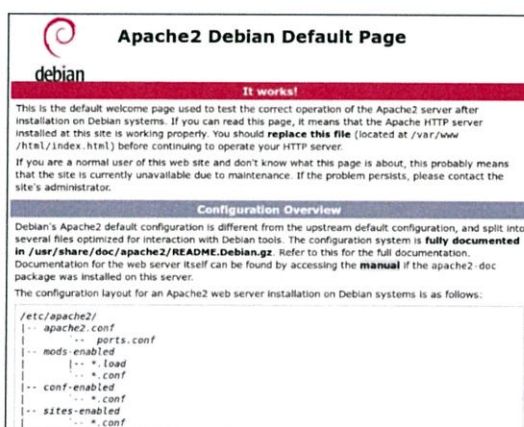
2.5 การติดตั้ง Apache บนอุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi) สำหรับสร้างเซิร์ฟเวอร์ (server)

1) ทำการติดตั้ง Apache ด้วยคำสั่ง `sudo apt-get install apache2 -y` (-y คือติดตั้งแพคเกจทันทีโดยไม่ต้องถาม) ดังแสดงในรูปที่ 2.52

```
pi@raspberrypi ~ $ sudo apt-get install apache2 -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  apache2-mpm-worker apache2-utils apache2.2-bin apache2.2-common libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap ssl-cert
Suggested packages:
  apache2-doc apache2-suexec apache2-suexec-custom openssl-blacklist
The following NEW packages will be installed:
  apache2 apache2-mpm-worker apache2-utils apache2.2-bin apache2.2-common libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 113 not upgraded.
Need to get 1,355 kB of archives.
After this operation, 4,929 kB of additional disk space will be used.
```

รูปที่ 2.52 การติดตั้ง Apache ด้วยคำสั่ง `sudo apt-get install apache2`

2) เมื่อติดตั้งเสร็จแล้วให้ลองเปิด Web Browser แล้วเข้าไปที่ IP Address ของ RaspberryPi เช่น 192.168.1.37 หรือถ้าเปิด Web Browser บน Raspberry Pi ก็ให้เข้าไปที่ `http://localhost` ก็ได้เช่นกัน จะเห็นว่ามีข้อความแสดงขึ้นต้นว่า It works! หมายความว่า Apache สามารถทำงานได้ปกติ ดังแสดงในรูปที่ 2.53



รูปที่ 2.53 Web Browser บน Raspberry Pi apache2

3) ทำการติดตั้ง PHP ด้วยคำสั่ง `sudo apt-get install php5 libapache2-mod-php5 -y` ดังแสดงในรูปที่ 2.54

```
pi@raspberrypi ~ $ sudo apt-get install php5 libapache2-mod-php5 -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  apache2-mpm-prefork libonig2 libqdbm14 lsof php5-cli php5-common
Suggested packages:
  php-pear
The following packages will be REMOVED:
  apache2-mpm-worker
The following NEW packages will be installed:
  apache2-mpm-prefork libapache2-mod-php5 libonig2 libqdbm14 lsof php5
  php5-cli php5-common
0 upgraded, 8 newly installed, 1 to remove and 113 not upgraded.
```

รูปที่ 2.54 ติดตั้ง PHP

4) เมื่อติดตั้งเสร็จแล้วก็ลองเข้าไปที่ Directory ที่ใช้เป็น Web Server โดยจะอยู่ที่ `/var/www` ดังนั้นให้พิมพ์คำสั่ง `cd /var/www` เพื่อไปยัง Directory ดังกล่าว ดังแสดงในรูปที่ 2.55

```
pi@raspberrypi ~ $ cd /var/www
```

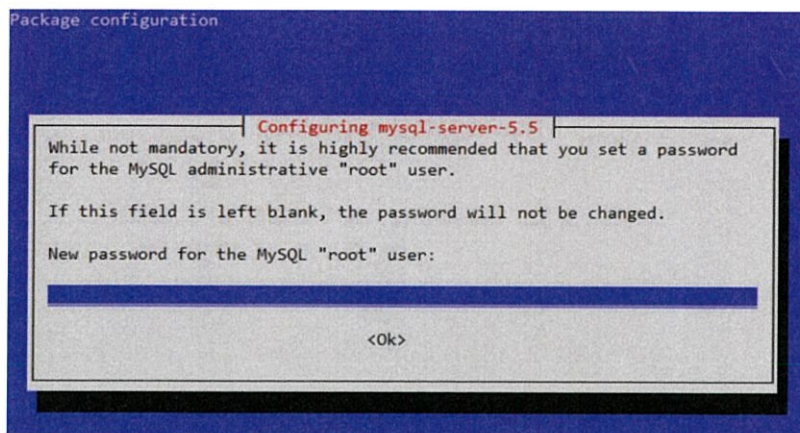
รูปที่ 2.55 พิมพ์คำสั่ง `cd /var/www` เพื่อไปยัง Directory

5) ติดตั้ง MySQL ได้ด้วยคำสั่ง `sudo apt-get install mysql-server php5-mysql -y` โดยจะใช้เวลาในการติดตั้งที่ค่อนข้างนานพอสมควร ดังแสดงในรูปที่ 2.56

```
pi@raspberrypi /var/www $ sudo apt-get install mysql-server php5-mysql -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  heirloom-mailx libaio1 libdbd-mysql-perl libdbi-perl libhtml-template-perl
  libmysqlclient16 libmysqlclient18 mysql-client-5.5 mysql-common
  mysql-server-5.5 mysql-server-core-5.5
Suggested packages:
  exim4 mail-transport-agent libipc-sharedcache-perl libterm-readkey-perl
  tinyc
Recommended packages:
  mailx
The following NEW packages will be installed:
  heirloom-mailx libaio1 libdbd-mysql-perl libdbi-perl libhtml-template-perl
  libmysqlclient16 libmysqlclient18 mysql-client-5.5 mysql-common mysql-server
  mysql-server-5.5 mysql-server-core-5.5 php5-mysql
0 upgraded, 13 newly installed, 0 to remove and 113 not upgraded.
```

รูปที่ 2.56 ติดตั้ง MySQL ได้ด้วยคำสั่ง `sudo apt-get install`

6) ระหว่างการติดตั้งจะมีการกำหนดรหัสผ่านด้วย โดยให้กำหนดรหัสผ่านสำหรับ MySQL ดังแสดงในรูปที่ 2.57



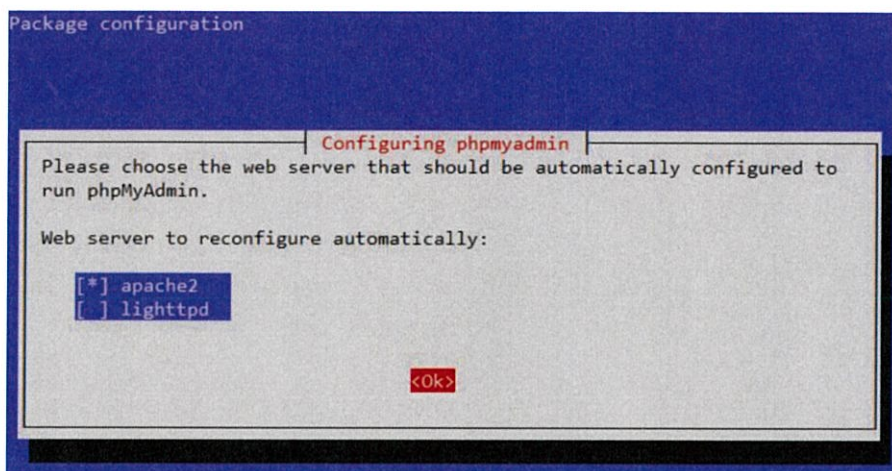
รูปที่ 2.57 กำหนดรหัสผ่านสำหรับ MySQL

7) ติดตั้ง PHPMyAdmin เพื่อเข้าไปจัดการฐานข้อมูลก็ให้ใช้คำสั่ง `sudo apt-get install phpmyadmin -y` ดังแสดงในรูปที่ 2.58

```
pi@raspberrypi /var/www $ sudo apt-get install phpmyadmin -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  dbconfig-common libmcrypt4 php5-gd php5-mcrypt
Suggested packages:
  libmcrypt-dev mcrypt
The following NEW packages will be installed:
  dbconfig-common libmcrypt4 php5-gd php5-mcrypt phpmyadmin
0 upgraded, 5 newly installed, 0 to remove and 113 not upgraded.
Need to get 6,112 kB of archives.
After this operation, 17.2 MB of additional disk space will be used.
```

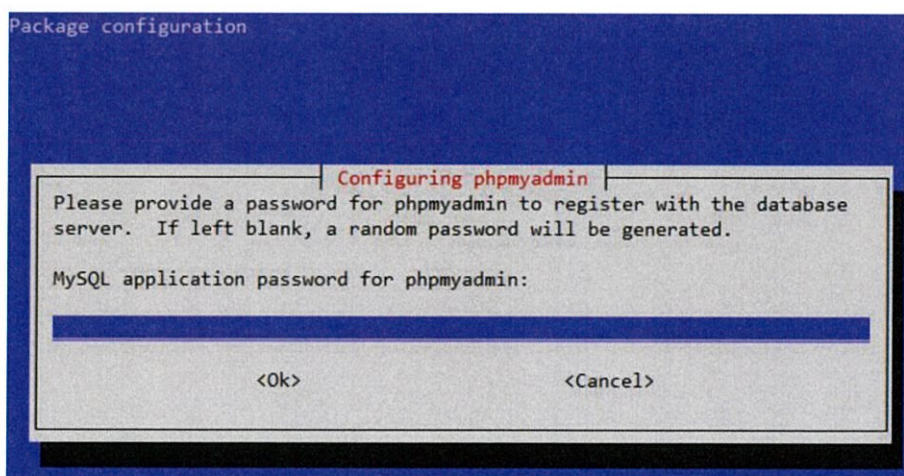
รูปที่ 2.58 ติดตั้ง PHPMyAdmin

8) ระหว่างการติดตั้งจะมีให้กำหนดว่าใช้ Web Server แบบใดอยู่ ให้เลือกเป็น Apache2 แล้วเลือกที่ OK (ใช้ Space Bar เพื่อเลือก และใช้ Tab เพื่อสลับ Cursor ไปมาระหว่างปุ่ม OK) ดังแสดงในรูปที่ 2.59



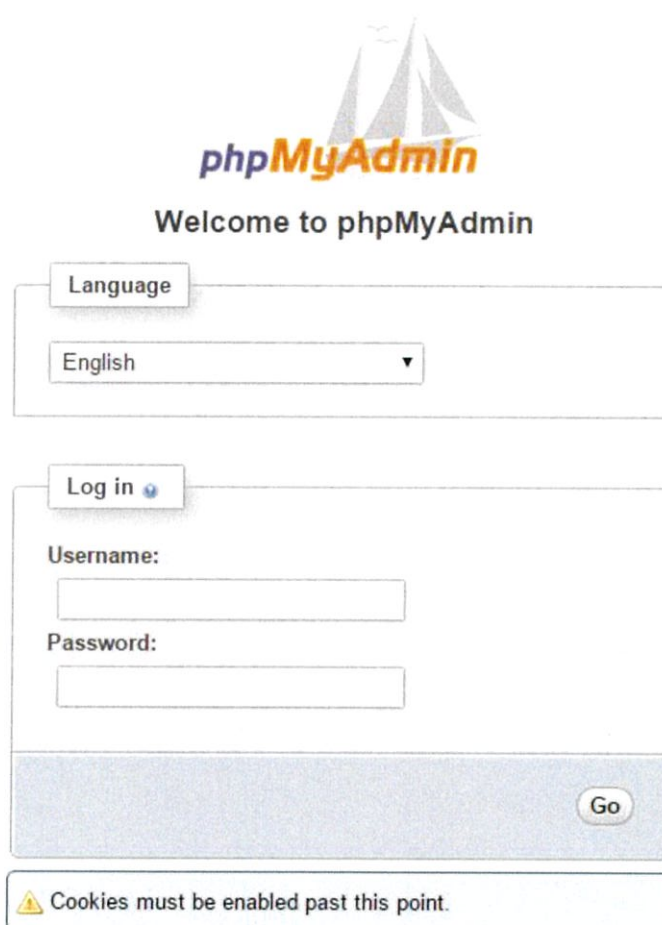
รูปที่ 2.59 กำหนด Web Server เป็น Apache2

9) กำหนดรหัสผ่านสำหรับ Administrator ของฐานข้อมูล ดังแสดงในรูปที่ 2.60



รูปที่ 2.60 กำหนดรหัสผ่านสำหรับ Administrator

10) เมื่อติดตั้งเสร็จแล้วให้ลองเปิด Web Browser ไปที่ PHPMYAdmin ถ้าเปิด Web Browser บน Raspberry Pi ก็จะพบกับหน้า Login ของ PHPMYAdmin สำหรับ Username ให้ใช้ว่า root และ Password ให้ใช้ตามที่กำหนดไว้ตอนติดตั้ง PHPMYAdmin ดังแสดงในรูปที่ 2.61



Language

English

Log in

Username:

Password:

Go

⚠ Cookies must be enabled past this point.

รูปที่ 2.61 เปิด Web Browser ไปที่ PHPMYAdmin

2.6 ภาษาพีเอชพี (php)

PHP คือภาษาโปรแกรมรูปแบบหนึ่งที่ทำงานอยู่บนเซิร์ฟเวอร์ โดยคำสั่งภาษา PHP นั้น จะถูกนำไปประมวลผลในฝั่งเซิร์ฟเวอร์ก่อนและส่งผลลัพธ์กลับมาแสดงผลให้ผู้ใช้งานผ่านทางหน้าจอของเครื่อง Client ที่ร้องขอไฟล์

ภาษา PHP ถูกพัฒนาอย่างต่อเนื่อง มี library และ Framwork ต่างๆ ให้เลือกใช้มากมาย เป็นภาษาที่ได้รับความนิยม มีผู้ใช้เป็นจำนวนมาก เหมาะสำหรับผู้เริ่มต้นในการเขียนโปรแกรม

2.6.1 เครื่องมือสำหรับการพัฒนาโปรแกรม

1) โปรแกรม Text editor เช่น Notepad++, sublime text 3, netbeans เป็นต้น

2) โปรแกรมจำลองเว็บเซิร์ฟเวอร์ เช่น Xampp, Wampp เป็นต้น โดยโปรแกรมจำลองเว็บเซิร์ฟเวอร์นั้น คือ โปรแกรมที่รวมโปรแกรมที่จำเป็นในการพัฒนา เช่น Apache , MySQL, PhpMyadmin และอื่นๆ เพื่อให้สามารถพัฒนาโปรแกรมได้สะดวกยิ่งขึ้น

3) โปรแกรมเว็บเบราว์เซอร์ เช่น Google Chrome หรือ Mozilla Firefox

2.7 HTML

HTML ย่อมาจากคำว่า Hypertext Markup Language เป็นภาษาหลักที่ใช้ในการสร้างไฟล์เว็บเพจ โดยมีแนวคิดจากการสร้างเอกสารไฮเปอร์เท็กซ์ (Hypertext Document) ซึ่งพัฒนาขึ้นมาจากภาษา SGML (Standard Generalized Markup Language) โดย Tim Berners-Lee เป็นภาษามาตรฐานที่ใช้พัฒนาเอกสารในรูปแบบของเว็บเพจเผยแพร่บนระบบเครือข่ายอินเทอร์เน็ต มีโครงสร้างการเขียนที่อาศัยตัวกำกับ เรียกว่า แท็ก (Tag) ควบคุมการแสดงผลของข้อความ, รูปภาพ หรือวัตถุอื่นๆ เรียกใช้เอกสารเหล่านี้โดยการใช้โปรแกรมเว็บเบราว์เซอร์ (Web Browser) เช่น Mozilla Firefox, Opera , Netscape navigator, Internet Explorer เป็นต้น

2.8 CSS

CSS ย่อมาจาก Cascading Style Sheets คือภาษาคอมพิวเตอร์ภาษาหนึ่งที่เป็นภาษาในกลุ่มภาษาสไตลชีต(style sheet) (ภาษาสไตลชีต เป็นภาษาที่มีการใช้งานมานานแล้วในวงการการพิมพ์ โดยภาษาสไตลชีตจะเป็นโครงสร้างเอกสารต้นฉบับที่มีการจัดรูปแบบและตัวอักษรไว้เรียบร้อยแล้ว) ภาษา CSS ใช้ในการจัดรูปแบบและโครงสร้างของเอกสารที่เขียนจากภาษา HTML โดยภาษา CSS สามารถใช้งานได้หลากหลายและมีความยืดหยุ่นสามารถใช้งานกับภาษา XML SVG และ XUL ภาษา CSS มีประโยชน์หลายอย่าง ทำให้การพัฒนาเว็บเพจด้วยภาษา HTML เป็นเรื่องที่ย่างมากขึ้น

1) ภาษา CSS จะช่วยในการจัดรูปแบบแสดงผลให้กับภาษา HTML ซึ่งจะช่วยลดการใช้ HTML ให้น้อยลง โดยเหลือเพียงแต่ส่วนที่เป็นเอกสารที่เป็นภาษา HTML เท่านั้นทำให้มีการแก้ไขและทำความเข้าใจได้ง่ายขึ้น

2) ทำให้ขนาดไฟล์ HTML น้อยลง เนื่องจากภาษา CSS จะช่วยลดการใช้ HTML ลง ทำให้ขนาดไฟล์นั้นก็เล็กลงไปด้วยเช่นกัน

3) ภาษา CSS เป็นภาษา Style Sheets โดย Style Sheets ชุดเดียวสามารถใช้กำหนดรูปแบบการแสดงผลให้เอกสาร HTML ทั้งหน้า หรือทุกหน้ามีผลเหมือนกันได้ จึงทำให้เวลาที่มีการแก้ไขก็จะแก้ไขได้ง่ายขึ้นเพียงแก้ไข Style Sheets ที่ใช้งานเพียงชุดเดียวเท่านั้น

4) ทำให้เว็บไซต์มีมาตรฐานเพราะการใช้งาน CSS นั้นจะทำให้การแสดงผลในสื่อต่างๆถูกปรับเปลี่ยนไปได้เหมาะสม เช่นการแสดงผลบนหน้าจอ และการแสดงผลในมือถือ

5) CSS สามารถที่จะใช้งานได้หลากหลาย เว็บเบราว์เซอร์ ทำให้การใช้งานนั้นสะดวกมากยิ่งขึ้น

2.9 จาวาสคริปต์ (Java Script)

จาวาสคริปต์ (Java Script) เป็นภาษาโปรแกรมเชิงวัตถุ ทำงานได้ข้ามระบบ มีขนาดเล็กใช้ง่ายกินทรัพยากรน้อย ไม่เหมาะที่จะทำงานด้วยตัวเอง แต่ถูกออกแบบมาให้ช่วยต่อการทำงาน

ในแอปพลิเคชันและผลิตภัณฑ์อื่น เช่น ในเว็บเบราว์เซอร์, ในสภาพแวดล้อมของโฮสต์, ทั้งยังสามารถเชื่อมต่อกับอ็อบเจกต์ในสภาพแวดล้อมต่างๆเพื่อควบคุมการทำงานผ่านโปรแกรมได้

ลักษณะที่สำคัญของจาวาสคริปต์ มีดังนี้

- 1) เป็นภาษาประเภท Scripting Language
- 2) ไม่ต้องอาศัยความรู้ในการเขียนโปรแกรมในระดับสูงนัก
- 3) การใช้งานจาวาสคริปต์จะใช้ในลักษณะการฝัง (embedded) ชุดคำสั่ง หรือโค้ดไว้ใน

ในหน้าเว็บ

- 4) จาวาสคริปต์ (Java Script) จะช่วยในส่วนของความเป็น Interactive ต่อผู้ใช้
- 5) เป็นภาษาที่ทำงานโดยอาศัยการแปลคำสั่ง (Interpret) ดังนั้นจึงไม่ต้องทำการคอมไพล์ (Compile) ชุดคำสั่งแต่อย่างใด
- 6) นักพัฒนาสามารถใช้จาวาสคริปต์ (Java Script) ได้ โดยไม่ต้องเสียค่าลิขสิทธิ์ (License)

รูปแบบการใช้งานจาวาสคริปต์

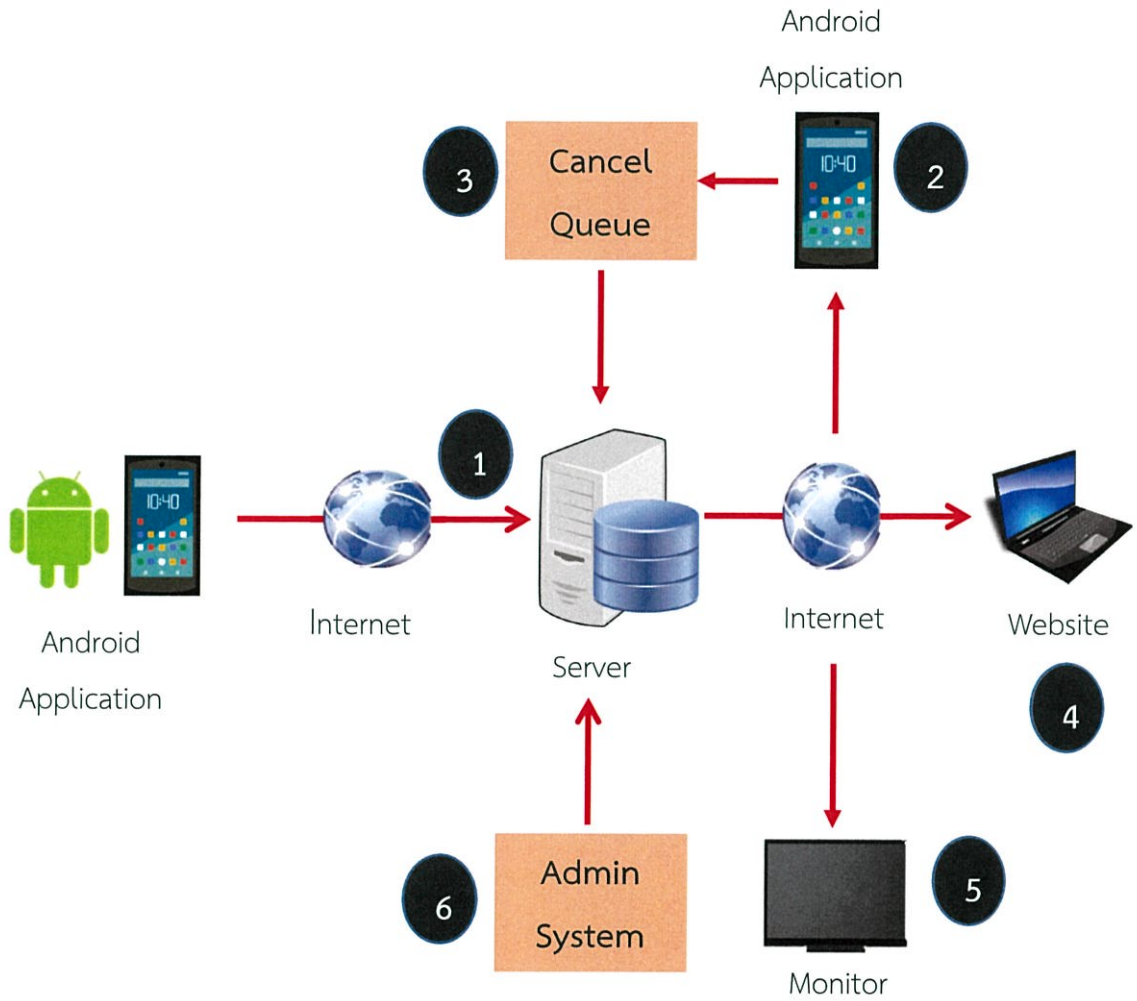
ก่อนจะเขียนชุดคำสั่งด้วยจาวาสคริปต์ (Java Script) ผู้เขียนควรมีความรู้และความเข้าใจว่าจาวาสคริปต์ (Java Script) นั้น สามารถนำไปประยุกต์ใช้ในลักษณะใดได้ ตามรายละเอียดต่อไปนี้

- 1) การใช้งานจาวาสคริปต์ (Java Script) ในลักษณะเครื่องมือของนักออกแบบเว็บไซต์ หรืออาจกล่าวได้ว่าจาวาสคริปต์เป็น Design Tools ที่สำคัญตัวหนึ่ง
- 2) ใช้เพื่อแสดง Dynamic Text บนหน้า HTML Page
- 3) ใช้ในกรณีที่ต้องการให้มีการโต้ตอบ (React) กับเหตุการณ์ (Event) ที่เกิดขึ้น หรือเหตุการณ์ที่ทำกับหน้าเพจดังกล่าว
- 4) ใช้ในการอ่าน (Read) และเขียน (Write) HTML Elements
- 5) ใช้เพื่อการตรวจสอบความถูกต้องของข้อมูลที่ได้รับเข้ามา (Validate)
- 6) ใช้เพื่อการตรวจสอบเบราว์เซอร์ของผู้ใช้
- 7) ใช้ในการสร้างคุกกี้ (Cookies)

บทที่ 3

การออกแบบและจัดทำโครงการ

3.1 การออกแบบ



รูปที่ 3.1 บล็อกไดอะแกรมของโครงการ

บล็อกไดอะแกรมภาพรวมของระบบติดตามรถไฟแสดงดังรูปที่ 3.1 โดยมีหลักการ
ทำงานของระบบจองร้านอาหารโดยผู้ใช้งานทำการจองคิวผ่านแอปพลิเคชัน (Application) สำหรับ
โทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ (Android Application) และส่งข้อมูลที่ได้นั้นเข้าสู่
เซิร์ฟเวอร์ (Server) ผ่านเครือข่ายอินเทอร์เน็ต (Internet) ซึ่งจะทำการประมวลผลข้อมูลสำหรับ
ประมวลผลคิว (หมายเลข 1) จากนั้นเซิร์ฟเวอร์ (Server) จะทำการส่งข้อมูลที่ได้ (คิว) ไปสู่แอป
พลิเคชัน (Application) สำหรับโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์ (Android
Application) ของผู้ใช้งาน (หมายเลข 2) รวมถึงผู้ใช้งานสามารถทำการยกเลิกคิวผ่านแอปลิเค
ชัน (Application) ได้ (หมายเลข 3) ผู้ใช้งานสามารถทราบการแสดงผลคิวปัจจุบันได้จากเว็บไซต์
(หมายเลข 4) และจอมอนิเตอร์หน้าร้าน (หมายเลข 5) รวมทั้งสร้างหน้าเว็บไซต์สำหรับผู้ดูแลระบบ
(หมายเลข 6) ขึ้นมาเพื่อบริหารจัดการคิว และดูแลความผิดพลาดที่เกิดขึ้นต่างๆ ได้อย่างง่ายขึ้น

3.1.1 การออกแบบฐานข้อมูลบนอุปกรณ์ราสเบอร์รี่พาย (raspberry pi)

ฐานข้อมูล (server) ที่ทำการออกแบบจะใช้ apache จัดทำเป็น web server (ขั้นตอนการติดตั้งกล่าวไว้ในบทที่ 2) สำหรับร้านอาหาร จะออกแบบให้มีหนึ่งฐานข้อมูล โดยหนึ่งเทเบิล (table) คือ หนึ่งสาขา และแต่ละเทเบิล (table) ประกอบไปด้วยจำนวนฟิลด์ (field) 13 ฟิลด์ (field) ได้แก่

- | | |
|------------------------------|---|
| 1) id (int 10) | คือ ลำดับข้อมูล |
| 2) time_in (timestamp) | คือ เวลาที่ลูกค้าจองคิวผ่านแอปพลิเคชัน (Application) |
| 3) time_on_serve (timestamp) | คือ เวลาที่ลูกค้าถูกเรียกคิว |
| 4) name (varchar 20) | คือ ชื่อผู้จอง |
| 5) people (int 5) | คือ จำนวนคนที่รับประทานทั้งหมด |
| 6) adult (int 3) | คือ จำนวนคนที่เป็นผู้ใหญ่ |
| 7) child (int 3) | คือ จำนวนคนที่เป็นเด็ก |
| 8) type (varchar 2) | คือ ประเภทของโต๊ะ โดยแบ่งเป็น 4 ประเภท ได้แก่
type "A" หมายถึง โต๊ะประเภท 1 - 2 ท่าน
type "B" หมายถึง โต๊ะประเภท 3 - 4 ท่าน
type "C" หมายถึง โต๊ะประเภท 5 - 6 ท่าน
type "D" หมายถึง โต๊ะประเภท 7 ท่านขึ้นไป |
| 9) sequence (int 5) | คือ หมายเลขลำดับคิวที่ได้ |
| 10) remenning_queue (int 3) | คือ จำนวนที่เหลือก่อนจะถึงคิวลูกค้า |
| 11) remenning_time (text) | คือ ระยะเวลารอคิวโดยประมาณ |

- 9) queue (varchar 5) คือ หมายถึงคิวที่ได้ เช่น คิว A1 ,B10 เป็นต้น กล่าวคือ เป็นการรวมฟิลด์ type และ sequence เข้าด้วยกันเพื่อให้แอดมินดูข้อมูลได้ง่าย
- 10) status (int 2) คือ สถานะของลูกค้า โดยแบ่งเป็น 2 ประเภท ได้แก่ status = “0” คือ ลูกค้าที่รอเรียกคิว status = “1” คือ ลูกค้าที่ถูกเรียกคิวแล้ว status = “2” คือ ลูกค้าที่ทำการยกเลิกคิว

รูปแบบฐานข้อมูล (database) เป็นดังแสดงในรูปที่ 3.2

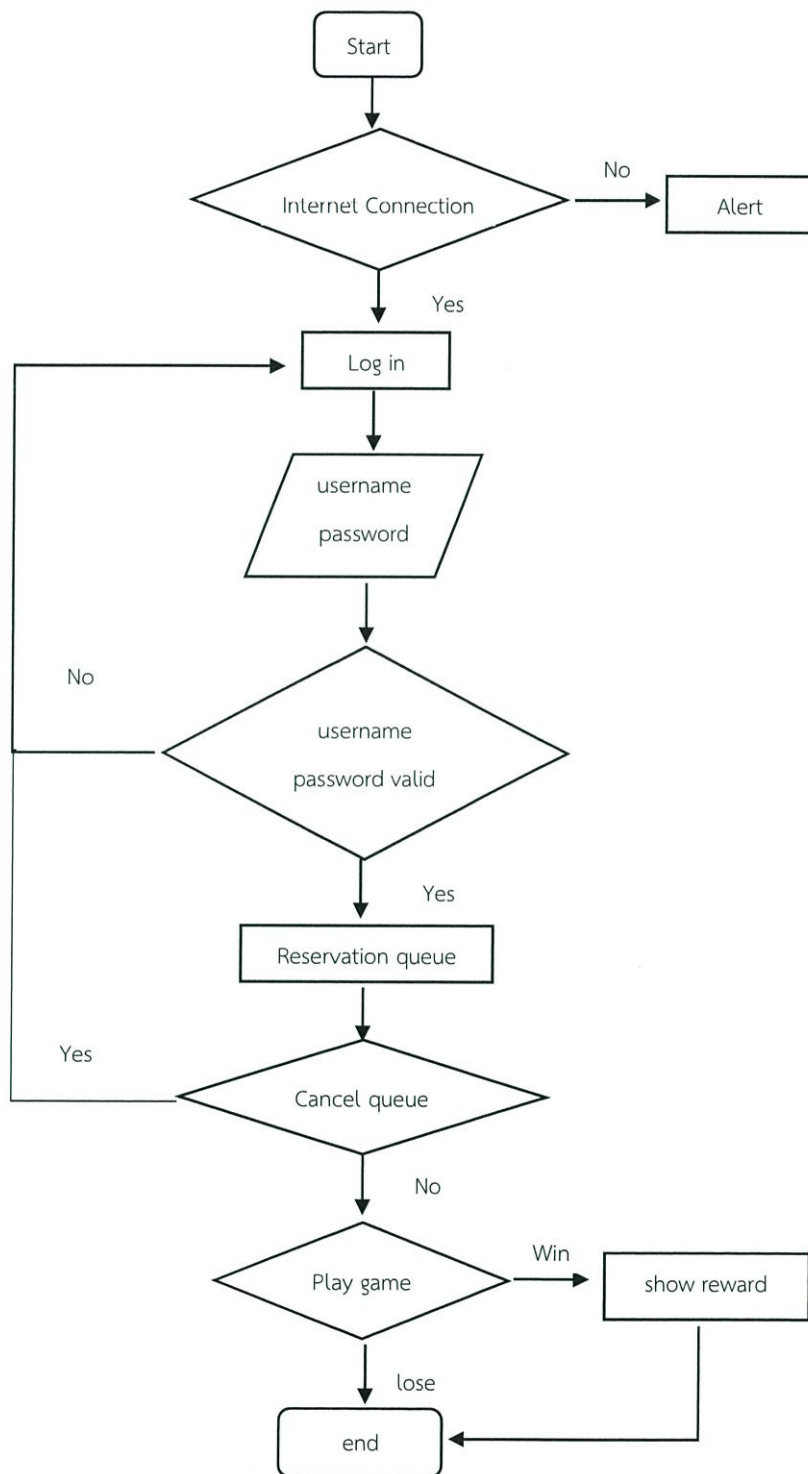
id	time_in	time_on_serive	name	people	adult	child	type	sequence	remenning_queue	remaining_time	queue	status
----	---------	----------------	------	--------	-------	-------	------	----------	-----------------	----------------	-------	--------

Query results operations

Create view

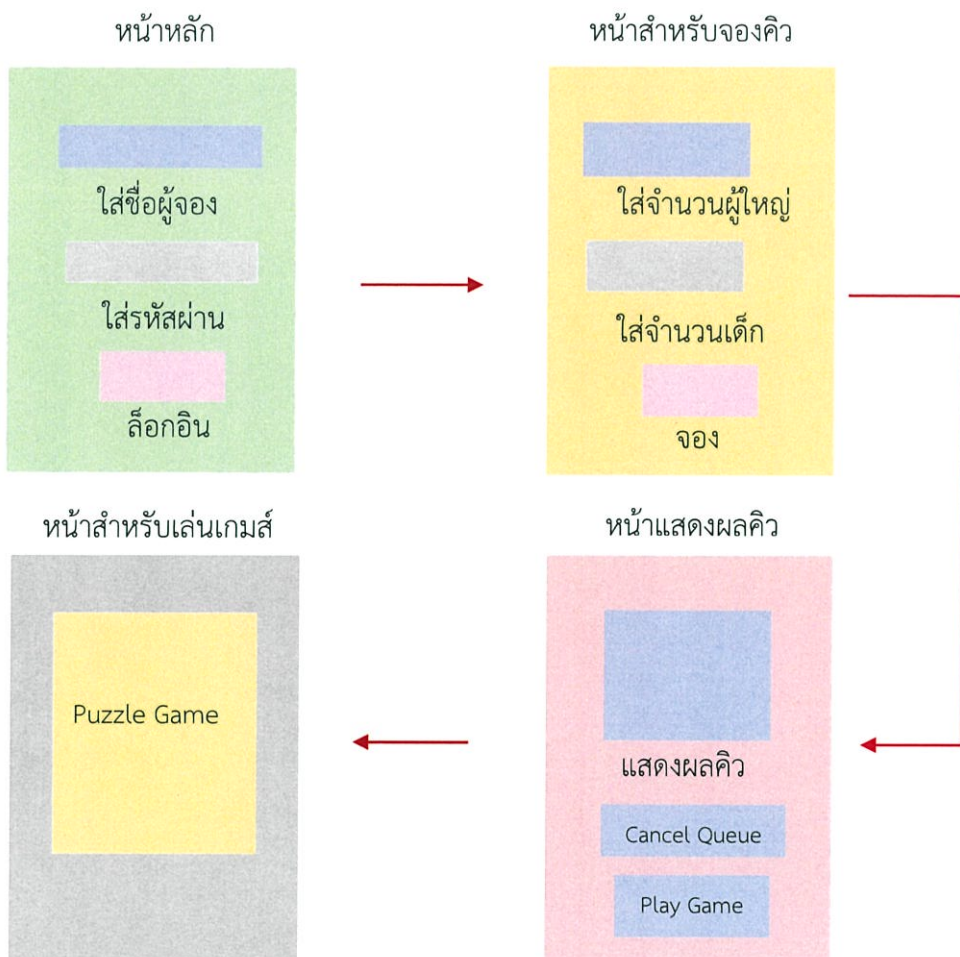
รูปที่ 3.2 รูปแบบฐานข้อมูล

3.1.2 การออกแบบและสร้างแอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์
บล็อกไดอะแกรมการทำงานของแอปพลิเคชัน (Application) เป็นไปดังรูปที่ 3.3



รูปที่ 3.3 บล็อกไดอะแกรมการทำงานของ
แอปพลิเคชัน (Application)

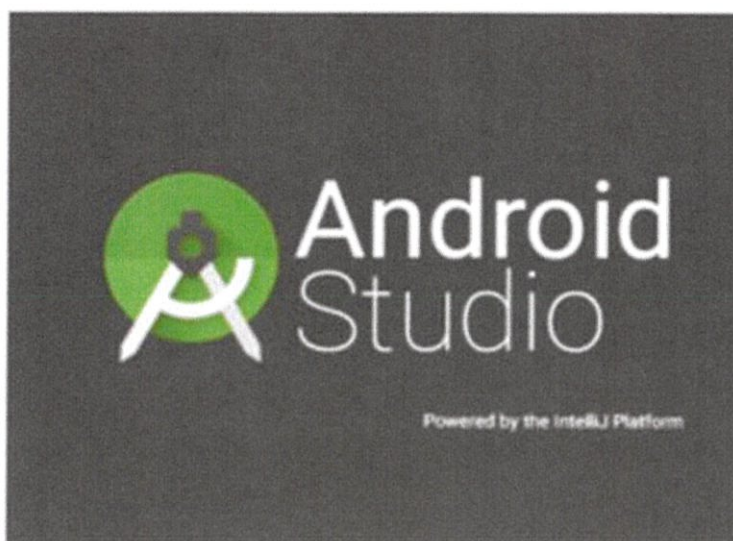
การสร้างแอปพลิเคชัน (Application) สำหรับระบบปฏิบัติการแอนดรอยด์ (Android) ในปริญญาโทฉบับนี้จะใช้โปรแกรมแอนดรอยด์สตูดิโอ (Android Studio) ซึ่งภาษาที่ใช้ในการเขียนแอปพลิเคชันคือภาษาจาวา (Java) ในการออกแบบส่วนผู้ใช้งาน (User Interface) ของแอปพลิเคชันนั้น ได้ออกแบบให้หน้าหลักสำหรับล็อกอิน (Login) หลังจากผู้ใช้งานทำการล็อกอิน (Login) แล้ว แอปพลิเคชัน (Application) จะนำไปสู่หน้าถัดไป แสดงหน้าสำหรับการจองคิว ซึ่งจะมีให้ใส่จำนวนคน โดยแยกประเภทเป็น ผู้ใหญ่ และ เด็ก เมื่อคลิกปุ่ม “Booking” ก็จะแสดงคิวที่ได้ เวลาที่จอง จำนวนคิวที่เหลือ และ ระยะเวลาประมาณการรอคิว โดยในระหว่างรอคิว ผู้ใช้งานสามารถเล่นเกมสล็อต โดยคลิกปุ่ม “Play Game” หากเล่นชนะ จะได้รับรางวัลจากทางร้าน นอกจากนี้ผู้วิจัยได้ทำการสร้างระบบยกเลิกคิว โดยผู้ใช้งานสามารถคลิกปุ่ม “Cancel” จากที่กล่าวไปข้างต้น User Interface ของแอปพลิเคชัน (Application) เป็นไปดังรูปที่ 3.4



รูปที่ 3.4 แผนผัง User Interface ของแอปพลิเคชัน

โดยขั้นตอนในการสร้างแอปพลิเคชันจะเป็นดังต่อไปนี้

3.1.2.1 ทำการโหลดโปรแกรมและติดตั้งโปรแกรมแอนดรอยด์สตูดิโอ (Android Studio) หลังจากติดตั้งแล้วทำการเปิดโปรแกรมจะได้โปรแกรมดังแสดงในรูปที่ 3.5



รูปที่ 3.5 โปรแกรมแอนดรอยด์สตูดิโอ (Android Studio)

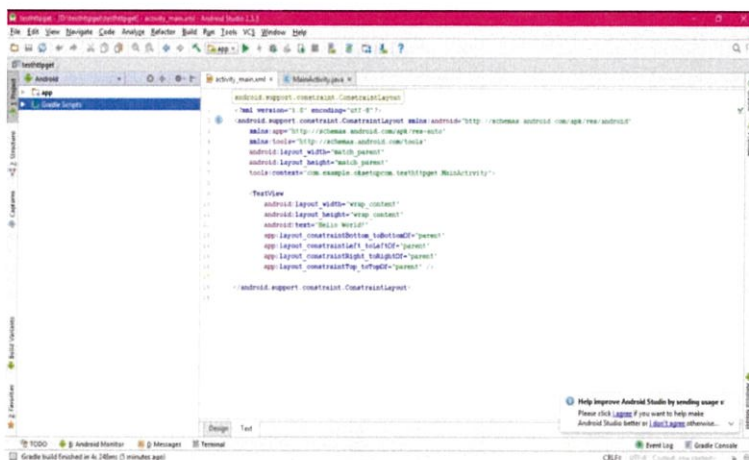
3.1.2.2 เมื่อเปิดโปรแกรมแล้ว ให้คลิกที่ Stat a new Android Studio project ทำการสร้างโปรเจค สำหรับ Android Studio จะมี Activity หลายรูปแบบให้เลือกใช้ จากรูป เราจะทำการเลือก Empty Activity



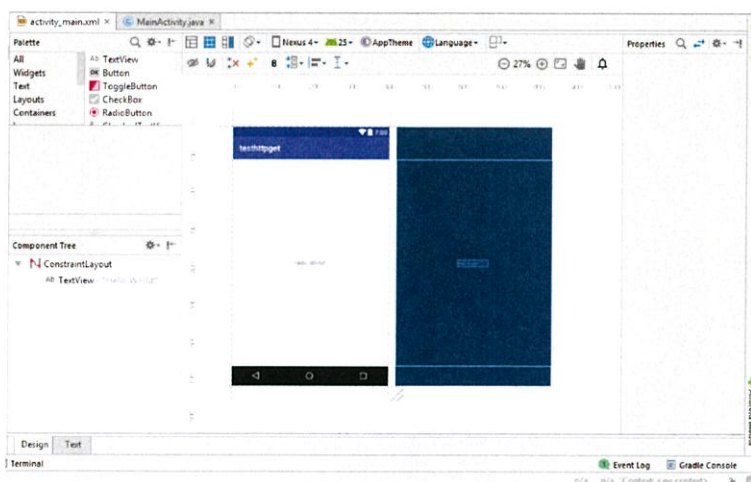
รูปที่ 3.6 เลือก Activity

ตั้งชื่อ Activity และ Layout จากรูป เราจะตั้งชื่อ Activity ว่า MainActivity และตั้งชื่อ Layout ว่า activity_main จาก นั้น คลิก Finish

3.1.2.3 หลังจากทำการสร้างโปรเจกขึ้นมาใหม่แล้วตัวโปรแกรมจะมีทั้งหน้าตาที่ไว้ใช้สำหรับเขียนโค้ดและหน้าตาที่ใช้สำหรับออกแบบตัวแอปพลิเคชันดังแสดงในรูปที่ 3.7 และ รูปที่ 3.8



รูปที่ 3.7 หน้าตาที่ใช้สำหรับเขียนโค้ดคำสั่ง



รูปที่ 3.8 หน้าตาที่ใช้สำหรับออกแบบ

3.1.2.4 เริ่มการสร้างแอปพลิเคชันที่หน้าตาสำหรับเขียนคำสั่ง โดยจะทำการส่งค่าข้อมูลค่า ชื่อผู้จอง (ตัวแปร name), จำนวนคน (ตัวแปร people), วันที่ และค่าข้อมูลอื่นๆ ส่งมาที่ฐานข้อมูล (database) โดยวิธีการส่งข้อมูลแบบ post method โดยแปลงข้อมูลเหล่านี้ให้อยู่ในรูปแบบของ Javascript Object Notation (JSON) เพื่อให้ Javascript สามารถแลกเปลี่ยนข้อมูลกับฐานข้อมูล (database) ได้ง่าย ดังแสดงในรูปที่ 3.9

```

1  {
2    "queue": [
3      {
4        "id": "21",
5        "time_in": "2017-11-20 22:47:17",
6        "time_on_server": "0000-00-00 00:00:00",
7        "name": "muey",
8        "people": "4",
9        "type": "B",
10       "sequence": "6",
11       "remanning_queue": "5",
12       "status": "0"
13     }
14   ]
15 }

```

รูปที่ 3.9 การส่งข้อมูลเข้าสู่ฐานข้อมูล (data base)

3.1.2.5 เมื่อส่งข้อมูลมาที่ดาต้าเบส (database) ข้อมูลที่ถูกส่งมาจะถูกนำไปประมวลผลคิวที่เซิร์ฟเวอร์ (server) ดังแสดงในรูปที่ 3.10 เพื่อประมวลผลคิวในลำดับถัดไป

```

function showpizza1()
{
    global $connect;

    $query = " SELECT * FROM pizza1 ORDER BY id DESC LIMIT 1 ; ";

    $result = mysqli_query($connect, $query);
    $number_of_rows = mysqli_num_rows($result);

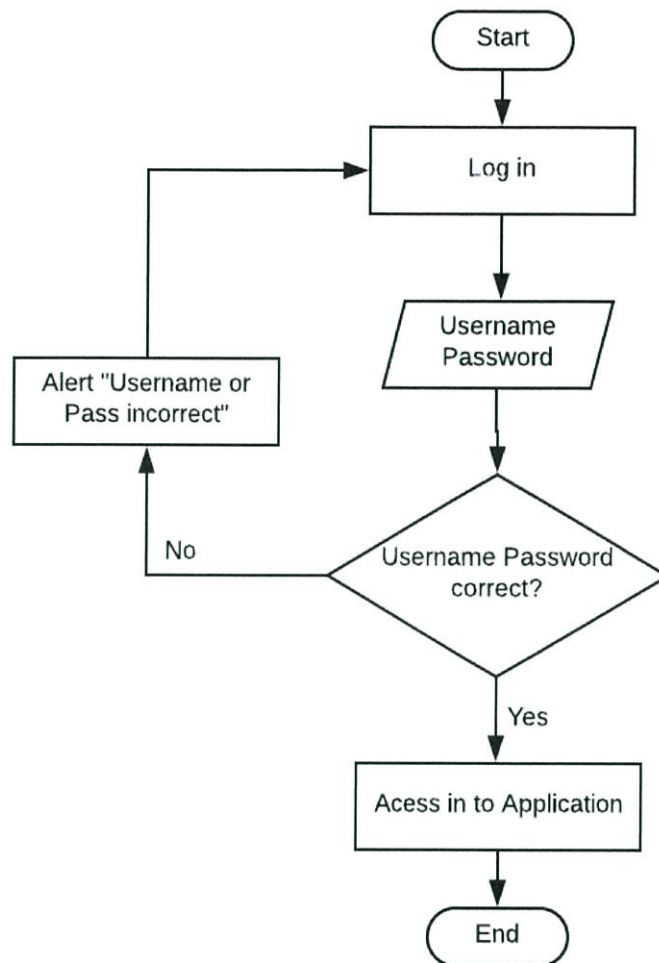
    $temp_array = array();

    if ($number_of_rows > 0){
        while ($row = mysqli_fetch_assoc($result)) {
            $temp_array[] = $row;
        }
    }
}

```

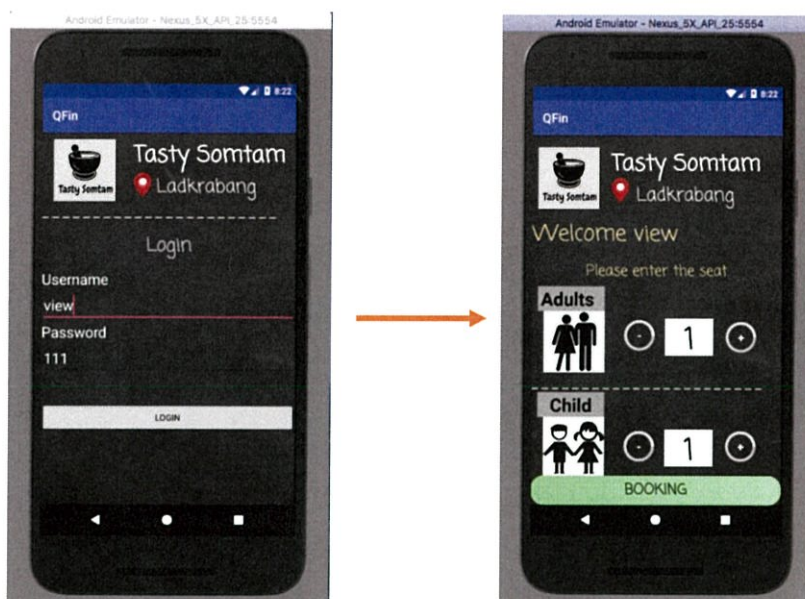
รูปที่ 3.10 การรับข้อมูลจากแอปพลิเคชันในฝั่งเซิร์ฟเวอร์ (server)

3.1.2.6 ออกแบบหน้า เมนูหลักให้สามารถล็อกอิน (Log in) เพื่อเข้าไปสู่หน้าการจองคิวได้ โดยมีการทำงานดังแผนผังงาน (flowchart) ดังแสดงในรูปที่ 3.11



รูปที่ 3.11 แผนผังงาน (flowchart) แสดงการทำงานของระบบล็อกอิน (Log in)

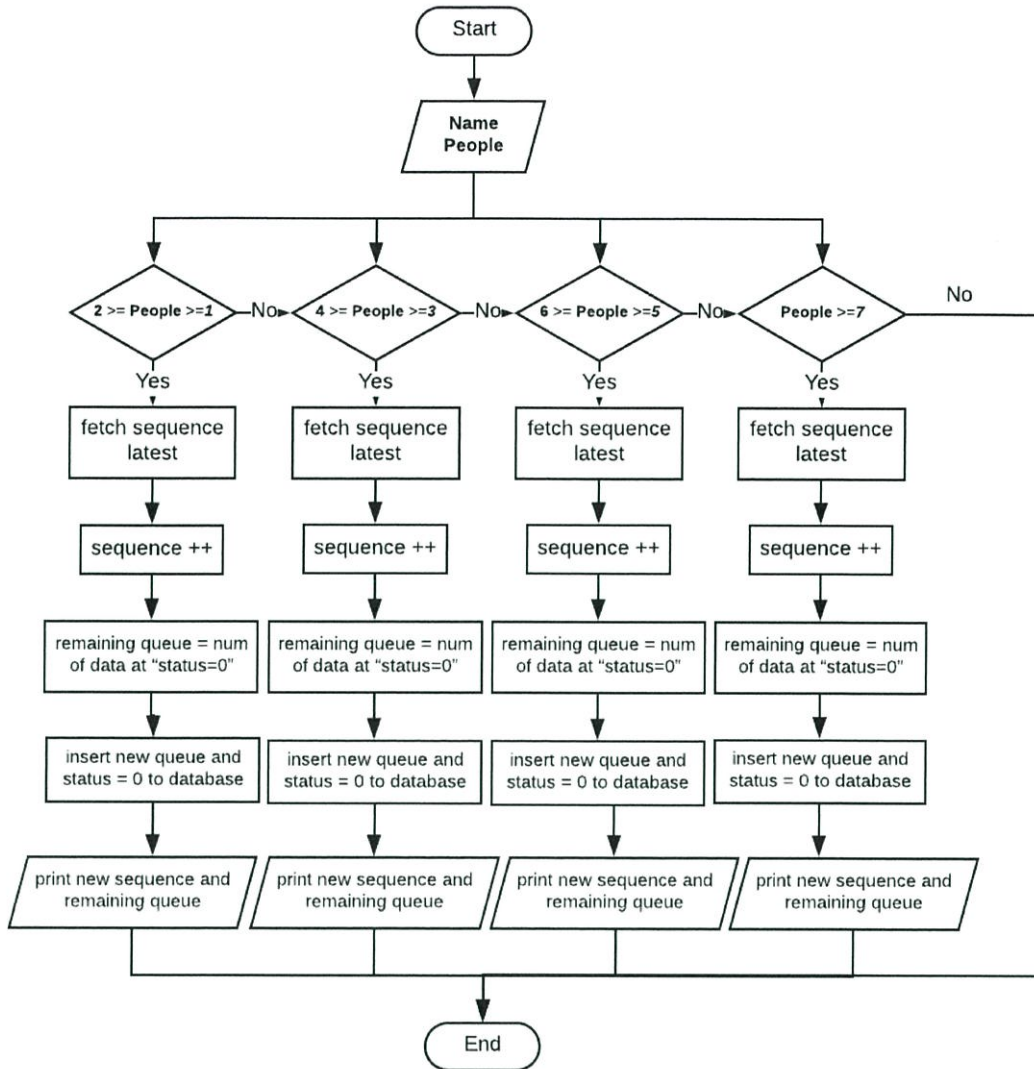
ผู้วิจัยได้ทำการออกแบบหน้าเมนูหลักล็อกอิน (Log in) สำหรับกรอก Username และ Password เพื่อเข้าสู่หน้าการจองคิว ดังแสดงในรูปที่ 3.12



รูปที่ 3.12 หน้าล็อกอิน (Log in) และหน้าการจองคิว ตามลำดับ

สำหรับการสร้างอัลกอริทึมการจองคิว จากที่เคยกล่าวในครั้งที่แล้ว จะแบ่งประเภทของคิวเป็น 4 ประเภท ได้แก่ type “A” หมายถึง โต๊ะประเภท 1 - 2 ท่าน ,type “B” หมายถึง โต๊ะประเภท 3 - 4 ท่าน , type “C” หมายถึง โต๊ะประเภท 5 – 6 ท่าน และ type “D” หมายถึง โต๊ะประเภท 7 ท่านขึ้นไป

3.1.2.7 การประมวลผลคิวนั้นจะใช้ตัวแปรจำนวนคน (people) มาพิจารณา เพื่อแยกประเภทของโต๊ะ และใช้ลำดับคิวของโต๊ะประเภทนั้นๆ ที่ดึงค่ามาจากฐานข้อมูล (database) ซึ่งมีการทำงานดังแผนผังงาน (flowchart) การทำงานของการประมวลผลคิว ดังแสดงในรูปที่ 3.13

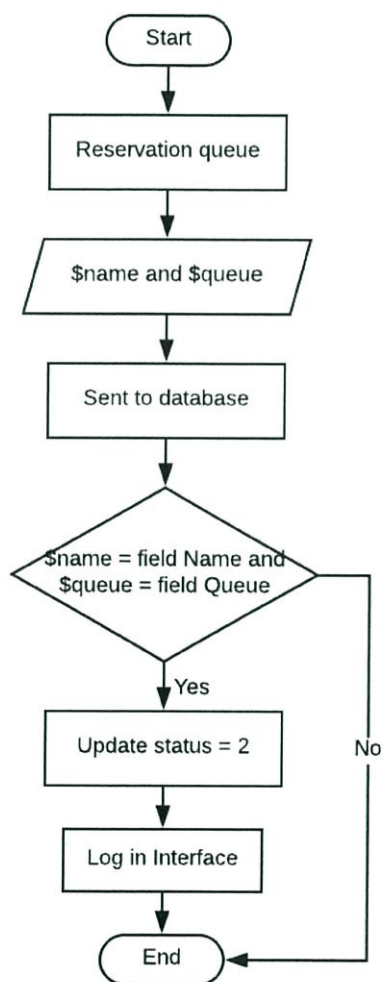


รูปที่ 3.13 Flowchart แสดงการทำงานของอัลกอริทึมการจองคิว

จากรูปที่ 3.13 อัลกอริทึมการจองคิวการผ่านแอปพลิเคชัน (Application) มีหลักการทำงานคือ ขั้นแรกผู้ใช้งานจะใส่ชื่อผู้จอง และจำนวนคนที่รับประทาน จากนั้นจะทำการส่งข้อมูลนี้มาที่เซิร์ฟเวอร์ (Server) เพื่อทำการประมวลผลคิว โดยเช็คจำนวนคนที่รับประทาน จำนวนคน 1-2 ท่าน ,3-4 ท่าน ,5-6 ท่าน และ 7 ท่านขึ้นไป จะจัดอยู่ในคิวประเภท A B C และ D ตามลำดับ แล้วทำการหาลำดับคิวโดยนำค่าลำดับคิวล่าสุดจากฐานข้อมูล (Database) ของประเภทคิวนั้นๆ บวกค่าเพิ่มอีกหนึ่ง ถัดไปจะเป็นการหาจำนวนคิวที่เหลือซึ่งหาได้จากลำดับคิวล่าสุดของประเภทนั้นๆ ที่มีค่า status = 0 หมายถึง รอเรียกคิว (Wait queue) แล้วนำลำดับคิวที่ได้กับจำนวนคิวที่เหลือ ส่ง

เข้าสู่ฐานข้อมูล (Server) สุดท้ายทำการดึงลำดับคิวนั้นกับจำนวนคิวที่เหลือมาที่แอปพลิเคชัน (Application) ของผู้ใช้งาน

3.1.2.8 หลังจากการจองคิวไปแล้ว ผู้วิจัยได้สร้างระบบยกเลิกคิว ผู้ใช้งานสามารถทำการยกเลิกโดยคลิกปุ่ม “Cancel” ซึ่งจะทำการส่งค่าชื่อผู้จองและคิวที่ได้ เข้าไปที่เซิร์ฟเวอร์ (server) แล้วทำการเปรียบเทียบว่าเหมือนค่าใดในฐานข้อมูล (database) แล้วทำการเปลี่ยนค่า status = 2 ซึ่งหมายถึง ยกเลิกคิวนี้ ดังนั้นคิวนี้จะไม่ถูกเรียก แล้วจะกลับไปยังหน้าล็อกอิน (Log in) อีกครั้งหนึ่ง ซึ่งมี การทำงานดังแผนผังงาน (flowchart) ดังแสดงในรูปที่ 3.14



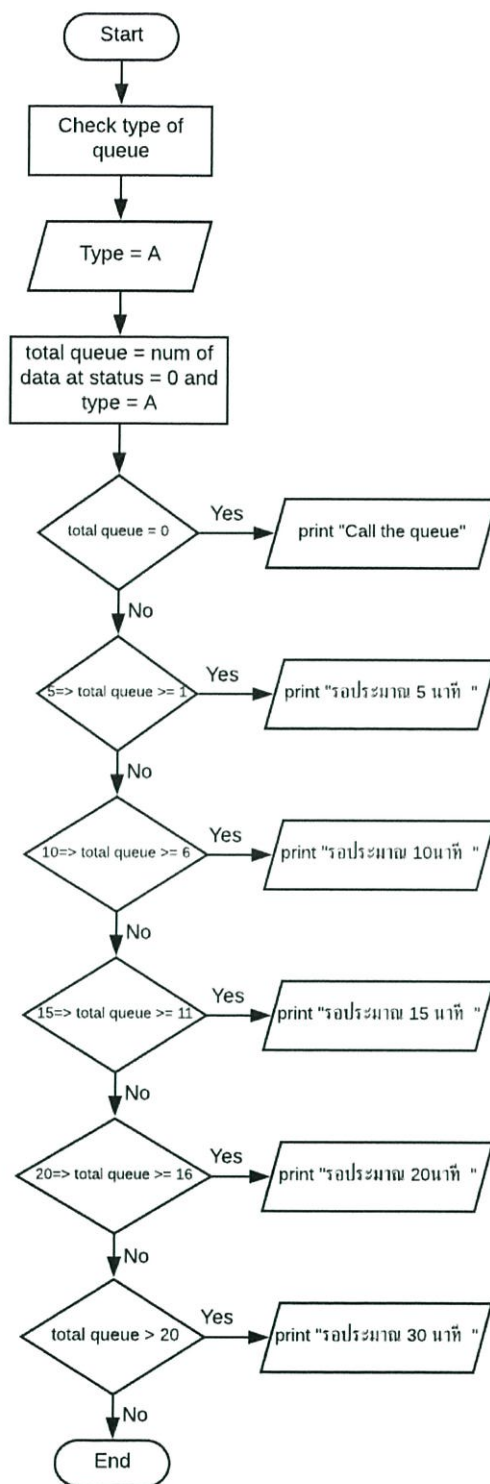
รูปที่ 3.14 แผนผังงาน (flowchart) การยกเลิกคิว

ผู้วิจัยได้ทำการออกแบบหน้าแสดงผลคิว ขณะเดียวกัน ผู้ใช้งานสามารถทำการยกเลิกคิวได้โดยคลิกปุ่ม “cancel” รวมทั้งในระหว่างรอคิวผู้ใช้งานสามารถเล่นเกมส Puzzle เพื่อรับรางวัลจากทางร้านโดยคลิกปุ่ม “Play Game” ดังแสดงในรูปที่ 3.15



รูปที่ 3.15 หน้าแสดงผลคิว

3.1.2.9 สำหรับการประมาณระยะเวลาการรอคิว ผู้วิจัยจะพิจารณาจากจำนวนคิวที่รอเรียกของคิวแต่ละประเภท โดยกำหนดให้ดังนี้ มีการทำงานดังแผนผังงาน (flowchart) ดังแสดงในรูปที่ 3.16 รอเรียกคิว



รูปที่ 3.16 แผนผังงาน (flowchart) การประมาณเวลารอคิว

จากรูปที่ 3.15 จะแสดงตัวอย่างการประมาณเวลาการรอคิว เริ่มต้นเช็คประเภทของคิว ในตัวอย่างคือประเภท A แล้วทำการเช็คจำนวนข้อมูลที่มีค่า status = 0 และ type =A ในฐานข้อมูล (database) ซึ่งจะจำนวนข้อมูลนี้จะเก็บไว้ในตัวแปร total queue แล้วนำตัวแปรนี้มาทำการเช็คว่าเข้ากับเงื่อนไขใด

3.1.2.10 สำหรับการออกแบบเกมส์ Puzzle ผู้วิจัยได้ทำการออกแบบโดยใช้ภาพจากทางร้าน โดยตัดแบ่งภาพขนาด 3 x 3 โดยผู้ใช้งานจะต้องเล่นเกมส์ให้ชนะภาพในระยะเวลาที่กำหนด จึงจะได้ของรางวัลจากทางร้านไป เช่น ส่วนลด 5 %, โบนัสขึ้นมา 4 จ่าย 3 เป็นต้น รูปที่ใช้ในการสร้างเกมส์ เป็นดังแสดงในรูปที่ 3.17



รูปที่ 3.17 หน้าแสดงผลคิว

เมื่อคลิกปุ่ม “Play Game” จะพบกับรูปแบบของเกมส์ดังแสดงในรูปที่ 3.18



รูปที่ 3.18 หน้าเกมส์

3.1.3 การออกแบบเว็บแสดงคิว

ในปฏิญานินพจน์นี้ มีการจัดทำเว็บไซต์สำหรับแสดงคิวโดยมีเป้าหมายเพื่อแสดงคิว, จำนวนคิวที่เหลือ และสำหรับโฆษณา โปรโมชันต่างๆ ให้แก่ผู้ใช้งานแบบเรียลไทม์ (real time) โดยผู้ใช้งานสามารถติดตามได้ผ่านทางเว็บไซต์และหน้าร้าน สำหรับการสร้างเว็บไซต์เพื่อแสดงผลคิวจะใช้ภาษา HTML ,CSS และ JavaScript ในการเขียน โดยได้ทำการออกแบบหน้าเว็บไซต์ดังแสดงในรูปที่ 3.19



รูปที่ 3.19 ออกแบบเว็บไซต์สำหรับแสดงผลคิว

สำหรับหน้าเว็บไซต์จะประกอบไปด้วยส่วนต่างๆ 6 ส่วนได้แก่

หมายเลข 1 คือ ชื่อร้านอาหาร

หมายเลข 2 คือ กล่องแสดงข้อความ

หมายเลข 3 คือ แสดงคิวที่ถูกรีเรียก และจำนวนคิวที่เหลือของโต๊ะประเภท 1-2 ท่าน

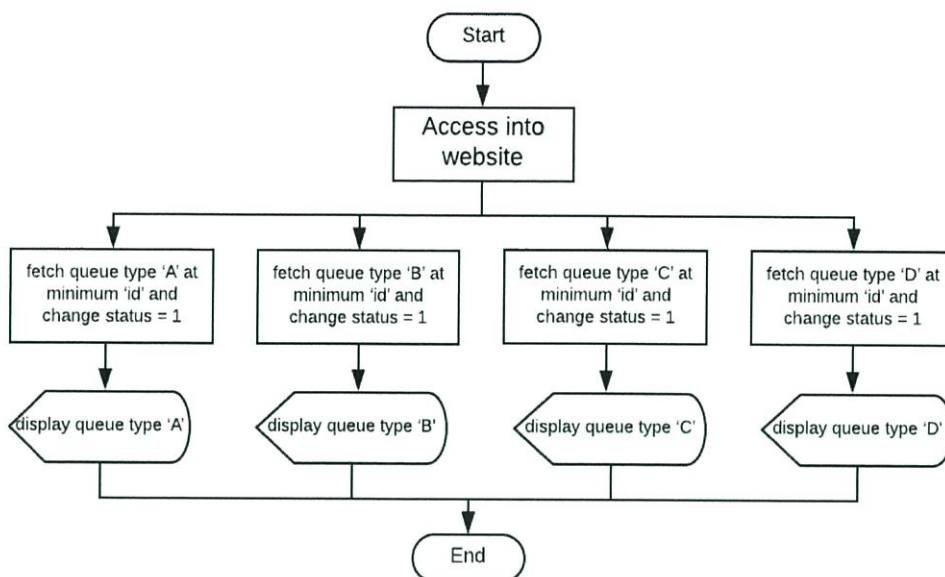
หมายเลข 4 คือ แสดงคิวที่ถูกรีเรียก และจำนวนคิวที่เหลือของโต๊ะประเภท 3-4 ท่าน

หมายเลข 5 คือ แสดงคิวที่ถูกรีเรียก และจำนวนคิวที่เหลือของโต๊ะประเภท 5-6 ท่าน

หมายเลข 6 คือ แสดงคิวที่ถูกรีเรียก และจำนวนคิวที่เหลือของโต๊ะประเภท 7 ท่านขึ้นไป

หมายเลข 7 คือ โฆษณาสินค้าและโปรโมชันต่างๆ

มีการทำงานดังแผนผังงาน (flowchart) ดังแสดงในรูปที่ 3.20



รูปที่ 3.20 Flowchart แสดงการทำงานของเว็บไซต์แสดงคิว

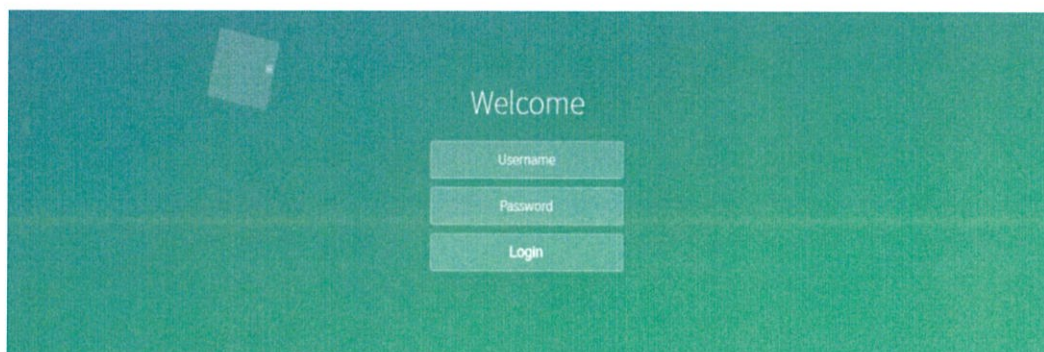
จากรูปที่ 3.17 เมื่อเข้าสู่เว็บไซต์แสดงผลคิว ค่าคิวที่แสดงนั้นจะดึงมาจากฐานข้อมูล เช่น โต๊ะประเภท 1-2 ท่าน จะจัดอยู่ใน type 'A' โดยจะดึงค่าคิวลำดับน้อยที่สุดและมีค่า status = 0 (รอเรียกคิว) ขึ้นมาแสดงบนหน้าเว็บไซต์ และจะเปลี่ยนค่า status = 1 ซึ่งหมายถึงถูกเรียกคิวแล้วนั่นเอง

3.1.4 การออกแบบเว็บไซต์สำหรับผู้ดูแลระบบ

ในโครงการนี้มีการจัดทำเว็บไซต์สำหรับผู้ดูแลระบบ โดยมีเป้าหมายเพื่อดูแล บริหารจัดการคิว การรันลำดับคิว รวมถึงเพื่อแจ้งข้อมูลข่าวสารต่างๆ แก่ผู้ใช้งาน โดยมีส่วนประกอบต่างๆ บนเว็บแอดมินดังนี้

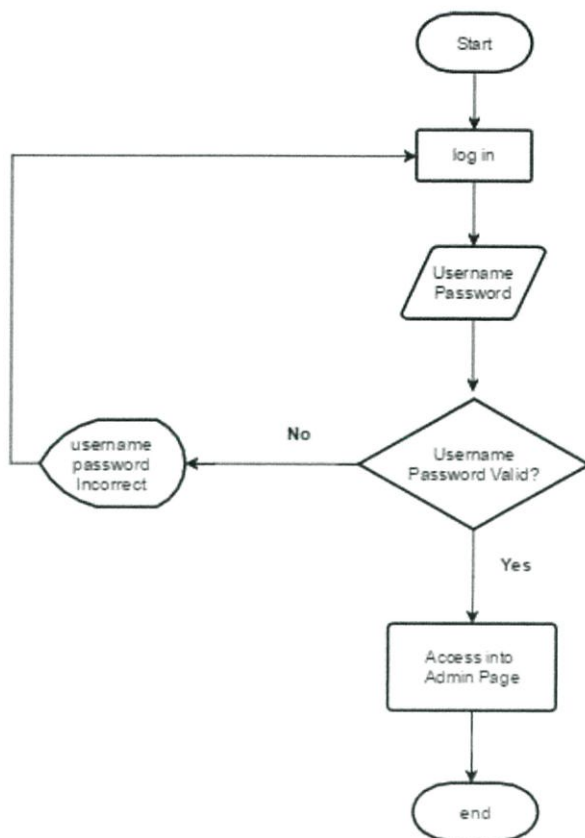
3.1.4.1 ระบบล็อกอิน เป็นตัวเข้าถึงเว็บแอดมิน ดังแสดงในรูปที่ 3.14 และเพื่อให้มีความปลอดภัย เว็บแอดมินจึงจำเป็นต้องมีการกรอก Username และ Password ที่ถูกต้องถึงจะสามารถเข้าถึงเว็บแอดมินได้ โดยเริ่มต้นจากการสร้างไฟล์ต่างๆ ที่เกี่ยวข้องกับระบบล็อกอินทั้งหมด 6 ไฟล์ ดังนี้

- 1) config.php เป็นไฟล์ที่ใช้ในการเชื่อมกับฐานข้อมูลให้ใช้งานได้
 - 2) login.html เป็นไฟล์ที่แสดงรูปแบบของหน้าล็อกอิน
 - 3) check_login.php เป็นไฟล์ที่เช็คเงื่อนไขว่ารหัสผ่านตรงกับในฐานข้อมูลหรือไม่ ถ้าตรงก็ให้ผ่าน
 - 4) admin.html เป็นไฟล์ที่แสดงรูปแบบหน้าเว็บแอดมิน
 - 5) logout.php เป็นไฟล์ที่ใช้ในการตัดการเชื่อมต่อจากที่เราอยู่ในระบบล็อกอิน
- โดยมีการแสดงรูปแบบการเข้าถึงดังนี้
- Login.php -> check_login.php -> admin.html -> logout.php หน้าล็อกอินเป็นดังแสดงในรูปที่ 3.21



รูปที่ 3.21 หน้าล็อกอินของเว็บแอดมิน

โดยระบบล็อกอินมีการทำงานดังรูปที่ 3.22

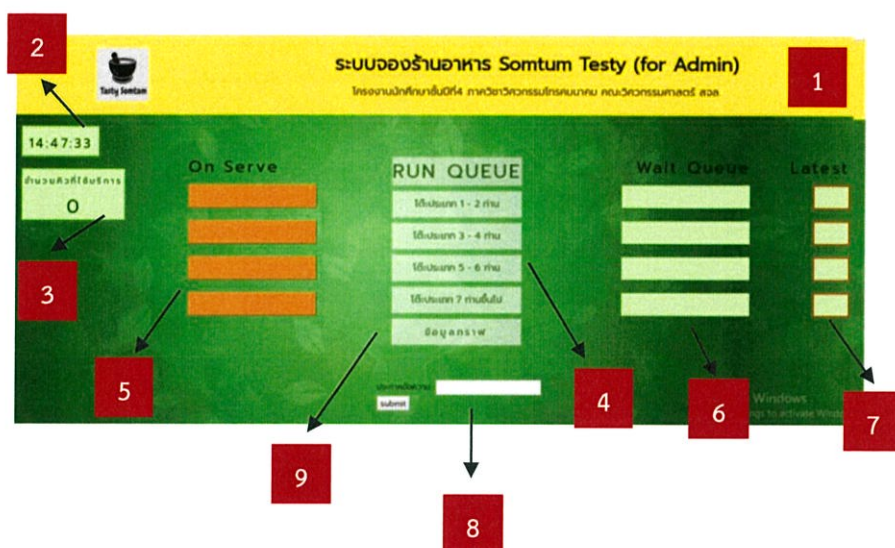


รูปที่ 3.22 Flowchart แสดงการทำงานของระบบล็อกอิน (Log in)

จากนั้นนำไฟล์ต่างๆ ข้างต้นไปเก็บไว้ที่เซิร์ฟเวอร์ (server) แล้วไปดึงข้อมูลที่สำคัญจากในฐานข้อมูล เช่น type, sequence, และ status ซึ่งหมายถึง ประเภทโต๊ะ ลำดับคิว และค่าสถานะของคิว ถ้า status = 0 คือ รอเรียกคิว status = 1 คือ ถูกเรียกคิวแล้ว ออกมา เพื่อนำไปแสดงผลที่หน้าหลักของเว็บแอดมินตามลำดับ

3.1.4.2 การออกแบบหน้าหลักของเว็บแอดมิน

สำหรับหน้าหลักของเว็บแอดมิน (admin web) มีเป้าหมายหลักคือ บริหารจัดการคิว การรันคิวเพื่อให้แสดงผลคิวที่ถูกเรียกผ่านหน้าเว็บไซต์ดังในหัวข้อ 3.1.4.1 การออกแบบและส่วนประกอบต่างๆในหน้าหลักเว็บแอดมิน (admin web) เป็นไปดังรูปที่ 3.23



รูปที่ 3.23 ออกแบบหน้าเว็บแอดมิน (admin web)

จากรูปที่ 3.20 หน้าเว็บแอดมิน (admin web) มีส่วนประกอบต่างๆ ดังนี้

หมายเลข 1 คือ ชื่อร้านอาหาร

หมายเลข 2 คือ เวลาปัจจุบัน

หมายเลข 3 คือ จำนวนคิวทั้งหมดที่เข้าใช้บริการ

หมายเลข 4 คือ ปุ่ม สำหรับรันคิว โดยจะแบ่งเป็น 4 ปุ่ม ตามประเภทของโต๊ะดังที่กล่าวไว้ข้างต้น เช่น เมื่อแอดมิน (พนักงานหรือเจ้าของร้าน) คลิกปุ่มรันคิวประเภทโต๊ะ 1-2 ท่าน จะทำการเลือกคิว type 'A' โดยจะเลือกคิวที่ลำดับน้อยสุด และมีค่า status = 0 (รอเรียกคิว) แล้วทำการอัปเดตค่า status = 1 (คิวนี้นี้ถูกเรียกแล้ว) ซึ่งคิวที่ถูกเรียกนี้ จะถูกนำไปแสดงบนหน้าเว็บไซต์ ดังหัวข้อที่ 3.1.3

หมายเลข 5 คือ ช่องแสดงลำดับคิวที่ถูกเรียกแล้ว (status = 1) ลำสุด 5 คิว โดยแยกตามประเภทของโต๊ะ

หมายเลข 6 คือ ช่องแสดงลำดับคิวที่รอเรียก (status = 0) ลำสุด 7 คิว โดยแยกตามประเภทของโต๊ะ

หมายเลข 7 คือ ช่องแสดงลำดับคิวสุดท้ายที่รอเรียก (status = 0) โดยแยกตามประเภทของโต๊ะ

หมายเลข 8 คือ ช่องสำหรับประกาศข้อความ ข่าวสารต่างๆ เช่น หยุดงดบริการ , ระบบจองคิวขัดข้อง เป็นต้น

หมายเลข 9 คือ ข้อมูลกราฟ

3.1.4.3 การสร้างกราฟสำหรับวิเคราะห์ข้อมูลจากฐานข้อมูล (server)

การสร้างกราฟโดยอ้างอิงจากข้อมูลในฐานข้อมูล (server) เพื่อให้แอดมินสามารถนำผลจากกราฟมาวิเคราะห์ได้ เช่น การเพิ่มจำนวนโต๊ะของแต่ละประเภท , ช่วงเวลาที่ลูกค้ามีจำนวนมาก เป็นต้น โดยภาษาที่ใช้ในการสร้างกราฟคือ ภาษา php , java script และ html โดยแอดมิน (admin) สามารถเลือกช่วงเวลาที่ต้องการได้ผ่านเว็บแอดมิน (admin web) แล้วคลิกปุ่ม “ข้อมูลกราฟ” ดังแสดงในรูปที่ 3.24

รูปที่ 3.24 หน้าเว็บแอดมิน สำหรับเลือกช่วงเวลาพล็อตข้อมูล

ประเภทของกราฟแบ่งเป็น 2 ประเภท คือ

- 1) กราฟแสดงความสัมพันธ์ระหว่างจำนวนผู้ใช้งานและประเภทของผู้ใช้งาน
- 2) กราฟแสดงความสัมพันธ์ระหว่างจำนวนโต๊ะและประเภทของโต๊ะ

การพล็อตกราฟ จะทำการการดึงข้อมูลจากฐานข้อมูล (database) ซึ่งฐานข้อมูลนี้ จะรวบรวมข้อมูลเป็นรายเดือน ของแต่ละปี ซึ่งจะแสดงจำนวนผู้ใช้งาน (people) , จำนวนผู้ใช้งานที่เป็นผู้ใหญ่ (adult), จำนวนผู้ใช้งานที่เป็นเด็ก (child), จำนวนโตะประเภทเอ (A) , จำนวนโตะประเภทบี (B) , จำนวนโตะประเภทซี (C) และ จำนวนโตะประเภทดี (D) ดังแสดงในรูปที่ 3.25

+ Options

			id	month	year	adult	child	people	A	B	C	D
<input type="checkbox"/>	Edit	Copy	1	Feb	2018	30	28	58	10	8	8	1
<input type="checkbox"/>	Edit	Copy	2	Mar	2018	48	14	62	19	5	3	1
<input type="checkbox"/>	Edit	Copy	3	April	2018	51	15	66	15	8	4	3

Check all With selected: Edit Copy Delete Export

รูปที่ 3.25 ข้อมูลจากฐานข้อมูล (database) ที่นำมาพล็อต

จากรูปจะนำข้อมูลจากฐานข้อมูลไปพ็อตกราฟ โดยให้แกน x คือ month แกน y จะประกอบไปด้วย A, B, C และ D โค้ดส่วนการดึงข้อมูล และส่วนรูปแบบกราฟเป็นดังในรูปที่ 3.26 และ 3.27 ตามลำดับ

```
<?php
$connect = mysqli_connect("localhost", "root", "view_1234", "test");

$month = $_POST['month'];
$month_1 = $_POST['month_1'];
$year = $_POST['year'];

$query = "SELECT * FROM graph_1 WHERE (id BETWEEN '$month' AND '$month_1') AND year = '$year'";
$result = mysqli_query($connect, $query);
$chart_data = '';
while($row = mysqli_fetch_array($result))
{
    $chart_data .= "{ month:'".$row["month"]."', people:".$row["people"].", adult:".$row["adult"].", child:".$row[""];
}
$chart_data = substr($chart_data, 0, -2);
?>
```

รูปที่ 3.26 โค้ดการดึงข้อมูลจากฐานข้อมูล (database) ที่นำมาพล็อต

```

17 <!DOCTYPE html>
18 <html>
19 <head>
20 <title>Webslesson Tutorial | How to use Morris.js chart with PHP & Mysql</title>
21 <link rel="stylesheet" href="//cdnjs.cloudflare.com/ajax/libs/morris.js/0.5.1/morris.css">
22 <script src="//ajax.googleapis.com/ajax/libs/jquery/1.9.0/jquery.min.js"></script>
23 <script src="//cdnjs.cloudflare.com/ajax/libs/raphael/2.1.0/raphael-min.js"></script>
24 <script src="//cdnjs.cloudflare.com/ajax/libs/morris.js/0.5.1/morris.min.js"></script>
25
26 </head>
27 <body>
28 <br /><br />
29 <div class="container" style="width:900px;">
30 <h2 align="center">Morris.js chart with PHP & Mysql</h2>
31 <h3 align="center">Last 10 Years Profit, Purchase and Sale Data</h3>
32 <br /><br />
33 <div id="chart"></div>
34 </div>
35 </body>
36 </html>
37
38 <script>
39 Morris.Bar({
40   element : 'chart',
41   data:[<?php   echo $chart_data; ?>],
42   xkey:'month',
43   ykeys:['people', 'adult', 'child'],

```

รูปที่ 3.27 โค้ดส่วนรูปแบบกราฟ

3.1.4.3 สร้างปฏิทินกิจกรรมผ่านหน้าเว็บแอดมิน (admin web)

สำหรับปฏิทินกิจกรรมผู้วิจัยมีจุดประสงค์เพื่อให้พนักงานได้ทราบรายละเอียดแผนงานต่างๆของร้าน ผ่านหน้าปฏิทินในหน้าเว็บแอดมิน (admin web) ที่พนักงานสามารถเข้าถึงได้ ซึ่งทำให้ง่ายต่อการสื่อสาร เช่น การนัดหมายต่างๆ , การตรวจร้าน, การประชุมร้าน ฯลฯ เป็นต้น

ผู้วิจัยสร้างปฏิทินกิจกรรม ด้วย fullcalendar และ jQuery เวอร์ชัน 3.9.0 ดังแสดงในรูปที่ 3.28 และสร้าง Bootstrap Modal เวอร์ชัน 4.0.0 สำหรับเพิ่มข้อมูล ลบข้อมูล และแก้ไขข้อมูลได้ ดังแสดงในรูปที่ 3.29 และ 3.30



รูปที่ 3.28 รูปแบบปฏิทินกิจกรรม

× New Fullcalendar Modal With MySQL

เรื่อง

วันที่เริ่มต้น

วันที่สิ้นสุด

บันทึกข้อมูล ปิด

รูปที่ 3.29 Bootstrap Modal สำหรับการเพิ่มข้อมูล

× Fullcalendar Modal With MySQL

เรื่อง

ประชุมร้าน

วันที่เริ่มต้น

2018-03-11

วันที่สิ้นสุด

2018-03-14

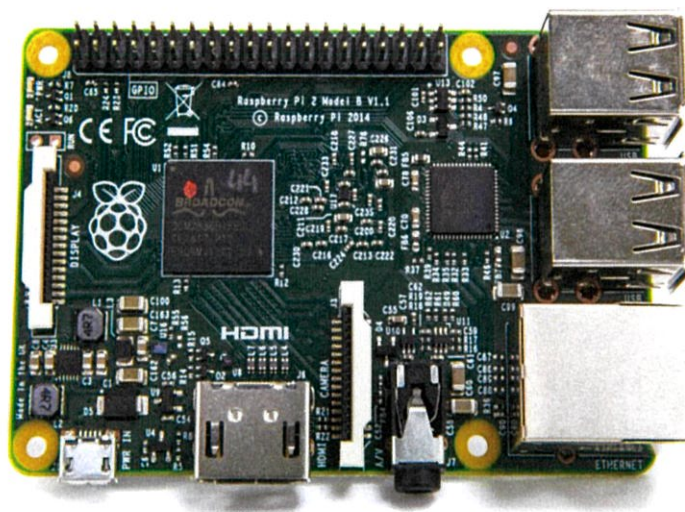
Delete Save changes Close

รูปที่ 3.30 Bootstrap Modal สำหรับการลบและแก้ไขข้อมูล

3.2 เครื่องมือที่ใช้ในการทดลอง

3.2.1 อุปกรณ์ราคาเบอรีพาย (Raspberry Pi) สำหรับการสร้างเซิร์ฟเวอร์ (Server)

Raspberry Pi คือ คอมพิวเตอร์ขนาดเล็กเท่ากับบัตรเครดิต สามารถใช้งานได้ด้วยทีวี หรือหน้าจอคอมพิวเตอร์ ในราคาที่ถูก (\$35) ถูกออกแบบมาโดยนักพัฒนาชาวอังกฤษ ดังแสดงในรูปที่ 3.27 เพื่อใช้ในการศึกษา ใช้ระบบปฏิบัติการลินุกซ์ (Linux) เช่น Raspbian (Debian), Pidora (Fedora) หรือล่าสุด Windows 10 ประกาศว่า Raspberry Pi จะสามารถใช้งาน Windows 10 ได้เช่นกัน



รูปที่ 3.30 ราคเบอรีพาย 2 (Raspberry Pi 2)

3.2.2 สเปกของ Raspberry Pi 2 Model B

- 1) หน่วยประมวลผล (CPU) : Broadcom BCM2836 900 MHz, Quad-core
- 2) กราฟฟิก (GPU) : Broadcom VideoCore IV
- 3) แรม (RAM) : 1 GB
- 4) หน่วยความจำ (Storage) : 64 GB SD Card (ขึ้นอยู่กับ SD card ที่ใช้)
- 5) Video Output : HDMI, Composite RCA
- 6) Audio Output : YES
- 7) Ethernet : YES

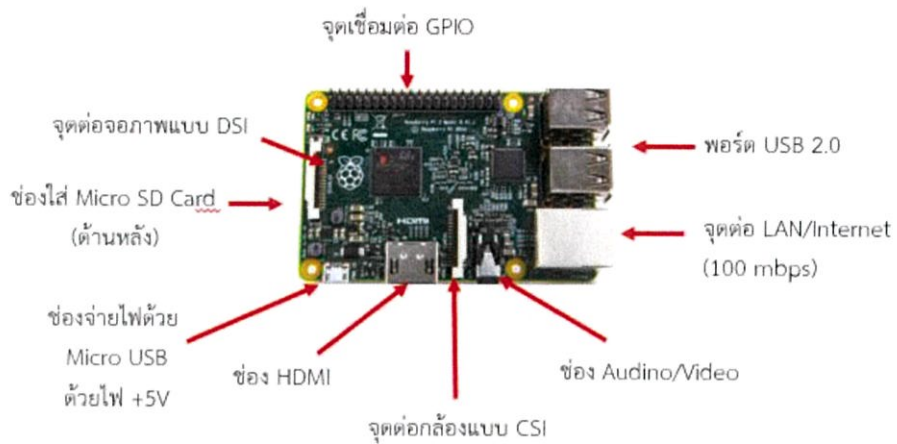
8) USB 2.0 : YES, 4 Ports

9) ระบบปฏิบัติการ (OS) : Windows/Linux

10) ขนาด : 8.6×5.4×1.7 ซม.

3.2.3 ส่วนประกอบของ Raspberry Pi

ส่วนประกอบของ Raspberry Pi เป็นดังแสดงในรูปที่ 3.28 และ 3.29



รูปที่ 3.31 ส่วนประกอบของ Raspberry Pi



รูปที่ 3.32 ส่วนประกอบของ Raspberry Pi

3.2.4 อุปกรณ์ที่ต้องใช้ในการเริ่มต้นกับ Raspberry Pi

อุปกรณ์ที่จำเป็น ได้แก่

- 1) USB Power Adapter (5V, 1A)
- 2) สาย Micro USB
- 3) Micro SD Card (อย่างน้อย 4 GB, Class 4)
- 4) สาย HDMI
- 5) สาย Ethernet หรือ WIFI USB dongle
- 6) หน้าจอที่พอร์ต HDMI
- 7) USB Mouse, USB Keyboard
- 8) คอมพิวเตอร์
- 9) Card Reader หรือ SD card Adapter

สำหรับผู้ที่ต้องการต่อยอด หรือทำให้ Raspberry Pi มีความสามารถเพิ่มขึ้นไปอีก ก็สามารถหาอุปกรณ์เสริมเหล่านี้มาใช้ได้

อุปกรณ์เสริม ได้แก่

- 1) Case
- 2) Heatsink
- 3) Camera Module
- 4) LCD Display
- 5) Real Time Clock

สำหรับการติดตั้งระบบปฏิบัติการลง Micro SD card ได้กล่าวไว้ในบทที่ 2

3.3 การจัดเก็บผลการทดลอง

3.3.1 การจัดเก็บผลการทดลองสำหรับแอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์

3.3.1.1 เมื่อแอปพลิเคชันถูกทำการทดสอบการทำงานด้วยการจำลองสมาร์ตโฟนในโปรแกรมแอนดรอยด์สตูดิโอ (Android Studio) ได้แล้ว จากนั้นจะนำแอปพลิเคชันมาติดตั้งลงบนสมาร์ตโฟนจริงเพื่อทดสอบการใช้งาน

3.3.2 การจัดเก็บผลการทดลองสำหรับเว็บไซต์แสดงผลคิว

3.3.2.1 เมื่อทำการสร้างไฟล์ที่เกี่ยวข้องกับเว็บไซต์แสดงผลคิวทั้งหมดเสร็จสิ้นแล้ว จากนั้นทำการทดลองแสดงการทำงานของเว็บผ่านทางคอมพิวเตอร์ เพื่อตรวจเช็คความถูกต้อง

3.3.3 การจัดเก็บผลการทดลองสำหรับเว็บแอดมิน

3.3.3.1 เมื่อทำการสร้างไฟล์ที่เกี่ยวข้องกับเว็บแอดมินทั้งหมดเสร็จสิ้นแล้ว จากนั้นทำการทดลองแสดงการทำงานของเว็บผ่านทางคอมพิวเตอร์ เพื่อตรวจเช็คความถูกต้อง

บทที่ 4

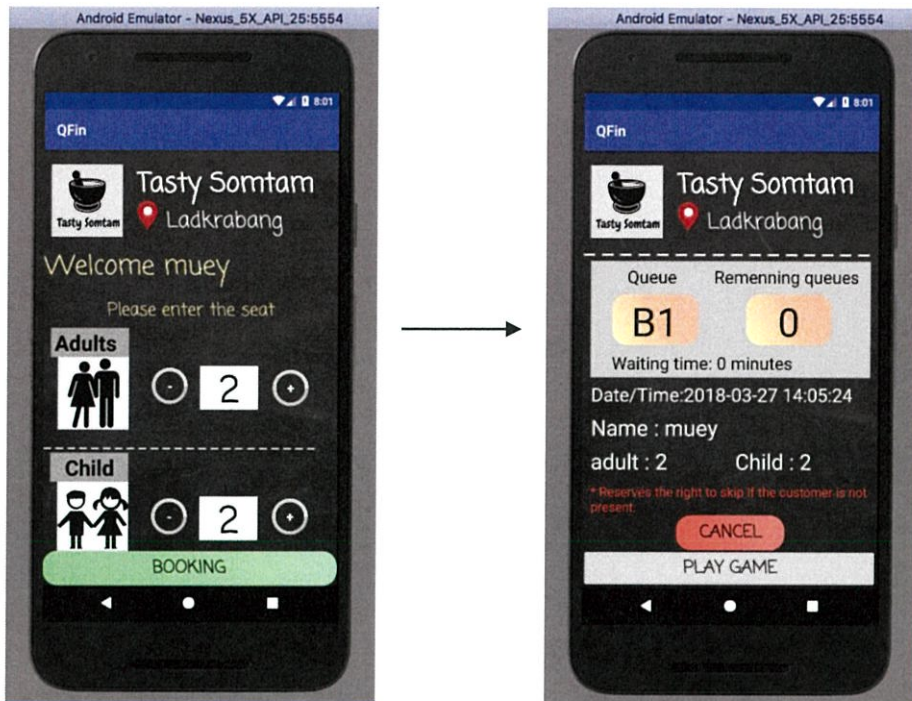
ผลการทดลอง

4.1 การทดสอบแอปพลิเคชัน

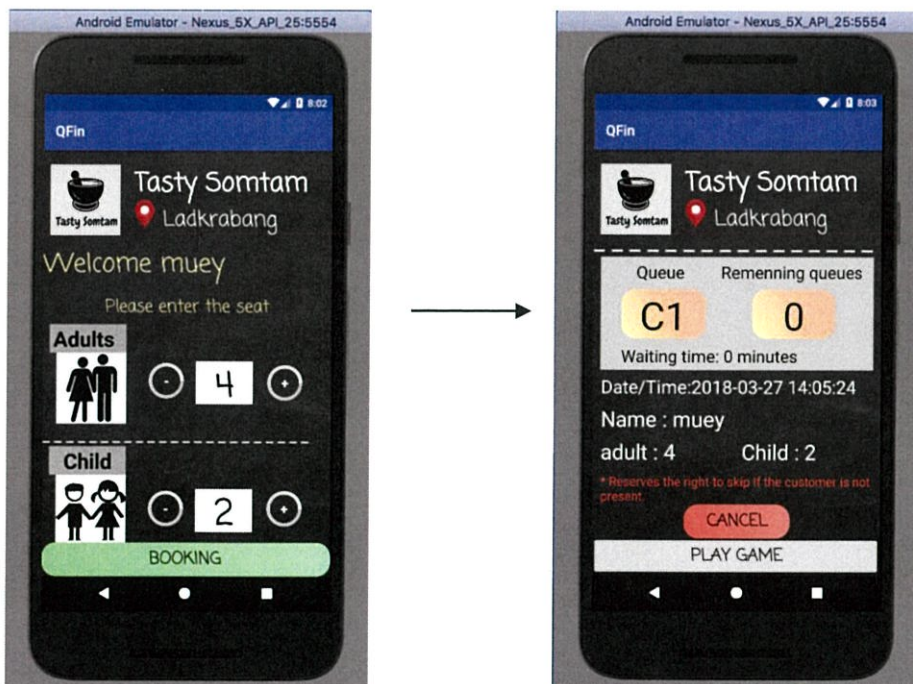
4.1.1 การทดสอบแอปพลิเคชัน (Application) บนโปรแกรมแอนดรอยด์สตูดิโอ (Android Studio) เริ่มแรกจะทำการล็อกอิน (Log in) กรอก username และ password โดยทำการคลิก “Log in” ในหน้า Register ถ้าถูกต้องจะเข้าสู่หน้าการจองคิว โดยใส่จำนวนเด็กและผู้ใหญ่ หลังจากทำการส่งข้อมูลจากแอปพลิเคชัน (Application) (ชื่อผู้จอง และจำนวนคน) มาที่เซิร์ฟเวอร์ (server) แล้วทำการประมวลผลผ่านอัลกอริทึม เพื่อแสดงผลคิวให้กับผู้ใช้งาน แล้วจะได้ดังรูปที่ 4.1(a) เป็นการแสดงผลคิวของโต๊ะประเภท 1-2 ท่าน รูปที่ 4.1(b) เป็นการแสดงผลคิวของโต๊ะประเภท 3-4 ท่าน รูปที่ 4.1(c) เป็นการแสดงผลคิวของโต๊ะประเภท 5-6 ท่าน และ รูปที่ 4.1(d)) เป็นการแสดงผลคิวของโต๊ะประเภท 7 ท่านขึ้นไป



รูปที่ 4.1 แสดงผลคิวของโต๊ะประเภท 1-2 ท่าน



รูปที่ 4.2 แสดงผลคิวของโต๊ะประเภท 3-4 ท่าน



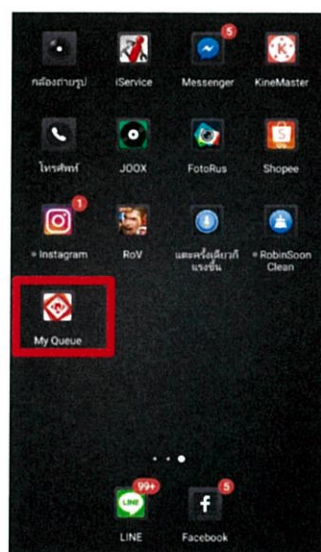
รูปที่ 4.3 แสดงผลคิวของโต๊ะประเภท 5-6 ท่าน



รูปที่ 4.4 แสดงผลคิวของโต๊ะประเภท 7 ท่านขึ้นไป

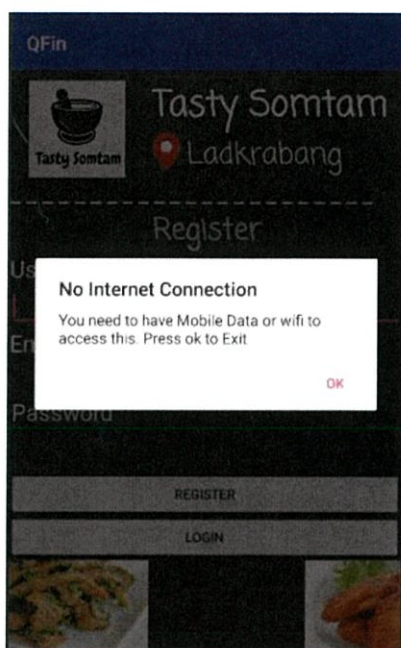
4.1.2 การทดสอบแอปพลิเคชัน (Application) บนโทรศัพท์จริง

ในส่วนของการทดสอบแอปพลิเคชัน (Application) ลงบนสมาร์ตโฟนในระบบปฏิบัติการแอนดรอยด์ (Android) จะเริ่มทดสอบจากการติดตั้งแอปพลิเคชันลงบนสมาร์ตโฟนโดยจะมีไอคอนของตัวแอปพลิเคชันขึ้นมาดังแสดงในรูปที่ 4.5

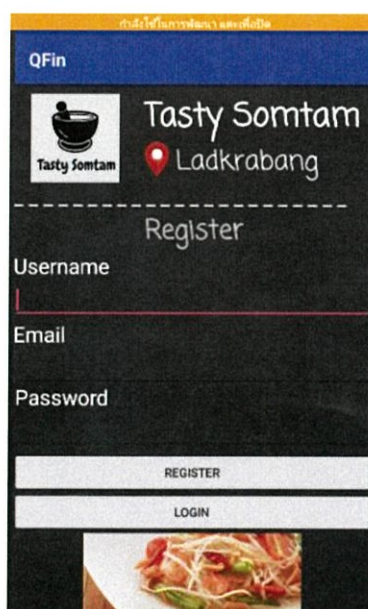


รูปที่ 4.5 ไอคอนของแอปพลิเคชัน (Application)

หากยังไม่เชื่อมต่ออินเทอร์เน็ต (Internet) จะขึ้นแจ้งเตือน ดังแสดงในรูปที่ 4.6

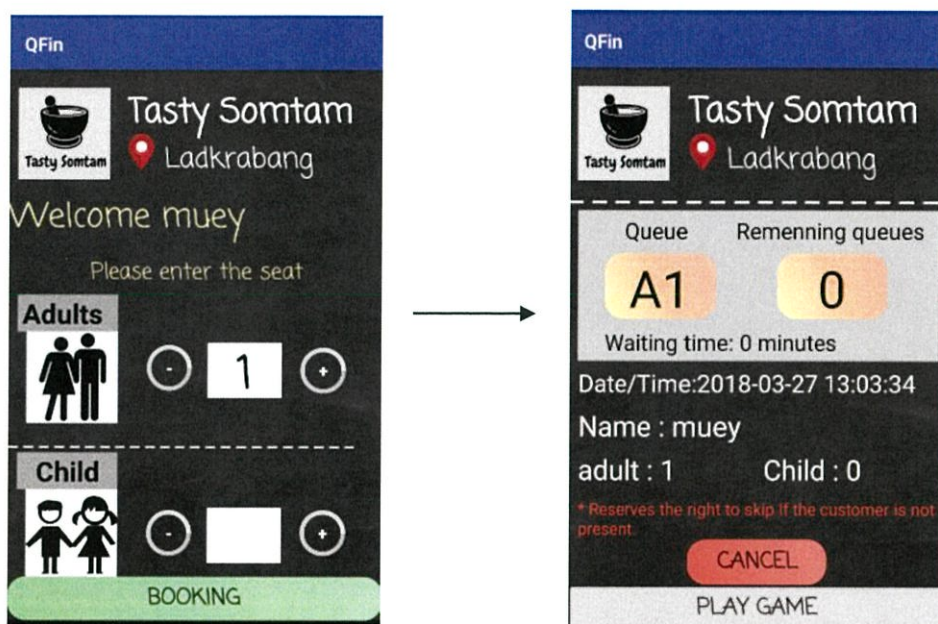


รูปที่ 4.6 การแจ้งเตือนการเชื่อมต่อ อินเทอร์เน็ต (Internet) เมื่อกดไอคอนแล้ว จะปรากฏหน้าสำหรับการล็อกอิน (Log in) ดังแสดงในรูปที่ 4.7



รูปที่ 4.7 หน้าเมนูหลัก

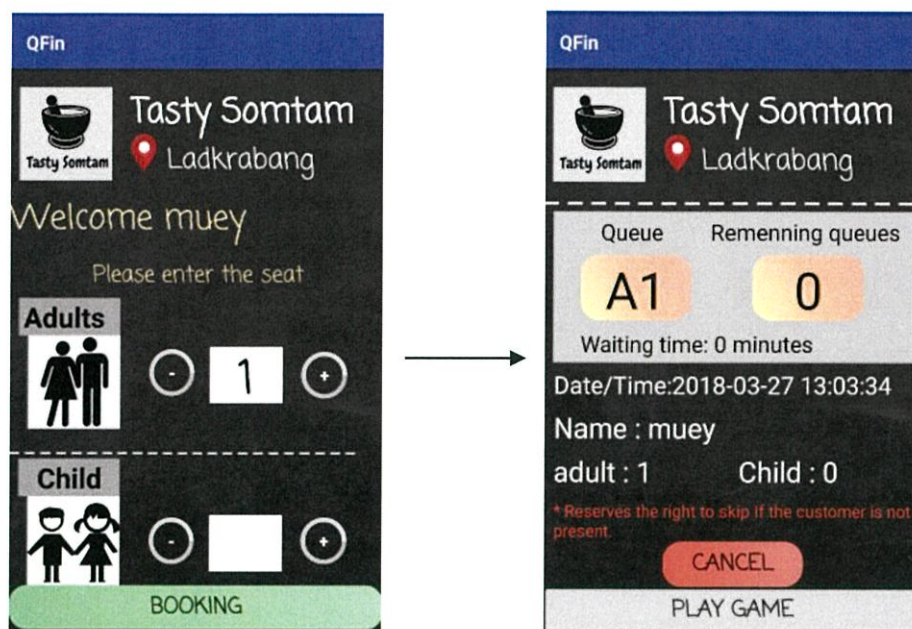
เมื่อผู้ใช้งานกรอก “จำนวนคน” แล้วคลิกปุ่ม “Booking” หน้าแอปพลิเคชัน (Application) จะแสดงคิวที่ได้ จำนวนคิวที่เหลือ เวลาที่จอง และระยะเวลาประมาณรอคิว ดังแสดงในรูปที่ 4.8



รูปที่ 4.8 หน้าแสดงผลคิวที่ได้

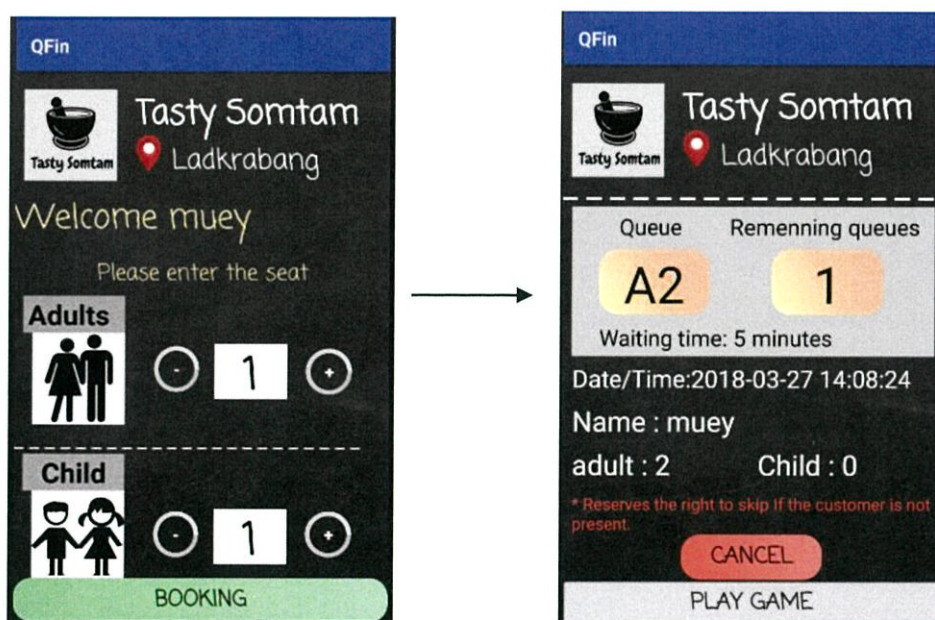
4.1.3 การทดสอบการประมาณระยะเวลารอคิว

จากที่กล่าวไปในบทที่ 3 การประมาณระยะเวลารอคิวจะพิจารณาจากจำนวนคิวที่รอเรียก (ค่า status = 0) ของคิวแต่ละประเภท ในที่นี้จะยกตัวอย่างผลการทดสอบของคิวประเภท A (โต๊ะประเภท 1-2 ท่าน) เมื่อทำการจองคิวครั้งแรกจะเห็นได้ว่าลำดับคิวที่ได้คือ A1 และระยะเวลาประมาณรอคิวเท่ากับ 0 นาที ดังแสดงในรูปที่ 4.9



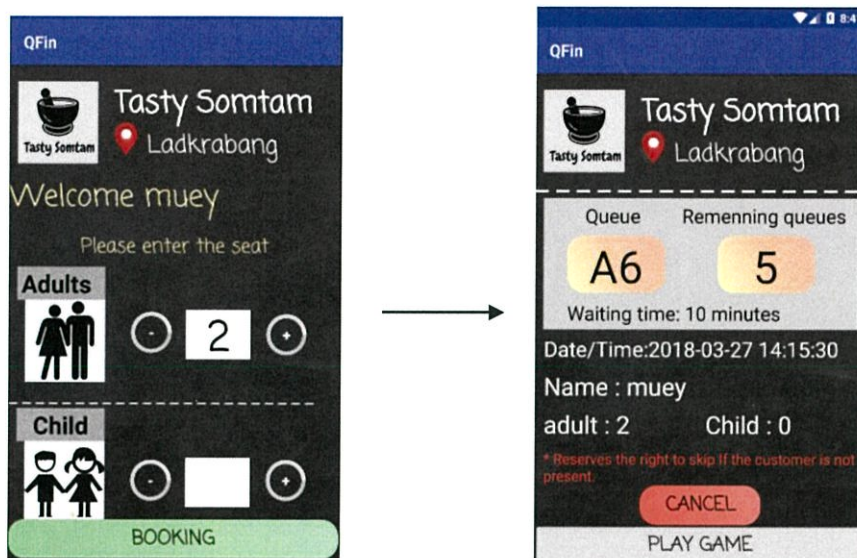
รูปที่ 4.9 หน้าแสดงผลระยะเวลารอคิว (จองครั้งที่ 1)

เมื่อทำการจองคิวครั้งที่ 2 จะเห็นว่ามียอดคิวที่รอเรียกอยู่ 1 คิว ดังนั้น ประมาณระยะเวลา
รอคิว คือ 5 นาที ดังแสดงในรูปที่ 4.10



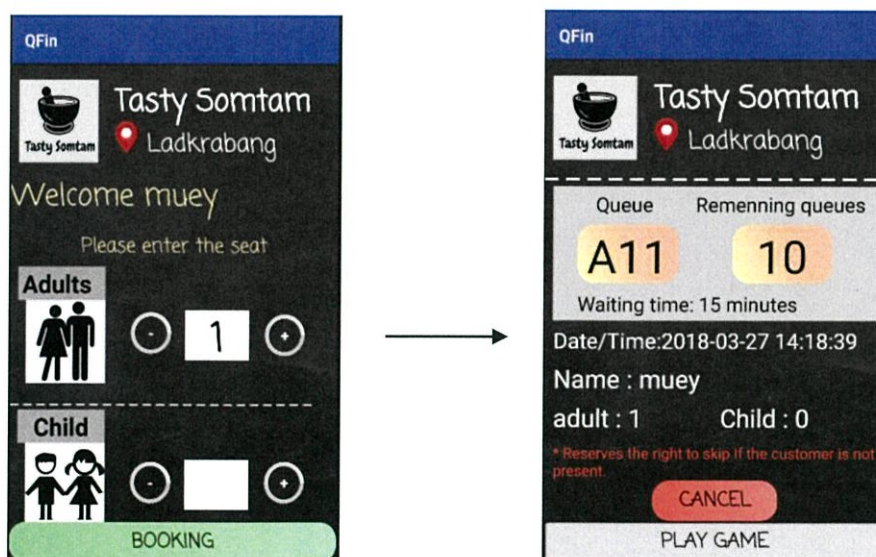
รูปที่ 4.10 หน้าแสดงผลระยะเวลารอคิว (จองครั้งที่ 2)

เมื่อทำการจองคิวครั้งที่ 6 จะเห็นว่าคิวที่รอเรียกอยู่ 5 คิว ดังนั้น ประมาณระยะเวลา
รอคิว คือ 10 นาที ดังแสดงในรูปที่ 4.11



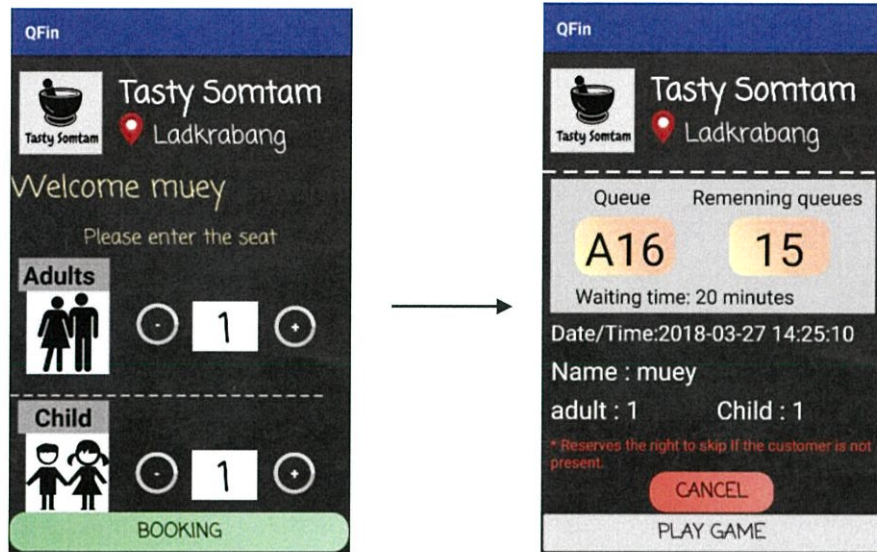
รูปที่ 4.11 หน้าแสดงผลระยะเวลารอคิว (จองครั้งที่ 6)

เมื่อทำการจองคิวครั้งที่ 11 จะเห็นว่าคิวที่รอเรียกอยู่ 10 คิว ดังนั้น ประมาณ
ระยะเวลารอคิว คือ 15 นาที ดังแสดงในรูปที่ 4.12



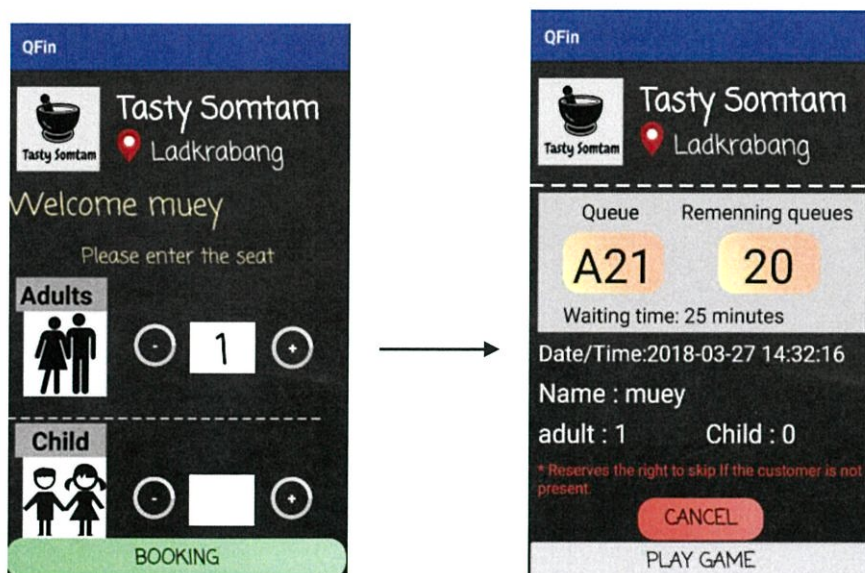
รูปที่ 4.12 หน้าแสดงผลระยะเวลารอคิว (จองครั้งที่ 11)

เมื่อทำการจองคิวครั้งที่ 16 จะเห็นว่าคิวที่รอเรียกอยู่ 15 คิว ดังนั้น ประมาณ
 ระยะเวลารอคิว คือ 20 นาที ดังแสดงในรูปที่ 4.13



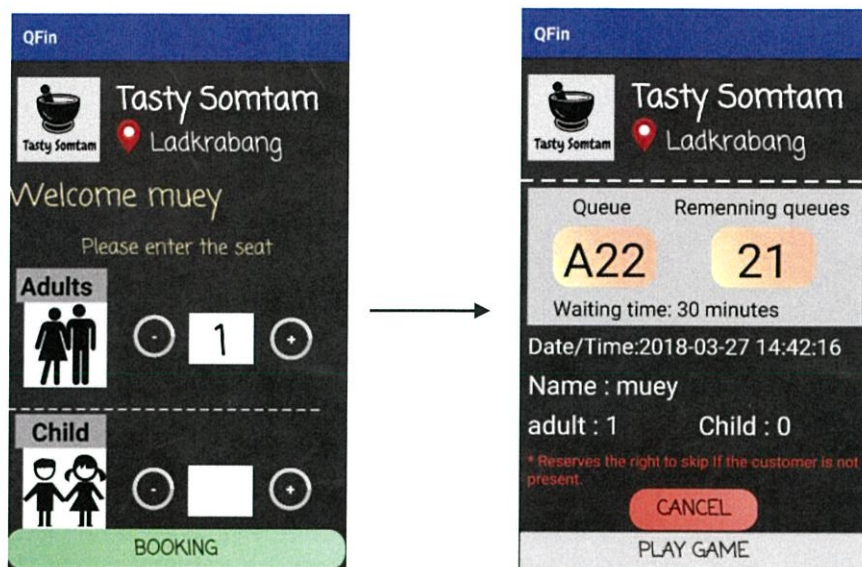
รูปที่ 4.13 หน้าแสดงผลระยะเวลารอคิว (จองครั้งที่ 16)

เมื่อทำการจองคิวครั้งที่ 21 จะเห็นว่าคิวที่รอเรียกอยู่ 20 คิว ดังนั้น ประมาณ
 ระยะเวลารอคิว คือ 25 นาที ดังแสดงในรูปที่ 4.14



รูปที่ 4.14 หน้าแสดงผลระยะเวลารอคิว (จองครั้งที่ 21)

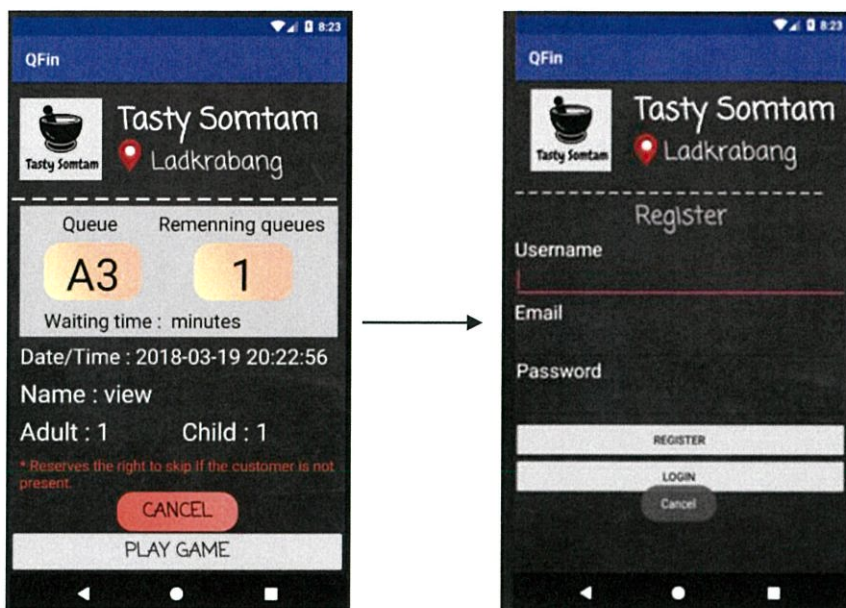
เมื่อทำการจองคิวครั้งที่ 22 จะเห็นว่าคิวที่รอเรียกอยู่มากกว่า 20 คิว ดังนั้น
ประมาณระยะเวลารอคิว คือ 30 นาที หรือมากกว่านั้น ดังแสดงในรูปที่ 4.15



รูปที่ 4.15 หน้าแสดงผลระยะเวลารอคิว (จองครั้งที่ 22)

4.1.4 การทดสอบการยกเลิกคิว

เมื่อผู้ใช้งานทำการล็อกอิน (log in) แล้วทำการจองคิว หลังจากได้คิวแล้ว ผู้ใช้งานต้องการยกเลิกคิว ผู้ใช้งานสามารถกดปุ่ม cancel เมื่อกดปุ่มนี้ ฝั่งแอปพลิเคชันจะทำการส่งค่าชื่อและคิวที่ได้ เข้าสู่เซิร์ฟเวอร์ (server) อีกครั้ง เพื่อหาข้อมูลที่มีค่าชื่อและคิวตรงกัน แล้วทำการอัปเดตสแตตัส (status) เท่ากับ 2 ซึ่งหมายถึงยกเลิกคิว เมื่อผู้ใช้งานทำการยกเลิกคิวแล้ว จะออกจากหน้าจองคิวแล้วมาที่หน้าล็อกอิน (log in) แล้วขึ้นข้อความว่า “cancel” ผลการทดสอบ ได้ผลดังแสดงในรูปที่ 4.16



รูปที่ 4.16 หน้าแสดงผลคิวและยกเลิกคิว

เมื่อกดปุ่ม “CANCEL” แล้ว จะเห็นว่าคิว A3 มีค่าสแตตัส (status) เปลี่ยนเป็น 2 ซึ่งหมายถึงยกเลิกคิว ดังแสดงในรูปที่ 4.17

	id	month	year	time_in	time_on_serve	name	people	adult	child	type	sequence	remenning_queue	queue	status	remennu
xy	Delete 43	Mar	2018	2018-03-19 20:17:50	0000-00-00 00:00:00	view	2	1	1	A	1	0	A1	0	
xy	Delete 44	Mar	2018	2018-03-19 20:18:14	0000-00-00 00:00:00	view	2	1	1	A	2	1	A2	2	
xy	Delete 45	Mar	2018	2018-03-19 20:22:56	0000-00-00 00:00:00	view	2	1	1	A	3	1	A3	2	
xy	Delete 46	Mar	2018	2018-03-19 20:23:39	0000-00-00 00:00:00	view	2	1	1	A	4	1	A4	0	

รูปที่ 4.17 คิว A3 เปลี่ยนค่าสแตตัส (status) เป็นยกเลิกคิว

4.1.5 การทดสอบเกมส์ Puzzle

สำหรับรูปแบบเกมส์บนแอปพลิเคชันที่ได้ออกแบบนั้น คือเกมส์ Puzzle ซึ่งจะเป็นส่วนหนึ่งของระบบจอร์นร้านอาหาร โดยใช้โปรแกรมแอนดรอยด์สตูดิโอ (android studio) ถ้าผู้ใช้งานสามารถชนะเกมส์ได้ ผู้ใช้งานจะสามารถรับรางวัลจากทางร้านอาหาร เช่น ส่วนลดราคาอาหาร , ของแถมจากทางร้าน เป็นต้น โดยตัวอย่างรูปที่ผู้ออกแบบใช้สำหรับสร้างเกมส์คือ รูปตรา

ประจำร้านอาหาร Tasty Somtum ได้ออกแบบให้มีขนาด 3 x 3 ดังนั้นจะต้องทำการแบ่งภาพออกเป็น 9 ส่วน ดังแสดงในรูปที่ 4.14 รูปแบบเกมส์ puzzle ได้ออกแบบให้มีขนาด 3 x 3 ดังนั้นจะต้องทำการแบ่งภาพออกเป็น 9 ส่วน ดังแสดงในรูปที่ 4.18



รูปที่ 4.18 ชิ้นส่วนรูปภาพขนาด 3 x 3

เมื่อผู้ใช้งานทำการเปิดเกมส์โดยคลิกปุ่ม “Play Game” จะเห็นได้ว่ารูปตราประจำร้านจะสลับกัน ดังแสดงในรูปที่ 4.19



รูปที่ 4.19 เกมส puzzle

ผู้ใช้งานจะต้องทำการสลับตำแหน่งรูปภาพภายในเวลาอันจำกัดให้ถูกต้อง เมื่อการเรียงภาพถูกต้องแล้ว จะมีข้อความแสดงขึ้นมาว่า “You Win” แล้วจะโชว์รางวัลที่ได้รับ ดังแสดงในรูปที่ 4.20



รูปที่ 4.20 เกมส puzzle เมื่อเล่นเกมส์ชนะ

4.2 การทดสอบการทำงานของเว็บแสดงคิว

การทำงานของเว็บแสดงคิวจะใช้วิธีการดึงค่าจากฐานข้อมูล เพื่อนำค่าลำดับคิว ของโต๊ะแต่ละประเภท และจำนวนคิวที่เหลือมาแสดงแก่ผู้ใช้งาน ดังรูปที่ 4.21

	id	time_in	time_on_serve	name	people	type	sequence	remenning_queue	queue	status
<input type="checkbox"/>	1	2017-12-08 17:25:40	0000-00-00 00:00:00	muey	1	A	1	0	A1	1
<input type="checkbox"/>	2	2017-12-08 17:26:12	0000-00-00 00:00:00	view	3	B	1	0	B1	1
<input type="checkbox"/>	3	2017-12-08 17:27:18	0000-00-00 00:00:00	Pim	6	C	1	0	C1	1
<input type="checkbox"/>	4	2017-12-08 17:27:51	0000-00-00 00:00:00	Big	10	D	1	0	D1	1
<input type="checkbox"/>	5	2017-12-08 17:45:10	0000-00-00 00:00:00	Fang	2	A	2	0	A2	0

รูปที่ 4.21 ข้อมูลในฐานข้อมูล (database) สำหรับทดสอบเว็บแสดงคิว

จากรูปที่ 4.17 จะเห็นได้ว่า ข้อมูลที่อยู่ในฐานข้อมูล (database) นี้ มาจากผู้ใช้งานที่ทำการจองคิวผ่านแอปพลิเคชัน ซึ่งผ่านการประมวลผลคิวที่เซิร์ฟเวอร์ดังที่ได้กล่าวไว้ในบทที่ 3 แล้วทำการส่งข้อมูลมาที่ฐานข้อมูล (database) ซึ่งในตอนแรกค่า status ของข้อมูลเท่ากับ 0 ซึ่งหมายถึง รอเรียกคิว โดยเมื่อแอดมิน (พนักงาน) กดปุ่มรันคิว ผ่านเว็บแอดมิน (admin web) ค่า status ของข้อมูลจะเปลี่ยนเป็น 1 ซึ่งหมายถึง คิวนี้ถูกเรียก ซึ่งค่าทั้ง 4 ค่านี้คือคิวที่แสดงผ่านหน้าเว็บแสดงคิว เพื่อเรียกผู้ใช้งานที่เป็นเจ้าของคิวเข้าร้าน ดังแสดงในรูปที่ 4.22



รูปที่ 4.22 เว็บแสดงคิว

จากรูปที่ 4.18 จะเห็นได้ว่า ช่องทั้ง 4 ช่อง จะแสดงผลคิวที่ถูกเรียก (status = 1) แบบเรียลไทม์ และพบว่า โต๊ะประเภท 1-2 ท่าน มีจำนวนคิวที่เหลืออีก 1 คิว เนื่องจากในฐานข้อมูล (database) มีคิวของโต๊ะประเภท 1-2 ท่าน อีกหนึ่งคิวที่รอเรียกคิวอยู่ (status = 0) ซึ่งก็คือคิว A2 นั้นเอง

4.3 การทดสอบการทำงานของเว็บแอดมิน

การทำงานของเว็บแสดงคิวจะใช้วิธีการดึงค่าจากฐานข้อมูล (database) เพื่อนำค่าลำดับคิว ของโต๊ะแต่ละประเภท และจำนวนคิวที่เข้าใช้บริการทั้งหมดมาแสดงแก่ผู้ดูแลระบบ ดังรูปที่ 4.23

	id	time_in	time_on_serve	name	people	type	sequence	remenning_queue	queue	status
<input type="checkbox"/>	22	2017-12-10 14:40:48	0000-00-00 00:00:00	Pong	1	A	1	0	A1	0
<input checked="" type="checkbox"/>	23	2017-12-10 14:41:03	0000-00-00 00:00:00	view	2	A	2	1	A2	0
<input type="checkbox"/>	24	2017-12-10 14:41:11	0000-00-00 00:00:00	Nut	2	A	3	2	A3	0
<input type="checkbox"/>	25	2017-12-10 14:41:22	0000-00-00 00:00:00	Dan	1	A	4	3	A4	0
<input type="checkbox"/>	26	2017-12-10 14:41:37	0000-00-00 00:00:00	Pang	1	A	5	4	A5	0
<input type="checkbox"/>	27	2017-12-10 14:41:48	0000-00-00 00:00:00	Poy	2	A	6	5	A6	0
<input type="checkbox"/>	28	2017-12-10 14:42:12	0000-00-00 00:00:00	Cham	2	A	7	6	A7	0
<input type="checkbox"/>	29	2017-12-10 14:43:02	0000-00-00 00:00:00	Ang	4	B	1	0	B1	0
<input type="checkbox"/>	30	2017-12-10 14:43:37	0000-00-00 00:00:00	Ping	3	B	2	1	B2	0
<input type="checkbox"/>	31	2017-12-10 14:43:49	0000-00-00 00:00:00	kai	3	B	3	2	B3	0
<input type="checkbox"/>	32	2017-12-10 14:44:04	0000-00-00 00:00:00	Downy	4	B	4	3	B4	0
<input type="checkbox"/>	33	2017-12-10 14:44:57	0000-00-00 00:00:00	Jan	5	C	1	0	C1	0
<input type="checkbox"/>	34	2017-12-10 14:45:20	0000-00-00 00:00:00	fang	5	C	2	1	C2	0
<input type="checkbox"/>	35	2017-12-10 14:45:44	0000-00-00 00:00:00	Gig	7	D	1	0	D1	0

รูปที่ 4.23 ข้อมูลในฐานข้อมูล (database) สำหรับทดสอบเว็บแอดมิน (admin web)

จากรูปที่ 4.23 การทำงานของเว็บแอดมินสามารถอ้างอิงได้จากฐานข้อมูล ดังในรูปที่ 4.24 ดังนี้

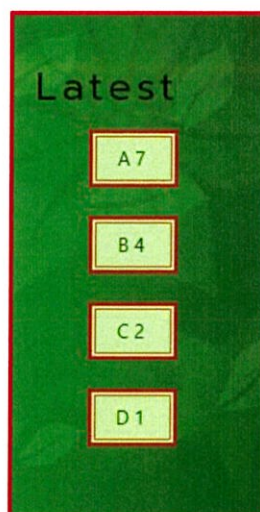


รูปที่ 4.24 การทำงานของเว็บแอดมิน (admin web)

จากรูปที่ 4.24 จะเห็นว่าการทำงานของเว็บแอดมินสัมพันธ์กับข้อมูลในฐานข้อมูล กล่าวคือ ในช่องสีเขียวซึ่งคือส่วนแสดงคิวที่รอเรียก (status ของข้อมูลมีค่าเป็น 0) แยกตามประเภทของโต๊ะ

โต๊ะประเภท 1-2 ท่าน (type 'A')	มีคิวที่รอเรียก 7 คิว
โต๊ะประเภท 3-4 ท่าน (type 'B')	มีคิวที่รอเรียก 4 คิว
โต๊ะประเภท 5-6 ท่าน (type 'C')	มีคิวที่รอเรียก 2 คิว
โต๊ะประเภท 7 ท่านขึ้นไป (type 'D')	มีคิวที่รอเรียก 1 คิว

สำหรับช่องสีเขียวกรอบสีแดงจะแสดงคิวสุดท้ายที่ทำการจองแยกตามโต๊ะแต่ละประเภท เพื่อให้พนักงานหรือเจ้าของร้านทราบว่า มีความหนาแน่นของคิวตามโต๊ะแต่ละประเภทมากน้อยเพียงใด ดังแสดงในรูปที่ 4.25



รูปที่ 4.25 แสดงคิวสุดท้ายที่ทำการจองของโต๊ะแต่ละประเภท

สำหรับช่องแสดงจำนวนคิวที่ใช้บริการ จะเห็นได้ว่ามีค่าเป็นศูนย์เนื่องจากยังไม่มีคิวใดที่ถูกเรียกเข้าร้าน แต่เมื่อแอดมิน (พนักงาน) ทำการกดปุ่มรันคิวประเภทโต๊ะ 1-2 ท่าน จะทำให้คิว

A1 ถูกเรียกเป็นคิวแรก แล้วจะถูกย้ายไปในช่องสี่เหลี่ยมซึ่งหมายถึงคิวที่ถูกเรียกแล้ว (status = 1)

รวมถึงช่องแสดงจำนวนคิวที่ให้บริการจะมีค่าเพิ่มขึ้นมาเป็น 1 ดังแสดงในรูปที่ 4.26 และ 4.27

	id	time_in	time_on_serve	name	people	type	sequence	remenning_queue	queue	status
<input type="checkbox"/>	22	2017-12-10 14:40:48	2017-12-10 15:18:25	Pong	1	A	1	0	A1	1
<input type="checkbox"/>	23	2017-12-10 14:41:03	0000-00-00 00:00:00	view	2	A	2	1	A2	0
<input type="checkbox"/>	24	2017-12-10 14:41:11	0000-00-00 00:00:00	Nut	2	A	3	2	A3	0
<input type="checkbox"/>	25	2017-12-10 14:41:22	0000-00-00 00:00:00	Dan	1	A	4	3	A4	0
<input type="checkbox"/>	26	2017-12-10 14:41:37	0000-00-00 00:00:00	Pang	1	A	5	4	A5	0
<input type="checkbox"/>	27	2017-12-10 14:41:48	0000-00-00 00:00:00	Poy	2	A	6	5	A6	0
<input type="checkbox"/>	28	2017-12-10 14:42:12	0000-00-00 00:00:00	Cham	2	A	7	6	A7	0
<input type="checkbox"/>	29	2017-12-10 14:43:02	0000-00-00 00:00:00	Ang	4	B	1	0	B1	0
<input type="checkbox"/>	30	2017-12-10 14:43:37	0000-00-00 00:00:00	Ping	3	B	2	1	B2	0
<input type="checkbox"/>	31	2017-12-10 14:43:49	0000-00-00 00:00:00	kai	3	B	3	2	B3	0
<input type="checkbox"/>	32	2017-12-10 14:44:04	0000-00-00 00:00:00	Downy	4	B	4	3	B4	0
<input type="checkbox"/>	33	2017-12-10 14:44:57	0000-00-00 00:00:00	Jan	5	C	1	0	C1	0
<input type="checkbox"/>	34	2017-12-10 14:45:20	0000-00-00 00:00:00	fang	5	C	2	1	C2	0
<input type="checkbox"/>	35	2017-12-10 14:45:44	0000-00-00 00:00:00	Gig	7	D	1	0	D1	0

รูปที่ 4.26 status ของคิว A1

จากรูปที่ 4.22 จะเห็นได้ว่าหลังจากกดปุ่มรันคิวโต๊ะประเภท 1-2 ท่าน ระบบจะทำการอัปเดต (update) คิว A1 ให้ค่า status เปลี่ยนเป็น 1 (หมายถึงคิวที่ถูกเรียก) รวมถึงอัปเดตเวลาที่เข้าร้าน ซึ่งก็คือ ค่า time_on_serve

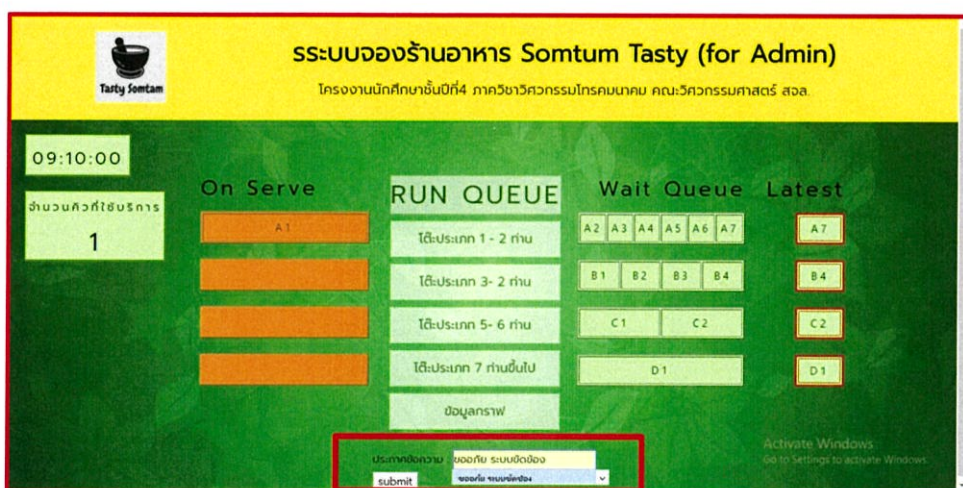
รูปที่ 4.27 การทำงานของเว็บแอดมินหลังคลิกปุ่มรันคิวประเภทโต๊ะ 1-2 ท่าน

จากรูปที่ 4.27 คิวที่ถูกเรียกคือคิว A1 ดังนั้น คิว A1 จะถูกนำไปแสดงบนเว็บไซต์ แสดงคิวด้วย ดังแสดงในรูปที่ 4.28



รูปที่ 4.28 เว็บไซต์แสดงผลคิวหลังกดปุ่มรันคิวโต๊ะประเภท 1-2 ท่าน

ในเว็บไซต์แอดมิน แอดมิน (พนักงานหรือเจ้าของร้าน) สามารถประกาศข้อความผ่านระบบได้ เช่น “ขออภัย ระบบขัดข้อง” ดังแสดงในรูปที่ 4.29



รูปที่ 4.29 ประกาศข้อความผ่านระบบแอดมิน

จากรูปที่ 4.29 เมื่อแอดมิน (พนักงานหรือเจ้าของร้าน) กดปุ่ม “submit” ข้อความที่ต้องการประกาศจะไปปรากฏ ที่เว็บแสดงผลคิว ดังรูปที่ 4.30



รูปที่ 4.30 เว็บแสดงผลคิวหลังจากประกาศข้อความผ่านเว็บแอดมิน

หากแอดมินต้องการดูข้อมูลกราฟ ให้คลิกปุ่ม “ข้อมูลกราฟ” บนหน้าเว็บแอดมิน (admin web) เมื่อเข้ามาแล้วแอดมินสามารถเลือกช่วงเวลาที่ต้องการพล็อตได้ โดยข้อมูลจากฐานข้อมูล เป็นไปดังรูปที่ 4.31

+ Options

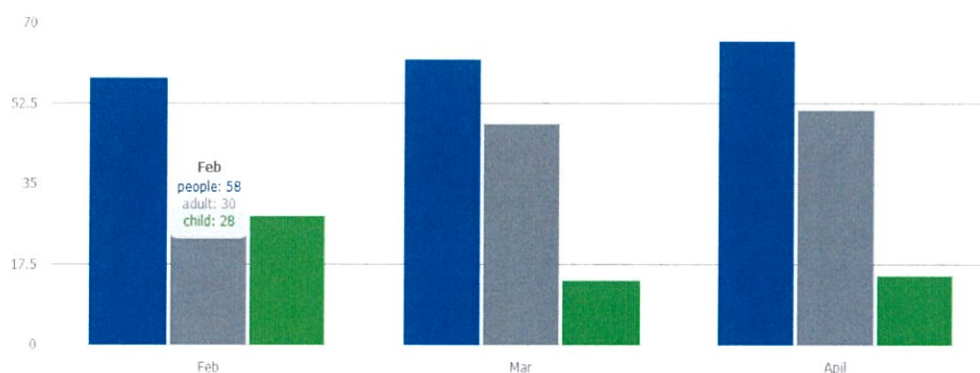
				id	month	year	adult	child	people	A	B	C	D
<input type="checkbox"/>	Edit	Copy	Delete	1	Feb	2018	30	28	58	10	8	8	1
<input type="checkbox"/>	Edit	Copy	Delete	2	Mar	2018	48	14	62	19	5	3	1
<input type="checkbox"/>	Edit	Copy	Delete	3	April	2018	51	15	66	15	8	4	3

↑ Check all With selected: Edit Copy Delete Export

รูปที่ 4.31 ข้อมูลจากฐานข้อมูล (database) ที่นำมาพล็อต

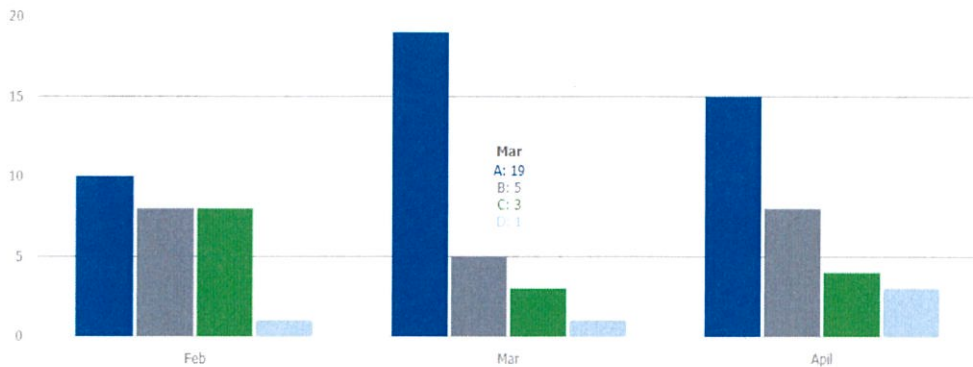
เมื่อแอดมินต้องการกราฟประเภทที่ 1 และ 2 (กล่าวไว้ในบทที่ 3) ในช่วงเดือนมกราคม ถึง เดือนเมษายน ปี 2017 แอดมินทำการเลือกช่วงเวลาที่ต้องการผ่านหน้าเว็บแอดมิน (admin web) ดังแสดงในรูปที่ 4.32

รูปที่ 4.32 เลือกช่วงเวลาที่ต้องการพล็อตผ่านหน้าเว็บแอดมิน (admin web) เมื่อกดปุ่ม “submit” จะได้กราฟดังแสดงในรูปที่ 4.33 และ 4.34



รูปที่ 4.33 กราฟประเภทที่ 1 แสดงความสัมพันธ์จำนวนคนแต่ละประเภทในแต่ละเดือน

จากรูปที่ 4.33 จะเห็นว่า แกน X คือ ช่วงเดือนที่แอดมินเลือก ส่วนแกน Y จะแสดงจำนวนคน ทั้งหมดในเดือนนั้น (ตัวแปร people) โดยจะแยกเป็นจำนวนผู้ใหญ่ (ตัวแปร adult) และจำนวนผู้เด็ก (ตัวแปร child) ทำให้ให้แอดมินได้ทราบแนวโน้มของจำนวนคนที่เข้าร้าน เพื่อใช้ในการวางแผนงานได้ เช่น การนำเสนอโปรโมชั่น, การเพิ่มหรือลดจำนวนพนักงาน เป็นต้น

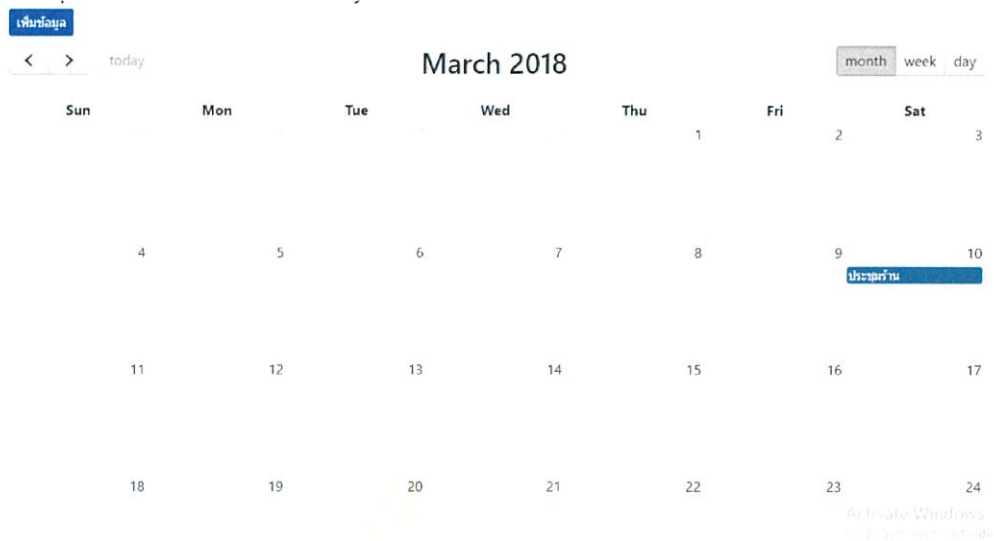


รูปที่ 4.34 กราฟประเภทที่ 2 แสดงความสัมพันธ์จำนวนโต๊ะแต่ละประเภทในแต่ละเดือน

จากรูปที่ 4.34 จะเห็นว่า แกน x คือ ช่วงเดือนที่แอดมินเลือก ส่วนแกน Y จะแสดงจำนวนโต๊ะ โดยจะแยกเป็นโต๊ะประเภท A , ประเภท B, ประเภท C และ ประเภท D ทำให้แอดมินทราบแนวโน้มประเภทโต๊ะที่มีการใช้งานมาก หรือน้อย

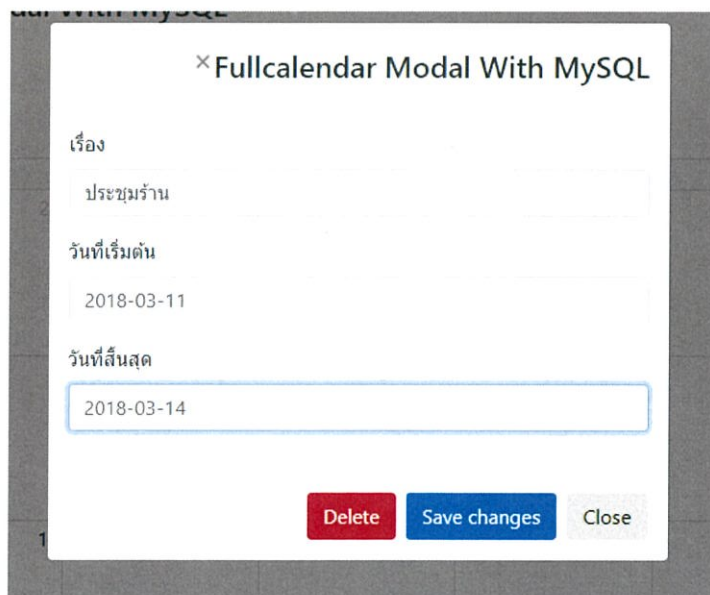
นอกจากนี้ในหน้าเว็บแอดมิน (admin web) ยังมีหน้าปฏิทินกิจกรรม เพื่อให้พนักงานได้ทราบรายละเอียดแผนงานต่างๆของร้าน เมื่อเพิ่มข้อมูล ใน Bootstrap Modal หัวข้อคือ “ประชุมร้าน” ในวันที่ 10 มีนาคม 2018 แล้วกดปุ่ม “บันทึกข้อมูล” จะขึ้นข้อความว่า “บันทึกข้อมูลเรียบร้อยแล้ว” ดังแสดงในรูปที่ 4.35 และ 4.36

รูปที่ 4.35 เพิ่มข้อมูลผ่าน Bootstrap Modal



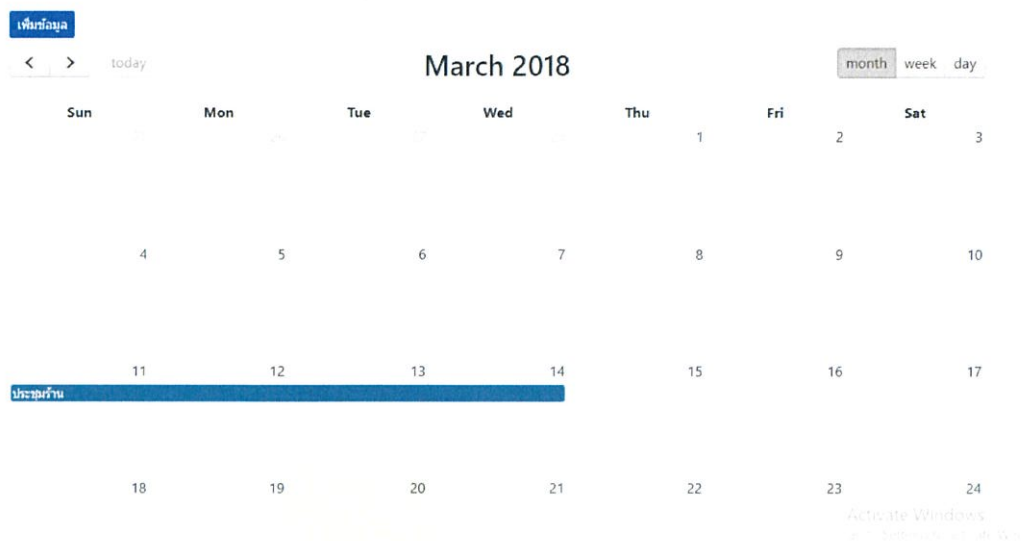
รูปที่ 4.36 ปฏิทินกิจกรรมหลังจากเพิ่มข้อมูล

แอดมินสามารถทำการแก้ไขข้อมูลและลบข้อมูลได้ โดยคลิกวันที่ต้องการแก้ไข/ลบข้อมูล ดังแสดงในรูปที่ 4.37 จะทำการแก้ไขข้อมูล โดยเปลี่ยนช่วงเวลาตั้งแต่วันที่ 11 ถึง 14 มีนาคม 2018 แล้วคลิกปุ่ม “Save change” หรือหากต้องการลบข้อมูลให้คลิกปุ่ม “Delete”

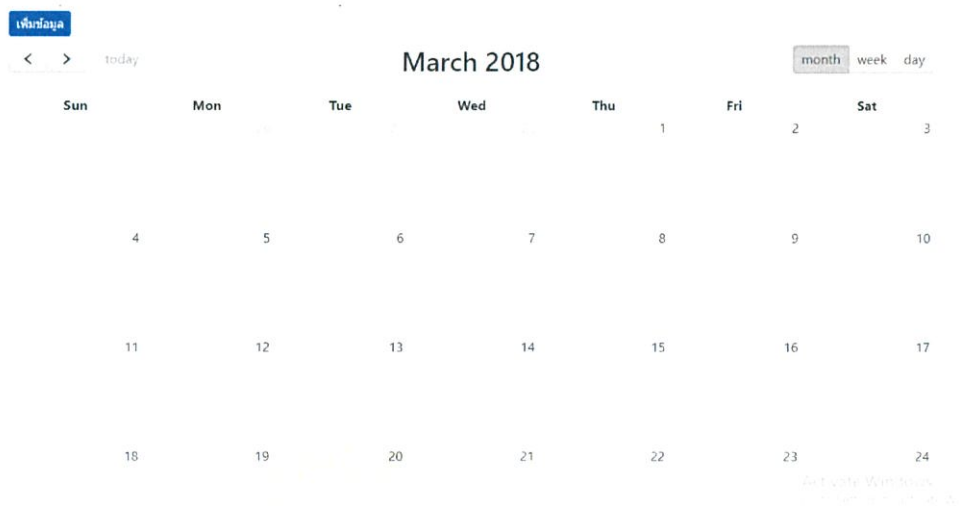


รูปที่ 4.37 แก้ไขข้อมูลในปฏิทินกิจกรรม

ผลการทดสอบหลังการแก้ไขข้อมูลได้ผลดังแสดงในรูปที่ 4.38



รูปที่ 4.38 ปฏิทินกิจกรรมหลังจากแก้ไขข้อมูล
ผลการทดสอบหลังการลบข้อมูลได้ผลดังแสดงในรูปที่ 4.39



รูปที่ 4.39 ปฏิทินกิจกรรมหลังจากลบข้อมูล

4.4 การทดสอบอุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi) สำหรับการสร้างเซิร์ฟเวอร์ (Server)

สำหรับการทดสอบอุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi) (สำหรับการติดตั้งระบบปฏิบัติการ ได้กล่าวไว้ในบทที่ 2) นั้นมีขั้นตอนดังต่อไปนี้

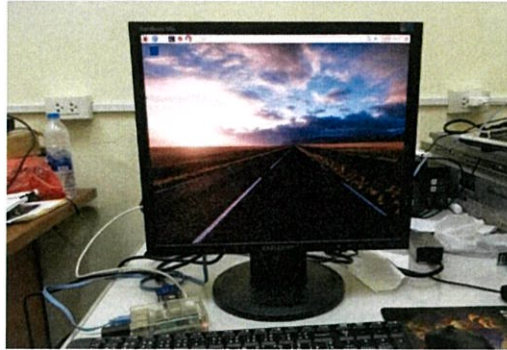
1) นำ sd card ที่เราได้ทำการลงระบบปฏิบัติการใส่ลงบนอุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi)

2) เปิดใช้งานราสเบอร์รี่พาย (Raspberry Pi) โดยเสียบ USB Mouse, USB Keyboard ,Ethernet เสียบสาย HDMI ให้ Raspberry Pi กับหน้าจอ และเสียบสาย Micro USB เข้ากับ USB Power Adapter และทำการจ่ายไฟ เมื่อเสียบอุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi) จะขึ้นไฟสีเขียวกะพริบและสีแดง ดังแสดงในรูปที่ 4.40



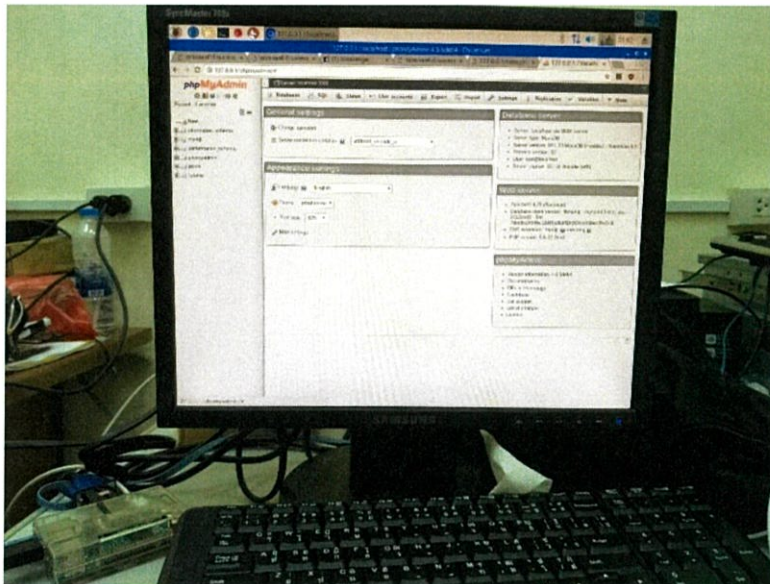
รูปที่ 4.40 ราสเบอร์รี่พาย (Raspberry Pi) เสียบอุปกรณ์ต่างๆ

หลังจากทำการเปิดอุปกรณ์เรียบร้อยแล้ว จะได้ผลเป็นดังรูปที่ 4.41



รูปที่ 4.41 หน้าจอของอุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi) เมื่อเปิดใช้งาน

หลังจากลงโปรแกรม Apache, phpmyadmin สำหรับสร้างเป็นเซิร์ฟเวอร์แล้ว จะ
ผลเป็นดังรูปที่ 4.42



รูปที่ 4.42 เซิร์ฟเวอร์บนอุปกรณ์ราสเบอร์รี่พาย (Raspberry Pi)

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

ในโครงการนี้มีการออกแบบและสร้างอัลกอริทึมสำหรับการประมวลผลคิว ระบบยกเลิควิวและเกมส์ Puzzle เพื่อสร้างระบบจองคิวร้านอาหารผ่านทางแอปพลิเคชันของสมาร์ตโฟนบนระบบปฏิบัติการแอนดรอยด์ (Android) โดยการจำลองร้านอาหารหนึ่งร้านที่แบ่งคิวตามประเภทของโต๊ะซึ่งใช้อุปกรณ์ราสเบอร์รี่พาย (Raspberry pi) สำหรับเป็นเซิร์ฟเวอร์ โครงการนี้มีการออกแบบการสร้างเว็บไซต์แสดงผลคิวสำหรับผู้ใช้งานและบอกจำนวนคิวที่เหลือของโต๊ะแต่ละประเภทรวมถึงการสร้างเว็บไซต์สำหรับผู้ดูแลระบบ (พนักงาน หรือ เจ้าของร้าน) เพื่อที่จะสามารถบริหารจัดการคิวได้

จากการทดสอบระบบจองคิวร้านอาหาร สามารถจองคิวผ่านทางแอปพลิเคชันแอนดรอยด์ (Android Application) ได้ สามารถแสดงผลคิว บอกจำนวนคิวที่เหลือของโต๊ะแต่ละประเภทผ่านเว็บไซต์ได้อย่างถูกต้อง สามารถทำการยกเลิควิวได้ และเล่นเกมส่ระหว่างรอคิวเพื่อให้ได้รางวัลได้ รวมถึงเว็บแอดมิน (Admin Web) สำหรับผู้ดูแลระบบ สามารถดูแล บริหารจัดการคิว และบอกจำนวนคิวที่เข้าใช้บริการทั้งหมด พร้อมทั้งสามารถประกาศข่าวสารและแจ้งเตือนแก่ผู้ใช้งานและพนักงานได้อย่างทั่วถึง

5.2 ข้อเสนอแนะ

- 1) ควรพัฒนารูปแบบแอปพลิเคชันแอนดรอยด์ (android) เพื่อให้สะดวกต่อการใช้งานมากขึ้น เช่น สามารถเลือกสาขาได้ ,การโฆษณาโปรโมชั่น เป็นต้น
- 2) ควรพัฒนารูปแบบเกมส์ให้มีความหลากหลายมากยิ่งขึ้น
- 3) ควรพัฒนาเว็บแสดงผลคิว เช่น สามารถเข้าไปดูเมนูอาหารได้, มีวิดีโอที่เกี่ยวข้องกับร้าน เป็นต้น รวมถึงออกแบบเว็บแสดงผลคิวให้มีความสวยงามมากขึ้น เพื่อดึงดูดลูกค้า

บรรณานุกรม

- [1] พร้อมเลิศ หล่อวิจิตร. *คู่มือเขียนแอป Android ด้วย Android Studio : การติดตั้ง Android Studio และภาษาจาวา (Java)*. กรุงเทพฯ : โปรวิชั่น, 2558.
- [2] นพ มหิษานนท์. *การติดตั้งและใช้งาน Raspberry Pi : การติดตั้งระบบปฏิบัติการสำหรับ Raspberry บน Micro SD Card*. กรุงเทพฯ : คอร์ฟิงก์ชั่น, 2557.
- [3] พร้อมเลิศ หล่อวิจิตร. *คู่มือเรียน PHP และ MySQL*. กรุงเทพฯ : โปรวิชั่น, บจก. 2558.
- [4] ศุภชัย สมพานิช. *คู่มือพัฒนาแอปพลิเคชันด้วย Android Studio ฉบับโปรแกรมเมอร์*. นนทบุรี : ไอดีซีฯ, 2559.
- [5] บัญชา ปะสีละเตสัง. *การเขียนโปรแกรม Java และ Android*. กรุงเทพฯ : ซีเอ็ดดูเคชั่น, 2559.
- [6] เกษม กมลชัยพิสิฐ. *รอบรู้ ประยุกต์ใช้ SQL Server*. กรุงเทพฯ : จุฬาลงกรณ์มหาวิทยาลัย, 2559.
- [7] ศุภชัย สมพานิช. *คู่มือการเขียนโปรแกรม Java สำหรับ Android Apps*. กรุงเทพฯ : สวีสวี ไอที, 2559.
- [8] บัญชา ปะสีละเตสัง. *พัฒนาเว็บแอปพลิเคชันด้วย PHP ร่วมกับ MySQL และ jQuery*. กรุงเทพฯ : ซีเอ็ดดูเคชั่น, 2558.
- [9] นพ มหิษานนท์. *การติดตั้งและใช้ Raspberry Pi*. นนทบุรี : คอร์ฟิงก์ชั่น, 2560.

ภาคผนวก ก

โค้ดสำหรับแอปพลิเคชันแอนดรอยด์ในการจองร้านอาหาร

MainActivity.java

```
public class MainActivity extends AppCompatActivity implements
View.OnClickListener{

private static final String REGISTER_URL = "http://10.0.2.2:8080/QQ/volleyRegister.php";
    public static final String KEY_USERNAME = "username";
    public static final String KEY_PASSWORD = "password";
    public static final String KEY_EMAIL = "email";
    private EditText editTextUsername;
    private EditText editTextEmail;
    private EditText editTextPassword;

    private Button buttonRegister,buttonLogin;

    ViewPager viewPager;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        //connect Internet

        if(!isConnected(MainActivity.this)) buildDialog(MainActivity.this).show();
        else { }

        editTextUsername = (EditText) findViewById(R.id.editTextUsername);
        editTextPassword = (EditText) findViewById(R.id.editTextPassword);
        editTextEmail = (EditText) findViewById(R.id.editTextEmail);
```

```
buttonRegister = (Button) findViewById(R.id.buttonRegister);
buttonLogin = (Button) findViewById(R.id.buttonLogin);
buttonRegister.setOnClickListener(this);
buttonLogin.setOnClickListener(this);
```

```
//view pager
```

```
viewPager = (ViewPager) findViewById(R.id.viewPager);
ViewPagerAdapter viewPagerAdapter = new ViewPagerAdapter(this);
viewPager.setAdapter(viewPagerAdapter);
```

```
Timer timer = new Timer();
timer.scheduleAtFixedRate(new MainActivity.MyTimerTask(), 2000, 2000);
```

```
}
```

```
//connect Internet
```

```
public boolean isConnected(Context context) {
```

```
    ConnectivityManager cm = (ConnectivityManager)
context.getSystemService(Context.CONNECTIVITY_SERVICE);
    NetworkInfo netinfo = cm.getActiveNetworkInfo();
```

```
    if (netinfo != null && netinfo.isConnectedOrConnecting()) {
        android.net.NetworkInfo wifi =
cm.getNetworkInfo(ConnectivityManager.TYPE_WIFI);
        android.net.NetworkInfo mobile =
cm.getNetworkInfo(ConnectivityManager.TYPE_MOBILE);
```

```
        if((mobile != null && mobile.isConnectedOrConnecting()) || (wifi != null &&
```

```

wifi.isConnectedOrConnecting())) return true;
    else return false;
} else
    return false;
}
//Alert box for connect internet

public AlertDialog.Builder buildDialog(Context c) {

    AlertDialog.Builder builder = new AlertDialog.Builder(c);
    builder.setTitle("No Internet Connection");
    builder.setMessage("You need to have Mobile Data or wifi to access this. Press
ok to Exit");

    builder.setPositiveButton("Ok", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int which) {

            finish();
        }
    });
    return builder; }

private void registerUser() {
    final String username = editTextUsername.getText().toString().trim();
    final String password = editTextPassword.getText().toString().trim();
    final String email = editTextEmail.getText().toString().trim();

//register user
StringRequest stringRequest = new StringRequest(Request.Method.POST,
REGISTER_URL,new Response.Listener<String>() {
    public void onResponse(String response) {
        Toast.makeText(MainActivity.this, response,

```

```

Toast.LENGTH_LONG).show();
    }
},
new Response.ErrorListener() {
    public void onErrorResponse(VolleyError error) {
        Toast.makeText(MainActivity.this,error.toString(),
Toast.LENGTH_LONG).show();
    }
}) {
    protected Map<String, String> getParams(){
        Map<String,String> params = new HashMap<String, String>();
        params.put(KEY_USERNAME,username);
        params.put(KEY_PASSWORD,password);
        params.put(KEY_EMAIL,email);
        return params;

    }
};
RequestQueue requestQueue = Volley.newRequestQueue(this);
requestQueue.add(stringRequest);
}
//next page to login
public void onClick(View v) {
    if (v == buttonRegister) {
        registerUser();
    }
    if(v == buttonLogin) {
        startActivity(new Intent(this, LoginActivity.class));
    }
}
}

```

```
//view pager
```

```
public class MyTimerTask extends TimerTask {  
  
    public void run() {  
  
        MainActivity.this.runOnUiThread(new Runnable() {  
  
            public void run() {  
                if (viewPager.getCurrentItem() == 0) {  
                    viewPager.setCurrentItem(1);  
                } else if (viewPager.getCurrentItem() == 1) {  
                    viewPager.setCurrentItem(2);  
                } else {  
                    viewPager.setCurrentItem(0);  
                }  
            }  
        });  
    }  
}
```

LoginActivity.java

```
public class LoginActivity extends AppCompatActivity implements  
View.OnClickListener {
```

```
    public static final String LOGIN_URL = "http://10.0.2.2:8080/QQ/login.php";
```

```
    public static final String KEY_USERNAME = "username";
```

```
    public static final String KEY_PASSWORD = "password";
```

```
    private EditText editTextUsername;
```

```
    private EditText editTextPassword;
```

```
    private Button buttonLogin;
```

```
    private String username;
```

```
    private String password;
```

```
@Override
```

```
protected void onCreate(Bundle savedInstanceState) {
```

```
    super.onCreate(savedInstanceState);
```

```
    setContentView(R.layout.activity_login);
```

```
    editTextUsername = (EditText) findViewById(R.id.editTextUsername);
```

```
    editTextPassword = (EditText) findViewById(R.id.editTextPassword);
```

```
    buttonLogin = (Button) findViewById(R.id.buttonLogin);
```

```
    buttonLogin.setOnClickListener(this);
```

```
}
```

```
private void userLogin(){
```

```
    username = editTextUsername.getText().toString().trim();
```

```
    password = editTextPassword.getText().toString().trim();
```

```
    StringRequest stringRequest = new StringRequest(Request.Method.POST,
```

```

LOGIN_URL,
    new Response.Listener<String>() {
        public void onResponse(String response) {
            if (response.trim().equals("success")) {
                openProfile();
            } else {
                Toast.makeText(LoginActivity.this, response,
Toast.LENGTH_LONG).show();
            }
        }
    },
    new Response.ErrorListener() {
        public void onErrorResponse(VolleyError error) {
            Toast.makeText(LoginActivity.this,error.toString(),
Toast.LENGTH_LONG).show();
        }
    }
){

    protected Map<String,String> getParams() throws AuthFailureError {
        Map<String,String> map = new HashMap<String, String>();
        map.put(KEY_USERNAME,username);
        map.put(KEY_PASSWORD,password);
        return map;
    }
};
RequestQueue requestQueue = Volley.newRequestQueue(this);
requestQueue.add(stringRequest);
}
public void openProfile(){

    Intent intent = new Intent(this, ActivityUserProfile.class);

```

```

        intent.putExtra(KEY_USERNAME, username);
        startActivity(intent);
    }

```

//click to login

```

    public void onClick(View v) {
        userLogin();
    }
}

```

ActivityUserProfile.java

```

public class ActivityUserProfile extends AppCompatActivity {

```

//count number of people

```

    int counter = 0; //click start from zero
    int counterk = 0;

```

```

    TextView adult,child;

```

```

    Button insert;

```

```

    RequestQueue requestQueue;

```

//URL

```

    final String insertUrl = "http://10.0.2.2:8080/QQ/insertStudent.php";

```

```

    final String showUrl = "http://10.0.2.2:8080/QQ/qshowPizza.php";

```

```

    @Override

```

```

    protected void onCreate(Bundle savedInstanceState) {

```

```

        super.onCreate(savedInstanceState);

```

```

        setContentView(R.layout.activity_user_profile);
    }
}

```

```

final TextView textView = (TextView) findViewById(R.id.textViewUsername);
final Intent intent = getIntent();
textView.setText("Welcome " +
intent.getStringExtra(LoginActivity.KEY_USERNAME));

//count number of people
adult = (TextView) findViewById(R.id.counterValue);
child = (TextView) findViewById(R.id.couterValuek);

//send data to database
insert = (Button) findViewById(R.id.insert);
requestQueue = Volley.newRequestQueue(getApplicationContext());
insert.setOnClickListener(new View.OnClickListener() {

    public void onClick(View view) {

        StringRequest request = new StringRequest(Request.Method.POST,
insertUrl, new Response.Listener<String>() {

            public void onResponse(String response) {
                Log.d("onResponse",response);

                Toast.makeText(ActivityUserProfile.this,"Success",Toast.LENGTH_SHORT).show();
            }
        }, new Response.ErrorListener() {

            public void onErrorResponse(VolleyError error) {
                Log.d("onError",error.toString());
                Toast.makeText(ActivityUserProfile.this,"Please try
again",Toast.LENGTH_SHORT).show();

```

```

        }
    }) {
        protected Map<String, String> getParams() throws AuthFailureError {
            Map<String, String> parameters = new HashMap<String, String>();
            parameters.put("name", intent.getStringExtra(LoginActivity.KEY_USERNAME));
            parameters.put("adult", adult.getText().toString());
            parameters.put("child", child.getText().toString());
            return parameters;
        }
    };

    //click to next page
        Intent intent = new Intent(ActivityUserProfile.this, Main2Activity.class);
        startActivity(intent);
        requestQueue.add(request);
    }
});
}

public void countIN(View view) {

    if ((counter >= 0) && (counter <= 6)) {
        counter++;
        adult.setText(Integer.toString(counter));
    }
}

public void countDE(View v) {
if ((counter <= 7) && (counter >= 1)) {
    counter--;
    adult.setText(Integer.toString(counter));
}
}

public void countINk(View view) {

```

```
        if ((counterk >= 0) && (counterk <= 6)) {  
            counterk++;  
            child.setText(Integer.toString(counterk));  
        }  
  
    }  
    public void countDEk(View v) {  
  
        if ((counterk <= 7) && (counterk >= 1)) {  
            counterk--;  
            child.setText(Integer.toString(counterk));  
        }  
  
    }  
}
```

Main2Activity.java

```
public class Main2Activity extends AppCompatActivity {

    TextView rqueue, rrqueue, rtime, rname, radult, rchild, rday, name;
    Button insert, cancel, game;
    com.android.volley.RequestQueue requestQueue;

    final String insertUrl = "http://10.0.2.2:8080/QQ/insertStudent.php";
    final String showUrl = "http://10.0.2.2:8080/QQ/showStudents.php";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);

        final TextView textView = (TextView) findViewById(R.id.textViewUsername);
        final Intent intent = getIntent();

        cancel = (Button) findViewById(R.id.cancel);
        game = (Button) findViewById(R.id.game);
        rqueue = (TextView) findViewById(R.id.rqueue);
        rrqueue = (TextView) findViewById(R.id.rrqueue);
        rtime = (TextView) findViewById(R.id.rtime);
        rname = (TextView) findViewById(R.id.rname);
        name = (TextView) findViewById(R.id.name);
        radult = (TextView) findViewById(R.id.radult);
        rchild = (TextView) findViewById(R.id.rchild);
        rday = (TextView) findViewById(R.id.rday);
```

```

requestQueue = Volley.newRequestQueue(getApplicationContext());

//Delay 100 ms
final Handler handler = new Handler();
handler.postDelayed(new Runnable() {
    public void run() {
        JsonObjectRequest jsonObjectRequest = new
JsonObjectRequest(Request.Method.POST, showUrl, new
Response.Listener<JSONObject>() {
    public void onResponse(JSONObject response) {
        try {
            JSONArray queue_queue =
response.getJSONArray("queue_queue");
            for (int i = 0; i < queue_queue.length(); i++) {
                JSONObject qq = queue_queue.getJSONObject(i);

                String name = qq.getString("name");
                String adult = qq.getString("adult");
                String child = qq.getString("child");
                String time_in = qq.getString("time_in");
                String remenning_queue = qq.getString("remenning_queue");
                String queue = qq.getString("queue");
                String remenning_time = qq.getString("remenning_time");

                rname.append(name);
                radult.append("Adult : " + adult);
                rchild.append("Child : " + child);
                rday.append("Date/Time : " + time_in);
                rrqueue.append(remenning_queue);
                rqueue.append(queue);
                rtime.append("Waiting time : " + remenning_time + " minutes ");
            }
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }
});
}
}

```

```

        }
    } catch (JSONException e) {
        e.printStackTrace();
    }
}
}, new Response.ErrorListener() {
    public void onErrorResponse(VolleyError error) {
    }
});

requestQueue.add(jsonObjectRequest);
}
}, 100);
//Insert name and queue to database for cancel

cancel.setOnClickListener(new View.OnClickListener() {

    public void onClick(View view) {

StringRequest request = new StringRequest(Request.Method.POST, insertUrl, new
Response.Listener<String>() {

    public void onResponse(String response) {
        Log.d("onResponse", response);
        Toast.makeText(Main2Activity.this, "Cancel",
Toast.LENGTH_SHORT).show();
    }
}, new Response.ErrorListener() {

    public void onErrorResponse(VolleyError error) {

```

```

        Log.d("onError", error.toString());
        Toast.makeText(Main2Activity.this, "Please try again",
Toast.LENGTH_SHORT).show();
    }
    }) {
        protected Map<String, String> getParams() throws AuthFailureError {
            Map<String, String> parameters = new HashMap<String, String>();
            parameters.put("name", rname.getText().toString());
            parameters.put("queue", rqueue.getText().toString());
            return parameters;
        }
    };
    //click to back to menu

    Intent intent = new Intent(Main2Activity.this, MainActivity.class);
    startActivity(intent);

    requestQueue.add(request);
}
});
//Button play game
game.setOnClickListener(new View.OnClickListener() {
    public void onClick(View view) {
        Intent intent = new Intent(Main2Activity.this, Main3Activity.class);
        startActivity(intent);
    }
});
}
}
}

```

ภาคผนวก ข

โค้ดสำหรับการประมวลผลคิว

queue.php

```
<?php
```

```
$con=mysqli_connect("127.0.0.1","root","view_1234","test");
```

```
if (mysqli_connect_errno($con))
```

```
{
```

```
    echo '{"query_result":"ERROR"}';
```

```
}// connect to server
```

```
$month = date("M");
```

```
$year = date("Y");
```

```
if($_SERVER["REQUEST_METHOD"]=="POST"){
```

```
    createsomtum_1();
```

```
}
```

```
function createsomtum_1(){
```

```
global $con;
```

```
$name = $_POST["Name"];
```

```
$adult = $_POST["Adult"];
```

```
$child = $_POST["Child"];
```

```
$people = $adult + $child;
```

```
$name="beer";
```

```
$people = 5;
```

```
$child = 0;
```

```
$adult = 5;
```

```
if( date("H:i") == "22:00"){
```

```
$insert = "INSERT INTO somtum_1(sequence, que) VALUES ( 0, 'A0')";
```

```

mysqli_query($con, $insert);
}

switch ($people) {
case ($people == 1) || ($people == 2):
    // ดึง sequence ล่าสุด ที่ type = A
    $query = "SELECT * FROM `test_1` WHERE type = 'A' ORDER BY `sequence`
DESC LIMIT 1";
    $result = mysqli_query($con, $query);
    $data = mysqli_fetch_assoc($result);
    $new_sequence = $data['sequence']+1;
    //หาจำนวนคิวที่เหลือ
    $query2 = "SELECT * FROM `test_1` WHERE type = 'A' AND status = '0' ";
    $result2 = mysqli_query($con, $query2);
    $num = mysqli_num_rows($result2);
    //insert to server
    $type = "A";
    $insert2 = "INSERT INTO test_1(mouth, year, name, people, adult, child, type,
sequence, remenning_queue, queue, status) VALUES ('$mouth', '$year', '$name',
'$people', '$adult', '$child', 'A', '$new_sequence', '$num', 'A$new_sequence', '0' )";
    mysqli_query($con, $insert2);
    break;

case ($people == 3) || ($people == 4):
    // ดึง sequence ล่าสุด ที่ type = B
    $query = "SELECT * FROM `test_1` WHERE type = 'B' ORDER BY `sequence`
DESC LIMIT 1";
    $result = mysqli_query($con, $query);
    $data = mysqli_fetch_assoc($result);
    $new_sequence = $data['sequence']+1;
    //หาจำนวนคิวที่เหลือ

```

```

$query2 = "SELECT * FROM `test_1` WHERE type = 'B' AND status = '0' ";
$result2 = mysqli_query($con, $query2);
$num = mysqli_num_rows($result2);
//insert to server
$type = "B";
$insert2 = "INSERT INTO test_1(mouth, year, name, people, adult, child, type,
sequence, remenning_queue, queue, status) VALUES ('$mouth', '$year', '$name',
'$people', '$adult', '$child', 'B', '$new_sequence', '$num', 'B$new_sequence', '0')";
mysqli_query($con, $insert2);
break;

```

```

case ($people == 5) || ($people == 6):

```

```

    // ดึง sequence ล่าสุด ที่ type = C

```

```

    $query = "SELECT * FROM `test_1` WHERE type = 'C' ORDER BY `sequence`
DESC LIMIT 1";

```

```

    $result = mysqli_query($con, $query);

```

```

    $data = mysqli_fetch_assoc($result);

```

```

    $new_sequence = $data['sequence']+1;

```

```

    //หาจำนวนคิวที่เหลือ

```

```

    $query2 = "SELECT * FROM `test_1` WHERE type = 'C' AND status = '0' ";

```

```

    $result2 = mysqli_query($con, $query2);

```

```

    $num = mysqli_num_rows($result2);

```

```

    //insert to server

```

```

    $type = "C";

```

```

    $insert2 = "INSERT INTO test_1(mouth, year, name, people, adult, child, type,
sequence, remenning_queue, queue, status) VALUES ( '$mouth', '$year', '$name',
'$people', '$adult', '$child', 'C', '$new_sequence', '$num', 'C$new_sequence', '0')";
    mysqli_query($con, $insert2);
    break;

```

```

case $people >= 7:

```

```

// ดึง sequence ล่าสุด ที่ type = D
$query = "SELECT * FROM `test_1` WHERE type = 'D' ORDER BY `sequence`
DESC LIMIT 1";

$result = mysqli_query($con, $query);
$data = mysqli_fetch_assoc($result);
$new_sequence = $data['sequence']+1;
//หาจำนวนคิวที่เหลือ
$query2 = "SELECT * FROM `test_1` WHERE type = 'D' AND status = '0' ";
$result2 = mysqli_query($con, $query2);
$num = mysqli_num_rows($result2);
//insert to server
$type = "D";
$insert2 = "INSERT INTO test_1(mouth, year, name, people, adult, child, type,
sequence, remenning_queue, queue, status) VALUES ( '$mouth', '$year', '$name',
'$people', '$adult', '$child', 'D', '$new_sequence', '$num', 'D$new_sequence', '0')";
mysqli_query($con, $insert2);
break;
}

}

$query = "SELECT SUM(people) as people, SUM(child) as child, SUM(adult) as adult
FROM test_1 WHERE mouth = '$mouth' AND year = '$year' ";
$result =mysqli_query($con, $query);
while($row = mysqli_fetch_array($result)){
    $people= $row['people'];
    $adult= $row['adult'];
    $child= $row['child'];
}
}

```

```
$query1 = "SELECT * FROM `test_1` WHERE type = 'A' AND mouth = '$mouth' AND  
year = '$year'";
```

```
$result1 = mysqli_query($con, $query1);
```

```
$num1 = mysqli_num_rows($result1);
```

```
$query2 = "SELECT * FROM `test_1` WHERE type = 'B' AND mouth = '$mouth' AND year  
= '$year'";
```

```
$result2 = mysqli_query($con, $query2);
```

```
$num2 = mysqli_num_rows($result2);
```

```
$query3 = "SELECT * FROM `test_1` WHERE type = 'C' AND mouth = '$mouth' AND  
year = '$year'";
```

```
$result3 = mysqli_query($con, $query3);
```

```
$num3 = mysqli_num_rows($result3);
```

```
$query4 = "SELECT * FROM `test_1` WHERE type = 'D' AND mouth = '$mouth' AND  
year = '$year'";
```

```
$result4 = mysqli_query($con, $query4);
```

```
$num4 = mysqli_num_rows($result4);
```

```
$update = "UPDATE graph_1 SET people = '$people', adult = '$adult', child = '$child',  
A = '$num1', B = '$num2', C = '$num3', D = '$num4' WHERE mouth = '$mouth' AND  
year = '$year'";
```

```
mysqli_query($con, $update);
```

```
?>
```