

BUFFET2GETHER



เสนอ

ดร.รัฐชัย ชาวอุทัย

ผศ.ดร.ชุตีเมษณี ศรีนิลภา

จัดทำโดย

61010432 ธนพล พัดพิน

61010588 นิรชา บุญยเนตร

61010821 ภิญดา ทองเต็ม

61011432 สุธาสีห์ อินทรวัตร

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชา

01076008 SOFTWARE DEVELOPMENT PROCESSES

ภาคเรียนที่ 2 ปีการศึกษา 2562

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

คำนำ

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชา 01076008 SOFTWARE DEVELOPMENT PROCESSES จัดทำขึ้นเพื่อศึกษาหลักการพัฒนาโปรแกรม ซึ่งในที่นี้เป็นการพัฒนาซอฟต์แวร์แอนดรอยด์แอปพลิเคชัน BUFFET2GETHER ที่เป็นซอฟต์แวร์สำหรับค้นหากลุ่มคนที่ต้องการไปรับประทานอาหารด้วยกันเพื่อใช้โปรโมชั่นของทางร้านให้คุ้มค่า ทำให้ผู้ใช้งานได้รับประทานอาหารในราคาที่ถูกลง และได้พบปะกับผู้คนใหม่ ๆ ที่มีความชอบคล้ายคลึงกันอีกด้วย

BUFFET2GETHER มีลักษณะโครงสร้างแบบ Mobile Application โดยพัฒนาขึ้นตามขั้นตอนการพัฒนาซอฟต์แวร์ตั้งแต่การรวบรวมและวิเคราะห์ความต้องการของผู้ใช้งาน (Requirements) การออกแบบระบบของแอปพลิเคชันทั้งในการรับส่งข้อมูล, การจัดเก็บในฐานข้อมูล, และความสัมพันธ์ของสิ่งต่าง ๆ ในระบบด้วยไดอะแกรมต่าง ๆ ได้แก่ BPMN (Business Process Model and Notation), DFD (Data Flow Diagram), และ ER (Entity-relationship Diagram) เพื่อให้สามารถมองเห็นการทำงานของระบบโดยรวมได้ชัดเจนและครอบคลุมยิ่งขึ้น การวางแผนการพัฒนาระบบในแต่ละช่วง การออกแบบ UI (User Interface) การออกแบบสถาปัตยกรรมซอฟต์แวร์แบบ 3-Tier Architecture เพื่อเป็นต้นแบบไปพัฒนาระบบและทำให้ซอฟต์แวร์ที่ได้ตอบโจทย์การใช้งาน, ความปลอดภัย, ความน่าเชื่อถือ, และการบำรุงรักษา การลงมือพัฒนาซอฟต์แวร์ด้วยเครื่องมือต่าง ๆ ได้แก่ Flutter framework รวมถึงการใช้ฐานข้อมูลเพื่อเก็บข้อมูลต่าง ๆ ภายในระบบด้วย Firebase ท้ายสุดจึงทำการทดสอบระบบด้วย Test requirements และ Test cases เพื่อให้ BUFFET2GETHER เป็นซอฟต์แวร์ที่ทำงานได้ตรงตามจุดประสงค์และมีความสมบูรณ์ที่สุด

ขอขอบคุณ ดร.รัฐชัย ขาวอุทัย และผศ.ดร.ชุตินิเมษฐ์ ศรีนิลทาที่คอยตรวจสอบและให้คำแนะนำต่าง ๆ และหากรายงานฉบับนี้มีข้อผิดพลาดประการใด ทางคณะผู้จัดทำต้องขออภัยมา ณ ที่นี้ด้วย

คณะผู้จัดทำ

สารบัญ

เรื่อง	หน้า
บทที่ 1 บทนำ	
ที่มาและความสำคัญ	1
วัตถุประสงค์	1
กลุ่มผู้ใช้	1
บทที่ 2 การวิเคราะห์ความต้องการของผู้ใช้และระบบโปรแกรม	
Project Scope	2
User Requirements	5
บทที่ 3 การออกแบบระบบ	
User Interface Design (UI)	11
Business Process Model and Notation (BPMN)	15
Data Flow Diagram (DFD)	15
Entity Relationship Diagram	20
Software Architecture	
3-Tier Architecture	25
Software Architecture: Stack	26
บทที่ 4 ผลการพัฒนา	
แอปพลิเคชัน	27
Url ของ GitHub	33
วิธีการใช้งานระบบ	34

บทที่ 5 การทดสอบระบบ

Test Report	37
Test Requirements และ Test Cases	38
ผลการทดสอบ	66

รายชื่อสมาชิก



61010432 ธนพล พัดพิน



61010588 นิรชา บุญยเนตร



61010821 ภาณุดา กองเต็ม



6101142 สุธสีย์ อินทรวัตร

บทที่ 1

บทนำ

ที่มาและความสำคัญ

ในปัจจุบัน ธุรกิจร้านอาหารเป็นอีกหนึ่งธุรกิจที่รุ่งเรือง มีร้านอาหารเปิดใหม่มากมายเพื่อให้บริการแก่ผู้คน แต่เมื่อมีจำนวนร้านอาหารมากขึ้นก็เกิดการแข่งขันมากขึ้น การแข่งขันนั้นจึงเป็นที่มาของโปรโมชั่นต่าง ๆ ที่ทางร้านจัดขึ้นเพื่อดึงดูดลูกค้า โดยส่วนใหญ่แล้วโปรโมชั่นนั้น ๆ จะคุ้มค่าก็ต่อเมื่อลูกค้าจับกลุ่มกันมารับประทานอาหารเป็นจำนวนมาก ซึ่งในบางครั้งการจะหาจำนวนคนให้ได้ครบตามที่ต้องการนั้นอาจจะเป็นปัญหาหนึ่งที่ใครหลาย ๆ คนต้องเคยพบเจอ จนทำให้บางครั้งพลาดโอกาสที่จะใช้โปรโมชั่นนั้น ๆ ไป หรือในกรณีของบางร้านก็มีราคาแพงและไม่คุ้มค่าต่อการไปรับประทานอาหารคนเดียว

ทางคณะผู้จัดทำได้เล็งเห็นถึงปัญหาดังกล่าวจึงมีความคิดที่จะออกแบบและพัฒนาแอปพลิเคชัน “BUFFET2GETHER” ขึ้นเพื่ออำนวยความสะดวกให้กับผู้ใช้งาน ช่วยค้นหากลุ่มคนที่ต้องการไปรับประทานอาหารด้วยกันเพื่อใช้โปรโมชั่นของทางร้านให้คุ้มค่า ทำให้ผู้ใช้งานได้รับประทานอาหารในราคาที่ถูกลง และได้พบปะกับผู้คนใหม่ ๆ ที่มีความชอบคล้ายคลึงกันอีกด้วย โดยภายในแอปพลิเคชันจะมีร้านอาหารเปิดให้บริการอย่างหลากหลาย มีการแนะนำโปรโมชั่น แนะนำร้าน และสามารถใช้ค้นหากลุ่มคนเพื่อไปรับประทานอาหารโดยไม่ต้องมีโปรโมชั่นได้อีกด้วย

วัตถุประสงค์

1. เพื่อศึกษาการพัฒนาซอฟต์แวร์ตามกระบวนการการพัฒนาซอฟต์แวร์ เช่น การคิดและวิเคราะห์ความต้องการของผู้ใช้งาน การออกแบบโครงสร้างซอฟต์แวร์ การออกแบบการทำงานของระบบ การออกแบบรูปลักษณ์ การวางแผนการพัฒนา การลงมือพัฒนา และการทดสอบ เป็นต้น
2. เพื่อศึกษาการประยุกต์ใช้โปรแกรมและฐานข้อมูลต่าง ๆ ในการพัฒนาซอฟต์แวร์ที่ต้องการ
3. เพื่อสร้างสรรค์ซอฟต์แวร์เพื่ออำนวยความสะดวกในความต้องการค้นหากลุ่มคนของผู้ใช้งาน
4. เพื่อศึกษาการวางแผนและการทำงานร่วมกับผู้อื่น

กลุ่มผู้ใช้งาน

ผู้ที่ต้องการค้นหากลุ่มคนเพื่อไปรับประทานอาหารด้วยกัน จะเพื่อใช้หรือไม่ใช้โปรโมชั่นของทางร้านให้คุ้มค่าก็ได้

บทที่ 2

การวิเคราะห์ความต้องการของผู้ใช้และระบบของโปรแกรม

Project Scope

ลำดับ	ขอบเขต	รายละเอียด
1	ส่วนสมัครสมาชิก	<p>สามารถสมัครสมาชิกได้ หากผู้ที่สมัครสมาชิกทำการกรอกข้อมูลสำหรับการสมัครสมาชิกครบถ้วนและถูกต้องตามเงื่อนไข ดังนี้</p> <ul style="list-style-type: none"> - ต้องกรอก Email ตามรูปแบบ Email ที่ถูกต้องและ Password ยาว 6 ตัวอักษรขึ้นไป (ไม่ปล่อยว่าง) - Email ต้องไม่นำหน้าและลงท้ายด้วยอักขระพิเศษ ต้องมีตัวอักษรภาษาอังกฤษ ต้องมีอักขระ '@' อย่างน้อย 1 ตัวอักษร ต้องมีอักขระ '.' อย่างน้อย 1 ตัวอักษร ตัวอักษรที่อยู่ด้านหน้าอักขระ '@' ต้องไม่ใช่อักขระพิเศษ (ยกเว้นอักขระ '_', '-') และตัวอักษรที่อยู่ด้านหลังอักขระ '@' จะต้องไม่ใช่อักขระพิเศษ และห้ามมีตัวอักษรภาษาอื่น - Email ที่ใช้สมัครสมาชิกต้องไม่ซ้ำกับ Email ที่เคยสมัครสมาชิกไปแล้ว - Password จะเป็นตัวอักษรภาษาอังกฤษหรือตัวเลขก็ได้ ที่มีความยาว 6 ตัวอักษรขึ้นไป - เปลี่ยนรูปภาพประจำตัวหรือจะใช้รูปภาพเริ่มต้นก็ได้ - ต้องกรอก Username ด้วยตัวอักษรภาษาอังกฤษ/ตัวอักษรภาษาไทย/ตัวเลข/อักขระพิเศษก็ได้ ความยาว 4 - 12 ตัวอักษร - ต้องเลือกเพศ <ul style="list-style-type: none"> ● เพศชาย ● เพศหญิง ● ไม่ประสงค์ระบุ - ต้องกรอกวันเกิด

2	ส่วนเข้าสู่ระบบ	สามารถทำการเข้าสู่ระบบได้ หากทำการเข้าสู่ระบบด้วย Email และ Password ที่ถูกต้อง และมีบัญชีอยู่ในฐานข้อมูล
3	ส่วนหน้าหลัก	<p>สามารถแสดงร้านอาหารที่ลงทะเบียนกับทางแอปพลิเคชันทั้งหมดได้</p> <p>สามารถค้นหาร้านอาหารจากชื่อร้านได้</p> <p>สามารถดูโปรโมชั่นแนะนำได้</p> <p>สามารถคลิกเข้าไปดูรายละเอียดของร้านได้</p> <ul style="list-style-type: none"> - ชื่อร้าน - รูปภาพร้าน - สถานที่ตั้งร้าน - เวลาเปิด – ปิดของร้าน - โปรโมชั่นของร้าน <p>สามารถเลือกกด Matching เพื่อค้นหาโต๊ะได้</p> <p>สามารถเลือกกดสร้างโต๊ะเพื่อสร้างโต๊ะได้</p>
4	ส่วนหน้าโต๊ะ	<p>สามารถแสดงรายละเอียดของโต๊ะได้</p> <ul style="list-style-type: none"> - หมายเลขโต๊ะ - ชื่อร้าน - รูปภาพร้าน - สถานที่ตั้งร้าน - เวลาเปิด - ปิดของร้าน - ช่วงอายุที่หัวหน้าโต๊ะต้องการ - จำนวนผู้ที่อยู่ในโต๊ะ - จำนวนคนที่หัวหน้าโต๊ะต้องการ - วันเวลาที่นัดหมาย - เพศที่หัวหน้าโต๊ะต้องการ - ความชอบร่วมกัน <ul style="list-style-type: none"> ● แพชั่น ● กีฬา ● เทคโนโลยี ● การเมือง ● วงการบันเทิง

		<ul style="list-style-type: none"> ● หนังสือ ● สัตว์เลี้ยง <p>สมาชิกสามารถออกจากโต๊ะได้จนกว่าสมาชิกจะเต็ม หัวหน้าโต๊ะสามารถกด Finish เมื่อรับประทานอาหารเสร็จแล้วได้เพื่อ ลบโต๊ะและเก็บประวัติ</p>
5	ส่วนหน้าแฉงเตอน	<p>สามารถแฉงเตอนหัวหน้าโต๊ะได้เมื่อมีผู้ที่มีคุณสมบัติตรงกับที่ต้องการมา match</p> <p>สามารถส่งคำเชิญหรือปฏิเสธไปให้ผู้ที่มา match ได้</p> <p>สามารถแฉงเตอนผู้ที่หาโต๊ะได้หลังจากหัวหน้าโต๊ะส่งคำเชิญให้</p> <p>สามารถตอบรับหรือปฏิเสธคำเชิญได้</p> <p>สามารถเข้าหน้าโต๊ะได้เมื่อตอบรับคำเชิญ</p>
6	ส่วนข้อมูลส่วนตัว	<p>สามารถแสดงข้อมูลส่วนตัวได้</p> <ul style="list-style-type: none"> - รูปภาพประจำตัว - Username - เพศ - อายุ (ที่คำนวณอัตโนมัติจากวันเกิดที่กรอกมา) - ความชอบ ● แฟชั่น ● กีฬา ● เทคโนโลยี ● การเมือง ● วงการบันเทิง ● หนังสือ ● สัตว์เลี้ยง <p>สามารถแก้ไขข้อมูลส่วนตัวได้</p> <p>สามารถดูประวัติที่เคยไปรับประทานอาหารได้</p>
7	ส่วนการสร้างโต๊ะ	<p>สามารถสร้างโต๊ะได้</p> <p>สามารถแก้ไขคุณสมบัติที่ต้องการได้</p> <ul style="list-style-type: none"> - ช่วงอายุที่หัวหน้าโต๊ะต้องการ

		<ul style="list-style-type: none"> - จำนวนผู้ที่อยู่ในโต๊ะ - จำนวนคนที่หัวหน้าโต๊ะต้องการ - วันเวลาที่นัดหมาย - เพศที่หัวหน้าโต๊ะต้องการ - ความชอบร่วมกัน <ul style="list-style-type: none"> ● แพฟชั่น ● กีฬา ● เทคโนโลยี ● การเมือง ● วงการบันเทิง ● หนังสือ ● สัตว์เลี้ยง
8	ส่วนการ Matching	สามารถจับคู่ผู้ที่ต้องการหาโต๊ะกับโต๊ะที่มีอยู่ตามคุณสมบัติที่ตรงกันได้

User Requirements

☐ Requirement ที่ทำ ☐ Requirement ที่ไม่ได้ทำ

ID	Details	Type	Priority
R1	แอป Buffet2Gether จะแสดงรายชื่อร้านทั้งหมดที่ได้ลงทะเบียนกับทางแอปไว้	<ul style="list-style-type: none"> ● Products ● Functional 	MustHave
R2	แอป Buffet2Gether จะแสดงรายการโปรโมชั่นแนะนำจากร้านที่ได้ลงทะเบียนกับทางแอปไว้	<ul style="list-style-type: none"> ● Products ● Functional 	MustHave
R3	แอป Buffet2Gether จะแสดงรายละเอียดของร้านค้าประกอบด้วยรูปภาพประกอบ, ชื่อ, สถานที่ตั้ง, เวลาเปิด-ปิด, โปรโมชั่น	<ul style="list-style-type: none"> ● Products ● Functional 	MustHave

R4	แอป Buffet2Gether จะแสดงรายละเอียดของโต๊ะที่ผู้ใช้งาน match ตรงกันประกอบด้วยหมายเลขโต๊ะ, รายละเอียดร้านค้าที่จะไป, สมาชิก, และคุณสมบัติที่โต๊ะต้องการประกอบด้วย เพศ, อายุ, เวลาที่จะไปใช้บริการ, และจำนวนคนที่ต้องการ	<ul style="list-style-type: none"> ● Products ● Functional 	MustHave
R5	แอป Buffet2Gether จะแสดงจำนวนคนในโต๊ะ โดยอัตโนมัติเมื่อมีสมาชิกโต๊ะ เข้า - ออก	<ul style="list-style-type: none"> ● Products ● Functional 	MustHave
R6	แอป Buffet2Gether จะกลับไปยังหน้าต่างให้ผู้ใช้งานเลือกระหว่าง สร้างโต๊ะ หรือ ค้นหาโต๊ะ ในกรณีที่หัวหน้าโต๊ะที่ได้รับคำเชิญทั้งหมดปฏิเสธ หรือครบเวลาที่กำหนด	<ul style="list-style-type: none"> ● Products ● Functional 	MustHave
R7	แอป Buffet2Gether จะเก็บประวัติการไปรับประทานอาหารครั้งล่าสุดให้กับผู้ใช้งานแต่ละคน	<ul style="list-style-type: none"> ● Products ● Functional 	ShouldHave
R8	แอป Buffet2Gether จะอนุญาตให้ผู้ใช้งานเลือกโปรโมชั่นที่ต้องการจากร้านค้าได้	<ul style="list-style-type: none"> ● User Interface ● Functional 	WantToHave
R9	แอป Buffet2Gether จะอนุญาตให้ผู้ใช้งานเลือกได้ว่าจะเป็นผู้สร้างโต๊ะ หรือใช้ระบบ Matching เพื่อค้นหาโต๊ะอื่น ๆ	<ul style="list-style-type: none"> ● User Interface ● Functional 	MustHave
R10	แอป Buffet2Gether จะอนุญาตให้หัวหน้าโต๊ะสามารถแก้ไขคุณสมบัติของโต๊ะได้จนกว่าสมาชิกจะครบ	<ul style="list-style-type: none"> ● User Interface ● Functional 	ShouldHave

R11	แอป Buffet2Gether จะอนุญาตให้ ผู้ใช้งานสามารถเลือกได้ว่าจะตกลงหรือ ปฏิเสธในการเข้าโต๊ะนั้น ๆ	<ul style="list-style-type: none"> ● User Interface ● Functional 	MustHave
R12	แอป Buffet2Gether จะอนุญาตให้ หัวหน้าโต๊ะสามารถเลือกได้ว่าจะตกลง หรือปฏิเสธในการรับผู้ที่มา match เข้า โต๊ะนั้น ๆ	<ul style="list-style-type: none"> ● User Interface ● Functional 	MustHave
R13	แอป Buffet2Gether จะอนุญาตให้ ผู้ใช้งานสามารถเข้าร่วมโต๊ะได้มากที่สุด 3 โต๊ะโดยที่วันที่และเวลานัดหมายของ แต่ละโต๊ะต้องไม่ชนกัน	<ul style="list-style-type: none"> ● User Interface ● Functional 	WantToHave
R14	แอป Buffet2Gether จะอนุญาตให้ สมาชิกสามารถออกจากโต๊ะได้จนกว่า จำนวนสมาชิกในโต๊ะจะครบ	<ul style="list-style-type: none"> ● User Interface ● Functional 	MustHave
R15	แอป Buffet2Gether จะอนุญาตให้ สมาชิกสามารถรายงานความประพฤติ ของสมาชิกท่านอื่น ๆ ได้ โดยต้องแนบ เหตุผลประกอบ	<ul style="list-style-type: none"> ● User Interface ● Functional 	ShouldHave
R16	แอป Buffet2Gether จะอนุญาตให้ สมาชิกภายในโต๊ะพูดคุยกันผ่านแชท	<ul style="list-style-type: none"> ● User Interface ● Functional 	WantToHave
R17	แอป Buffet2Gether จะอนุญาตให้ หัวหน้าโต๊ะกดสิ้นสุดการรับประทาน อาหารได้เมื่อรับประทานเสร็จแล้ว และ ระบบจะทำการลบโต๊ะไป	<ul style="list-style-type: none"> ● User Interface ● Functional 	MustHave
R18	แอป Buffet2Gether จะอนุญาตให้ ผู้ใช้งานค้นหาร้านค้าที่ต้องการได้ด้วย การพิมพ์ชื่อร้าน	<ul style="list-style-type: none"> ● User Interface ● Functional 	MustHave
R19	แอป Buffet2Gether จะอนุญาตให้ ผู้ใช้งานค้นหาโต๊ะที่ต้องการได้ด้วยการ พิมพ์หมายเลขโต๊ะ	<ul style="list-style-type: none"> ● User Interface ● Functional 	MustHave

R20	แอป Buffet2Gether จะอนุญาตให้สมาชิกสามารถดูประวัติการไปรับประทานอาหารของตนเองได้	<ul style="list-style-type: none"> ● User Interface ● Functional 	ShouldHave
R21	แอป Buffet2Gether จะอนุญาตให้ผู้ใช้งานที่เลือกสร้างโต๊ะเองเป็นหัวหน้าโต๊ะ และสามารถระบุคุณสมบัติที่ต้องการได้ประกอบด้วยเพศ, ช่วงอายุ, เวลารับประทาน, และจำนวนคน	<ul style="list-style-type: none"> ● User Interface ● Functional 	MustHave
R22	แอป Buffet2Gether จะอนุญาตให้ผู้ใช้งานที่ลงทะเบียนเป็นตัวแทนจากร้านค้าสามารถดูและแก้ไขข้อมูลร้านค้าของตนเองได้ เช่น ชื่อร้าน, รูปร้าน, สถานที่ตั้ง, เวลาเปิด-ปิดร้าน, โปรโมชั่น, และช่องทางการติดต่อ	<ul style="list-style-type: none"> ● User Interface ● Functional 	ShouldHave
R23	แอป Buffet2Gether จะจับคู่ผู้ใช้งานที่เลือก Matching โดยคัดเลือกอัตโนมัติโดยคำนึงถึงร้านค้า และคุณสมบัติที่ตรงกันของผู้ใช้งานกับที่หัวหน้าโต๊ะนั้น ๆ ระบุไว้	<ul style="list-style-type: none"> ● System Algorithm ● Functional 	MustHave
R24	แอป Buffet2Gether จะแจ้งเตือนผู้ใช้งานเมื่อมีโต๊ะที่ match โดยอัตโนมัติ	<ul style="list-style-type: none"> ● Notification ● Functional 	MustHave
R25	แอป Buffet2Gether จะแจ้งเตือนผู้ใช้งานที่เป็นหัวหน้าโต๊ะก่อนถึงเวลานัด 15 นาที ในกรณีสมาชิกในโต๊ะยังไม่ครบตามจำนวนที่กำหนดว่าจะแก้ไขคุณสมบัติหรือลบโต๊ะ	<ul style="list-style-type: none"> ● Notification ● Functional 	ShouldHave
R26	แอป Buffet2Gether จะแจ้งเตือนผู้ใช้งานเมื่อมีโต๊ะที่ match โดยอัตโนมัติ แม้ว่าผู้ใช้งานจะตกลงกับโต๊ะใดโต๊ะหนึ่ง	<ul style="list-style-type: none"> ● Notification ● Functional 	MustHave

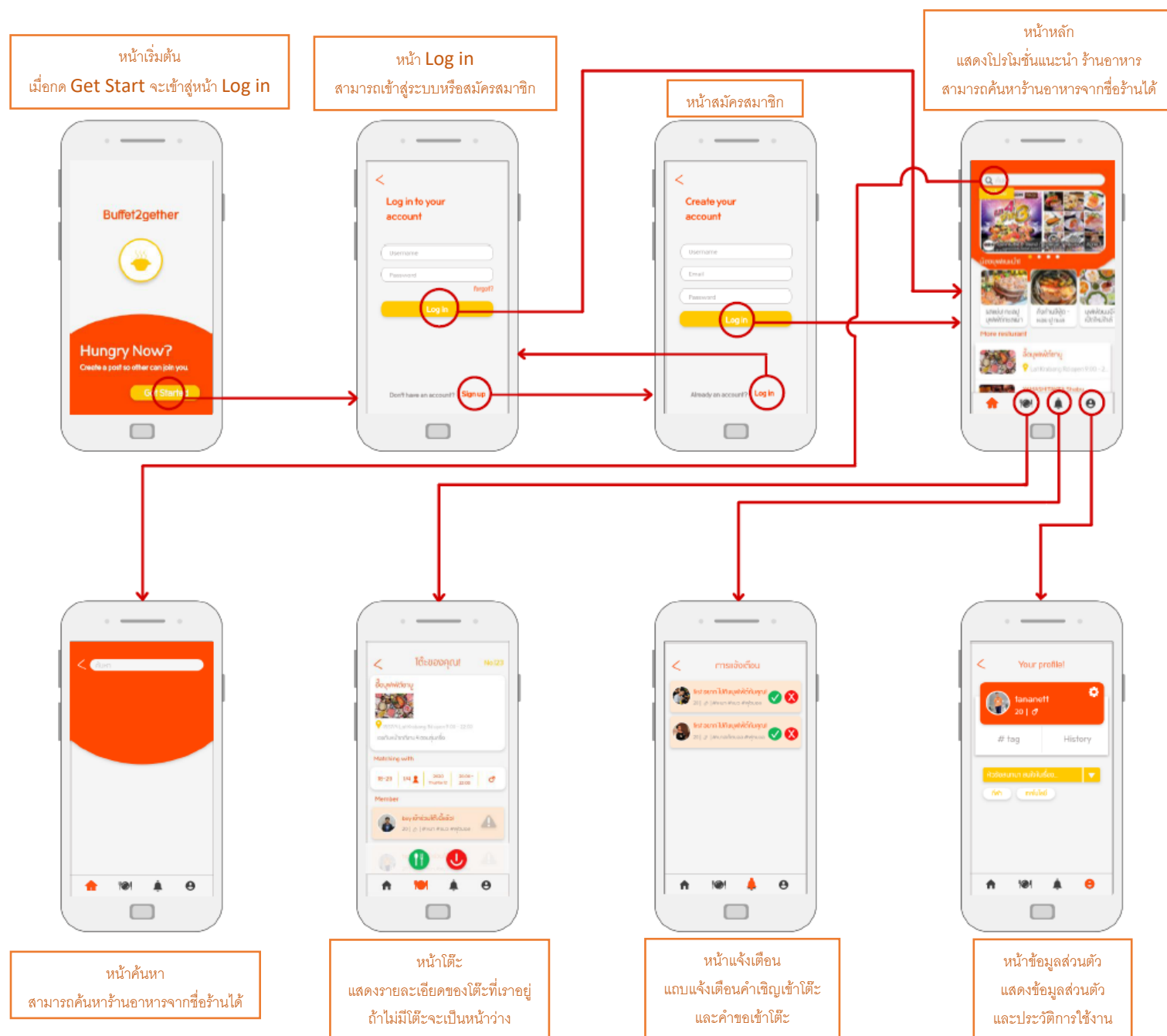
	ไปแล้วก็ตาม แต่จะหยุดแจ้งเตือนเมื่อสมาชิกในโต๊ะที่อยู่มีจำนวนครบแล้ว		
R27	แอป Buffet2Gether จะแจ้งเตือนหัวหน้าโต๊ะเมื่อมีผู้มา match โดยอัตโนมัติ	<ul style="list-style-type: none"> ● Notification ● Functional 	MustHave
R28	แอป Buffet2Gether จะทำการแบนผู้ใช้งานที่ได้รับแจ้งจากผู้ใช้งานอื่น ระยะเวลาตามความเหมาะสม	<ul style="list-style-type: none"> ● Security ● Functional 	WantToHave
R29	แอป Buffet2Gether จะอนุญาตให้ผู้ใช้งานสมัครสมาชิกที่หน้าแรกของแอปพลิเคชัน	<ul style="list-style-type: none"> ● Registration ● Functional 	MustHave
R30	แอป Buffet2Gether จะต้องให้ผู้ใช้งานตั้ง Email และ password	<ul style="list-style-type: none"> ● Registration ● Functional 	MustHave
R31	แอป Buffet2Gether จะเก็บข้อมูลผู้ใช้งาน ประกอบด้วย username, password, Email, รูปโปรไฟล์, เพศ, อายุ, และความชอบ	<ul style="list-style-type: none"> ● Registration ● Functional 	MustHave
R32	แอป Buffet2Gether จะอนุญาตให้ผู้ใช้งานสามารถดูและแก้ไขข้อมูลส่วนตัวของตนเองได้	<ul style="list-style-type: none"> ● Registration ● Functional 	MustHave
R33	แอป Buffet2Gether จะอนุญาตให้ผู้ใช้งานสามารถแก้ไขรหัสผ่านของตนเองได้ใน	<ul style="list-style-type: none"> ● Registration ● Functional 	MustHave
R34	แอป Buffet2Gether จะอนุญาตให้ผู้ใช้งานสามารถดลิมรหัสผ่านได้ และ จะทำการยืนยันตัวตนผ่านทาง Email ก่อนจะให้ผู้ใช้งานเปลี่ยนรหัสผ่านใหม่	<ul style="list-style-type: none"> ● Registration ● Functional 	MustHave
R35	แอป Buffet2Gether จะให้บริการตลอด 24 ชั่วโมงต่อวัน, 365 วันต่อปี	<ul style="list-style-type: none"> ● Availability ● NonFunctional 	MustHave

R36	แอป Buffet2Gether จะรองรับบน ระบบปฏิบัติการแอนดรอยด์	<ul style="list-style-type: none">● ComplianceToStandards● NonFunctional	MustHave
-----	---	---	----------

บทที่ 3

การออกแบบระบบ

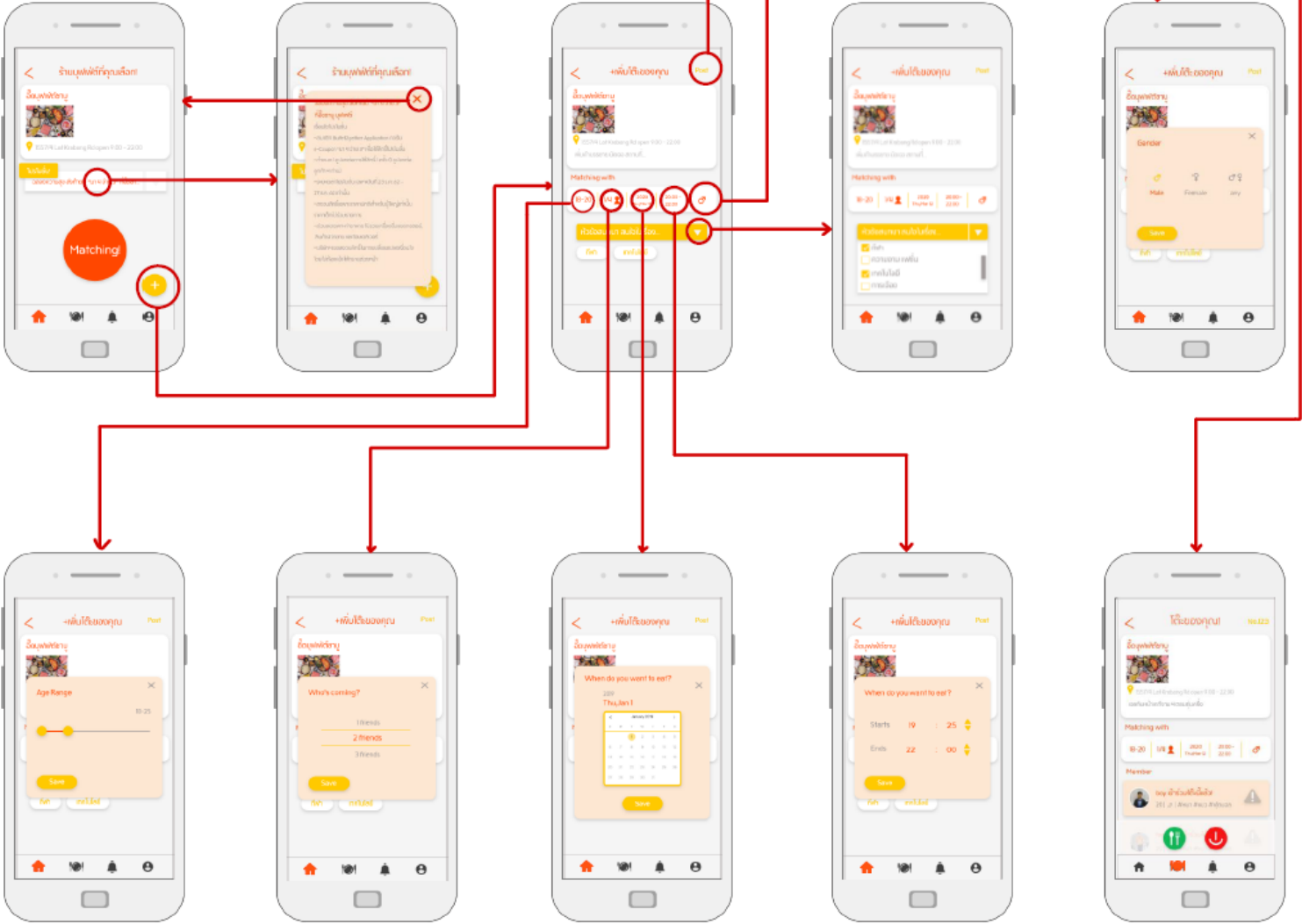
User Interface Design



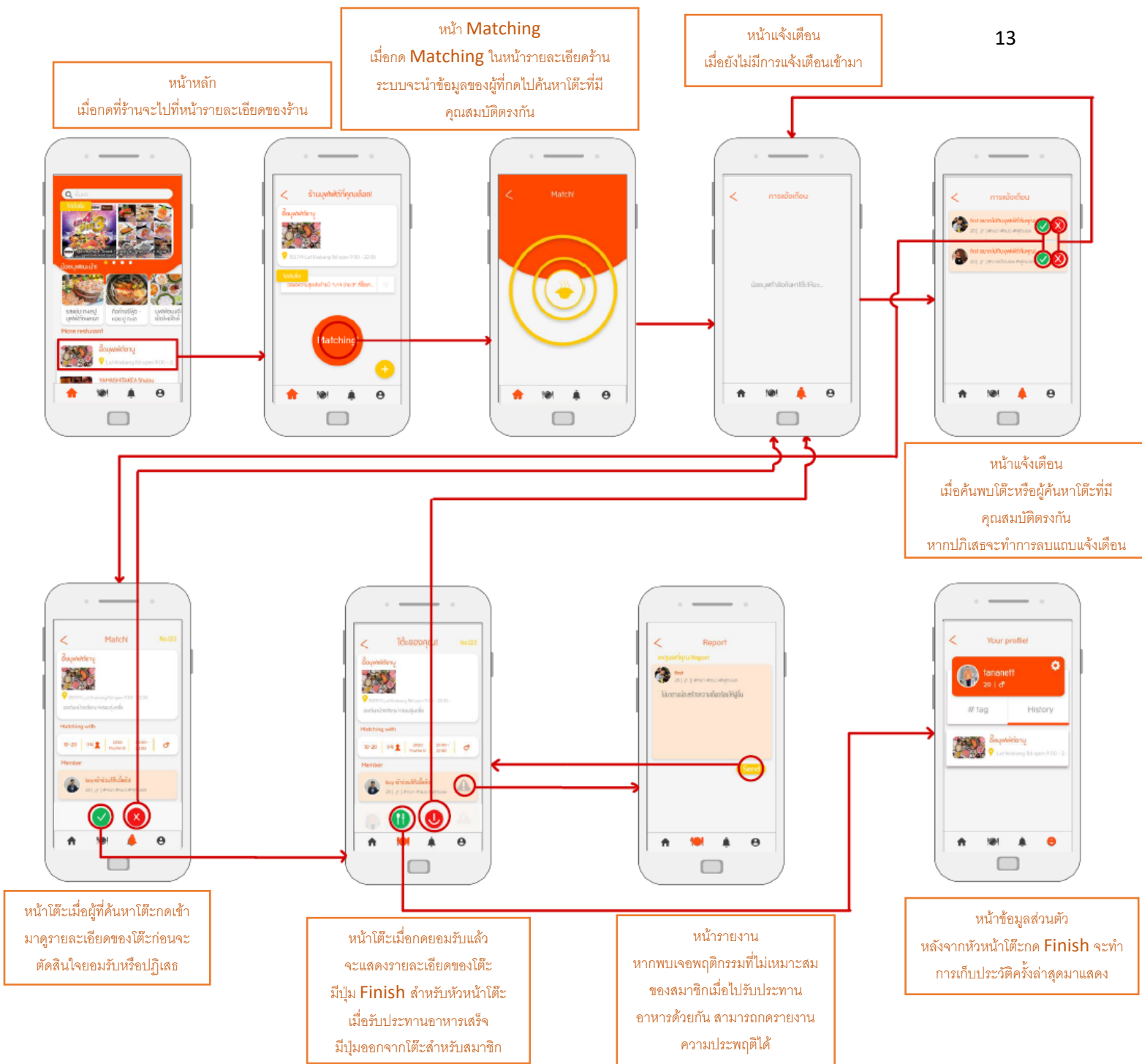
หน้ารายละเอียดร้าน
แสดงรายละเอียดต่าง ๆ ของร้าน
สามารถกดปุ่ม **Matching**
หรือ สร้างโต๊ะได้

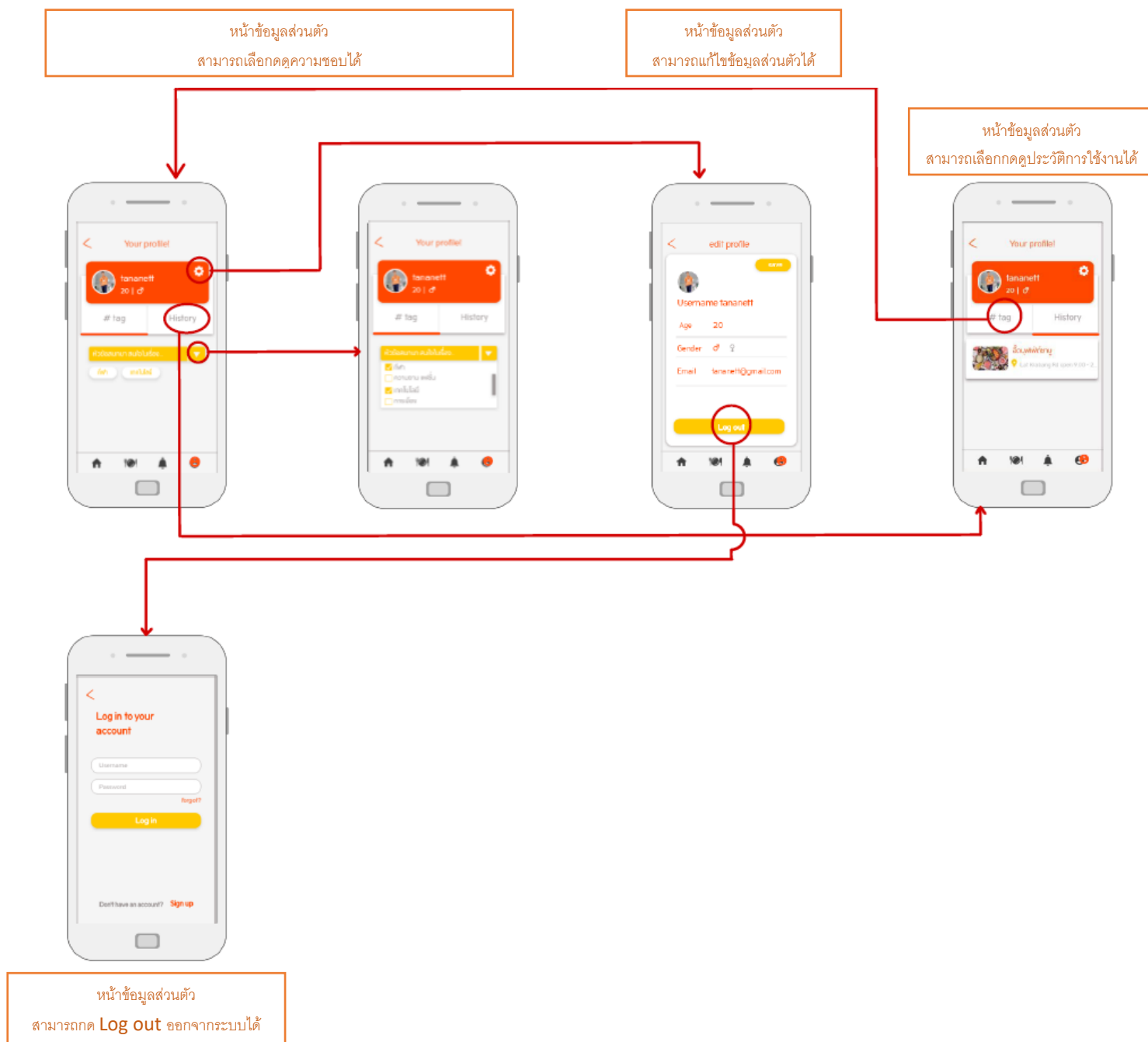
หน้ารายละเอียดร้านเมื่อกดดูโปรโมชัน

หน้าสร้างโต๊ะ
สามารถแก้ไขคุณสมบัติที่ต้องการและความชอบได้
สามารถกด **Post** เพื่อสร้างโต๊ะได้

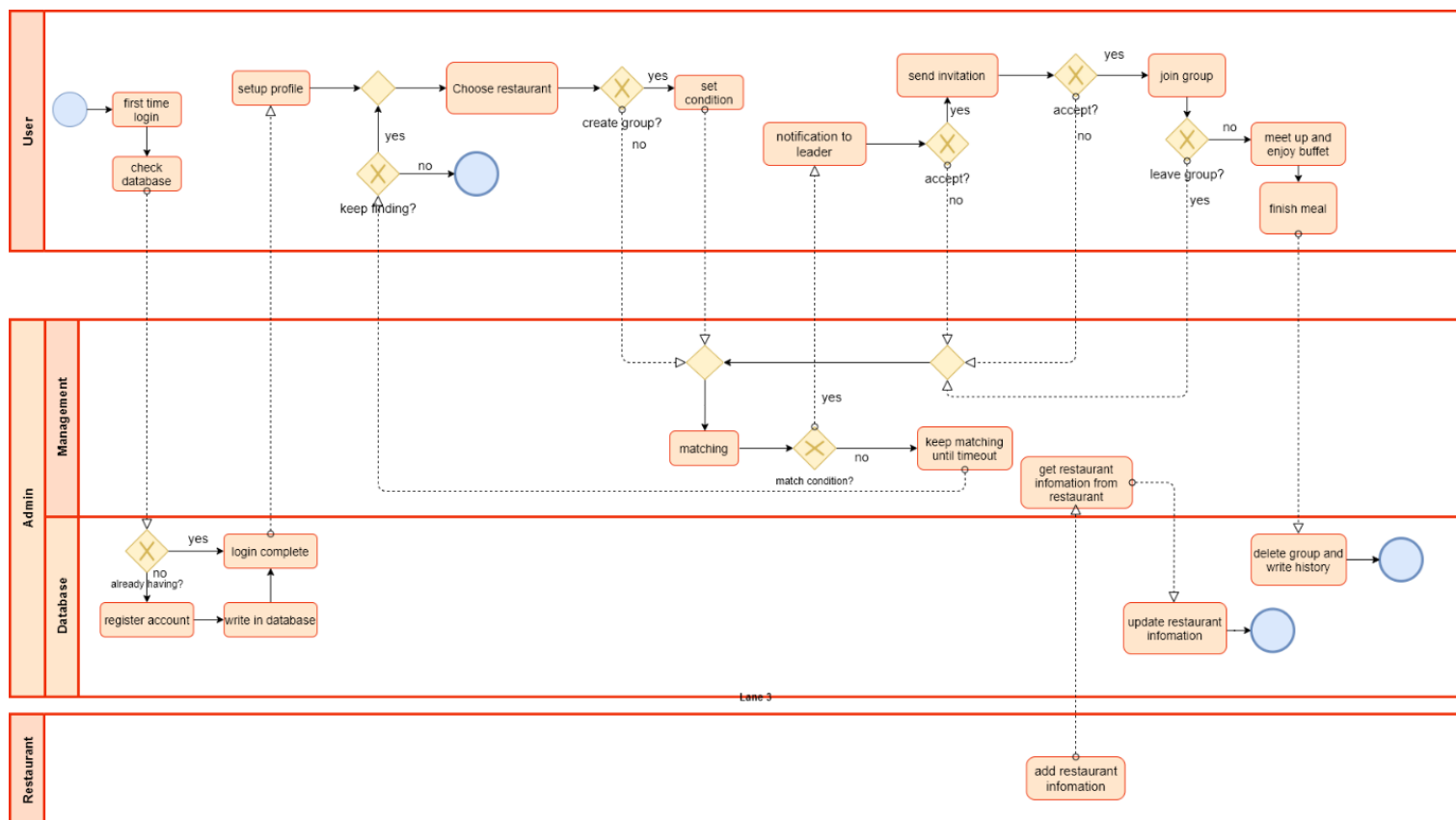


หน้าสร้างโต๊ะ
สามารถแก้ไขคุณสมบัติที่ต้องการและความชอบได้
สามารถกด **Post** เพื่อสร้างโต๊ะได้





Business Process Model and Notation (BPMN)



ระบบนี้จะแบ่งเป็น 3 ส่วนคือ User Admin และ restaurant โดย Admin จะดูแลเรื่องของระบบ matching และจัดการ Database ด้วย

1. โดยเริ่มต้นที่ User ทำการ login หากมี account อยู่แล้วก็จะสามารถเข้าสู่ระบบได้เลย แต่หากไม่มีก็ต้องไป register ก่อนถึงจะสามารถเข้าสู่ระบบได้ โดยก่อนที่จะไป Match จะต้องตั้งค่าโปรไฟล์เพื่อนำมาใช้ในการหาโต๊ะให้เรียบร้อยก่อน
2. เลือกร้านบุฟเฟต์ที่ต้องการจากนั้นเลือกที่จะกดปุ่ม Match เพื่อคนหาโต๊ะที่ถูกสร้างไว้ในร้านที่มีเงื่อนไขกลุ่มตรงกับคุณสมบัติของเรา หรือสร้างโต๊ะใหม่ หากมีการสร้างโต๊ะใหม่ จะต้องกำหนดเงื่อนไขต่างๆที่ใช้ในการค้นหาสมาชิกคนอื่นๆที่มีคุณสมบัติตรงกับเงื่อนไขแล้วจึงโพสต์กลุ่มทิ้งไว้
3. เมื่อทำการ Match เรียบร้อย การแจ้งเตือนจะถูกส่งไปที่คนที่สร้างโต๊ะหรือที่เรียกว่าหัวหน้าโต๊ะ เพื่อทำการตัดสินใจว่าจะเชิญใครที่มีคุณสมบัติตรงกับโต๊ะเข้าร่วมโต๊ะบ้าง เมื่อหัวหน้าโต๊ะกดเชิญ การแจ้งเตือนเชิญเข้าร่วมโต๊ะนั้นจะถูกส่งไปที่คนๆนั้น

4. เมื่อมีการแจ้งเตือนเชิญเข้าร่วมโต๊ะส่งเข้ามา หากผู้ค้นหาโต๊ะตัดสินใจยอมรับ ผู้ค้นหาผู้นั้นก็จะสามารถเข้าร่วมโต๊ะสำเร็จ แต่ถ้าผู้ค้นหาปฏิเสธก็ต้องรอการแจ้งเตือนเชิญเข้าร่วมโต๊ะต่อไป จนกว่าจะหมด time out หรือ กดปุ่มยกเลิก Match เอง

5. หากยังไม่ถึงเวลานัดหมายคนในกลุ่มสามารถออกจากกลุ่ม และทำการ Match กับกลุ่มอื่นได้

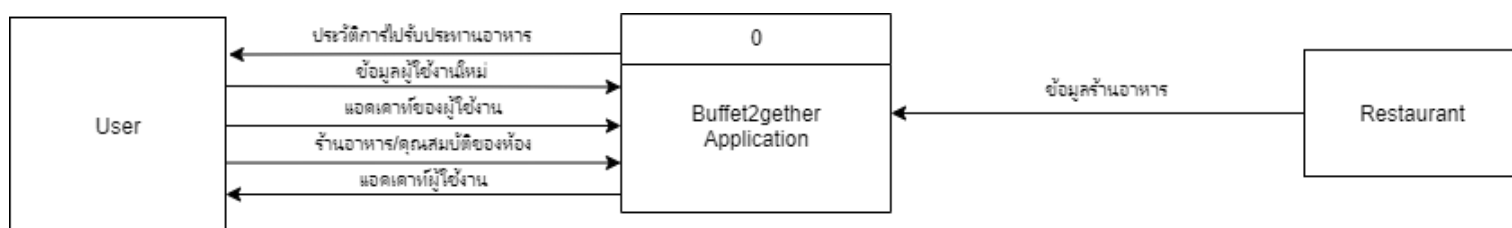
6. เมื่อถึงเวลานัดหมาย สมาชิกในกลุ่มไปรับประทานอาหารบุฟเฟ่ต์ด้วยกันตามที่นัดหมายไว้ และเมื่อเสร็จสิ้นหัวหน้าโต๊ะจะกดปุ่มลบกลุ่มและบันทึกประวัติการไปรับประทานอาหารบุฟเฟ่ต์ในครั้งนี้ไว้ที่หน้า history ของทุกคน

โดยที่ร้านบุฟเฟ่ต์จะติดต่อและส่งข้อมูลของร้านมาที่ Admin เพื่อทำการเพิ่มหรือปรับเปลี่ยนร้านในฐานข้อมูลของระบบ

Data Flow Diagram (DFD)

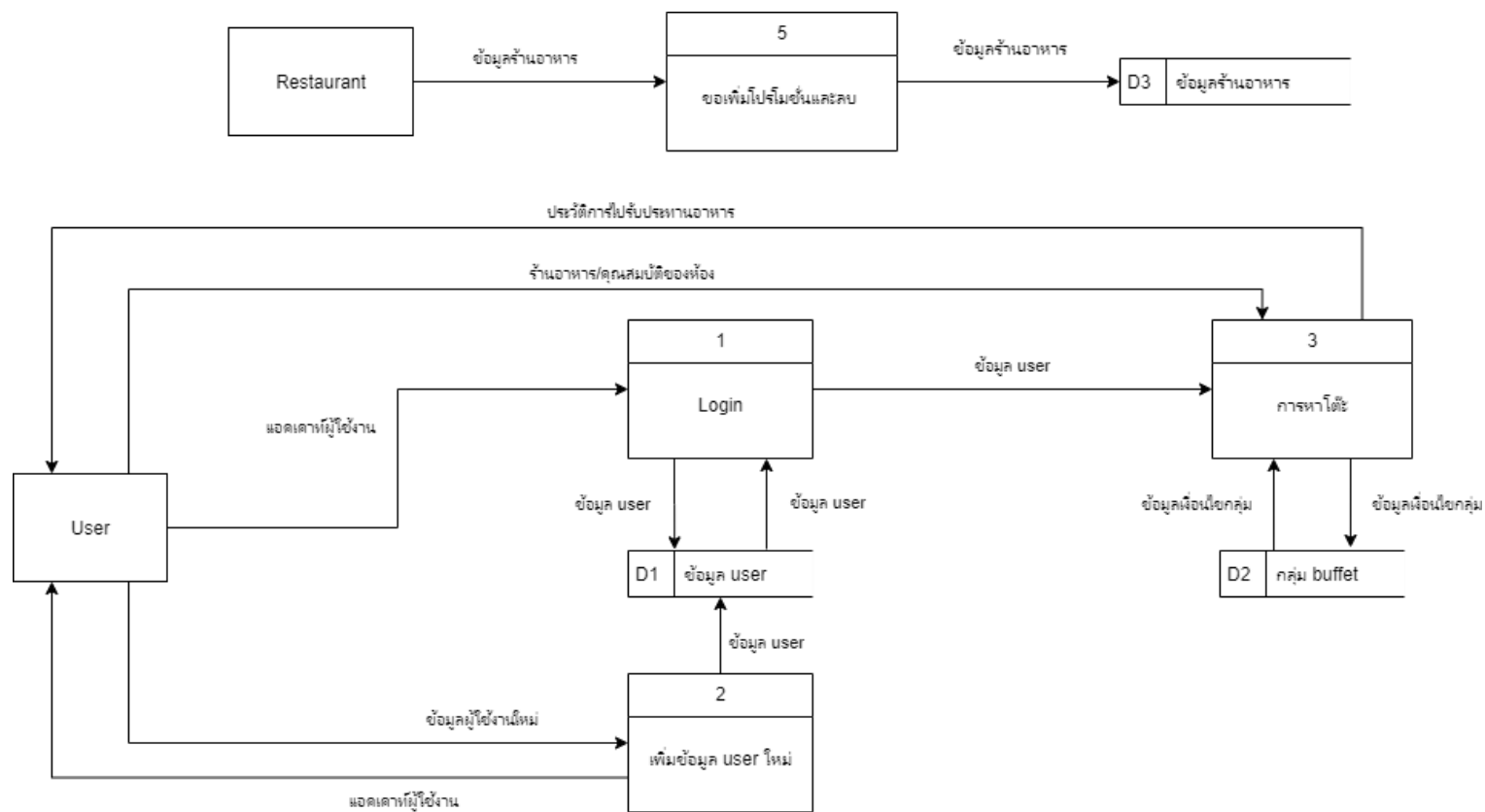
Buffet2gether

Context Diagram



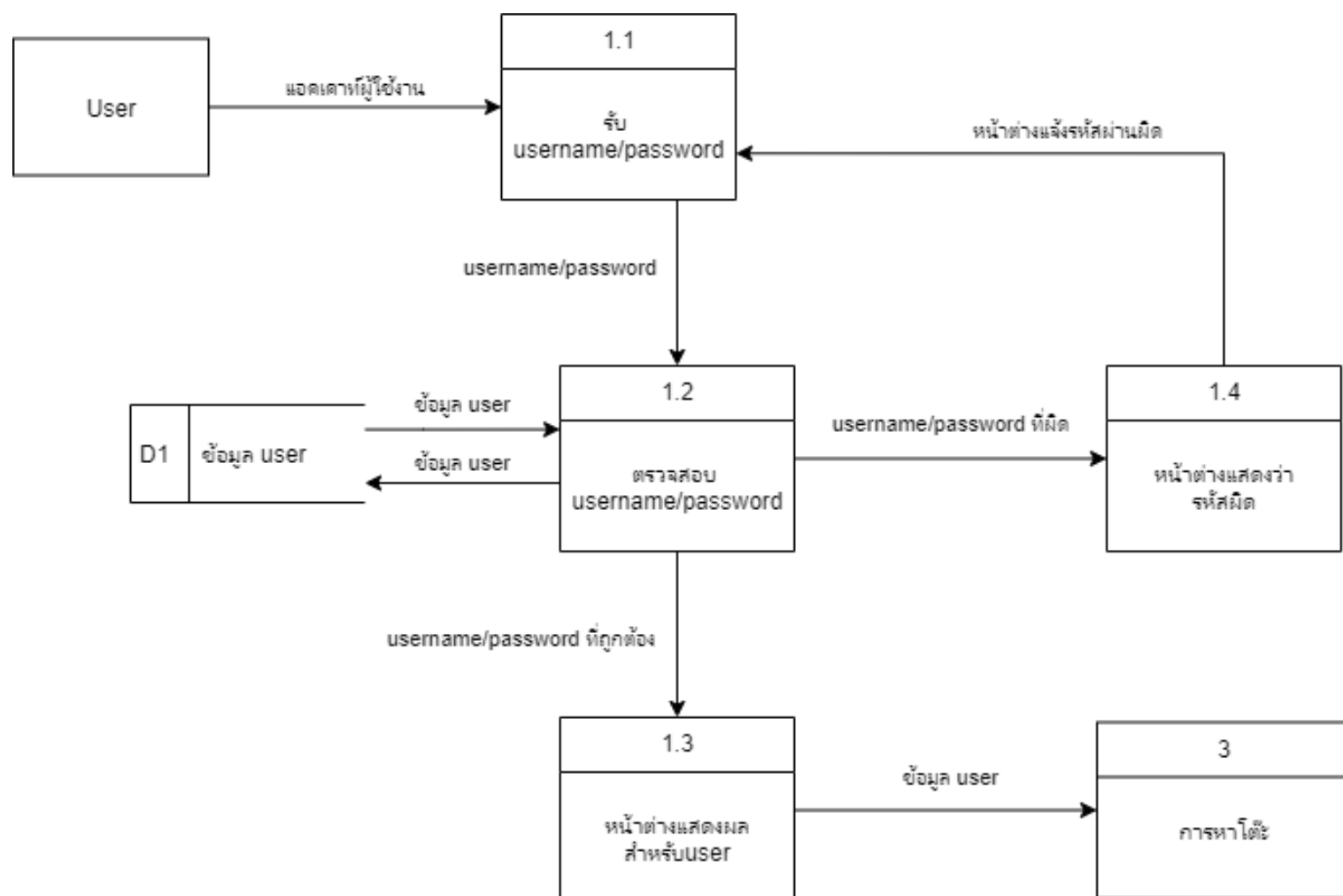
Buffet2gether

Data Flow Diagram Level 0



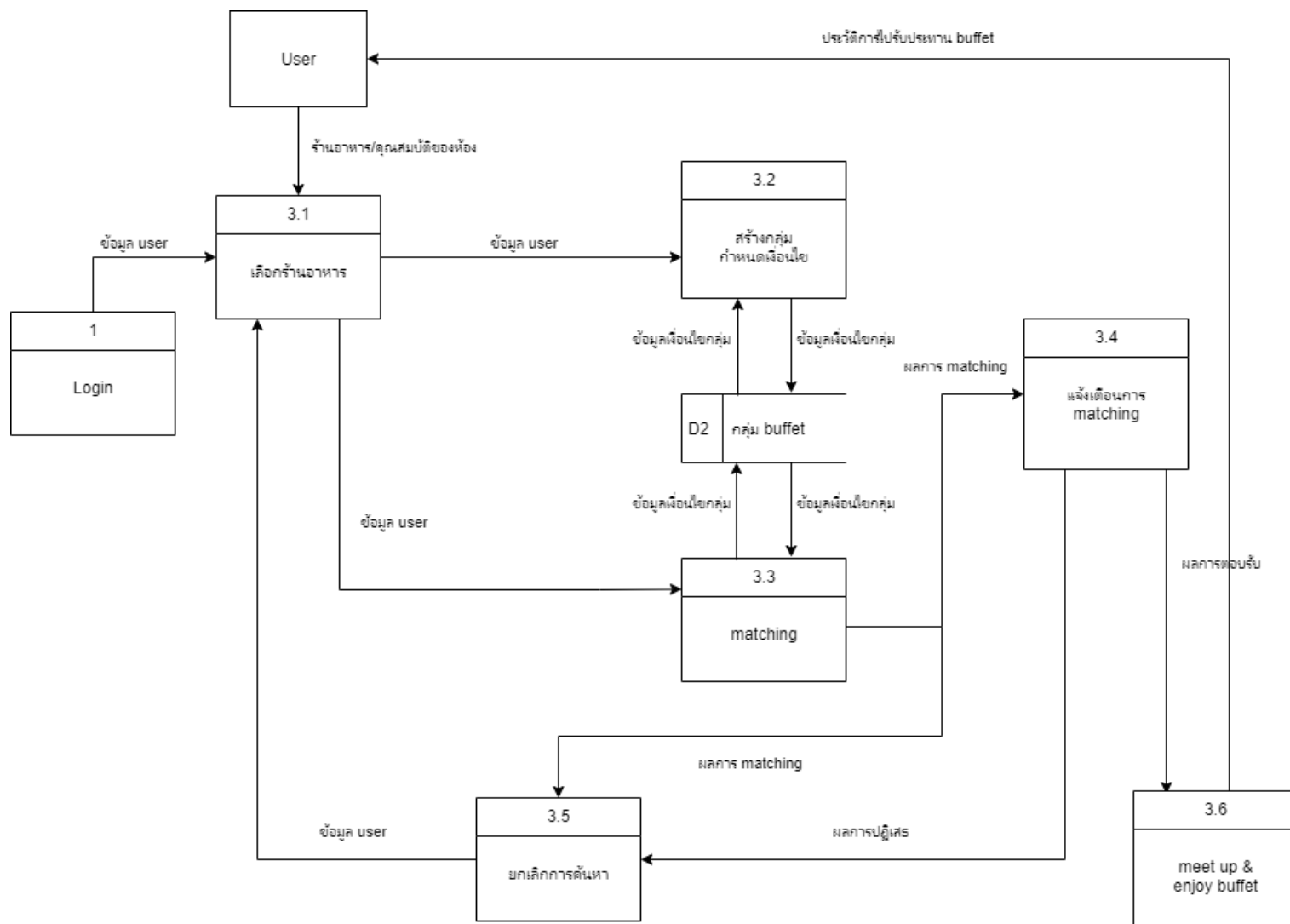
Buffet2gether

Data Flow Diagram Level 1



Buffet2gether

Data Flow Diagram Level 3



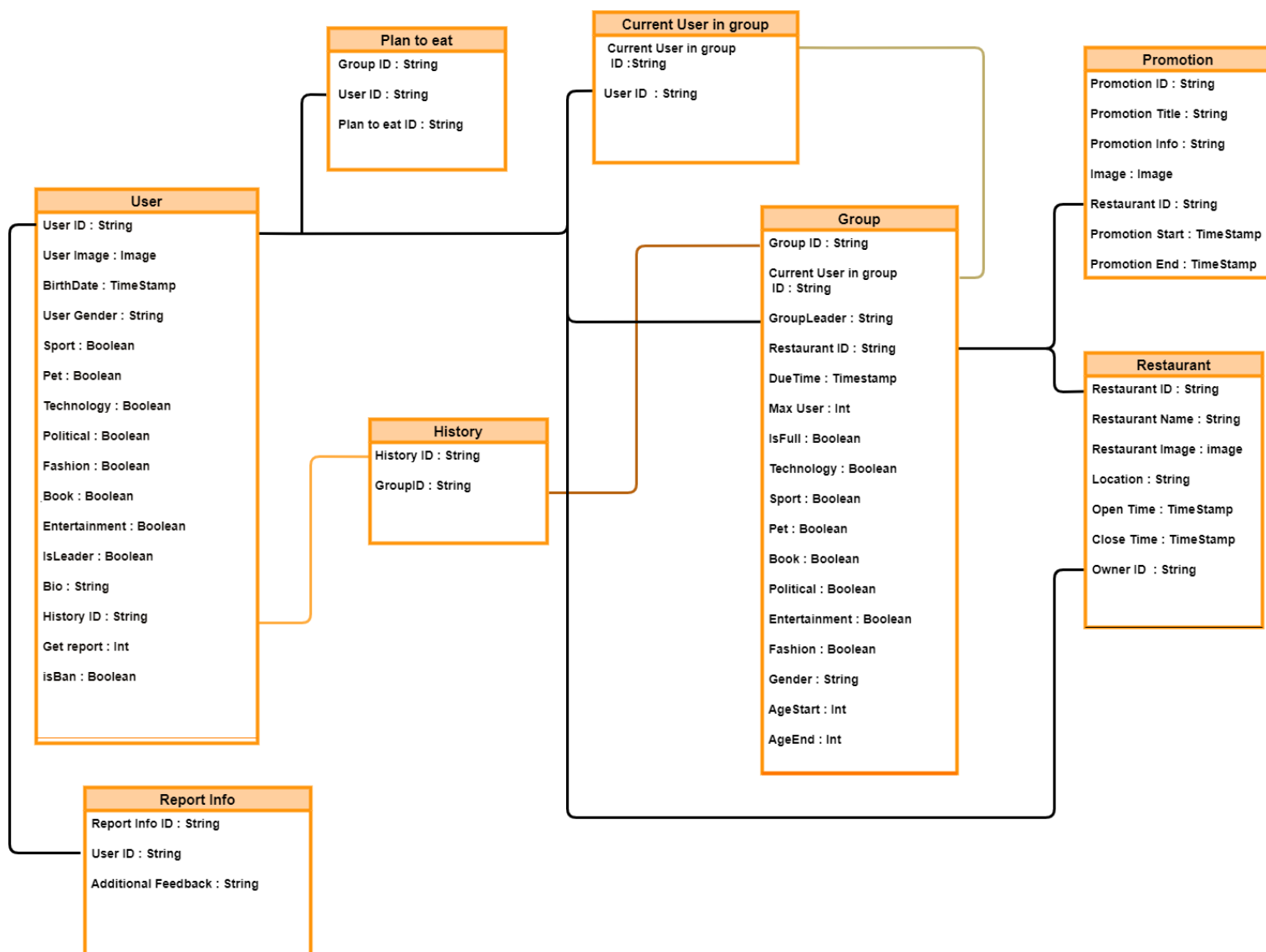
ในแผนภาพนี้จะเป็นแผนภาพแสดงขั้นตอนการทำงานของระบบ โดยจะมีสิ่งที่อยู่ภายนอก External Entity เพียง 2 อย่าง คือ ผู้ใช้งาน(user) และร้านอาหาร (restaurant) โดยมีการนำข้อมูลเข้าจากผู้ใช้งาน 3 อย่างคือ ข้อมูลผู้ใช้งานใหม่ บัญชีผู้ใช้งาน และ ร้านอาหาร/คุณสมบัติของโต๊ะ และจะส่งออกข้อมูลไปยังผู้ใช้งาน 2 อย่างคือ ประวัติการไปรับประทานอาหาร และ บัญชีผู้ใช้งาน นอกจากนี้ระบบจะรับข้อมูลจากร้านอาหารอีก 1 อย่าง คือ ข้อมูลร้านอาหาร

เมื่อผู้ใช้งานเข้าสู่ระบบ ระบบจะส่งข้อมูลไปยังระบบการหาโต๊ะที่รับข้อมูลจากผู้ใช้งาน เมื่อรับประทานเสร็จแล้วก็จะส่งประวัติการรับประทานอาหารไปให้ผู้ใช้งาน ในกรณีที่ผู้ใช้งานเข้าสู่ระบบครั้งแรก ระบบจะสร้างข้อมูลผู้ใช้งานใหม่ และเก็บข้อมูลผู้ใช้งานลงไปในระบบ

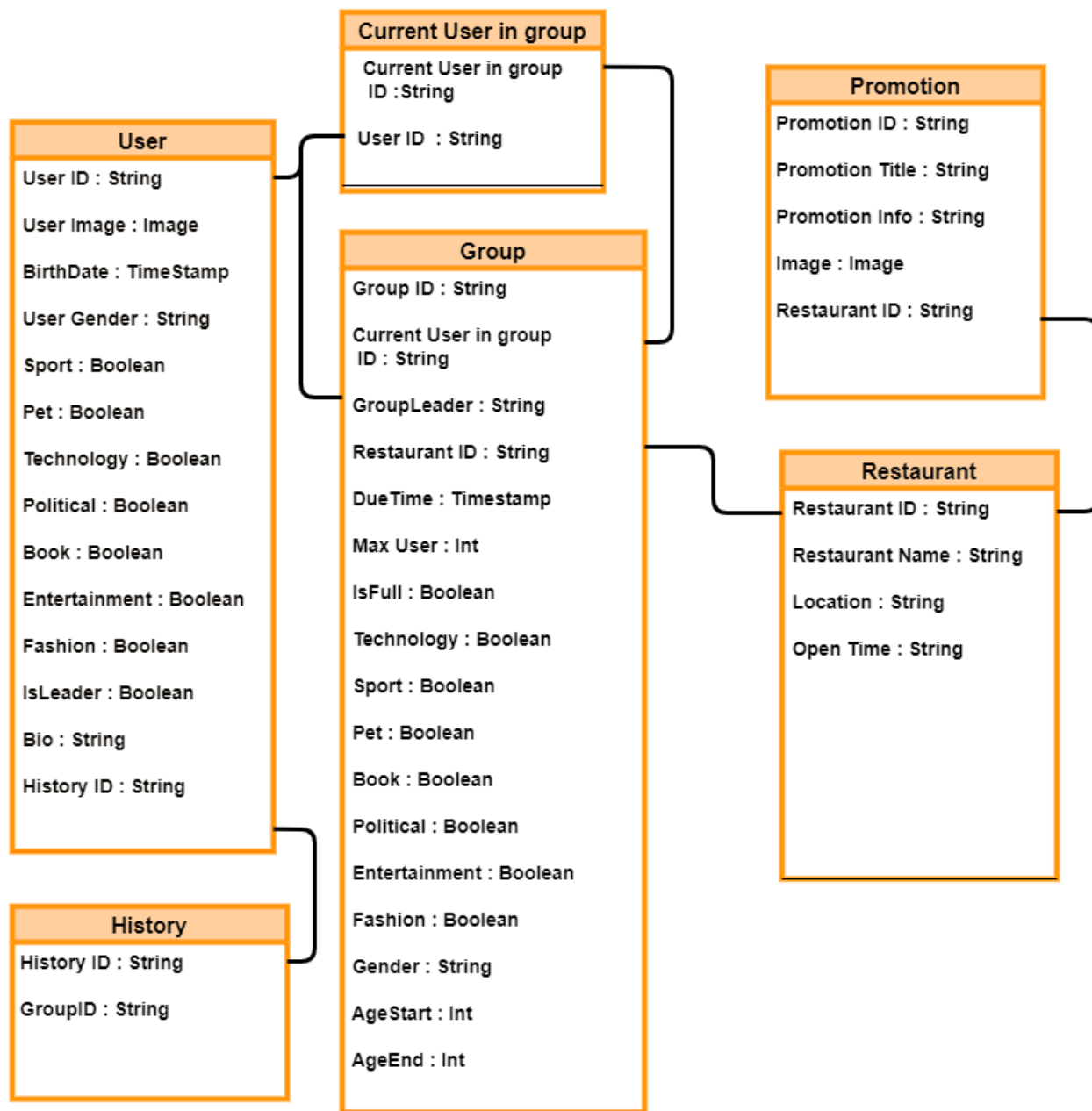
เจ้าของร้านอาหารจะสามารถขอเพิ่มหรือลบโปรโมชั่นได้ และจะเก็บข้อมูลนั้นในระบบ

Entity Relationship Diagram (ER Diagram)

ER Diagram ที่ออกแบบสมบูรณ์



ER Diagram ที่ทำจริง



User collection

Description : เป็นcollection สำหรับเก็บข้อมูลของUser

User ID - เป็นIDของแต่ละUser

UserImage - เก็บรูปภาพประจำตัว

BirthDate - เก็บวันเกิดของ

User Gender - เก็บเพศ

Sport - เก็บความชอบด้านกีฬา

Pet - เก็บความชอบด้านสัตว์เลี้ยง

Technology - เก็บความชอบด้านเทคโนโลยี

Political - เก็บความชอบด้านการเมือง

Book - เก็บความชอบด้านหนังสือ

Entertainment - เก็บความชอบด้านสื่อบันเทิง

Fashion - เก็บความชอบด้านแฟชั่น

IsLeader - เก็บข้อมูลว่าเป็นหัวหน้าโต๊ะอยู่หรือไม่

Bio - เก็บข้อความสำหรับบอกเล่าเกี่ยวกับตัวเอง

History ID - เก็บข้อมูลของHistory IDสำหรับตั้งประวัติการกินร้านอาหาร

History collection

Description : เป็นcollectionสำหรับเก็บประวัติการทานอาหารของแต่ละUser

History ID - เป็น ID ของแต่ละข้อมูลประวัติร้านอาหารที่เก็บ

Group ID - เก็บ ID ของ Group ID สำหรับตั้งรายละเอียดโต๊ะอาหารที่เคยไปรับประทาน

Current User In Group collection

Description : เป็นcollectionสำหรับเก็บข้อมูลสมาชิกในกลุ่ม

Current User in Group ID - เป็น ID ของแต่ละ Current User Ingroup

User ID - เป็น ID ของแต่ละ user ที่เข้าร่วมกลุ่ม

Group collection

Description : เป็นcollectionสำหรับเก็บข้อมูลกลุ่ม

Group ID - เป็น ID ของแต่ละกลุ่ม

Current User in Group ID- เก็บไอดีของ Current User In Group สำหรับดึงข้อมูลสมาชิกในกลุ่ม

Group Leader - เก็บUser IDของหัวหน้าโต๊ะ

Restaurant ID - เก็บ ID ของร้านบุฟเฟ่ต์ที่กลุ่มตั้งเป้าจะไปรับประทาน

DueTime - เก็บเวลานัดหมายสำหรับกินบุฟเฟ่ต์

Max User - เก็บจำนวนสูงสุดของสมาชิกในโต๊ะ

IsFull - เก็บข้อมูลว่ากลุ่มเต็มหรือยัง

Technology - เก็บการตั้งค่าโต๊ะหมวดความชอบเทคโนโลยี

Sport - เก็บการตั้งค่าโต๊ะหมวดความชอบกีฬา

Pet - เก็บการตั้งค่าโต๊ะหมวดความชอบสัตว์เลี้ยง

Book - เก็บการตั้งค่าโต๊ะหมวดความชอบหนังสือ

Political - เก็บการตั้งค่าโต๊ะหมวดความชอบการเมือง

Entertainment - เก็บการตั้งค่าโต๊ะหมวดความชอบสื่อบันเทิง

Fashion - เก็บการตั้งค่าโต๊ะหมวดความชอบแฟชั่น

Gender - เก็บการตั้งค่าโต๊ะหมวดเพศ

AgeStart - เก็บการตั้งค่าโต๊ะหมวดอายุขั้นต่ำ

AgeEnd - เก็บการตั้งค่าโตะหมวดอายุมากที่สุด

Promotion collection

Description : เป็นcollectionสำหรับเก็บโปรโมชั่นต่างๆของร้านบุฟเฟ่

Promotion ID : เก็บID ของแต่ละโปรโมชั่น

Promotion Title : เก็บหัวข้อของโปรโมชั่น

Promotion Info : เก็บรายละเอียดของโปรโมชั่น

Image : เก็บรูปภาพของโปรโมชั่น

Restaurant ID : เก็บID ของร้านอาหารสำหรับนำไปใช้แสดงโปรโมชั่นของร้านอาหารนั้นๆ

Restaurant collection

Description : เป็นcollectionสำหรับเก็บข้อมูลร้านอาหาร

Restaurant ID : เป็นไอดีของแต่ละร้านบุฟเฟ่

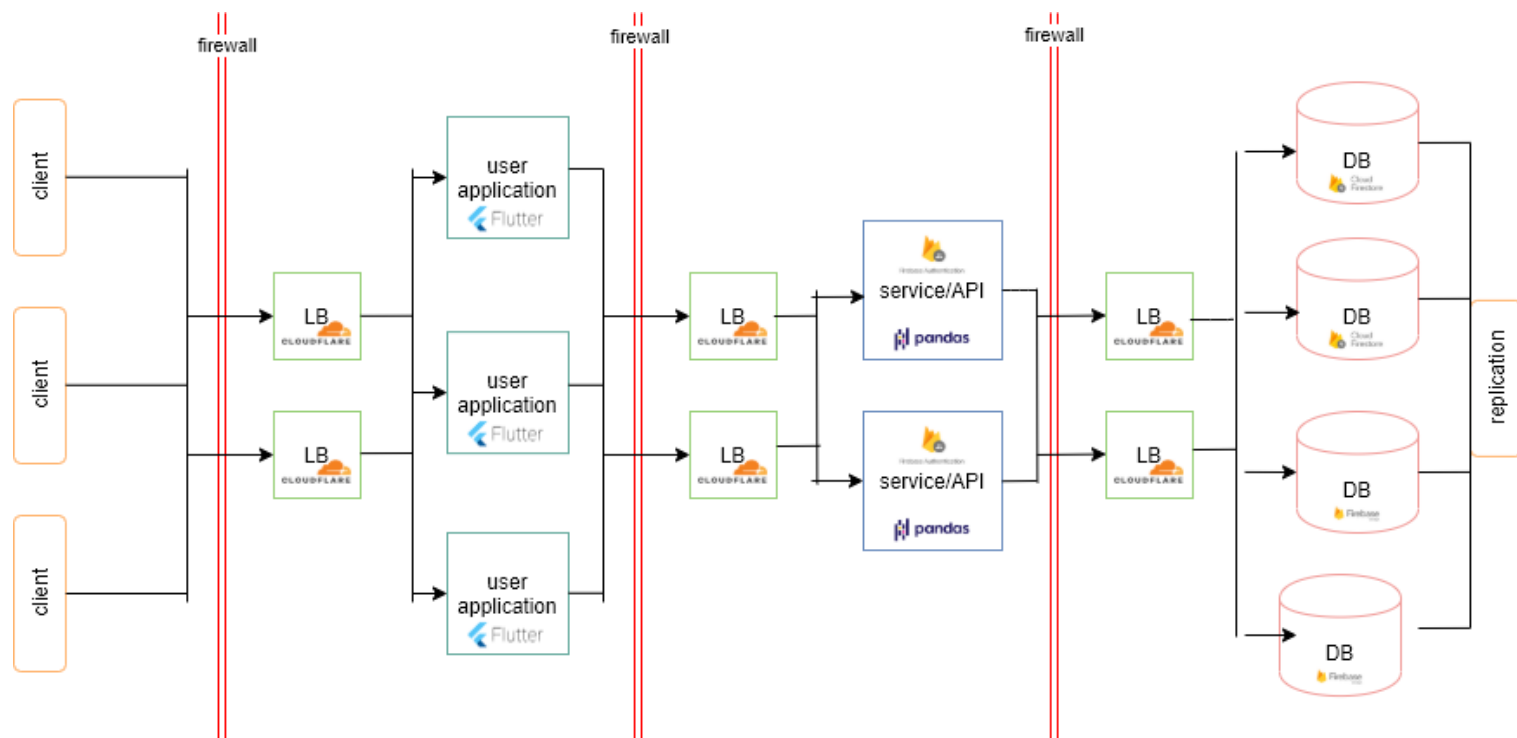
Restaurant Name : เก็บชื่อของร้านบุฟเฟ่

Location : แสดงที่อยู่ของร้านอาหาร

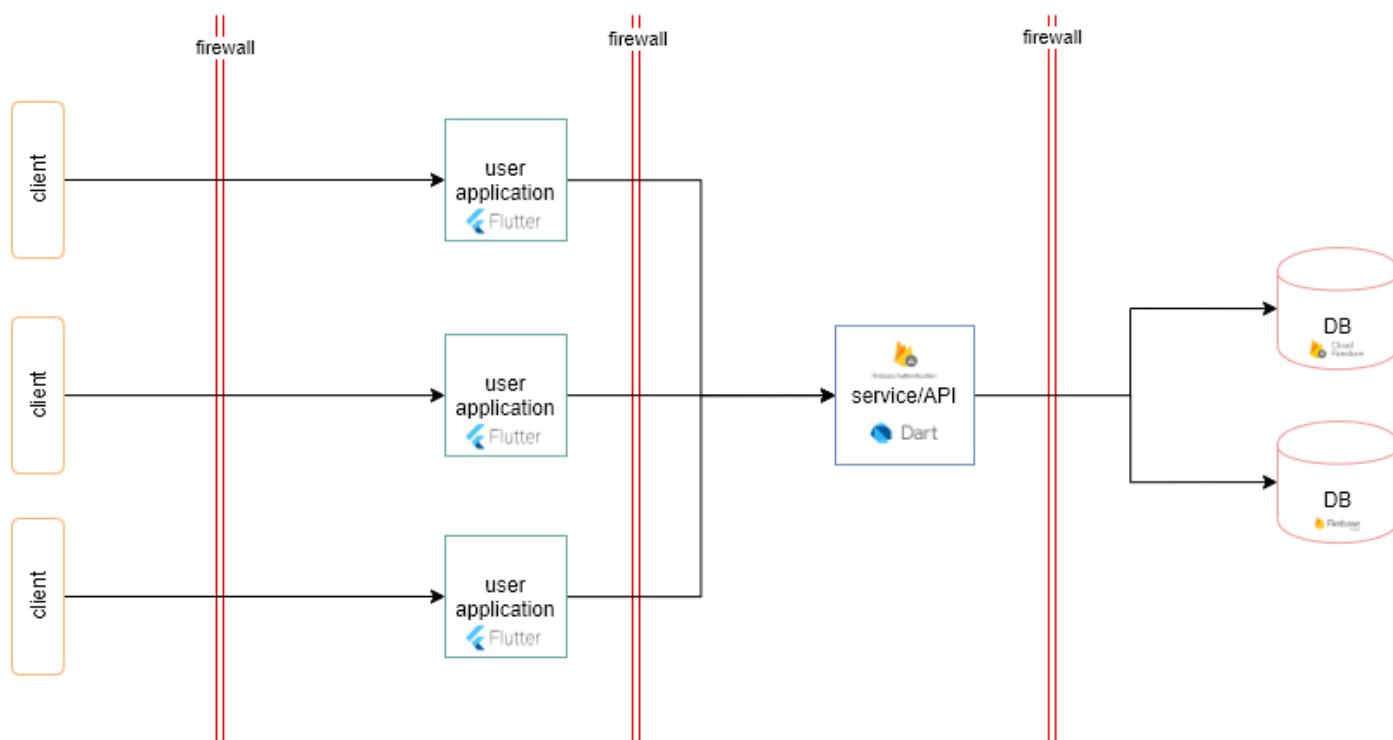
Open Time :แสดงเวลาเปิดปิดของร้าน

Software Architecture

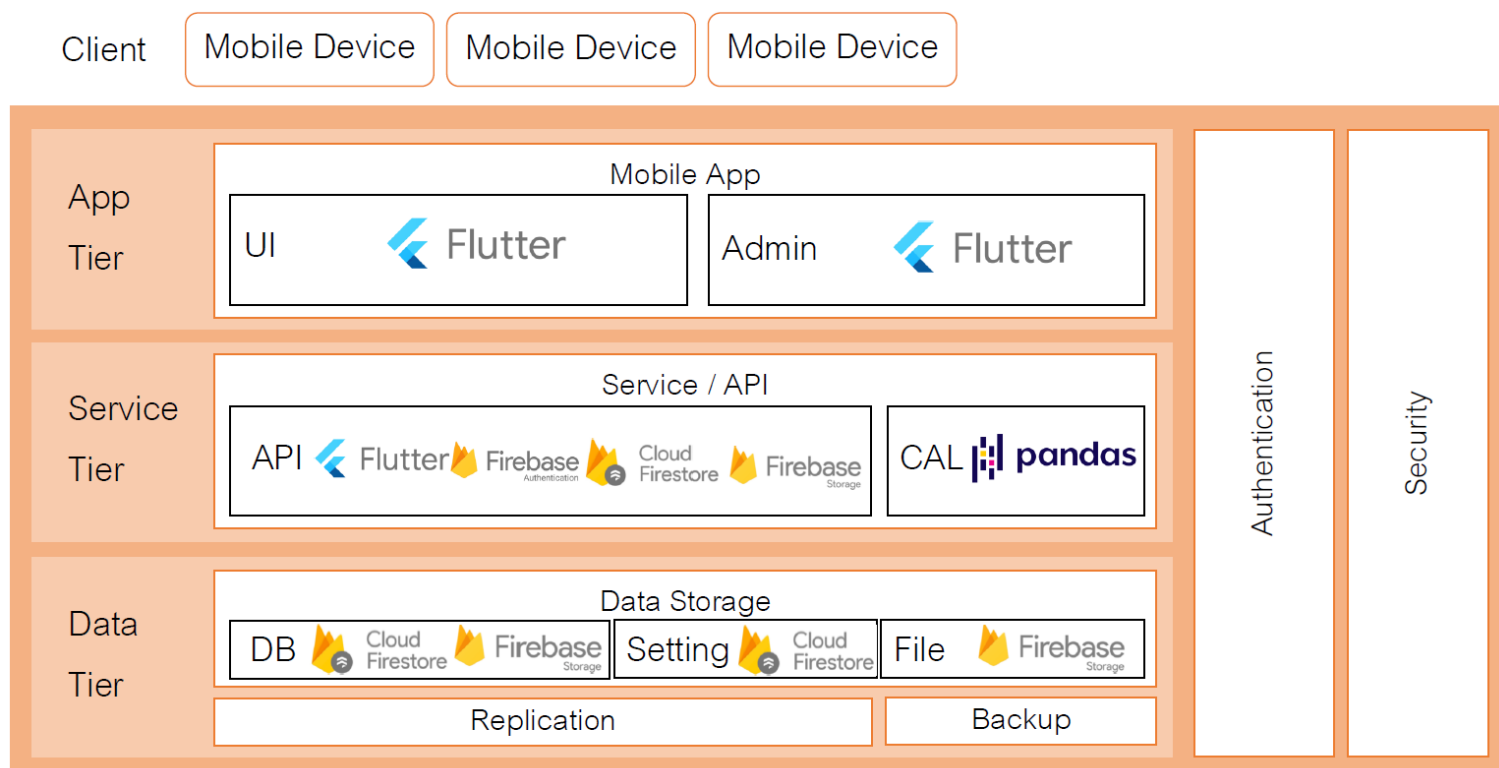
3-Tier Architecture ที่ออกแบบสมบูรณ์



3-Tier Architecture ที่ทำจริง



Software Architecture: Stack

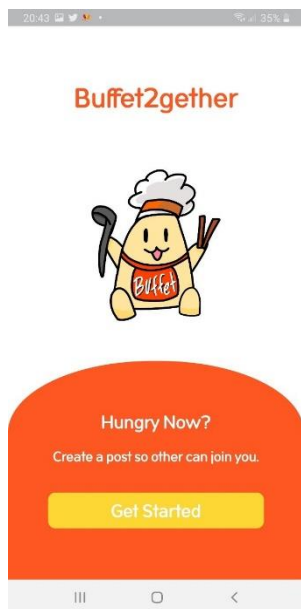


ผลการพัฒนา

แอปพลิเคชัน

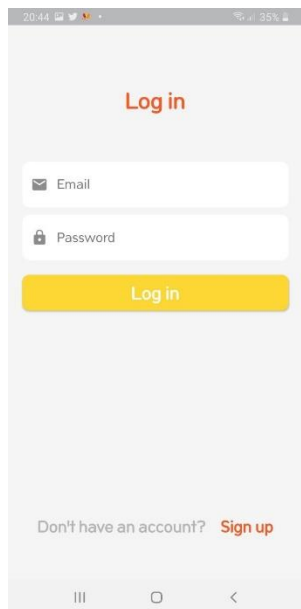
หน้าเริ่มต้น

คลิก **Get Started**
เพื่อเข้าสู่หน้า **Log in**



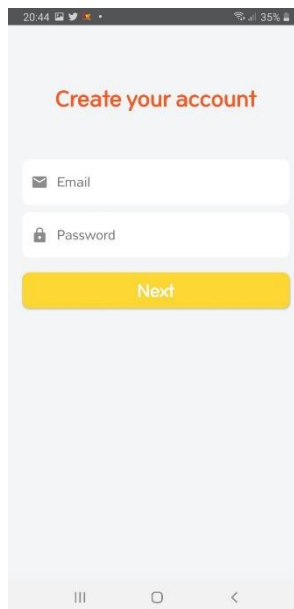
หน้า **Log in**

หากสมัครสมาชิกแล้วสามารถกรอก **Email**
และ **Password** เพื่อเข้าสู่ระบบได้
หากยังไม่สมัครสมาชิกสามารถกด **Sign up**
เพื่อสมัครสมาชิกได้

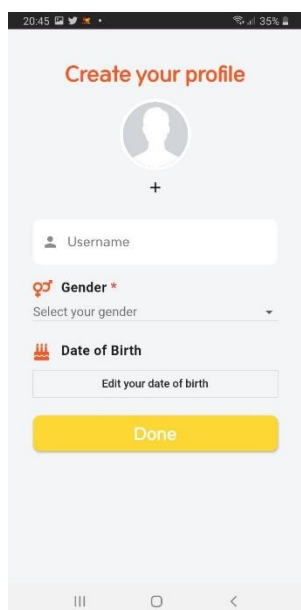


หน้าสมัครสมาชิก

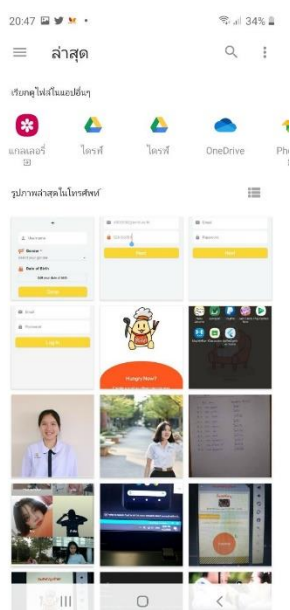
กรอก **Email** และ **Password** ที่ไม่มีในระบบเพื่อสมัครสมาชิก
คลิก **Next** เพื่อเข้าสู่หน้ากรอกข้อมูลส่วนตัว



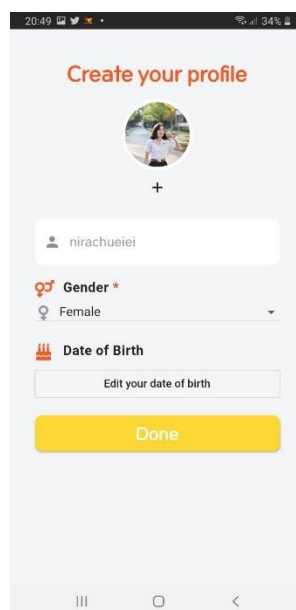
หน้ากรอกข้อมูลส่วนตัว



คลิกไอคอน + เพื่อเปลี่ยน
รูปภาพประจำตัวโดยเลือก
จากภาพในเครื่องได้

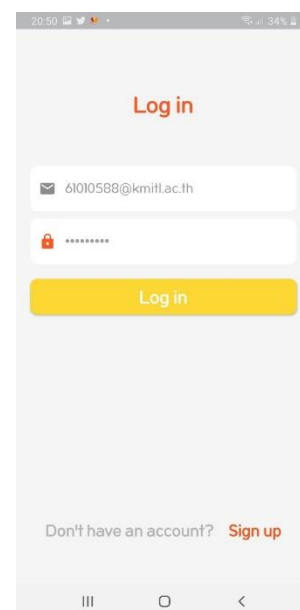


กรอกข้อมูลส่วนตัวให้ครบถ้วน
คลิก **Done** เพื่อเข้าสู่หน้า **Log in**



หน้า **Log in**

กรอก **Email** และ **Password**
ที่สมัครสมาชิกไว้แล้วกด **Log in**
เพื่อเข้าสู่หน้าหลัก



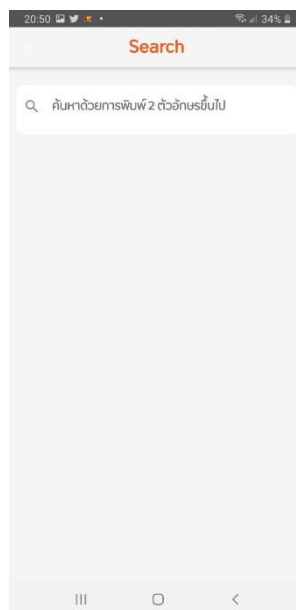
หน้าหลัก
สามารถเลื่อนดูโปรโมชันแนะนำ
ร้านอาหารที่แนะนำ หรือ
ร้านอาหารต่าง ๆ ได้



หน้ารายละเอียดร้าน
สามารถคลิกที่ร้านอาหารในหน้า
หลักเพื่อเข้าสู่หน้ารายละเอียด
ร้านอาหารเพื่อดูข้อมูลของร้านได้



หน้าค้นหา
สามารถคลิกไอคอนแว่นขยายใน
หน้าหลักเพื่อเข้าสู่หน้าค้นหาได้



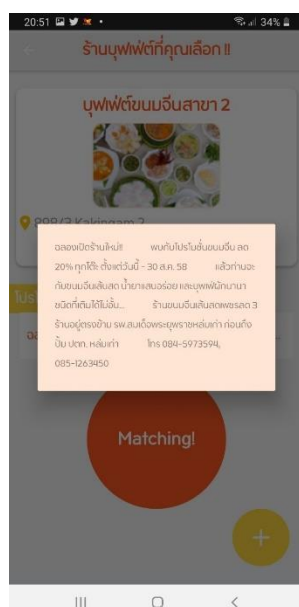
หน้าค้นหา
สามารถค้นหาร้านอาหาร
จากชื่อร้านได้



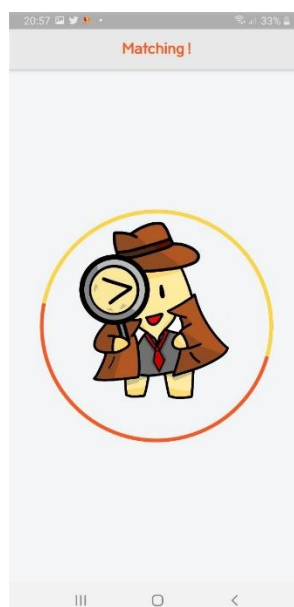
หน้ารายละเอียดร้าน
สามารถอ่านอาหารในหน้าค้นหา
เพื่อเข้าสู่หน้ารายละเอียด
ร้านอาหารเพื่อดูข้อมูลของร้านได้



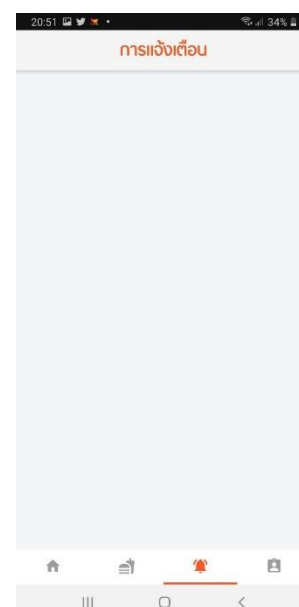
หน้ารายละเอียดร้าน
สามารถคลิกที่โปรโมชันเพื่อดู
โปรโมชั่นของร้านได้



หน้า Matching
สามารถกด Matching ใน
หน้ารายละเอียดร้านเพื่อค้นหา
โต๊ะได้ อยู่ในสถานะผู้ค้นหาโต๊ะ



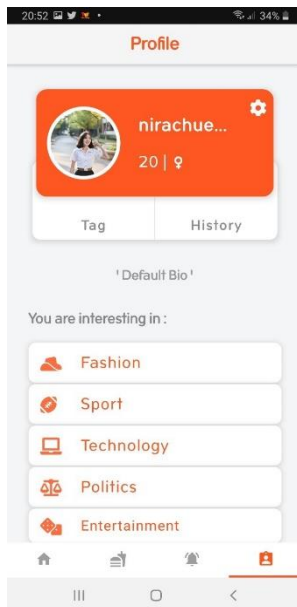
หน้าแจ้งเตือน
หลังจากกด Matching แล้ว
หากมีหัวหน้าโต๊ะใดส่งคำเชิญมา
ให้ก็จะมีการแจ้งเตือนขึ้นมา



หน้ารายละเอียดร้าน
หลังจากกด **Matching** ไป
แล้วจะขึ้นสถานะในหน้า
รายละเอียดร้านว่ากำลังค้นหาได้
หากต้องการยกเลิกการ **Match**
สามารถกดไอคอนดังชยะได้



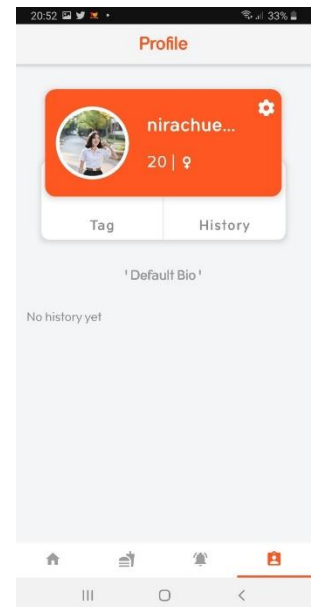
หน้าข้อมูลส่วนตัว
สามารถข้อมูลส่วนตัวต่าง ๆ ได้
เช่น รูปภาพประจำตัว,
Username, อายุ, เพศ,
ความชอบ



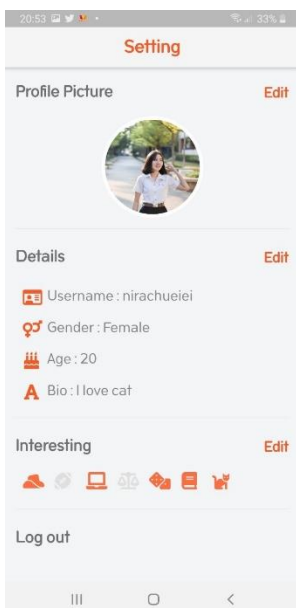
หน้าข้อมูลส่วนตัว
สามารถกดที่รูปภาพประจำตัวเพื่อ
ดูรูปภาพใหญ่ได้



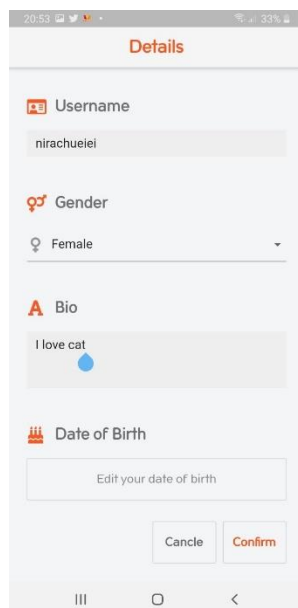
หน้าข้อมูลส่วนตัว
สามารถคลิกที่ **History** เพื่อดู
ประวัติการไปรับประทานอาหารได้



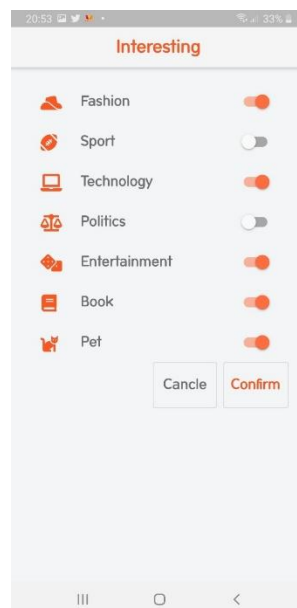
หน้าแก้ไขข้อมูลส่วนตัว
สามารถกดที่ไอคอนฟันเฟืองจาก
หน้าข้อมูลส่วนตัวเพื่อเข้าสู่หน้า
แก้ไขข้อมูลส่วนตัวได้



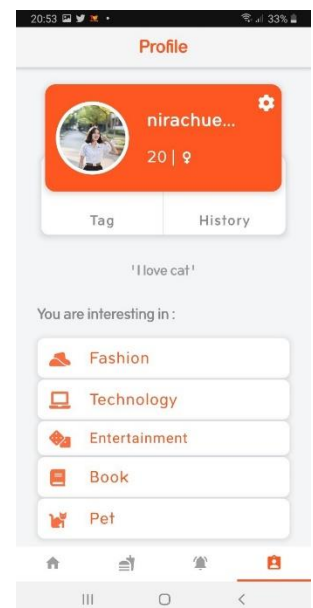
หน้าแก้ไขข้อมูลส่วนตัว
สามารถกด **Edit** จาก **Detail**
เพื่อแก้ไขข้อมูลส่วนตัวได้



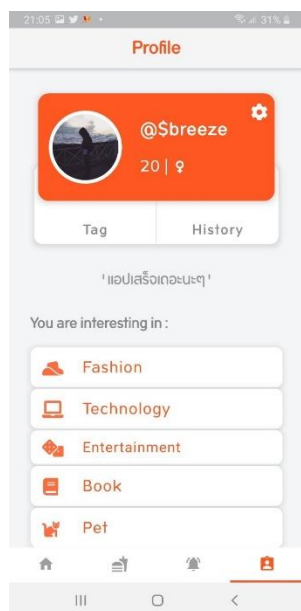
หน้าแก้ไขข้อมูลส่วนตัว
สามารถกด **Edit** จาก
Interesting เพื่อแก้ไข
ความชอบได้



หน้าข้อมูลส่วนตัว
หลังจากแก้ไขแล้ว ข้อมูลส่วนตัว
จะเปลี่ยนไป



หน้าข้อมูลส่วนตัวของบัญชีที่ 2
ซึ่งจะเป็นผู้สร้างไอดี



หน้ารายละเอียดร้าน
สามารถคลิกไอคอน +
เพื่อสร้างไอดีได้



หน้าสร้างไอดี
สามารถแก้ไขคุณสมบัติตามที่
ต้องการได้



หน้าสร้างไอดี
สามารถคลิกที่ช่วงอายุ เพื่อแก้ไข
ช่วงอายุที่ต้องการ



หน้าสร้างไอดี
สามารถคลิกที่จำนวนคน เพื่อ
แก้ไขจำนวนคนที่ต้องการ



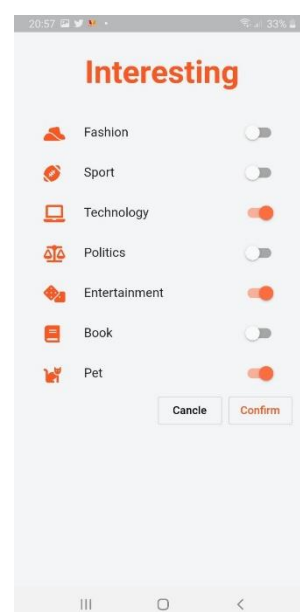
หน้าสร้างไอดี
สามารถคลิกที่วันเวลา เพื่อแก้ไข
วันเวลานัดหมายที่ต้องการ



หน้าสร้างไอดี
สามารถคลิกที่เพศ เพื่อแก้ไขเพศที่
ต้องการ



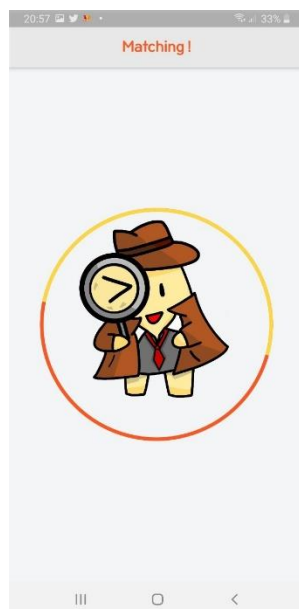
หน้าสร้างไอดี
สามารถกด Edit ที่
Interesting เพื่อแก้ไข
ความชอบตามต้องการ



หน้าสร้างโຕะหลังจากที่แก้ไข
คุณสมบัติต่าง ๆ แล้ว



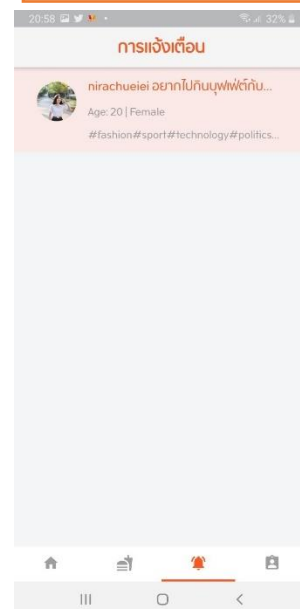
หน้าสร้างโຕะ
หลังจากกด Post



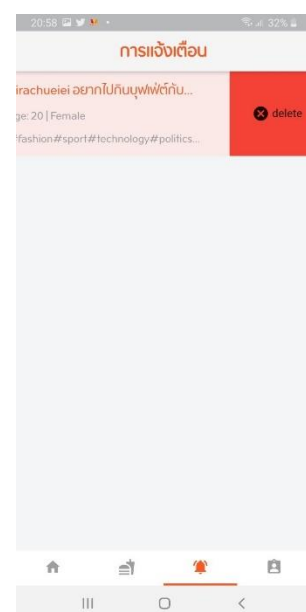
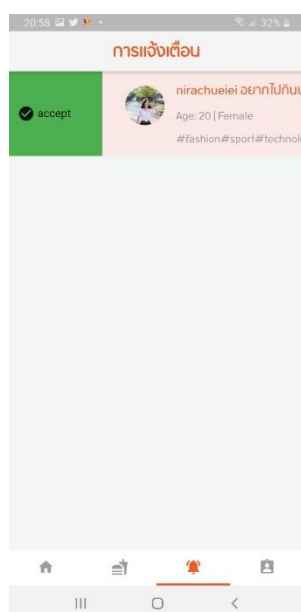
หน้าโຕะของหัวหน้าโຕะ
แสดงรายละเอียดของโຕะเรา



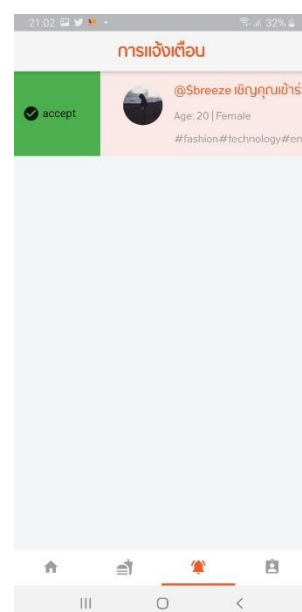
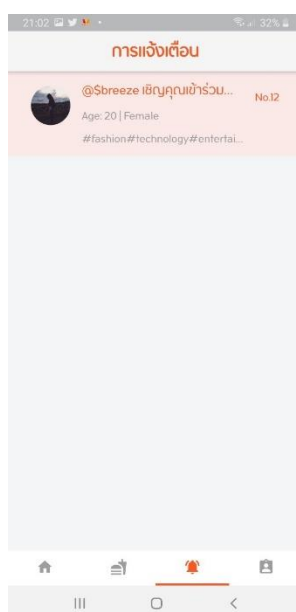
หน้าแจ้งเตือนของหัวหน้าโຕะ
เมื่อมีผู้ค้นหาโຕะที่มีคุณสมบัติ
ตรงกันจะแจ้งเตือนที่หัวหน้าโຕะ



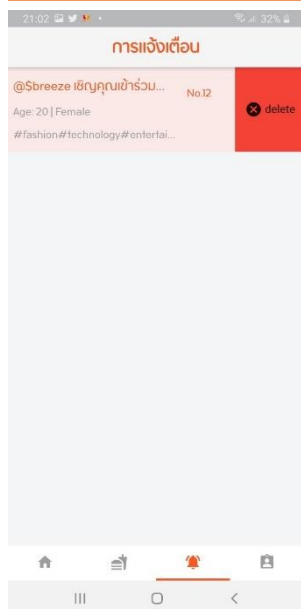
หน้าแจ้งเตือนของหัวหน้าโຕะ
หัวหน้าโຕะสามารถตอบรับและส่งคำเชิญ
หรือปฏิเสธเพื่อลบแถบแจ้งเตือนได้



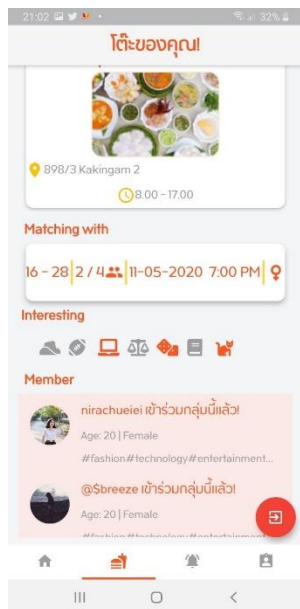
หน้าแจ้งเตือนของผู้ค้นหาโຕะ
ผู้ค้นหาโຕะสามารถตอบรับคำเชิญและเข้าร่วมโຕะ
หรือปฏิเสธเพื่อลบแถบแจ้งเตือนได้



หน้าแจ้งเตือนของผู้ค้นหาโต๊ะ
ผู้ค้นหาโต๊ะสามารถตอบรับคำเชิญ
และเข้าร่วมโต๊ะ
หรือปฏิเสธเพื่อลบแจ้งเตือนได้



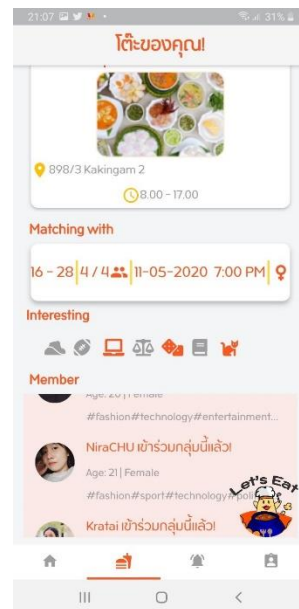
หน้าโต๊ะของผู้ค้นหาโต๊ะ
เมื่อผู้ค้นหาโต๊ะตอบรับคำเชิญและ
เข้าร่วมโต๊ะแล้ว ที่หน้าโต๊ะก็จะ
แสดงรายละเอียดของโต๊ะที่อยู่
มีปุ่มออกจากโต๊ะหากสมาชิกใน
โต๊ะยังไม่ครบ



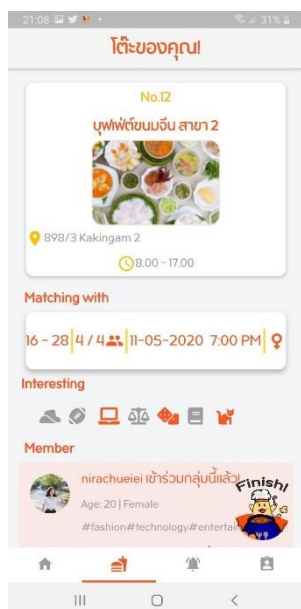
หน้าแจ้งเตือนของหัวหน้าโต๊ะ
เมื่อมีผู้ค้นหาโต๊ะที่มีคุณสมบัติ
ตรงกันจะแจ้งเตือนที่หัวหน้าโต๊ะ



หน้าโต๊ะของสมาชิก
เมื่อจำนวนสมาชิกในโต๊ะครบแล้ว
ปุ่มออกจากโต๊ะจะเปลี่ยนเป็น
รูปภาพและกดออกจากโต๊ะไม่ได้



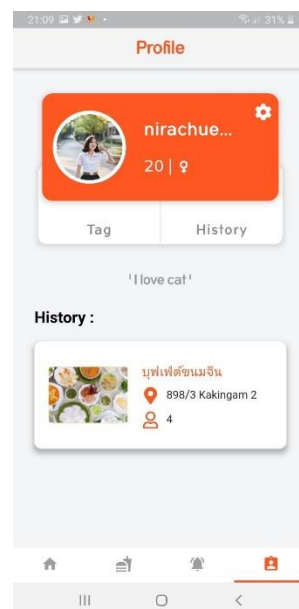
หน้าโต๊ะของหัวหน้าโต๊ะ
เมื่อรับประทานอาหารเสร็จแล้ว
หัวหน้าโต๊ะสามารถกดปุ่ม
Finish เพื่อลบโต๊ะและเก็บ
ประวัติ



หน้าโต๊ะของสมาชิกทุกคนในโต๊ะ
หลังจากกดปุ่ม **Finish** เพื่อลบ
โต๊ะและเก็บประวัติแล้ว



หน้าข้อมูลส่วนตัวของสมาชิกทุกคนในโต๊ะ
หลังจากกดปุ่ม **Finish** เพื่อลบโต๊ะและเก็บประวัติแล้ว จะมีประวัติการรับประทานอาหาร
อาหารครั้งสุดท้ายแสดงที่ **History**



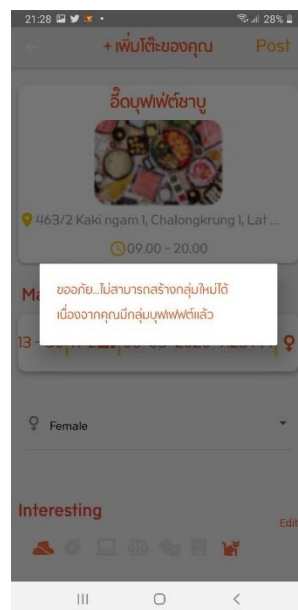
หน้าแจ้งเตือนของผู้ค้นหาโต๊ะ
ค่าเตือนเมื่อกดตอบรับคำเชิญเข้า
โต๊ะที่เต็มแล้ว



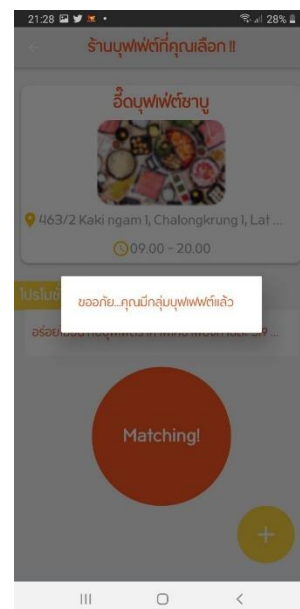
หน้าแจ้งเตือนของหัวหน้าโต๊ะ
ค่าเตือนเมื่อกดส่งคำเชิญเมื่อโต๊ะ
ของตนเองเต็มแล้ว



หน้าสร้างโต๊ะ
ค่าเตือนเมื่อกด Post เพื่อสร้างโต๊ะ
แต่มีโต๊ะอยู่แล้ว



หน้ารายละเอียดร้าน
ค่าเตือนเมื่อกด Matching เพื่อ
ค้นหาโต๊ะ แต่มีโต๊ะอยู่แล้ว



Url ของ github: <https://github.com/Buffer2gether/buffet2gether>

วิธีการใช้งานระบบ

1. เข้าแอปพลิเคชัน จะพบกับหน้าเริ่มต้น
2. ที่หน้าเริ่มต้น คลิก Get Start เพื่อเข้าสู่หน้า Log in
3. ที่หน้า Log in
 - หากยังไม่ได้สมัครสมาชิก
 - คลิก Sign Up เพื่อเข้าสู่หน้าสมัครสมาชิก
 - ที่หน้าสมัครสมาชิก กรอก Email และ Password
 - คลิก Next เพื่อเข้าสู่หน้ากรอกข้อมูลส่วนตัว
 - ที่หน้ากรอกข้อมูลส่วนตัว ต้องกรอกข้อมูลส่วนตัวโดยจะเปลี่ยนรูปภาพประจำตัวหรือไม่ก็ได้, กรอก Username, เลือกเพศ, ตั้งวันเกิด
 - คลิก Done เพื่อกลับมาหน้า Log in
 - หากสมัครสมาชิกแล้ว
 - กรอก Email และ Password ตามที่ได้สมัครสมาชิกไว้
 - คลิก Log in เพื่อเข้าสู่หน้าหลัก
4. ที่หน้าหลัก สามารถเลือกร้านอาหารได้ตามต้องการ
 - คลิกที่ไอคอนแว่นขยายเพื่อค้นหาร้านอาหารที่ต้องการ
 - คลิกที่ร้านอาหารเพื่อดูรายละเอียดของร้านก่อน
 - เมื่อได้ร้านที่ต้องการแล้วให้คลิกเข้าไปที่หน้ารายละเอียดของร้านอาหารนั้น
 - เลือกจะเป็นหัวหน้าโต๊ะ
 - คลิกไอคอน + เพื่อเข้าสู่หน้าสร้างโต๊ะ
 - กำหนดคุณสมบัติ (ช่วงอายุ, จำนวนคน, วันเวลา, เพศ) ตามต้องการ
 - คลิก Post เพื่อสร้างโต๊ะและเข้าสู่หน้าโต๊ะของตัวเอง
 - จะคลิก Post ได้ก็ต่อเมื่อไม่มีโต๊ะอยู่ก่อนแล้ว
 - เลือกจะเป็นผู้ค้นหาโต๊ะ
 - คลิก Matching เพื่อค้นหาโต๊ะที่มีคุณสมบัติตรงกันและเข้าสู่หน้าแจ้งเตือน
 - จะคลิก Matching ได้ก็ต่อเมื่อไม่มีโต๊ะอยู่ก่อนแล้วและไม่ได้กด Matching ที่ร้านใดไว้

5. หลังจากที่ได้สร้างโต๊ะหรือค้นหาโต๊ะแล้ว หากมีโต๊ะหรือผู้ค้นหาที่มีคุณสมบัติตรงกัน ก็จะมีแถบแจ้งเตือนขึ้นที่หน้าแจ้งเตือน โดยมีหลักการดังนี้

- หัวหน้าโต๊ะจะได้รับแจ้งเตือนว่าพบผู้ค้นหาโต๊ะที่มีคุณสมบัติตรงกับที่โต๊ะต้องการ
 - หากหัวหน้าโต๊ะปฏิเสธ ให้กดค้างที่แถบแจ้งเตือน และเลื่อนแถบแจ้งเตือนไปทางซ้ายเพื่อลบแถบแจ้งเตือนนั้นไป
 - หากหัวหน้าโต๊ะตอบรับ ให้กดค้างที่แถบแจ้งเตือน และเลื่อนแถบแจ้งเตือนไปทางขวาเพื่อทำการส่งคำเชิญเข้าโต๊ะไปให้ผู้ค้นหาโต๊ะคนนั้น
 - หากสุดท้ายแล้วไม่มีผู้ค้นหาโต๊ะคนใดมีคุณสมบัติตรงกับโต๊ะเลย หรือหัวหน้าโต๊ะรอนานเกินไป หัวหน้าโต๊ะต้องคลิกปุ่ม Finish เพื่อลบโต๊ะด้วยตนเอง
- ผู้ค้นหาโต๊ะจะได้รับการแจ้งเตือนก็ต่อเมื่อมีหัวหน้าโต๊ะส่งคำเชิญมาให้เท่านั้น เมื่อได้รับคำเชิญสามารถกดดูรายละเอียดของโต๊ะนั้นก่อนตัดสินใจตอบรับหรือปฏิเสธได้
 - หากผู้ค้นหาโต๊ะปฏิเสธ ให้กดค้างที่แถบแจ้งเตือน และเลื่อนแถบแจ้งเตือนไปทางซ้ายเพื่อลบแถบแจ้งเตือนนั้นไป
 - หากผู้ค้นหาโต๊ะตอบรับ ให้กดค้างที่แถบแจ้งเตือน และเลื่อนแถบแจ้งเตือนไปทางขวาเพื่อทำตอบรับคำเชิญนั้น และเข้าสู่หน้าโต๊ะ โดยจะเป็นการยกเลิกการค้นหาโต๊ะโดยอัตโนมัติ
 - หากสุดท้ายแล้วไม่มีโต๊ะใดส่งคำเชิญมาให้ผู้ค้นหาโต๊ะ หรือผู้ค้นหาโต๊ะรอนานเกินไป ผู้ค้นหาโต๊ะต้องคลิกไอคอนถังขยะในหน้ารายละเอียดของร้านนั้น ๆ เพื่อยกเลิกการค้นหาโต๊ะด้วยตนเอง

6. หลังจากมีโต๊ะของตัวเองแล้ว ที่หน้าโต๊ะ

- สมาชิกสามารถกดออกจากโต๊ะได้จนกว่าจำนวนคนในโต๊ะจะครบตามต้องการ
- เมื่อจำนวนสมาชิกครบตามจำนวนคนที่โต๊ะต้องการแล้วจะไม่สามารถกดออกจากโต๊ะได้
- เมื่อถึงวันทีนัดหมาย สมาชิกทุกคนก็ไปเจอกันที่ร้านตามเวลาที่นัดหมาย และรับประทานอาหารด้วยกันได้เลย
- หลังจากรับประทานอาหารเสร็จแล้ว หัวหน้าโต๊ะจะทำการกดปุ่ม Finish เพื่อลบโต๊ะและเก็บประวัติการไปรับประทานอาหารไว้ที่หน้าข้อมูลส่วนตัว

7. ที่หน้าข้อมูลส่วนตัวสามารถดูข้อมูลส่วนตัวของตนเองได้

- คลิก Tag เพื่อดูความชอบ
- คลิก History เพื่อดูประวัติการไปรับประทานอาหารเช้าครั้งสุดท้าย
 - คลิกที่ประวัติการไปรับประทานอาหารเช้าครั้งสุดท้ายเพื่อดูรายละเอียดของโต๊ะที่ไปรับประทานอาหารเช้าในครั้งนั้น
- คลิกไอคอนฟันเฟืองเพื่อเข้าสู่หน้าแก้ไขข้อมูลส่วนตัว

8. ที่หน้าแก้ไขข้อมูลส่วนตัว

- คลิก Edit เพื่อแก้ไขข้อมูลส่วนตัวของตนเอง
- คลิก Confirm เพื่อบันทึกการแก้ไข
- คลิก Cancel เพื่อยกเลิกการแก้ไข
- คลิก Log out เพื่อออกจากระบบ

บทที่ 5

การทดสอบระบบ

Test Report

TEST REPORT	
Project Name	Buffet2Gether
Description	https://github.com/Buffer2gether/buffet2gether
Version	1.0
Test Feature	ทุกระบบ
Test Requirement / Scenario	ทดสอบการทำงานทั้งแอปพลิเคชัน
Written By	ธนพล พัดพิน นิรชา บุญยเนตร ภิญดา ทองเต็ม สุรสีห์ อินทรวัตร
Written Date	2020-3-21
Test Strategy	Manual Test
Test Environment	OS: Windows 10 Test Tool: <ul style="list-style-type: none"> Python version 3.8.1 Robot Framework version 3.1.2 Robotframework-selenium2library version3.0.0 Robotframework-seleniumlibrary version 3.3.1 Selenium version 3.141.0

	Browser: ChromeDriver 80.0.3987.149
Tested By	ธนพล พัดพิน นิรชา บุญยเนตร ภิญญา ทองเต็ม สุรสิทธิ์ อินทรวัตร
Tested Date	2020-5-4

Test Requirements และ Test cases

Test Requirement: Register

1. ตรวจสอบว่าสามารถสมัครสมาชิกได้
2. ตรวจสอบว่าสามารถป้อนอีเมลในช่องEmail ได้
3. ตรวจสอบว่าสามารถป้อนรหัสผ่านในช่องPassword ได้
4. ตรวจสอบว่าสามารถป้อน Username ได้
5. ตรวจสอบว่าสามารถระบุ เพศได้
6. ตรวจสอบว่าสามารถระบุ วันเดือนปีเกิดได้
7. ตรวจสอบว่าสามารถอัปโหลดรูปภาพโปรไฟล์ได้
7. ตรวจสอบว่าหากกรอกemailไม่ถูกต้องจะขึ้นแจ้งเตือน
8. ตรวจสอบว่าหากกรอกรหัสผ่านไม่ครบ6ตัวจะขึ้นแจ้งเตือน
9. ตรวจสอบว่าหากไม่ป้อนUsername จะขึ้นแจ้งเตือน
10. ตรวจสอบว่าหากไม่ระบุเพศจะไม่สามารถสมัครได้
11. ตรวจสอบว่าหากไม่ระบุวันเดือนปีเกิดจะไม่สามารถสมัครได้

Test Req.	ตรวจสอบว่าสามารถ Register ได้				
Testcase ID	Test Description	Step	Input	Expect result	Test result
1	กรอก email ถูกต้อง และ password ไม่น้อยกว่า 6 ตัวอักษร	1. กรอก email 2. กรอก password 3. กด next	Email: test@gmail.com Password: 123456	ดำเนินการไปยังหน้ากรอกข้อมูลส่วนตัว	PASSED
2.	กรอก email ไม่ถูกต้อง แต่รหัสผ่านมากกว่าหรือเท่ากับ 6 ตัวอักษร	1. กรอก email 2. กรอก รหัส 3. กด next	Email: a Password: 123456	แสดงข้อความ "Invalid email"	PASSED
3	กรอก email ถูกต้อง แต่ password น้อยกว่า 6 ตัวอักษร	1. กรอก email 2. กรอก รหัส 3. กด next	Email: test@gmail.com Password: 1234	แสดงข้อความ "Enter a password 6+ chars long"	PASSED
4	กรอก email ไม่ถูกต้อง และ รหัสไม่ถูกต้อง	1. กรอก email	Email: a Password: 1234	แสดงข้อความ "Invalid Email " และ	PASSED

		2. กรอก รหัส 3. กด next		“Enter a password 6+ chars long”	
5	ไม่กรอก email	กด next		แสดงข้อความ “Invalid Email ” และ	PASSED
6	ไม่กรอก password	กด next		แสดงข้อความ “Enter a password 6+ chars long ”	PASSED
7	สมัครโดยใช้emailที่เคย สมัครไปแล้ว	1. กรอก email 2. กรอก password 3. กด next	Email: a@hotmail.com Password: 123456	แสดงข้อความ “This email is already use”	PASSED
8	กรอกUsername ระบุ เพศ และ วันเดือนปีเกิด และ ใส่รูปภาพประจำตัว ถูกต้องทั้งหมด	1. กรอก Username 2. ระบุเพศ 3. ระบุวัน เดือนปีเกิด กด register 4. อัปโหลด รูปภาพ	Username: Bright Gender: Male Date Of Birth: 11 05 2000	ดำเนินการไปหน้า ล็อกอิน	PASSED

		4. กด finish			
9	กรอก Username ผิด	1. กรอก Username	Username: B	แสดงข้อความ “Enter a username more than 4 character”	PASSED
10	ไม่กรอกUsername	กดfinish		แสดงข้อความ “Enter a username more than 4 character”	PASSED
11	ไม่ระบุเพศ	กดfinish		ไม่สามารถดำเนินการไปยังหน้า loginได้	PASSED
12	ไม่ระบุวันเดือนปีเกิด	กดfinish		ไม่สามารถดำเนินการไปยังหน้า loginได้	PASSED
13	ไม่อัปโหลดรูปภาพ	กดfinish		ระบบจะทำการใส่รูปโปรไฟล์พื้นฐานให้	PASSED
14	อัปโหลดรูปภาพ	1. อัปโหลดรูปภาพ 2. กด finish		ระบบจะใส่รูปที่อัปโหลดไปยังหน้าโปรไฟล์	PASSED

Test Requirement: Login

1. ตรวจสอบว่าสามารถสมัครสมาชิกได้
2. ตรวจสอบว่าสามารถป้อนอีเมลในช่องEmail ได้
3. ตรวจสอบว่าสามารถป้อนรหัสผ่านในช่องPassword ได้
4. ตรวจสอบว่าหากกรอกemailไม่ถูกต้องจะขึ้นแจ้งเตือน
5. ตรวจสอบว่าหากกรอกรหัสผ่านไม่ครบ6ตัวจะขึ้นแจ้งเตือน
6. ตรวจสอบว่าหากemail และ password ที่กรอกไปไม่มีในฐานข้อมูล

Test Req.	ตรวจสอบว่าสามารถ Login ได้				
Testcase ID	Test Description	Step	Input Data	Expect result	Test result
1	กรอก email และ password ถูกต้อง	1. กรอก email 2. กรอก password 3. กด Login	Email: test@gmail.com Password: 123456	ดำเนินการไปยังหน้ากรอกข้อมูลส่วนตัว	PASSED
2.	กรอก email ไม่ถูกต้อง แต่รหัสผ่านถูกต้อง	1. กรอก email 2. กรอกรหัส	Email: a Password: 123456	แสดงข้อความ "Invalid email"	PASSED

		3. กด Login			
3	กรอก email ถูกต้องแต่ รหัส ไม่ถูกต้อง	1. กรอก email 2. กรอก รหัส 3. กด Login	Email: test@gmail.com Pass: 1234	แสดงข้อความ “Enter a password 6+ chars long”	PASSED
4	กรอกemailไม่ถูกต้อง และ รหัสไม่ถูกต้อง	1 กรอก email 2 กรอก รหัส 3. กด Login	Email: a Password: 12345	แสดงข้อความ “Invalid Email ” และ “Enter a password 6+ chars long ”	PASSED
5	ไม่กรอก email	กด Login		แสดงข้อความ “Invalid Email ”	PASSED
6	ไม่กรอก password	กด Login		แสดงข้อความ“Enter a password 6+ chars long ”	PASSED
7	กรอก email และ password ที่ไม่มีในระบบ	กด Login	Email: notregister@gmail.com Password: 123456	แสดงข้อความ “Could not sign in ,Please check your	PASSED

				email or password”	
--	--	--	--	-----------------------	--

Test Requirement: Home

1. ตรวจสอบว่าสามารถเข้าหน้า Home ได้
2. ตรวจสอบว่าสามารถเลื่อนดูรูปภาพโปรโมชั่นได้
3. ตรวจสอบว่าสามารถคลิกที่รูปภาพโปรโมชั่น แล้วไปที่หน้าแสดงข้อมูลของร้านนั้น ๆ ได้
4. ตรวจสอบว่าสามารถเลื่อนดูร้านที่แนะนำได้
5. ตรวจสอบว่าสามารถคลิกที่ร้านแนะนำ แล้วไปที่หน้าแสดงข้อมูลของร้านนั้น ๆ ได้
6. ตรวจสอบว่าสามารถเลื่อนดูร้านต่าง ๆ ได้
7. ตรวจสอบว่าสามารถคลิกไอคอนแว่นขยายเพื่อค้นหา แล้วไปที่หน้าค้นหาได้
8. ตรวจสอบว่าสามารถเข้าหน้าค้นหาได้
9. ตรวจสอบว่าสามารถคลิกที่ร้านต่าง ๆ แล้วไปที่หน้าแสดงข้อมูลของร้านนั้น ๆ ได้
10. ตรวจสอบว่าสามารถค้นหาร้านต่าง ๆ ได้ ถ้าช่องค้นหาใส่ด้วยข้อมูลภาษาอังกฤษ
11. ตรวจสอบว่าสามารถค้นหาร้านต่าง ๆ ได้ ถ้าช่องค้นหาใส่ด้วยข้อมูลภาษาไทย
12. ตรวจสอบว่าสามารถค้นหาร้านต่าง ๆ ได้ ถ้าช่องค้นหาใส่ด้วยตัวอักษรตั้งแต่ 2 ตัวอักษรขึ้นไป
13. ตรวจสอบว่าไม่สามารถค้นหาร้านต่าง ๆ ได้ ถ้าช่องค้นหาใส่ด้วยตัวเลข
14. ตรวจสอบว่าไม่สามารถค้นหาร้านต่าง ๆ ได้ ถ้าช่องค้นหาใส่ด้วยอักขระพิเศษ
15. ตรวจสอบว่าสามารถคลิกที่รายการร้านอาหารที่ขึ้นในหน้าค้นหา แล้วไปที่หน้าแสดงข้อมูลของร้านนั้น ๆ ได้
16. ตรวจสอบว่าสามารถคลิกที่ไอคอนย้อนกลับในหน้าค้นหา แล้วกลับไปหน้า Home ได้

Test Req	ตรวจสอบว่าสามารถค้นหาร้านต่าง ๆ ได้				
Testcase ID	Test Description	Step	Input Data	Expect Result	Test Result
1	กรอกตัวอักษรภาษาอังกฤษ 2 ตัวอักษรขึ้นไป	1. คลิกไอคอนแว่นขยายเพื่อไปหน้าค้นหา 2. กรอกตัวอักษรภาษาอังกฤษ 2 ตัวอักษรขึ้นไปในช่องค้นหา	oi	แสดงรายการร้านอาหารที่มีชื่อขึ้นต้นด้วย “oi”, “Oi”, “oi”, และ “OI” หากไม่มีร้านอาหารที่ขึ้นต้นด้วย “oi” หรือ “Oi” หรือ “oi” หรือ “OI” จะไม่แสดงรายการใด	PASSED
2	กรอกตัวอักษรภาษาอังกฤษ 1 ตัวอักษร	1. คลิกไอคอนแว่นขยายเพื่อไปหน้าค้นหา 2. กรอกตัวอักษรภาษาอังกฤษ 1 ตัวอักษรในช่องค้นหา	a	ไม่แสดงรายการใด	PASSED
3	กรอกตัวอักษรภาษาไทย 2 ตัวอักษรขึ้นไป	1. คลิกไอคอนแว่นขยายเพื่อไปหน้าค้นหา 2. กรอกตัวอักษรภาษาไทย 2 ตัวอักษรขึ้นไปในช่องค้นหา	บุ	แสดงรายการร้านอาหารที่มีชื่อขึ้นต้นด้วย “บุ” หากไม่มีร้านอาหารที่ขึ้นต้นด้วย “บุ” จะไม่แสดงรายการใด	PASSED

4	กรอกตัวอักษร ภาษาไทย 1 ตัวอักษร	1. คลิกไอคอนแว่น ขยายเพื่อไปหน้าค้นหา 2. กรอกตัวอักษร ภาษาไทย 1 ตัวอักษร ในช่องค้นหา	ร	ไม่แสดงรายการใด	PASSED
5	กรอกตัวเลข 1 ตัว	1. คลิกไอคอนแว่น ขยายเพื่อไปหน้าค้นหา 2. กรอกตัวเลข 1 ตัว ในช่องค้นหา	1	ไม่แสดงรายการใด	PASSED
6	กรอกตัวเลข 2 ตัว	1. คลิกไอคอนแว่น ขยายเพื่อไปหน้าค้นหา 2. กรอกตัวเลข 2 ตัว ในช่องค้นหา	12	ไม่แสดงรายการใด	PASSED
7	กรอกอักขระพิเศษ 1 ตัว	1. คลิกไอคอนแว่น ขยายเพื่อไปหน้าค้นหา 2. กรอกอักขระพิเศษ 1 ตัวในช่องค้นหา	*	ไม่แสดงรายการใด	PASSED
8	กรอกอักขระพิเศษ 2 ตัว	1. คลิกไอคอนแว่น ขยายเพื่อไปหน้าค้นหา 2. กรอกอักขระพิเศษ 2 ตัวในช่องค้นหา	*#	ไม่แสดงรายการใด	PASSED
9	ไม่กรอก	1. คลิกไอคอนแว่น ขยายเพื่อไปหน้าค้นหา		ไม่แสดงรายการใด	PASSED

		2. คลิกที่ช่องค้นหาแต่ ไม่กรอกอะไร			
--	--	---------------------------------------	--	--	--

Test Requirement: Restaurant Info

1. ตรวจสอบว่าสามารถเข้าหน้าข้อมูลของร้านได้
2. ตรวจสอบว่าสามารถคลิกดูโปรโมชั่นของร้านได้
3. ตรวจสอบว่าสามารถคลิก Matching ได้ หากผู้ใช้งานยังไม่มีโต๊ะ
4. ตรวจสอบว่าสามารถคลิก Matching ได้ หากผู้ใช้งานยังไม่ได้คลิก Matching ร้านอื่นไว้
5. ตรวจสอบว่าสามารถคลิก Matching แล้วบันทึกข้อมูลผู้ค้นหาโต๊ะได้
6. ตรวจสอบว่าสามารถคลิก Matching แสดงหน้า Matching ก่อนจะไปหน้าจอแจ้งเตือนได้
7. ตรวจสอบว่าสามารถคลิกไอคอนบวกเพื่อสร้างโต๊ะ แล้วไปหน้าจอสร้างโต๊ะได้
8. ตรวจสอบว่าสามารถคลิกไอคอนย้อนกลับ

Test Req	ตรวจสอบว่าสามารถเข้าหน้าข้อมูลของร้านได้				
Testcase ID	Test Description	Step	Input Data	Expect Result	Test Result
1	เข้าหน้าร้านอาหาร ผ่านหน้าแนะนำ	1. กดรูปร้านอาหาร แนะนำที่หน้าจอ หลัก 2. แสดงผลข้อมูล ร้าน		แสดงผลร้านอาหารที่เลือก	PASSED

2	เข้าหน้าร้านอาหาร ผ่านการพิมพ์ค้นหา ชื่อร้าน	1. พิมพ์ชื่อร้านใน ช่องค้นหา 2. กดเลือก ร้านอาหาร	sukishi	แสดงผลร้านอาหารที่เลือก	PASSED
3	กดอ่านโปรโมชันของ ร้านได้	1. เข้าสู่หน้าข้อมูล ร้าน 2. กดที่กล่อง ข้อความด้านล่าง ตัวหนังสือ “โปรโมชันน้องบุฟ”		แสดงผลโปรโมชัน	PASSED
4	กดmatchingเพื่อทำ การหากลุ่ม	1. เข้าสู่หน้าข้อมูล ร้าน 2. กดmatching		แสดงผลหน้าจอmatching จากนั้น กลับไปแสดงผล หน้าจอnotification	PASSED
5	กดไอคอนบวกเพื่อ เข้าสู่หน้าสร้างกลุ่ม	1. เข้าสู่หน้าข้อมูล ร้าน 2. กดไอคอนบวก		แสดงผลหน้าจอสำหรับการ สร้างกลุ่ม	PASSED
6	กดลูกศรย้อนกลับ	1. กดลูกศร ย้อนกลับ		ย้อนกลับไปยังหน้าหลัก	PASSED
7	กดmatchingจากนั้น กลับมายังหน้าข้อมูล ร้านที่เคยกด Matchingไว้	1. เข้าสู่หน้าข้อมูล ร้าน 2. กดmatching 3. เข้าสู่หน้าข้อมูล ร้าน		แสดงผลข้อความ“ระบบ กำลังMatchกลุ่มในร้านนี้ที่ เหมาะกับคุณ” และ ปุ่ม “ยกเลิกการ Match”	PASSED

8	กดmatchingจากนั้น เข้าสู่หน้า Restaurant Info ของร้านอื่น	1. เข้าสู่หน้า Restaurant Info ของร้านที่ 1 2. กด Matching 3. เข้าสู่หน้า Restaurant Info ของร้านที่2		แสดงผลหน้าRestaurant Info ของร้านที่ 2 โดย สามารถเลือกที่จะmatching หรือ เข้าสู่หน้า create Table ก็ได้	PASSED
---	--	---	--	---	--------

Test Requirement: Create Table

1. ตรวจสอบว่าสามารถเข้าหน้าสร้างโต๊ะได้
2. ตรวจสอบว่าสามารถคลิกที่คุณสมบัติอายุ เพื่อเลือกช่วงอายุที่ต้องการได้
3. ตรวจสอบว่าสามารถคลิกที่คุณสมบัติจำนวนคน เพื่อเลือกจำนวนคนที่ต้องการได้
4. ตรวจสอบว่าสามารถคลิกที่คุณสมบัติวันเวลา เพื่อเลือกวันที่และเวลาที่จะนัดหมายได้
5. ตรวจสอบว่าสามารถคลิกที่คุณสมบัติเพศ เพื่อเลือกเพศที่ต้องการได้
6. ตรวจสอบว่าสามารถเลือกช่วงอายุได้ เมื่อคลิกที่คุณสมบัติอายุ
7. ตรวจสอบว่าสามารถเลือกจำนวนคนที่ต้องการได้ เมื่อคลิกที่คุณสมบัติจำนวนคน
8. ตรวจสอบว่าสามารถเลือกวันที่และเวลาที่จะนัดหมายได้ เมื่อคลิกที่คุณสมบัติวันเวลา
9. ตรวจสอบว่าสามารถเลือกเพศได้ เมื่อคลิกที่คุณสมบัติเพศ
10. ตรวจสอบว่าสามารถคลิกแก้ไขความชอบ แล้วไปที่หน้าแก้ไขความชอบได้
11. ตรวจสอบว่าสามารถเข้าหาแก้ไขความชอบได้
12. ตรวจสอบว่าสามารถเลือกความชอบต่าง ๆ ได้
13. ตรวจสอบว่าสามารถคลิก Confirm แล้วกลับที่หน้าสร้างโต๊ะโดยบันทึกความชอบได้
14. ตรวจสอบว่าสามารถคลิก Cancel แล้วกลับไปหน้าสร้างโต๊ะโดยไม่บันทึกความชอบได้
15. ตรวจสอบว่าสามารถคลิก POST แล้วบันทึกข้อมูลโต๊ะได้
16. ตรวจสอบว่าสามารถคลิก POST ได้ หากผู้ใช้งานยังไม่มีโต๊ะ
17. ตรวจสอบว่าสามารถคลิก POST ได้หากผู้ใช้งานยังไม่คลิก Matching ที่ร้านใดไว้

18. ตรวจสอบว่าสามารถคลิก POST แล้วแสดงหน้า Matching ก่อนจะไปหน้าจอแจ้งเตือนได้

19. ตรวจสอบว่าไม่สามารถคลิก POST ได้ หากผู้ใช้งานมีโต๊ะแล้ว

20. ตรวจสอบว่าไม่สามารถคลิก POST ได้ หากผู้ใช้งานคลิก Matching ร้านใดก็ตามไว้

Test Req	ตรวจสอบว่าสามารถเข้าหน้าสร้างโต๊ะได้				
Testcase ID	Test Description	Step	Input Data	Expect Result	Test Result
1	กำหนดช่วงอายุสำหรับคนเข้ากลุ่ม	1. กดแถบช่วงอายุ 2. กำหนดช่วงอายุในการรับเข้ากลุ่ม 3. กดPost		สร้างโต๊ะที่มีข้อกำหนดช่วงอายุตามที่กำหนด	PASSED
2	กำหนดจำนวนคนเข้ากลุ่ม	1. กดแถบจำนวนสมาชิก 2. กำหนดจำนวนสมาชิก 3. กดPost		สร้างโต๊ะที่มีจำนวนสมาชิกตามที่กำหนด	PASSED
3.	กำหนดเวลาสำหรับนัดกินบุฟเฟต์	1. กดแถบเวลา 2. กำหนดเวลาสำหรับกินบุฟเฟต์ 3. กดPost		สร้างโต๊ะที่มีข้อมูลเวลานัดพบตามที่กำหนด	PASSED

4.	กำหนดเพศสำหรับ คนเข้ากลุ่ม	1. กดแถบเพศ 2. กำหนดเพศ สำหรับคนเข้า กลุ่ม 3. กดPost		สร้างโตะที่มีข้อกำหนดเพศตามที่ กำหนด	PASSED
5.	กำหนดความชอบ สำหรับคนเข้ากลุ่ม	1. กดแถบ Interesting 2. เลือก ความชอบต่าง ๆ 3. กดPost		สร้างโตะที่มีค่าความชอบตามที่ กำหนด	PASSED
6.	กดสร้างกลุ่มแต่มี กลุ่มอยู่แล้ว	1. กำหนดข้อมูล ต่าง ๆ 2. กดPost		แสดงผลข้อความ”ขอภัยค่ะ ไม่ สามารถสร้างกลุ่มใหม่ได้ เนื่องจากคุณมีกลุ่มบุฟเฟต์อยู่ แล้ว”	PASSED
7.	กดconfirm โดยที่ ยังไม่กำหนดเพศใน การเข้ากลุ่ม	1. กดPost		แสดงผลข้อความ”Please select gender”	PASSED
8	กดสร้างโตะหลังจาก ที่กดmatchingร้าน เดียวกัน	1. กดสร้างโตะ 2. กำหนด รายละเอียดโตะ 3. กดpost		แสดงผลข้อความ”ระบบกำลัง Matchกลุ่มในร้านนี้ที่เหมาะสมกับ คุณ” และ ปุ่ม “ยกเลิกการMatch”	PASSED

		4. กวดูหน้า Restaurant Info			
--	--	-----------------------------------	--	--	--

Test Requirement: Notification

1. ตรวจสอบว่าสามารถเข้าหน้าการแจ้งเตือนได้
2. ตรวจสอบว่าสามารถเลื่อนแถบการแจ้งเตือนไปทางขวาเพื่อเชิญผู้ค้นหาโต๊ะคนนั้น ๆ เข้าร่วมโต๊ะได้ (ในฐานะเป็นหัวหน้าโต๊ะ)
3. ตรวจสอบว่าสามารถเลื่อนแถบการแจ้งเตือนไปทางขวาเพื่อตอบรับคำเชิญเข้าร่วมโต๊ะนั้น ๆ และเข้าร่วมโต๊ะได้ (ในฐานะเป็นผู้ค้นหาโต๊ะ)
4. ตรวจสอบว่าสามารถเลื่อนแถบการแจ้งเตือนไปทางซ้ายเพื่อปฏิเสธผู้ค้นหาโต๊ะคนนั้น ๆ ได้ (ในฐานะเป็นหัวหน้าโต๊ะ)
5. ตรวจสอบว่าสามารถเลื่อนแถบการแจ้งเตือนไปทางซ้ายเพื่อปฏิเสธคำเชิญเข้าร่วมโต๊ะนั้น ๆ ได้ (ในฐานะเป็นหัวหน้าโต๊ะ)
6. ตรวจสอบว่าสามารถลบแถบแจ้งเตือนได้ เมื่อเลื่อนแถบแจ้งเตือน
7. ตรวจสอบว่าไม่สามารถเชิญคนเข้าร่วมโต๊ะได้ถ้าคน ๆ นั้นมีโต๊ะแล้ว
8. ตรวจสอบว่าไม่สามารถเชิญคนเข้าร่วมโต๊ะได้ถ้าโต๊ะของผู้ใช้งานเองเต็มแล้ว
9. ตรวจสอบว่าไม่สามารถตอบรับคำเชิญเข้าร่วมโต๊ะและเข้าร่วมโต๊ะได้ถ้าโต๊ะนั้นเต็มแล้ว
10. ตรวจสอบว่าสามารถคลิกที่แถบแจ้งเตือนเชิญเข้าร่วมโต๊ะ แล้วแสดงข้อมูลของโต๊ะนั้น ๆ ได้
11. ตรวจสอบว่าสามารถไปที่หน้าโต๊ะได้ หากตอบรับคำเชิญเข้าร่วมโต๊ะและเข้าร่วมโต๊ะ

Test Req	ตรวจสอบว่าสามารถเข้าหน้าการแจ้งเตือนได้				
Testcase ID	Test Description	Step	Input Data	Expect Result	Test Result
1	เชิญผู้ค้นหาโต๊ะคนนั้น ๆ เข้าร่วมโต๊ะ(ในฐานะเป็นหัวหน้าโต๊ะ)	1. สร้างโต๊ะ 2. เลือกผู้ค้นหาที่ต้องการให้เข้าโต๊ะ 3. เลื่อนแถบการแจ้งเตือนไปทางขวา		มีการแจ้งเตือนเชิญเข้าร่วมโต๊ะส่งไปที่ผู้ค้นหาคนนั้น	PASSED
2	ตอบรับคำเชิญเข้าร่วมโต๊ะนั้น ๆ และเข้าร่วมโต๊ะ (ในฐานะเป็นผู้ค้นหาโต๊ะ)	1. กดปุ่ม match 2. เลือกโต๊ะที่ต้องการเข้า 3. เลื่อนแถบการแจ้งเตือนไปทางขวา		แสดงผลหน้าข้อมูลของโต๊ะโดยมีผู้ใช้เป็นสมาชิกร่วมอยู่ด้วย	PASSED
3	ปฏิเสธผู้ค้นหาโต๊ะคนนั้น ๆ (ในฐานะเป็นหัวหน้าโต๊ะ)	1. สร้างโต๊ะ 2. เลือกผู้ค้นหาที่ต้องการปฏิเสธ		แถบแจ้งเตือนหายไป	PASSED

		3. เลื่อนแถบการ แจ้งเตือนไป ทางซ้าย			
4	ปฏิเสธคำเชิญเข้าร่วมโต๊ะ นั้น ๆ (ในฐานะเป็นหัวหน้า โต๊ะ)	1. กดปุ่ม match 2. เลือกโต๊ะที่ ต้องการปฏิเสธ 3. เลื่อนแถบการ แจ้งเตือนไป ทางซ้าย		แถบแจ้งเตือนหายไป	PASSED
5	เชิญคนเข้าร่วมโต๊ะ เมื่อคน ๆ นั้นมีโต๊ะแล้ว	1. สร้างโต๊ะ 2. เลือกผู้ค้นหา ที่มีโต๊ะแล้ว 3. เลื่อนแถบการ แจ้งเตือนไป ทางขวา		แสดงผลเตือนว่า 'ขอ อภัย...ไม่สามารถเชิญได้ เนื่องจาก...มีกลุ่มแล้ว' +	PASSED
6	เชิญคนเข้าร่วมโต๊ะ เมื่อโต๊ะของผู้ใช้งานเองเต็ม แล้ว	1. สร้างโต๊ะ 2. รับคนเข้าร่วม โต๊ะจนเต็ม 3. เลือกผู้ค้นหา ที่ต้องการให้เข้า ร่วมโต๊ะเพิ่ม		แสดงผลเตือนว่า 'ขอ อภัย...กลุ่มของคุณเต็ม แล้ว'	PASSED

		4. เลื่อนแถบการ แจ้งเตือนไป ทางขวา			
7	ตอบรับคำเชิญเข้าร่วมโต๊ะ เมื่อโต๊ะนั้นเต็มแล้ว	1. กดปุ่ม match 2. เลือกโต๊ะที่ เต็มแล้ว 3. เลื่อนแถบการ แจ้งเตือนไป ทางขวา		แสดงผลเตือนว่า 'ขอ อภัย...ไม่สามารถเข้ากลุ่ม ได้เนื่องจากกลุ่มนี้เต็ม แล้ว'	PASSED
8	ตอบรับคำเชิญเข้าร่วมโต๊ะ อื่นเพิ่ม เมื่อผู้ใช้งานมีโต๊ะ แล้ว	1. กดปุ่ม match 2. เข้าร่วมโต๊ะ 3. เลือกโต๊ะที่ ต้องการเข้าเพิ่ม 4. เลื่อนแถบการ แจ้งเตือนไป ทางขวา		แสดงผลเตือนว่า 'ขอ อภัย...ไม่สามารถเข้ากลุ่ม ได้เนื่องจากคุณมีกลุ่ม แล้ว'	PASSED
9	คลิกที่แถบแจ้งเตือนเชิญเข้า ร่วมโต๊ะ ดูข้อมูล รายละเอียดของโต๊ะนั้นๆ	1. สร้างโต๊ะ หรือกดปุ่ม Match 2. เลือกโต๊ะที่ ต้องการ		แสดงผลข้อมูลของโต๊ะ นั้น ๆ	PASSED

		3. คลิกที่แถบ แจ้งเตือนเชิญ เข้าร่วมโต๊ะ			
10	ตอบรับคำเชิญเข้าร่วมโต๊ะ และเข้าร่วมโต๊ะ ดูหน้าโต๊ะที่ ผู้ใช้งานเป็นสมาชิกอยู่	1. กดปุ่ม match 2. เลือกโต๊ะที่ ต้องการเข้า 3. เลื่อนแถบการ แจ้งเตือนไป ทางขวา		แสดงผลหน้าข้อมูลของ โต๊ะโดยมีผู้ใช้เป็นสมาชิก ร่วมอยู่ด้วย	PASSED
11	ลบข้อมูลของแถบแจ้งเตือน	1. กดปุ่ม match หรือ สร้างโต๊ะ 2. เลือกแถบแจ้ง เตือนที่ต้องการ ลบ 3. เลื่อนแถบการ แจ้งเตือนไป ทางซ้าย		ข้อมูลของแถบแจ้งเตือน ถูกลบไปจากฐานข้อมูล	PASSED

Test Requirement: Table

1. ตรวจสอบว่าสามารถลบข้อมูลของโต๊ะได้ ถ้ากดปุ่มกินเสร็จ(ในฐานะเป็นหัวหน้าโต๊ะ)
2. ตรวจสอบว่าสามารถกดออกจากโต๊ะได้ ถ้าโต๊ะยังไม่เต็ม(ในฐานะเป็นผู้ค้นหาโต๊ะ)
3. ตรวจสอบว่าไม่สามารถกดออกจากโต๊ะได้ ถ้าโต๊ะเต็มแล้ว(ในฐานะเป็นผู้ค้นหาโต๊ะ)
4. ตรวจสอบว่าสามารถดูคุณสมบัติต่างๆของโต๊ะ และสมาชิกในโต๊ะได้

Test Req	ตรวจสอบว่าสามารถเข้าหน้าโต๊ะได้				
Testcase ID	Test Description	Step	Input Data	Expect Result	Test Result
1	ลบข้อมูลของโต๊ะ(ในฐานะเป็นหัวหน้าโต๊ะ)	1. สร้างโต๊ะ 2. รับคนเข้าร่วมโต๊ะจนเต็ม 3. กดปุ่มกินเสร็จ		โต๊ะถูกลบไปและบันทึกข้อมูลโต๊ะที่ไปกินล่าสุดลงในhistory	PASSED
2	กดออกจากโต๊ะ เมื่อโต๊ะยังไม่เต็ม(ในฐานะเป็นผู้ค้นหาโต๊ะ)	1. กดปุ่ม match 2. เข้าร่วมโต๊ะ 3. กดปุ่มออกจากโต๊ะ ในขณะที่โต๊ะยังไม่เต็ม		ข้อมูลของผู้ใช้งานเองในโต๊ะนั้นถูกลบไป สมาชิกในโต๊ะนั้นลดลง	PASSED
3	กดออกจากโต๊ะ เมื่อโต๊ะเต็มแล้ว(ในฐานะเป็นผู้ค้นหาโต๊ะ)	1. กดปุ่ม match 2. เข้าร่วมโต๊ะ 3. กดปุ่มออกจากกลุ่ม ในขณะที่โต๊ะเต็มแล้ว		ไม่แสดงปุ่มออกจากปุ่มให้กด	PASSED
4	ดูคุณสมบัติต่างๆของโต๊ะ และสมาชิกในโต๊ะ	1. สร้างโต๊ะ หรือ กดปุ่มMatch 2. เข้าร่วมโต๊ะ		แสดงผลหน้าข้อมูลของโต๊ะโดยมีผู้ใช้เป็นสมาชิกร่วมอยู่ด้วย	PASSED

		3. คลิกที่แถบหน้า โต๊ะ			
--	--	---------------------------	--	--	--

Test Requirement: Profile

1. ตรวจสอบว่าสามารถเข้าหน้าข้อมูลส่วนตัวได้
2. ตรวจสอบว่าสามารถคลิกไอคอนฟันเฟืองเพื่อแก้ไขข้อมูลส่วนตัว แล้วไปที่หน้าแก้ไขข้อมูลส่วนตัวได้
3. ตรวจสอบว่าสามารถคลิกที่แถบ Tag แล้วแสดงความชอบได้
4. ตรวจสอบว่าสามารถคลิกที่แถบ History แล้วแสดงประวัติการไปรับประทานอาหารของผู้ใช้งานเองได้
5. ตรวจสอบว่าสามารถเข้าหน้าแก้ไขข้อมูลส่วนตัวได้
6. ตรวจสอบว่าคลิกปุ่ม Logout แล้วออกจากระบบได้
7. ตรวจสอบว่าสามารถ เปลี่ยนรูปภาพประจำตัวได้

Test Req	ตรวจสอบว่าสามารถเข้าหน้าข้อมูลส่วนตัวได้				
Testcase ID	Test Description	Step	Input Data	Expect Result	Test Result
1	ดูข้อมูลส่วนตัวในหน้า Profile	1. คลิกแถบหน้า Profile		แสดงผลข้อมูลส่วนตัว รูปภาพ Username อายุ เพศ ความชอบ และประวัติการไป กินที่ร้านอาหารครั้งล่าสุด	PASSED
2	แก้ไขรูป profile ในหน้า profile	1. คลิกไอคอนฟันเฟือง	รูปแมวในเครื่อง	แสดงผลรูปภาพ profile เป็นรูปแมว	PASSED

		2. คลิกแก้ไขรูป profile 3. เลือกรูป profile ที่ต้องการแก้ไข 4. คลิกปุ่มบันทึก			
3	แก้ไข username ในหน้า profile	1. คลิกไอคอนฟันเฟือง 2. คลิกแก้ไข username 3. พิมพ์ username ที่ต้องการแก้ไข 4. คลิกปุ่มบันทึก	tananet	แสดง Username tananet	PASSED
4	แก้ไขอายุ ในหน้า profile	1. คลิกไอคอนฟันเฟือง 2. คลิกแก้ไขอายุ 3. พิมพ์เลขที่ต้องการแก้ไข	20	แสดงอายุ 20 ปี	PASSED

		4. คลิกปุ่ม บันทึก			
5	แก้ไขเพศ ในหน้า profile	1. คลิกไอคอน ฟันเฟือง 2. คลิกแก้ไข เพศ 3. เลือกเพศที่ ต้องการแก้ไข หรือเลือกไม่ ระบุ 4. คลิกปุ่ม บันทึก	male	แสดงผล ไอคอนเป็น male	PASSED
6	แก้ไขความชอบใน หน้า profile	1. คลิกไอคอน ฟันเฟือง 2. คลิกแก้ไข interesting 3. เลือก ความชอบที่ ต้องการ 4. คลิกปุ่ม บันทึก	Pet Sport Technology	แสดงผลไอคอนต่างๆตาม ความชอบที่เลือก	PASSED
7	ดูความชอบใน หน้า profile	1. คลิกที่แถบ Tag		แสดงความชอบ	PASSED

8	ดูประวัติการไป รับประทานอาหารของ ผู้ใช้งานเองใน หน้า profile	1. คลิกที่แถบ History		แสดงประวัติการไปรับประทานอาหาร ของผู้ใช้งานเองครั้ง ล่าสุด	PASSED
9	แก้ไขความชอบใน หน้า profile โดย ไม่ใส่อะไรเลย	1. คลิกไอคอน ฟันเฟือง 2. คลิกแก้ไข interesting 3. แก้ไข ความชอบ 4. คลิกปุ่ม บันทึก		ไม่แสดงผลไอคอนความชอบใด ๆ	PASSED
10	กดดูรายละเอียด ของโต๊ะล่าสุดที่ไป กินมา	1. คลิกที่แถบ History 2. คลิกแถบ แสดงผลร้านที่ ไปกินมาล่าสุด		แสดงรายละเอียดของโต๊ะ คุณ สมบัติ ของร้านที่เพิ่งไปกินมา	PASSED

Test Requirement: Matching

1. ตรวจสอบว่าสามารถ Match ผู้ค้นหาที่มีช่วงอายุตรงกับที่ต้องการมาที่โต๊ะของผู้ใช้งานได้
2. ตรวจสอบว่าสามารถ Match ผู้ค้นหาที่มีเพศตรงกับที่ต้องการมาที่โต๊ะของผู้ใช้งานได้
3. ตรวจสอบว่าสามารถ Match ผู้ค้นหาที่มีความชอบตรงกับที่ต้องการมาที่โต๊ะของผู้ใช้งานได้

4. ตรวจสอบว่าสามารถ Match ผู้ค้นหาที่มีเพศ อายุ และความชอบ ทั้ง 3 ตรงกับที่ต้องการมาที่โต๊ะของผู้ใช้งานได้
5. ตรวจสอบว่าไม่สามารถ Match ผู้ค้นหาที่มีเพศ อายุ และความชอบ ทั้ง 3 ไม่ตรงกับที่ต้องการมาที่โต๊ะของผู้ใช้งานได้
6. ตรวจสอบว่าไม่สามารถ Match ผู้ค้นหาที่มีเพศ อายุ และความชอบ อย่างใดอย่างหนึ่งไม่ตรงกับที่ต้องการมาที่โต๊ะของผู้ใช้งานได้
7. ตรวจสอบว่าไม่สามารถ Match ผู้ค้นหาที่มีเพศ อายุ และความชอบ เพียงอย่างใดอย่างหนึ่งตรงกับที่ต้องการมาที่โต๊ะของผู้ใช้งานได้

Test Req.	ตรวจสอบว่าสามารถ Match ได้				
Testcase ID	Test Description	Step	Input Data	Expect result	Test result
1	ผู้ค้นหาที่มีช่วงอายุตรงกับที่ต้องการ	1. ผู้ค้นหา กดปุ่ม Match 2. ผู้ใช้งาน สร้างโต๊ะ โดยกำหนดอายุให้ตรงกับผู้ค้นหา 3. ผู้ใช้งาน กดปุ่ม Post	ข้อมูลผู้ค้นหา อายุ: 18 เพศ: หญิง ความชอบ: pet ข้อมูลที่โต๊ะ อายุ: 14-20 เพศ: หญิง	แสดงการแจ้งเตือนของผู้ค้นหาปรากฏที่หัวหน้าโต๊ะ	PASSED

			ความชอบ: pet sport		
2.	ผู้ค้นหามีเพศตรงกับที่ ต้องการ	1. ผู้ค้นหาทดปุ่ม Match 2. ผู้ใช้งานสร้างโตะโดย กำหนดเพศให้ตรงกับผู้ ค้นหา 3. ผู้ใช้งานกดปุ่ม Post	ข้อมูลผู้ ค้นหา อายุ:25 เพศ: หญิง ความชอบ: pet ข้อมูลที่โตะ อายุ: 19-25 เพศ: หญิง ความชอบ: pet sport	แสดงการแจ้ง เตือนของผู้ ค้นหาปรากฏที่ หัวหน้าโตะ	PASSED
3	ผู้ค้นหาที่มีความชอบ ตรงกับที่ต้องการ	1. ผู้ค้นหาทดปุ่ม Match 2. ผู้ใช้งานสร้างโตะโดย กำหนดความชอบให้ตรง กับผู้ค้นหา 3. ผู้ใช้งานกดปุ่ม Post	ข้อมูลผู้ ค้นหา อายุ:30 เพศ: ชาย ความชอบ: political ข้อมูลที่โตะ	แสดงการแจ้ง เตือนของผู้ ค้นหาปรากฏที่ หัวหน้าโตะ	PASSED

			อายุ: 19-25 เพศ: ชาย ความชอบ: political		
4	ผู้ค้นหาที่มีเพศ อายุ และ ความชอบ ทั้ง 3 ตรง กับที่ต้องการ	1. ผู้ค้นหาทดปุ่ม Match 2. ผู้ใช้งานสร้างโตะโดย กำหนดความชอบ อายุ เพศให้ตรงกับผู้ค้นหา 3. ผู้ใช้งานกดปุ่ม Post	ข้อมูลผู้ ค้นหา อายุ:22 เพศ: ชาย ความชอบ: sport ข้อมูลที่โตะ อายุ: 22-25 เพศ: ชาย ความชอบ: sport	แสดงการแจ้ง เตือนของผู้ ค้นหาปรากฏที่ หัวหน้าโตะ	PASSED
5	ผู้ค้นหาที่มีเพศ อายุ และความชอบ ทั้ง 3 ไม่ ตรงกับที่ต้องการ email	1. ผู้ค้นหาทดปุ่ม Match 2. ผู้ใช้งานสร้างโตะโดย กำหนดความชอบ อายุ เพศให้แตกต่างกับผู้ค้นหา 3. ผู้ใช้งานกดปุ่ม Post	ข้อมูลผู้ ค้นหา อายุ:22 เพศ: ชาย ความชอบ: sport	ไม่มีแจ้งเตือน ของผู้ค้นหาคน นี้ปรากฏ	PASSED

			<p>ข้อมูลที่โต๊ะ</p> <p>อายุ: 23-24</p> <p>เพศ: หญิง</p> <p>ความชอบ: fashion</p>		
6	ผู้ค้นหาที่มีเพศ อายุ และความชอบ อย่างไม่อย่างหนึ่งไม่ตรงกับที่ต้องการ	<p>1. ผู้ค้นหาทูปุ่ม Match</p> <p>2. ผู้ใช้งานสร้างโต๊ะโดยกำหนดความชอบ อายุ ให้ตรงกับผู้ค้นหา แต่กำหนดเพศให้แตกต่างกับผู้ค้นหา</p> <p>3. ผู้ใช้งานกดปุ่ม Post</p>	<p>ข้อมูลผู้ค้นหา</p> <p>อายุ:34</p> <p>เพศ: ชาย</p> <p>ความชอบ: sport book</p> <p>ข้อมูลที่โต๊ะ</p> <p>อายุ: 22-40</p> <p>เพศ: หญิง</p> <p>ความชอบ: book</p>	ไม่มีแจ้งเตือนของผู้ค้นหาคนนี้ปรากฏ	PASSED
7	ผู้ค้นหาที่มีเพศ อายุ และความชอบ เพียงอย่างเดียวอย่างหนึ่งตรงกับที่ต้องการ	<p>1. ผู้ค้นหาทูปุ่ม Match</p> <p>2. ผู้ใช้งานสร้างโต๊ะโดยกำหนด เพศให้ตรงกับผู้ค้นหา แต่กำหนดอายุ</p>	<p>ข้อมูลผู้ค้นหา</p> <p>อายุ:27</p> <p>เพศ: หญิง</p>	ไม่มีแจ้งเตือนของผู้ค้นหาคนนี้ปรากฏ	PASSED

		ความชอบให้แตกต่างกับผู้ ค้นหา 3. ผู้ใช้งานกดปุ่ม Post	ความชอบ: book ข้อมูลที่ได้ะ อายุ: 22-26 เพศ: หญิง ความชอบ: pet		
--	--	---	--	--	--

ผลการทดสอบ

จากการพัฒนาและทดสอบแอปพลิเคชัน BUFFET2GETHER ได้ผลการทดสอบคือสามารถใช้งานในขั้นพื้นฐานได้อย่างครบถ้วนสมบูรณ์ และตอบสนองตามความต้องการของผู้ใช้งาน(Requirements) เป็นส่วนใหญ่ ถึงแม้จะยังมีบางความต้องการที่ยังไม่ได้พัฒนาขึ้นในปัจจุบัน แต่ในอนาคตทางคณะผู้จัดทำหวังว่าจะสามารถนำมาพัฒนาให้แอปพลิเคชัน BUFFET2GETHER สมบูรณ์ยิ่งขึ้น เป็นที่รู้จัก และทำให้ผู้ใช้งานเพลิดเพลินกับการรับประทานอาหารมากขึ้น

