



กรณีศึกษา การพัฒนาฟรอนท์เอนด์ของโมบายแอปพลิเคชันด้วย Ionic Framework
Case Study of The Front-End Development for Mobile Application with Ionic Framework

ภาคย์ ผดุงพัฒนดิษฐ์

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีไทย-ญี่ปุ่น

พ.ศ. 2561

กรณีศึกษา การพัฒนาฟรอนต์เอนด์ของโมบายแอปพลิเคชันด้วย Ionic Framework
Case Study of The Front-End Development for Mobile Application with Ionic Framework

ภาคย์ ผดุงพัฒน์ดิษฐ์

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย - ญี่ปุ่น
ปีการศึกษา 2561

คณะกรรมการสอบ

..... ประธานกรรมการสอบ

(อาจารย์ อมรพันธ์ ชมกลีน)

..... กรรมการสอบ

(อาจารย์ โอฬาร รื่นชื่น)

..... อาจารย์ที่ปรึกษา

(อาจารย์ สติลา ชีวกิตการ)

..... ประธานสหกิจศึกษาสาขาวิชา

(อาจารย์ สติลา ชีวกิตการ)

ลิขสิทธิ์ของสถาบันเทคโนโลยีไทย - ญี่ปุ่น

ชื่อโครงการ

กรณีศึกษา การพัฒนาฟรอนต์เอนด์ของโมบายแอปพลิเคชันด้วย

Ionic Framework

Case Study of The Front-End Development for Mobile

Application with Ionic Framework

ผู้เขียน

ภาคย์ ผดุงพัฒนดิษฐ์

คณะวิชา

เทคโนโลยีสารสนเทศ สาขาวิชา เทคโนโลยีสารสนเทศ

อาจารย์ที่ปรึกษา

อาจารย์ สติลา ชีวกิตาการ

พนักงานที่ปรึกษา

นายณัฐวุฒิ พิทักษ์นรเศรษฐ์

ชื่อบริษัท

A-Host Company Limited

ประเภทธุรกิจ/สินค้า

ให้บริการทางด้าน Oracle Product และ Hosting Service

บทสรุป

ในการสหกิจศึกษาได้รับมอบหมายในตำแหน่งโปรแกรมเมอร์ ซึ่งช่วยในส่วนของการพัฒนาระบบต่างๆ โดยจะเน้นการใช้งานภาษาแองกูล่าเจเอส 2 (AngularJS 2) และไอออนิกเฟรมเวิร์คเวอร์ชัน 2 (Ionic Framework Version 2) ซึ่งเป็นสิ่งที่ใช้คู่กันในการพัฒนาเว็บแอปพลิเคชันและ โมบายแอปพลิเคชัน เป้าหมายการทำงานในครั้งนี้คือ การพัฒนาแอปพลิเคชันให้ได้ตรงตามความต้องการของบริษัท สามารถนำไปใช้งานได้จริงมีประสิทธิภาพและง่ายต่อการนำระบบไปพัฒนาต่อ

ผลที่ได้รับจากการดำเนินงานในส่วนของการพัฒนาแอปพลิเคชันทำให้เห็นและเข้าใจสภาพการทำงานจริง สร้างความรับผิดชอบในหน้าที่ของตนเอง งานที่ได้รับมอบหมาย รวมถึงการทำงานร่วมกับผู้อื่น การบริการเวลาเมื่อเกิดภาวะกดดันในการทำงาน นับว่าเป็นประสบการณ์ที่ดีในการนำไปใช้ในการปฏิบัติงานในวันข้างหน้าและการพัฒนาตนเอง นอกจากนี้ยังได้รู้ถึงหลักการทำงานเป็นทีมอย่างมีประสิทธิภาพ คือ การปรับตัวให้เข้ากับเพื่อนร่วมทีมและการสื่อสารในสิ่งที่คิดออกมาให้เข้าใจอย่างมีประสิทธิภาพ รวมถึงการออกแบบแอปพลิเคชันให้มีการใช้งานง่ายขึ้น แลพสมบูรณ์ เพื่อให้ผู้ใช้งานเข้าใจได้รวดเร็วและผู้พัฒนาคนอื่นนำไปพัฒนาต่อได้

กิตติกรรมประกาศ

ขอขอบคุณ นายณัฐวุฒิ พัทธนัยเศรษฐ พนักงานที่ปรึกษา(พี่เลี้ยง) และพี่ๆพนักงานทุกคนในบริษัทที่ได้ให้ความรู้ ความสามารถ ด้านการพัฒนาโมบายแอปพลิเคชัน และเทคนิคต่างๆที่ใช้ในการทำงานและสามารถนำไปใช้ในชีวิตจริงได้ ทำให้การปฏิบัติงานสหกิจศึกษา ตลอด 4 เดือนนี้เป็นการทำงานที่น่าจดจำประทับใจและได้รับความรู้มากมาย และขอขอบคุณอาจารย์ศศิลา ชีวกิตติการ ที่คอยให้คำปรึกษาทั้งในเรื่องของการทำโครงการและรูปเล่ม ทำให้งานทุกอย่างผ่านไปได้อย่างสำเร็จลุล่วง และขอขอบคุณทุกท่านที่มีส่วนช่วยเหลือข้าพเจ้าในการปฏิบัติงานสหกิจศึกษา ณ โอกาสนี้ด้วย ขอขอบคุณครับ

นายภาคย์ ผดุงพัฒนดิษฐ์
ผู้จัดทำ

TNI

สารบัญ

หน้า

บทสรุป.....	ก
กิตติกรรมประกาศ.....	ง
สารบัญภาพประกอบ.....	ช
สารบัญตาราง.....	ญ

บทที่

1 บทนำ.....	1
1.1 ชื่อและที่ตั้งของสถานประกอบการ	1
1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร	2
1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร	3
1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย	4
1.5 พนักงานที่ปรึกษา และ ตำแหน่งของพนักงานที่ปรึกษา.....	5
1.6 ระยะเวลาที่ปฏิบัติงาน.....	5
1.7 ที่มาและความสำคัญของปัญหา	6
1.8 วัตถุประสงค์หรือจุดมุ่งหมายของโครงการ	6
1.9 ผลที่คาดว่าจะได้รับการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย	6
1.10 นิยามศัพท์เฉพาะ	7
2 ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน	8
2.1 ภาษาที่ใช้ในการพัฒนา.....	8
2.1.1 TypeScript.....	8
2.1.2 JavaScript.....	9
2.1.3 CSS	10
2.2 เฟรมเวิร์กที่ใช้ในการพัฒนา	11
2.2.1 Ionic Framework.....	11

สารบัญ

หน้า

2.3 ทฤษฎีที่ใช้ในการปฏิบัติงาน	12
2.3.1 Test Driven Development (TDD)	12
2.3 เทคโนโลยีที่ใช้ในการปฏิบัติงาน	12
2.3.1 โปรแกรม Visual Studio Code	12
3 แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน	13
3.1 แผนงานการฝึกงาน	13
3.2 รายละเอียดที่นักศึกษาปฏิบัติในการฝึกงาน	13
3.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติงาน	14
3.3.1 ศึกษาการใช้งานไอออนิกเฟรมเวิร์ก เวอร์ชัน 2	14
3.3.2 ศึกษาโครงสร้างการทำงานของแอปพลิเคชัน	14
3.3.3 สร้างโครงสร้างของแอปพลิเคชันการจองระบบขนส่งสินค้า	14
3.3.4 พัฒนา User Interface ของหน้าต่างภายในแอปพลิเคชัน	15
3.3.5 ตรวจสอบการทำงานของแอปพลิเคชัน	15
3.3.6 จัดทำเอกสาร Test Case & Unit Test	15
3.3.7 ศึกษา Automate Test Hybrid App	15
3.3.8 จัดทำตัวต้นแบบ Automate Test for Ionic	15
3.3.9 จัดทำเอกสารที่เกี่ยวข้อง	16
4 สรุปผลการดำเนินงาน การวิเคราะห์และสรุปผลต่างๆ	17
4.1 ขั้นตอนและผลการดำเนินงาน	17
4.1.1 ศึกษาวิธีการพัฒนาแอปพลิเคชันในส่วนต่างๆของ Ionic Framework	17
4.1.2 เริ่มพัฒนาแอปพลิเคชันในส่วนต่างๆที่ได้รับมอบหมาย	18
4.1.3 จัดทำเอกสาร Test Case & Unit Test	34
4.1.3 ศึกษาวิธีการทำ Automate Test สำหรับ Ionic Framework	35
ขั้นตอนการติดตั้งและใช้งาน Protractor Library และ Jasmine Framework	39

สารบัญ

หน้า

ขั้นตอนการติดตั้ง Karma Library & Jasmine Framework	44
5 บทสรุปและข้อเสนอแนะ.....	54
5.1 สรุปผลการดำเนินงาน	54
5.2 แนวทางการแก้ไขปัญหา.....	55
5.3 ข้อเสนอแนะจากการดำเนินงาน	55
เอกสารอ้างอิง	56
ภาคผนวก.....	57
ประวัติผู้จัดทำโครงการ.....	58

สารบัญภาพประกอบ

ภาพที่	หน้า
1.1	แผนที่ บริษัท เอ-โฮสจำกัด.....1
1.2	คณะผู้บริหารบริษัท เอ-โฮสต์ จำกัด แต่ละแผนก.....3
2.1	โลโก้ของ TypeScript.....8
2.2	โลโก้ของ JavaScript.....9
2.3	โลโก้ของ CSS.....10
2.4	โลโก้ของ Ionic Framework.....11
4.1	หน้า CSS Utilities และคำสั่งใน Ionic Framework.....17
4.2	หน้า SCSS Variables และคำสั่งใน Ionic Framework.....18
4.3	หน้า Login Page.....19
4.4	หน้า Sign Up Page.....19
4.5	หน้า Home Page.....20
4.6	หน้า History.....21
4.7	หน้า History Detail.....21
4.8	หน้า Schedule Page.....22
4.9	หน้า News & Promotion Page.....23
4.10	หน้า Account Page.....24
4.11	หน้า Edit Account Page.....24
4.12	หน้า Pickup Address Page.....25
4.13	หน้า Consignee Address Page.....26
4.14	หน้า Transport Address Page.....27
4.15	หน้า Agent Network Page.....28
4.16	หน้า Contact Us Page.....29
4.17	หน้า Help Page.....30
4.18	หน้า Confirm Bank Transfer Page.....31
4.19	หน้า LCL Booking Page.....32

สารบัญภาพประกอบ (ต่อ)

ภาพที่	หน้า
4.20	หน้า Trailer Booking Page.....33
4.21	เอกสาร Test Case.....34
4.22	เอกสาร Unit Test.....35
4.23	การทำงานของ Ionic Framework.....36
4.24	การทำงานของ Cordova.....37
4.25	โลโก้ของ Protractor Library.....38
4.26	โลโก้ของ Jasmine Framework.....38
4.27	โลโก้ของ Karma Library.....39
4.28	ไฟล์ภายในโปรเจก.....41
4.29	ผลลัพธ์การทดสอบ.....43
4.30	ที่อยู่ของรายงานผลลัพธ์การทดสอบ.....43
4.31	รายงานผลการทดสอบ.....44
4.32	ขั้นตอนการสร้างไฟล์ Karma.conf.js.....45
4.33	ไฟล์ Config ทั้งหมด.....50
4.34	ตัวอย่างที่แก้ไขในไฟล์ package.json.....53
4.35	ผลการทดสอบ.....53

สารบัญตาราง

ตาราง

หน้า

1.1	สิ่งที่ได้รับมอบหมายในการสหกิจศึกษามีดังนี้.....	4
3.1	แผนการปฏิบัติงาน.....	15



บทที่ 1

บทนำ

1.1 ชื่อและที่ตั้งของสถานประกอบการ

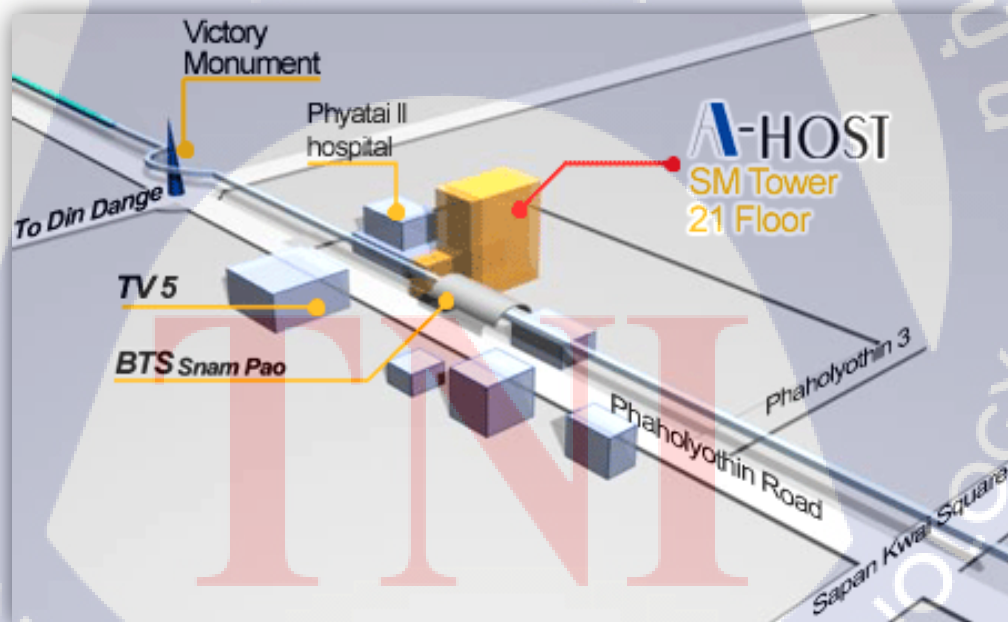
ชื่อสถานประกอบการ : บริษัท เอ-โฮสต์ จำกัด

A-Host Company Limited

ที่ตั้งสถานประกอบการ : 979/53-55 ชั้น 21 อาคาร SM Tower ถนนหลโยธิน
แขวงสามเสนใน เขตพญาไท กรุงเทพฯ 10400

โทรศัพท์ : (66) 2298-0625-32

โทรสาร : (66)2298-0053



ภาพที่ 1.1 แผนที่ บริษัท เอ-โฮสต์ จำกัด

1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร

ธุรกิจหลักของบริษัท เอ-โฮสต์ คือ การให้บริการโฮสติ้ง และบริการระบบไอทีด้วยผลิตภัณฑ์ของออรากิล (Oracle) และไอบีเอ็ม (IBM) ซึ่งเป็นซอฟต์แวร์สำหรับการวางแผนบริหารทรัพยากรขององค์กร (ERP) ระดับแนวหน้าของโลก

เอ-โฮสต์ถือกำเนิดขึ้นจากกลุ่มผู้เชี่ยวชาญด้านไอทีท่ามกลางภาวะเศรษฐกิจตกต่ำทั่วภูมิภาคเอเชียแต่ เอ-โฮสต์ ก็สามารถเติบโตได้อย่างรวดเร็ว และมั่นคงตั้งแต่แรกก่อตั้ง ด้วยจุดแข็งในฐานะผู้บุกเบิกธุรกิจโฮสติ้งเซอวิซ พร้อมทั้งนำธุรกิจแนวใหม่อย่างการให้บริการแอปพลิเคชัน หรือ ASP (Application Services Providing) เข้ามาให้บริการแก่องค์กรธุรกิจเป็นรายแรกในเมืองไทย

ธุรกิจการให้บริการแอปพลิเคชัน ในรูปแบบ ASP ของเอ-โฮสต์ไม่เพียงแต่ให้บริการด้านแอปพลิเคชันด้านการดำเนินธุรกิจทางอิเล็กทรอนิกส์ระดับโลกของออรากิลพร้อมโครงสร้างพื้นฐานทางเทคโนโลยีสารสนเทศเท่านั้น แต่ยังมีบริการที่ครอบคลุมตั้งแต่การให้คำปรึกษา การสนับสนุนและการให้บริการทั่วไปอย่างพร้อมครบบครัน รวมทั้งยังมีความยืดหยุ่นสูงมีการปรับเปลี่ยนบริการและทรัพยากรให้เหมาะสมกับความต้องการ และสภาพงานที่แตกต่างกันของลูกค้าในรายได้ระดับต่างๆ

ในการดำเนินธุรกิจของเอ-โฮสต์ตลอดระยะเวลา 10 ปี ไม่เพียงแต่ในฐานะผู้บุกเบิกธุรกิจโฮสติ้งและธุรกิจการให้บริการแอปพลิเคชันในรูปแบบ ASP เท่านั้น แต่เอ-โฮสต์ยังได้ทำการติดตั้งระบบไอที รวมทั้งผลิตภัณฑ์ของออรากิลให้กับลูกค้าจนประสบความสำเร็จมาแล้วเป็นจำนวนมาก ซึ่งหลายรายเป็นหนึ่งในร้อยบริษัทชั้นนำของประเทศไทย แต่ที่สำคัญกว่านั้นก็คือการที่ เอ-โฮสต์ ได้สานสัมพันธ์กับลูกค้า และพันธมิตรทางธุรกิจอย่างแนบแน่นจนกลายเป็นหุ้นส่วนทางกลยุทธ์ และเป็นผู้สนับสนุนสำคัญที่มีส่วนช่วยผลักดันให้ธุรกิจของลูกค้าเติบโตสู่ความสำเร็จ

ปัจจุบันเอ-โฮสต์เป็นหนึ่งในบริษัทลูกของบริษัท ดราก้อนวัน จำกัด ซึ่งเป็นบริษัทมหาชนที่อยู่ในตลาดหลักทรัพย์แห่งประเทศไทย

เป็นเวลากว่า 10 ปีที่เอ-โฮสต์ และออรากิลได้ดำเนินธุรกิจร่วมกันอย่างใกล้ชิด และถือเป็นพันธมิตรทางธุรกิจที่มีความแนบแน่นกันมานับตั้งแต่ก่อตั้งบริษัทปี 2542 จวบจนกระทั่งในปัจจุบัน

ในปี 2011 A-HOST ได้ก้าวไปข้างหน้าเพื่อความท้าทายทางธุรกิจใหม่ที่จะเป็นพันธมิตรทางธุรกิจของไอบีเอ็มพรีเมียร์ (IBM Premier Business Partner) เป็นตัวแทนจำหน่ายฮาร์ดแวร์ของไอบีเอ็มและผลิตภัณฑ์ซอฟต์แวร์ ที่จะสามารถเสริมสร้างผลิตภัณฑ์และผลงานบริการของเราเพื่อให้

ลูกค้าได้รับเทคโนโลยีที่ดีที่สุดอยู่ตลอดเวลาและช่วยส่งเสริมการเจริญเติบโตของลูกค้าอย่างรวดเร็วและมั่นคง

ด้วยความมุ่งมั่นในการนำเสนอผลิตภัณฑ์ของออร่าเคิล และ ไอบีเอ็ม ผ่านการให้บริการแอปพลิเคชันในรูปแบบของ ASP ในฐานะที่เอ-โฮสต์เป็นผู้บุกเบิกธุรกิจดังกล่าว และเพิ่มศักยภาพในการดำเนินธุรกิจของลูกค้าได้อย่างเต็มประสิทธิภาพ และเหมาะสม ทำให้ได้รับรางวัลแห่งความสำเร็จและได้รับการยกย่องมาอย่างต่อเนื่อง

1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร



ภาพที่ 1.2 คณะผู้บริหารบริษัท เอ-โฮสต์ จำกัด แต่ละแผนก

1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย

ตำแหน่งงานที่ได้รับมอบหมายในการปฏิบัติงานสหกิจศึกษาครั้งนี้คือผู้พัฒนาระบบ (Programmer) โดยจะมอบขอบเขตงานและหน้าที่หลักดังนี้

ตารางที่ 1.1 สิ่งที่ได้รับมอบหมายในการสหกิจศึกษามีดังนี้

KPI Description	Expected Results
1.ช่วยสนับสนุน ในการทำ Mobile Application (Angular and IONIC)	- สามารถช่วยกำหนด Test case ได้ - สามารถทดสอบระบบได้อย่างมีประสิทธิภาพ - สามารถเขียนโปรแกรมได้อย่างมีประสิทธิภาพ - เมื่อระบบมีปัญหา สามารถช่วยแก้ไขปัญหาคได้
2.ทำหน้าที่ผู้ทดสอบระบบ (Tester) ให้กับระบบต่างๆที่เกิดขึ้นของแผนก	- สามารถช่วยกำหนด Test case ได้ - สามารถทดสอบระบบได้อย่างมีประสิทธิภาพ
3.พัฒนา Mobile Application การจองระบบขนส่งสินค้า - Angular JS / Typescript - IONIC 2 - Laravel Framework (Restful API) - Oracle Database	- สามารถพัฒนาระบบได้อย่างมีประสิทธิภาพ - โปรแกรมผ่านการทดสอบบนเครื่อง UAT (ทำเอกสาร Check List) สามารถช่วย Deploy Application บน Production ได้ - ทำเอกสารสำหรับการพัฒนาระบบเช่น Code Diff. หรือ programming flow
พัฒนา LEO Phase 2 (Mobile) ด้วย IONIC Framework V.2 และ พัฒนา PHP Web	- สามารถช่วยกำหนด test case ได้ - จัดทำเอกสาร User Manual

Service ด้วย Laravel Framework 5.4 เพื่อเชื่อมต่อกับ Oracle Database พร้อมทั้งปรับปรุง User Interface ตามที่ Designer ออกแบบสำหรับ version ใหม่	- จัดทำเอกสาร Programmer Manual
4.ศึกษา Technology ใหม่ หรือหัวข้อที่น่าสนใจ เพื่อสนับสนุนงานโครงการต่างๆที่ได้รับมอบหมาย	- สามารถนำไปใช้งานได้จริง - สามารถถ่ายทอดให้ผู้อื่นได้ - ศึกษาเทคโนโลยี และจัดทำ Programmer Manual - 1. Automated Tests for Ionic Application
5.จัดทำ Individual Work List	- นำเสนอ IWL เป็นประจำทุกเดือน

1.5 พนักงานที่ปรึกษา และ ตำแหน่งของพนักงานที่ปรึกษา

พนักงานที่ปรึกษา : นาย ณัฐวุฒิ พัทธกันสรเศรษฐ
 ตำแหน่ง : Programmer
 แผนก : Business Solution

1.6 ระยะเวลาที่ปฏิบัติงาน

ระยะเวลาที่ปฏิบัติงานสหกิจศึกษาเป็นเวลาประมาณ 4 เดือน
 เริ่มต้นปฏิบัติงานสหกิจศึกษา วันที่ 4 มิถุนายน 2561
 สิ้นสุดปฏิบัติงานสหกิจศึกษา วันที่ 28 กันยายน 2561

1.7 ที่มาและความสำคัญของปัญหา

เนื่องจากทางเอ-โฮสต์มีโปรเจกต์ในการทำแอปพลิเคชันเกี่ยวกับการจองการขนส่งสินค้า และการซื้อขายสินค้าออนไลน์ประเภทอาหารแช่แข็งและอาหารพร้อมรับประทาน โดยใช้เซิร์ฟเวอร์ของ มาเจนโต้ในการเก็บข้อมูลทั้งหมดของแอปพลิเคชัน ซึ่งสามารถนำไปใช้ได้บนเว็บไซต์และโทรศัพท์มือถือ

1.8 วัตถุประสงค์หรือจุดมุ่งหมายของโครงการ

ในการปฏิบัติงานสหกิจศึกษาครั้งนี้มีขึ้นเพื่อให้นักศึกษาได้สามารถลงมือปฏิบัติงานในโครงการจริงในบริษัทซึ่งทางบริษัทเอ-โฮสต์มีความประสงค์ที่รับนักศึกษาสหกิจศึกษาเข้ามาเป็นส่วนหนึ่งในการพัฒนาระบบในโครงการซึ่งทำให้สามารถมองเห็นภาพรวมในการปฏิบัติงานตั้งแต่ขั้นเริ่มต้นโครงการจนถึงเสร็จสิ้นโครงการโดยนำความรู้ที่ได้จากการฝึกอบรมมาประยุกต์ใช้ในการทำงานได้จริงอีกทั้งยังเป็นการเรียนรู้ในการทำงานพัฒนาระบบจริงรวมถึงขั้นตอนการพัฒนาระบบเพื่อนำไปเป็นประสบการณ์และพัฒนาประยุกต์ใช้กับงานในภาคหน้าได้

1.9 ผลที่คาดว่าจะได้รับการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย

- 1) มีความรับผิดชอบในการทำงาน และการบริหารจัดการเวลาในการทำงาน
- 2) สามารถเข้าใจกระบวนการในการปฏิบัติงานตั้งแต่ต้นจนจบ
- 3) ได้รับเทคนิคใหม่ๆ ที่ใช้ในการทำงาน และสามารถศึกษาหาความรู้เพิ่มเติมด้วยตนเองได้
- 4) สามารถทำงานภายใต้ความกดดันของสถานการณ์ และปรับตัวให้เข้ากับการทำงาน
- 5) สามารถศึกษาค้นคว้า ความรู้ใหม่มาประยุกต์ใช้กับงานที่ได้รับมอบหมายได้

1.10 นิยามศัพท์เฉพาะ

- 1.10.1 Framework หมายถึง โครงสร้างของการเขียนโปรแกรม ที่มีการวางไว้อย่างเป็นระบบ มีรูปแบบแผน และลักษณะการเขียนเป็นมาตรฐานเดียวกันในกรณีที่มีผู้พัฒนาอยู่หลายคน ในขณะเดียวกันก็ช่วยให้การทำงานมีความรวดเร็วมากยิ่งขึ้น
- 1.10.2 Test Case หมายถึง กรณีการทดสอบโปรแกรมโดยอ้างอิงจากการทำงานของโปรแกรมที่ตรงตามความต้องการของผู้ใช้ระบบ
- 1.10.3 Individual Work List หมายถึง รายการการทำงานของแต่ละบุคคลที่ใช้ในการวางแผนการทำงานล่วงหน้า
- 1.10.4 User Interface (UI) หมายถึง สิ่งที่ใช้แสดงออกมาเพื่อให้ผู้ใช้มีปฏิสัมพันธ์กับตัวโปรแกรมหรือแอปพลิเคชันนั้นๆ ไม่ว่าจะเป็นปุ่มกด หรือข้อความที่แสดงขึ้นหลังจากที่กระทำบางอย่าง
- 1.10.5 Front-end หมายถึง ส่วนที่ติดต่อกับผู้ใช้ เช่น การแสดงผลต่างๆของแอปพลิเคชัน
- 1.10.6 Back-end หมายถึง ส่วนที่ใช้ติดต่อข้อมูลในฐานข้อมูลหรือจากเซิร์ฟเวอร์
- 1.10.7 User Manual หมายถึง คู่มือการใช้ระบบ
- 1.10.8 Programmer Manual หมายถึง คู่มือการพัฒนาระบบ

บทที่ 2

ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

ในการปฏิบัติงานสหกิจศึกษาครั้งนี้ เป็นการนำความรู้ทางด้านทฤษฎีและเทคโนโลยี มาใช้ในการปฏิบัติงานทุกส่วนตลอดการปฏิบัติงานสหกิจศึกษา ซึ่งเป็นการนำความรู้ทั้งที่เคยเรียน มาประยุกต์ใช้และเป็นการศึกษาเรียนรู้สิ่งใหม่ ๆ ที่ได้จากการปฏิบัติงาน

2.1 ภาษาที่ใช้ในการพัฒนา

2.1.1 TypeScript



ภาพที่ 2.1 โลโก้ของ TypeScript

ไทป์สคริปต์ คือ ภาษาที่ถูกใช้ในแองกูล่าเจเอส เวอร์ชัน2 แทนที่ภาษาจาวาสคริปต์ ถูกพัฒนามาให้ง่ายต่อการใช้งานมากขึ้น เป็นเครื่องมือที่ใช้ควบคุมระบบเบื้องหน้าของเว็บไซต์และช่วยให้นักพัฒนาเว็บไซต์เข้าใจและจัดการส่วนระบบเบื้องหน้าเหมือนจาวาสคริปต์ แองกูล่าเจเอสเป็นภาษาที่มีโครงสร้างแบบแอปพลิเคชันหน้าเดียว (Single Page Application) ซึ่งจะมีความรวดเร็ว

ในการทำงานเพราะไม่จำเป็นต้องโหลดใหม่ทั้งหน้าเพียงแค่โหลดข้อมูลใหม่มาแทนที่โดยมีหลักการทำงานแบบเอ็มวีซี (Model View Controller) ซึ่งเป็นหัวใจสำคัญในการทำงานที่ช่วยเชื่อมโยงโครงสร้างต่างๆ เข้าด้วยกัน โดยโมเดล (Model) จะควบคุมในส่วนของการเชื่อมต่อข้อมูลวิว (View) จะควบคุมการแสดงผลโดยใช้ภาษาเอชทีเอ็มแอลและตัวควบคุม (Controller) จะควบคุมการทำงานของแอปพลิเคชันทั้งหมด

2.1.2 JavaScript



ภาพที่ 2.2 โลโก้ของ JavaScript

JavaScript เป็นภาษาสคริปต์เชิงวัตถุหรือเรียกว่าอ็อบเจกต์โอเรียนเต็ลด์ (Object Oriented Programming) ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ตสำหรับผู้เขียนเอกสารด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ทำงานร่วมกับภาษา HTML และภาษาจาวาได้ทั้งทางฝั่งลูกข่าย (Client) และทางฝั่งแม่ข่าย (Server) โดยมีลักษณะการทำงานดังนี้

1) Navigator JavaScript เป็น Client-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่งไคลเอนต์จึงมีความเหมาะสมต่อการใช้งานของผู้ใช้ทั่วไปเป็นส่วนใหญ่

2) Live Wire JavaScript เป็น Server-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลงทางฝั่งเซิร์ฟเวอร์

2.1.3 CSS



ภาพที่ 2.3 โลโก้ของ CSS

CSS ย่อมาจาก Cascading Style Sheets เป็นภาษาที่มีรูปแบบการเขียน Syntax ที่เฉพาะและถูกกำหนดมาตรฐานโดย W3C (World Wide Web Consortium) เช่นเดียวกับ HTML และ XHTML ใช้สำหรับตกแต่งเอกสาร HTML/ XHTML ให้มีหน้าตา สี สัน ตัวอักษร เส้นขอบ พื้นหลัง ระยะห่าง ฯลฯ อย่างที่เราต้องการด้วยการกำหนดคุณสมบัติให้กับ Element ต่าง ๆ ของ HTML

TNI

2.2 เฟรมเวิร์คที่ใช้ในการพัฒนา

2.2.1 Ionic Framework



ภาพที่ 2.4 โลโก้ของ Ionic Framework

ไอออนิกเฟรมเวิร์ค คือเฟรมเวิร์คที่ใช้ภาษาเอชทีเอ็มแอล ซีเอสเอสและแองกูล่าเจส ในการพัฒนาซึ่งถูกสร้างขึ้นโดยบริษัทดีในปีพ.ศ. 2556 เนื่องจากปัจจุบันการพัฒนานะทีฟ แอปพลิเคชัน (Native Application) มีความยากลำบากและใช้เวลามากรวมถึงเมื่อพัฒนาเสร็จแล้วสามารถใช้ได้เพียงแพลตฟอร์ม (Platform) เดียว ทางไอออนิกจึงมีเป้าหมายที่จะสร้างแอปพลิเคชันบนมือถือที่มีลักษณะเป็นแบบไฮบริด (Hybrid) หรือก็คือเขียนแอปพลิเคชันครั้งเดียวสามารถใช้แพลตฟอร์มไหนก็ได้โดยมีโฟนแก๊ป (PhoneGap) หรือคอร์ดว่า (Cordova) เป็นตัวช่วยในการเชื่อมต่อเข้ากับฮาร์ดแวร์ของโทรศัพท์มือถือเพื่อที่จะใช้งานให้ได้ใกล้เคียงกับเนทีฟแอปพลิเคชัน (Native Application) ให้มากที่สุด นอกจากนี้จะช่วยเหลือในเรื่องของการทำแอปพลิเคชันแบบไฮบริดแล้วไอออนิกยังมีเทมเพลตที่สวยงามและใช้งานง่ายเพื่ออำนวยความสะดวกให้แก่ผู้พัฒนาแอปพลิเคชัน

2.3 ทฤษฎีที่ใช้ในการปฏิบัติงาน

2.3.1 Test Driven Development (TDD)

TDD คือ กระบวนการพัฒนาซอฟต์แวร์ที่อาศัยหลักการทำซ้ำๆ ในวงจรพัฒนาที่สั้นๆ โดยสิ่งแรกที่ สิ่งแรกที่ developer จะทำคือเขียน automated test case (ที่จะ fail) เพื่อที่จะกำหนดคุณสมบัติ ของฟังก์ชันใหม่ที่ต้องการ และจากนั้นจะใช้วิธีการเขียนโค้ดให้น้อยที่สุดที่จะผ่าน test แล้วจึงค่อย ปรับปรุงโค้ดให้ได้มาตรฐานอีกที

ขั้นตอนการทำ TDD

- 1) Quickly add a test - เขียน test case
- 2) Run the tests - รัน test เพื่อให้เห็นว่าไม่สามารถรัน test case ใหม่ได้
- 3) Update the code - แก้ไขโค้ดให้พอเท่าที่สามารถรัน test ได้
- 4) Re-run tests - รัน test และแก้ไขโค้ดจนกว่าจะผ่าน test
- 5) Refactor - แก้ไขโค้ดให้ดีขึ้น และได้มาตรฐานและลบสิ่งที่ไม่จำเป็นออก
- 6) Repeat - ทำขั้นตอน 1 ถึง 5 ซ้ำ

2.3 เทคโนโลยีที่ใช้ในการปฏิบัติงาน

2.3.1 โปรแกรม Visual Studio Code

Visual Studio Code คือเครื่องมือสำหรับนักพัฒนาโปรแกรม เครื่องมือชิ้นนี้จะตอบรับกับความต้องการระดับพื้นฐานอย่างเต็มรูปแบบ สามารถใช้งานได้บนวินโดวส์ แมค และลินุกซ์ ซึ่งทางไมโครซอฟท์ให้ใช้ฟรี โปรแกรมใช้งานง่ายและไม่ซับซ้อน มีความเป็นมืออาชีพ

รองรับมากกว่า 30 โปรแกรมภาษาอะไรก็ได้ เช่น C++, C#, CSS, Dockerfile, HTML, JavaScript, JSON, Less, Markdown, PHP, Python, Sass, TypeScript ที่สำคัญรองรับภาษา Java ด้วย

บทที่ 3

แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน

3.1 แผนงานการฝึกงาน

ตารางที่ 3.1 แผนการปฏิบัติงาน

หัวข้องาน	เดือนที่ 1	เดือนที่ 2	เดือนที่ 3	เดือนที่ 4
1.ศึกษา Ionic Framework เวอร์ชัน 2				
2.สร้างโครงสร้างของแอปพลิเคชันจอร์บบขนส่งสินค้า				
3.ตรวจสอบและแก้ไข UI ของ Application				
4.จัดทำเอกสาร Test Case & Unit Test				
5.ศึกษา Automate Test Hybrid App				
6.จัดทำตัวต้นแบบ Automate Test for Ionic				
7.จัดทำเอกสารที่เกี่ยวข้อง				

3.2 รายละเอียดที่นักศึกษาปฏิบัติในการฝึกงาน

การปฏิบัติงานสหกิจศึกษาทางบริษัทได้มอบหมายให้ทำตำแหน่งโปรแกรมเมอร์ของแผนกบิซิเนสโซลูชั่น(Business Solution) โดยงานที่ได้รับมอบหมายให้ปฏิบัติงานไปในทางด้านการพัฒนาระบบที่สามารถตอบสนองต่อความต้องการของธุรกิจโดยได้พัฒนาโมบายแอปพลิเคชันเกี่ยวกับการจองระบบขนส่งสินค้าและได้ศึกษาการทำทดสอบโมบายแอปพลิเคชัน (Automate Test)

การทำแอปพลิเคชันเกี่ยวกับการจองระบบขนส่งสินค้า ได้เริ่มด้วยการศึกษาในเรื่องของวิธีการพัฒนาโค้ด และโครงสร้างของไอออนิกเฟรมเวิร์ค และได้ศึกษาการพัฒนาโค้ด และโครงสร้างของซีเอสเอส และเจเอสเอส และเมื่อเข้าใจการทำงานและการพัฒนาโค้ด ในระดับหนึ่งแล้ว จึงได้เริ่มศึกษาการทำงานของแอปพลิเคชันการจองระบบขนส่งสินค้าด้วยไอออนิกเฟรมเวิร์ค ว่าในแต่ละไฟล์ทำงานอะไรบ้างและต้องแสดงผลอย่างไรบ้าง จากนั้นได้เริ่มพัฒนาแอปพลิเคชันตามงานที่ได้รับมอบหมาย เมื่อพัฒนาแอปพลิเคชันเสร็จสิ้น หลังจากนั้นได้ทำการ

ตรวจสอบการทำงานของแอปพลิเคชัน และได้ศึกษาการทำทดสอบ (Automate Test) เพื่อนำมาใช้ทดสอบแอปพลิเคชัน

3.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติงาน

3.3.1 ศึกษาการใช้งานไอโอไนคเฟรมเวิร์ก เวอร์ชัน 2

ในช่วงแรกของการทำโครงงานนี้คือการลงโปรแกรมที่ใช้พัฒนาแอปพลิเคชันโดยโปรแกรมและเทคโนโลยีที่ใช้จะมี 6 อย่างคือ

- 1) Visual Studio Code ใช้ในการเขียนและแก้ไขโค้ดของไอโอไนคเฟรมเวิร์ก
- 2) Git ใช้ในการควบคุมเวอร์ชันของแอปพลิเคชัน
- 3) Node.js ใช้สำหรับเป็นตัวช่วยในการติดตั้ง Package
- 4) Ionic Framework ใช้สำหรับพัฒนาแอปพลิเคชัน
- 5) Wampserve ใช้สำหรับในการจำลองเว็บเซิร์ฟเวอร์
- 6) Laravel ใช้สำหรับทำเป็น API ในการดึงข้อมูลจากฐานข้อมูล

จากนั้นจึงได้ศึกษาการใช้งานไอโอไนคเฟรมเวิร์ก เวอร์ชัน 2 โดยศึกษาจากเว็บไซต์ของผู้พัฒนาที่อธิบายถึงวิธีใช้งานว่าสามารถเรียกใช้อะไรได้บ้าง คำสั่งนี้ใช้งานอย่างไรและได้ทำการฝึกเขียนตัวต้นแบบเพื่อทบทวนความรู้ไปด้วย

3.3.2 ศึกษาโครงสร้างการทำงานของแอปพลิเคชัน

ในการศึกษาเรื่องของโครงสร้างการทำงานของแอปพลิเคชันที่พัฒนาด้วยไอโอไนคเฟรมเวิร์กเวอร์ชัน 2 ก่อนหน้านี้แต่เป็น User Interface เก่า ในอันดับแรกจะทำการตรวจสอบการทำงานของโค้ดเลขที่เอ็มแอลทุกหน้าว่าแสดงอะไร ใช้ฟังก์ชันอะไรจากนั้นจะทำการจดบันทึกไว้เป็นแหล่งอ้างอิงในการพัฒนาแอปพลิเคชันด้วยไอโอไนคเฟรมเวิร์ก เวอร์ชัน 2 New Design

3.3.3 สร้างโครงสร้างของแอปพลิเคชันการจองระบบขนส่งสินค้า

ในส่วนของการสร้างโครงสร้างของแอปพลิเคชัน เพื่อเป็นพื้นฐานของแอปพลิเคชัน ทั้งหน้าต่างๆ การเตรียมหน้าต่างๆ ให้พร้อมเรียกใช้งานเมื่อถูกเรียกใช้

3.3.4 พัฒนา User Interface ของหน้าต่างภายในแอปพลิเคชัน

ในส่วนของการพัฒนา User Interface ของแอปพลิเคชัน เปรียบเสมือนหน้าตาของแอปพลิเคชัน เป็นส่วนที่ต้องมีปฏิสัมพันธ์กับผู้ใช้ เพื่อให้ผู้ใช้สามารถกระทำการต่างๆภายในแอปพลิเคชันได้

3.3.5 ตรวจสอบการทำงานของแอปพลิเคชัน

ในส่วนของการตรวจสอบแอปพลิเคชันคือ จะทำการตรวจสอบการทำงานของฟังก์ชันและการแสดงผลทั้งหมดว่ามีข้อผิดพลาดตรงไหน มีส่วนไหนที่ยังไม่ได้ทำการแก้ไข แอปพลิเคชันสามารถใช้งานได้ทั้งระบบปฏิบัติการแอนดรอยด์และไอโอเอส จากนั้นจึงทำการจดบันทึกเป็นหัวข้อๆ เพื่อที่จะสามารถแก้ไขได้ถูกจุดและช่วยลดเวลาในตรวจสอบการทำงานของแอปพลิเคชัน

3.3.6 จัดทำเอกสาร Test Case & Unit Test

ในส่วนของการจัดทำเอกสาร คือจะต้องทำการทดสอบแอปพลิเคชันและทำการจดบันทึกผลลัพธ์ว่าผลลัพธ์ที่ออกมาตรงตามความคาดหวังหรือไม่ และต้องจัดทำให้อยู่ในรูปแบบเอกสารเมื่อผู้พัฒนาคอนอื่นมาทำทดสอบต่อผลลัพธ์จะต้องถูกต้องตามเอกสารที่ได้ทำการจดบันทึกไว้

3.3.7 ศึกษา Automate Test Hybrid App

ในส่วนการศึกษาของเรื่องการทำ Automate Test คืออะไร ใช้เครื่องมืออะไรในการทำทดสอบและได้ศึกษาเกี่ยวกับเรื่องข้อจำกัดของไอโอไนคเฟรมเวิร์ค

3.3.8 จัดทำตัวต้นแบบ Automate Test for Ionic

ในส่วนของการจัดทำตัวต้นแบบ Automate Test เมื่อศึกษาการทำ Automate Test และมีความเข้าใจในระดับหนึ่ง จึงได้เริ่มสร้างตัวต้นแบบในการทำ Automate Test เพื่อหาความแตกต่างในแต่ละเครื่องมือว่าเครื่องมือไหนเร็วที่สุดแล้วสามารถเข้าถึงการทำงานของไอโอไนคเฟรมเวิร์คได้ดีที่สุด

3.3.9 จัดทำเอกสารที่เกี่ยวข้อง

ในส่วนสุดท้ายของการพัฒนาระบบจะต้องมีการจัดทำเอกสารที่เกี่ยวข้องกับโครงการขึ้น เช่น

- 1) จัดทำคู่มือสำหรับโปรแกรมเมอร์เพื่อเป็นคู่มือให้สำหรับผู้พัฒนาระบบคนอื่น ๆ และสามารถใช้ต่อขยายระบบในภายภาคหน้าได้
- 2) จัดทำเอกสารที่รวบรวมการติดตั้งเครื่องมือ และการใช้งานตัวทำทดสอบ เพื่อเป็นข้อมูลให้แก่โปรแกรมเมอร์คนอื่น ๆ ที่ต้องมาพัฒนาต่อ เพื่อช่วยให้สามารถทำงานรวดเร็วได้มาก

THAI

TNI

NICHI INSTITUTE OF TECHNOLOGY

บทที่ 4

สรุปผลการดำเนินงาน การวิเคราะห์และสรุปผลต่าง ๆ

4.1 ขั้นตอนและผลการดำเนินงาน

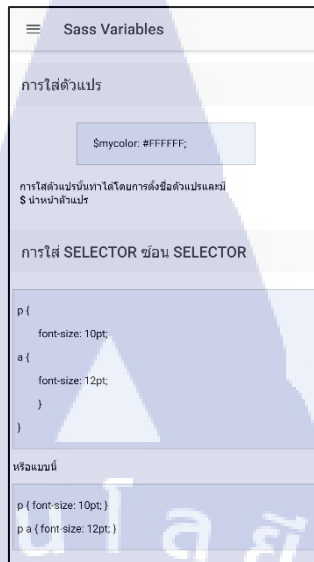
ในช่วงระยะแรกเวลา 2 สัปดาห์แรกของการปฏิบัติงานสหกิจศึกษา จะเป็นช่วงของการศึกษาข้อมูลเกี่ยวกับการเขียนโค้ดและทำความเข้าใจโครงสร้างของไอโอไนคเฟรมเวิร์ค และทำความเข้าใจกับการทำงานของแอปพลิเคชันการจอร์บบนขนส่งสินค้า (User Interface เก่า) อย่างละเอียด โดยในช่วง 2 สัปดาห์หลังเป็นต้นไป ได้เริ่มทำการพัฒนาแอปพลิเคชันตามรูปแบบที่ได้รับมอบหมาย

4.1.1 ศึกษาวิธีการพัฒนาแอปพลิเคชันในส่วนต่างๆของ Ionic Framework

อย่างที่ได้อธิบายไปข้างต้น ในช่วงระยะเวลา 2 สัปดาห์แรก จะเป็นการศึกษาข้อมูลเกี่ยวกับการพัฒนาแอปพลิเคชันโดยใช้ Ionic Framework รวมถึงภาษาต่างๆที่จำเป็นต้องใช้ในการพัฒนา เช่น CSS, SCSS, HTML เพื่อเตรียมพร้อมต่อการพัฒนาแอปพลิเคชันต่อไป และจัดทำตัวต้นแบบควบคู่ไป



ภาพที่ 4.1 CSS Utilities และคำสั่งใน Ionic Framework



ภาพที่ 4.2 SCSS Variables และคำสั่งใน Ionic Framework

4.1.2 เริ่มพัฒนาแอปพลิเคชันในส่วนต่างๆที่ได้รับมอบหมาย

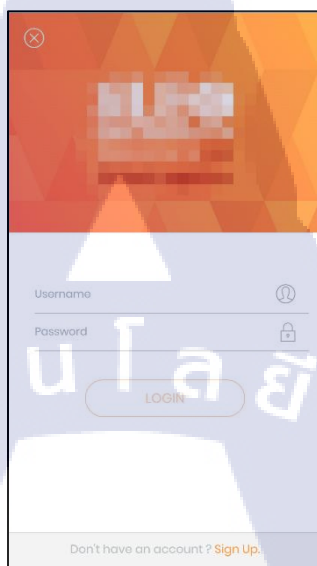
เริ่มขั้นตอนการพัฒนาแอปพลิเคชันในส่วนต่างๆ ตามรูปแบบที่ได้รับมอบหมายในรูปแบบ

User Interface

TNI

4.1.2.1 หน้า Login Page

จัดทำหน้า Login Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



ภาพที่ 4.3 Login Page

4.1.2.2 หน้า Sign Up Page

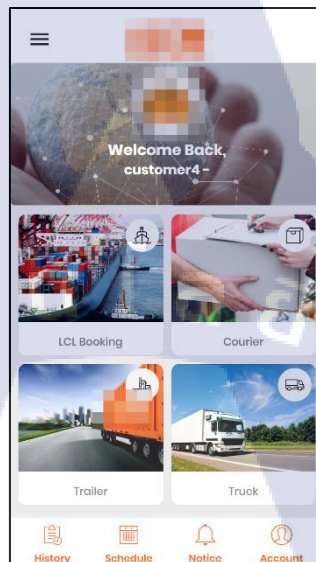
จัดทำหน้า Sign Up page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย

A screenshot of a mobile application's sign-up page. The page has a white background with a blue header bar at the top. Below the header, there is a large, stylized 'TNI' logo in the background. The sign-up form consists of several input fields: 'Firstname', 'Lastname', 'Email', 'Mobile Phone', 'Password', and 'Confirm Password'. Below these fields is a section titled 'Booking Service' with three radio button options: 'Courier Service' (selected), 'Truck & Trailer Service', and 'LCL Service'. Each option has a small orange circle next to it and the text 'Waiting for Approval' below it. At the bottom, there is a blue 'SUBMIT' button.

ภาพที่ 4.4 Sign Up Page

4.1.2.3 หน้า Home Page

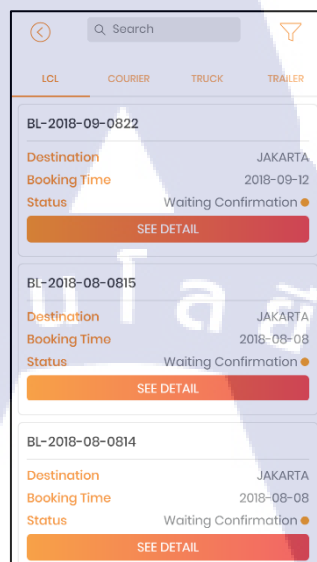
จัดทำหน้า Home Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



ภาพที่ 4.5 Home Page

4.1.2.4 หน้า History Page

จัดทำหน้า History Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



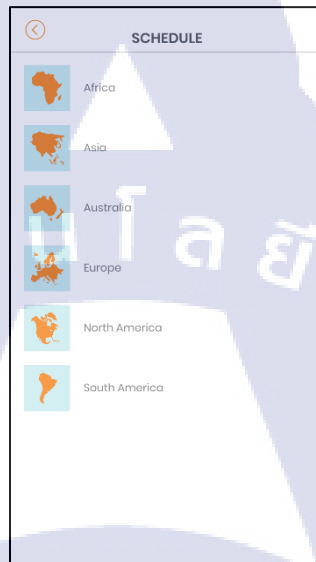
ภาพที่ 4.6 History



ภาพที่ 4.7 History Detail

4.1.2.5 หน้า Schedule Page

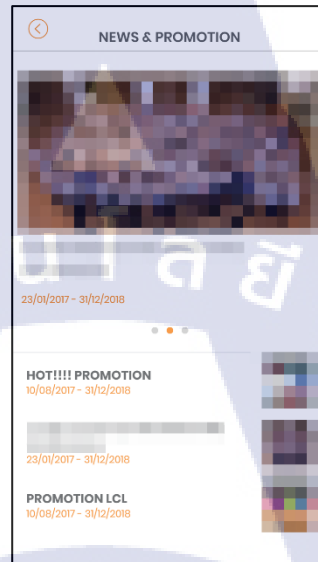
จัดทำหน้า Schedule Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



ภาพที่ 4.8 Schedule Page

4.1.2.6 หน้า News & Promotion Page

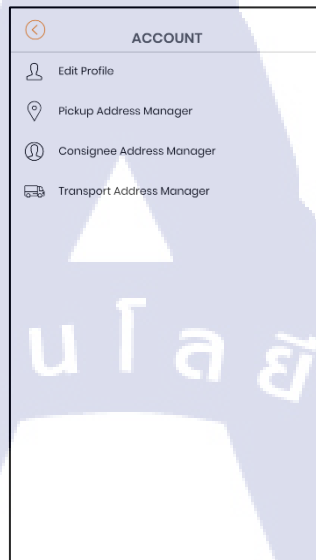
จัดทำหน้า News & Promotion Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



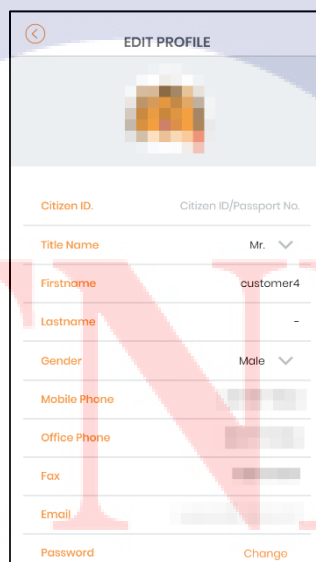
ภาพที่ 4.9 News & Promotion Page

4.1.2.7 หน้า Account Page

จัดทำหน้า Account Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



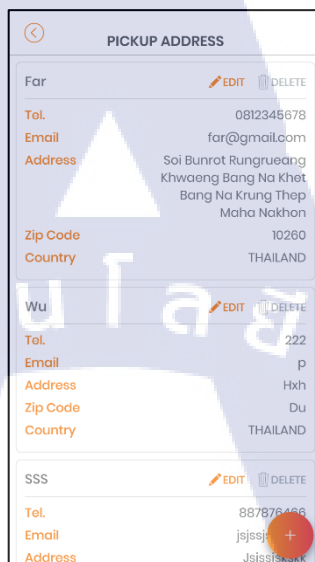
ภาพที่ 4.10 Account Page



ภาพที่ 4.11 Edit Account Page

4.1.2.8 หน้า Pickup Address Page

จัดทำหน้า Pickup Address Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



The screenshot displays a mobile application interface titled "PICKUP ADDRESS". It contains three address entries, each with a set of fields and action buttons. The first entry, "Far", has a large white triangle icon. The second entry, "Wu", has a large white circle icon. The third entry, "SSS", has a large red circle icon with a white plus sign. Each entry includes fields for "Tel.", "Email", "Address", "Zip Code", and "Country", as well as "EDIT" and "DELETE" buttons.

Far	Wu	SSS
Tel. 0812345678	Tel. 222	Tel. 887876466
Email far@gmail.com	Email p	Email jsjsj
Address Soi Bunrot Rungruedang Khwaeng Bang Na Khet Bang Na Krung Thep Maha Nakhon	Address Hxh	Address Jsjsjskakk
Zip Code 10260	Zip Code Du	
Country THAILAND	Country THAILAND	

ภาพที่ 4.12 Pickup Address Page

4.1.2.9 หน้า Consignee Address Page

จัดทำหน้า Consignee Address Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



The screenshot displays a web application titled "CONSIGNEE ADDRESS". It contains three sections, each for a different consignee. Each section has an "EDIT" button (pencil icon) and a "DELETE" button (trash icon). The fields are color-coded: orange for Name, Contact, and Remark; blue for Tel; and light blue for Address and Zip Code.

Consignee	Consignee Name	Contact	Tel.	Address	Zip Code	Remark
Consignee 1	Chanchira	Frame	0801112222	10 Phothalo	66130	
Consignee 2	Peera	Top	0801086452	8/1 test "" ????	18130	ทดสอบ
Consignee 3	Surapun		08675			

ภาพที่ 4.13 Consignee Address Page

4.1.2.10 หน้า Transport Address Pag

จัดทำหน้า Transport Address Pag โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



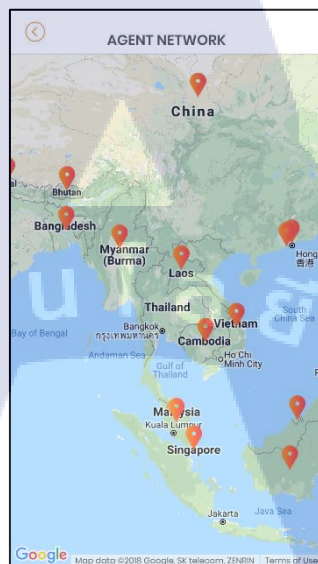
The screenshot displays a mobile application interface titled "TRANSPORT ADDRESS". It features a list of two entries, each with a back arrow, an "EDIT" button, and a "DELETE" button. The first entry is for "Bc'Dui" with a telephone number "0944890114" and an address "167/1 Phahonyothin Rd, Khwaeng Samsen Nai, Khet Phaya Thai Krung, Thep Maha Nakhon 10400". The second entry is for "Far" with a telephone number "0812345678" and an address "76 Soi Pradiphati 25, Khwaeng Samsen Nai, Khet Phaya Thai Krung, Thep Maha Nakhon 10400". Both entries have a "District" field set to "Phayathai" and a "Zipcode" field set to "10400". A red circular button with a white plus sign is located at the bottom right of the list.

Name	Tel	Address	District	Zipcode	Remark
Bc'Dui	0944890114	167/1 Phahonyothin Rd Khwaeng Samsen Nai Khet Phaya Thai Krung Thep Maha Nakhon 10400	Phayathai	10400	
Far	0812345678	76 Soi Pradiphati 25 Khwaeng Samsen Nai Khet Phaya Thai Krung Thep Maha Nakhon 10400	Dindaeng	10400	

ภาพที่ 4.14 Transport Address Page

4.1.2.11 หน้า Agent Network Page

จัดทำหน้า Agent Network Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



ภาพที่ 4.15 Agent Network Page

4.1.2.12 หน้า Contact Us Page

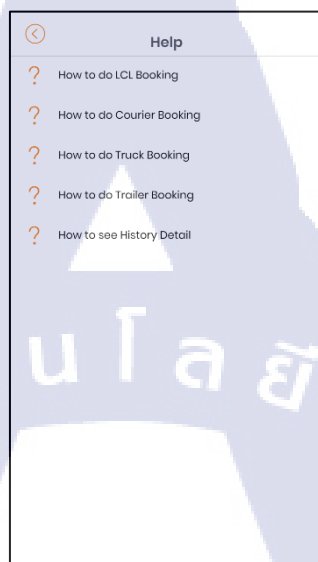
จัดทำหน้า Contact Us Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



ภาพที่ 4.16 Contact Us Page

4.1.2.13 หน้า Help Page

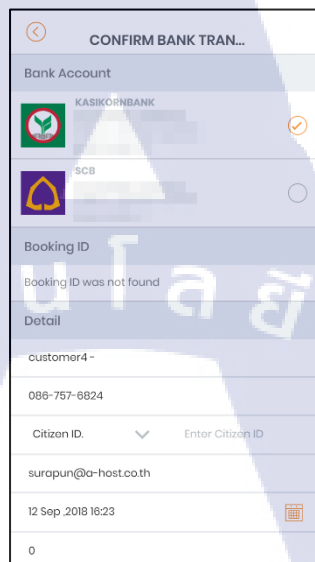
จัดทำหน้า Help Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



ภาพที่ 4.17 Help Page

4.1.2.14 หน้า Confirm Bank Transfer Page

จัดทำหน้า Confirm Bank Transfer Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



The screenshot shows a mobile application interface for confirming a bank transfer. The title bar at the top is labeled "CONFIRM BANK TRAN...". Below the title bar, there is a section for "Bank Account" with two options: "KASIKORN BANK" (selected with a checkmark) and "SCB" (unselected with a circle). Below this, there is a "Booking ID" section with the text "Booking ID was not found". The "Detail" section contains the following information: "customer4-", "086-757-6824", "Citizen ID." with a dropdown arrow and the text "Enter Citizen ID", "surapun@a-host.co.th", "12 Sep. 2018 16:23", and "0".

ภาพที่ 4.18 Confirm Bank Transfer Page

4.1.2.15 หน้า LCL Booking Page

จัดทำหน้า LCL Booking page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย

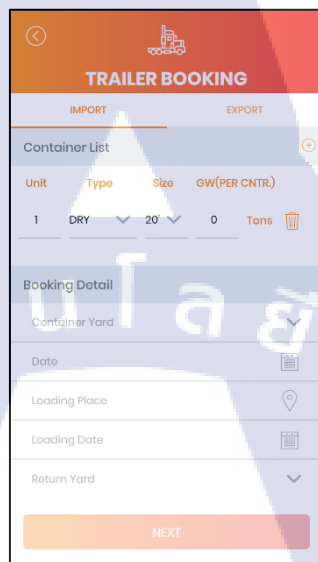


The screenshot shows a mobile application interface for LCL Booking. At the top is a red header bar with a back arrow icon on the left, a warehouse icon in the center, and the text "LCL BOOKING" on the right. Below the header is a form with several fields: "POD" with a dropdown arrow, "Wed 12 Sep 18" with a calendar icon, "Volume" with a dropdown arrow, "Gross weight" and "kg" with a dropdown arrow, "Commodities" with a dropdown arrow, "Quantity" and "carton(s)" with a dropdown arrow, and "Additional Details" with a text input field. At the bottom of the form is a red button labeled "NEXT".

ภาพที่ 4.19 LCL Booking Page

4.1.2.16 หน้า Trailer Booking Page

จัดทำหน้า Trailer Booking Page โดยทำการจัดวาง Element ต่างๆ ตามตำแหน่งที่ได้ถูกมอบหมาย



The screenshot displays the 'TRAILER BOOKING' mobile application interface. At the top, there is a red header bar with a back arrow icon on the left and a truck icon on the right. Below the header, the title 'TRAILER BOOKING' is centered. Underneath the title, there are two tabs: 'IMPORT' (active) and 'EXPORT'. The main content area is divided into two sections. The first section, 'Container List', features a table with the following columns: 'Unit', 'Type', 'Size', and 'GW(PER CNTR.)'. The table contains one row with the values '1', 'DRY', '20', and '0'. To the right of the '0' value, there is a 'Tons' label and a trash can icon. The second section, 'Booking Detail', contains five input fields: 'Container Yard' (with a dropdown arrow), 'Date' (with a calendar icon), 'Loading Place' (with a location pin icon), 'Loading Date' (with a calendar icon), and 'Return Yard' (with a dropdown arrow). At the bottom of the form, there is a large orange button labeled 'NEXT'.

ภาพที่ 4.20 Trailer Booking Page

4.1.3 จัดทำเอกสาร Test Case & Unit Test

วิธีจัดทำเอกสาร Test Case & Unit Test อันดับแรกจะต้องจัดทำเอกสาร Test Case ก่อน การทำเอกสาร Test Case นั้นจะเป็นการตรวจสอบการทำงานของแอปพลิเคชันทั้งระบบ ว่าสามารถทำงานความที่คาดหวังได้หรือไม่ เหมือนเป็นการทดสอบแอปพลิเคชันที่เลียนแบบเหมือนผู้ใช้งานจริง โดยจะตรวจการแสดงผล, การทำงานของแต่ละโมดูล, การทำงานของแต่ละหน้าของแอปพลิเคชัน

1. TC-001 หน้า Home

Test Case No.	001
Test Case Type	Normal Case
Purpose	เพื่อทดสอบการเข้าใช้งานและการแสดงผลข้อมูลจากหน้า Home
Initial System State	เข้าแอปพลิเคชันผ่านสมาร์ทโฟน ระบบปฏิบัติการ IOS
Test Action	1. สามารถเข้าแอปพลิเคชันได้ หน้า Home แสดงผลถูกต้อง 
Expected Output	- สามารถเข้าใช้งานได้ปกติ ระยะเวลาการคอยไม่เกิน 5 วินาที - ไม่พบข้อความที่สะกดผิดพยางค์ - การแสดงผลเมนูย่อยครบถ้วน
Remarks	- หากมีการกดที่ปุ่ม LCL BOOKING , COURIER , TRUCK TRAILER จะต้องทำการ login เพื่อเข้าสู่ระบบก่อน

ภาพที่ 4.21 ตัวอย่างเอกสาร Test Case

เมื่อทำเอกสาร Test Case เสร็จก็จะสามารถทำเอกสาร Unit Test ได้ ทำไมถึงต้องทำเอกสาร Test Case ก่อน เอกสาร Unit Test ก็เพราะว่าเอกสาร Test Case เปรียบเสมือนการกำหนดว่าจะทดสอบหน้าไหนและจะเกิดความผิดพลาดตรงไหนบ้าง ส่วนเอกสาร Unit Test จะเป็นการกำหนดขั้นตอนในการทำทดสอบทุกขั้นตอนเช่น การคลิกปุ่ม ,การระบุข้อมูล ,การตรวจสอบผลการแสดง เมื่อมีการส่งข้อมูลไปให้ โดยเอกสาร Unit Test จะมีรายละเอียดที่มากกว่าและจะต้องทำการทดสอบทั้งกรณีที่ใช้งานแอปพลิเคชันสำเร็จ และไม่สำเร็จ

Case Name	Test Step Description	Test Data	Expected Result
ction : XXXXXXXX			
Home	- ทดสอบการเข้าใช้งาน และการแสดงผลข้อมูล หน้า Home	-	- สามารถเข้าใช้งานได้
Log-in (Success Case)	- ระบุ Username และกรอก Password เพื่อเข้าสู่ระบบ	- Username = customer4 - Password = 123456	- สามารถเข้าสู่ระบบได้
Log-in (Error Case)	1. แจ้งเตือนเมื่อระบุ Username แต่ไม่ระบุ Password 2. แจ้งเตือนเมื่อระบุ Password แต่ไม่ระบุ Username 3. หน้าจอแจ้งเตือนเมื่อระบุ Username, Password ไม่ถูกต้อง	1. Username = customer4 Password = 2. Username = Password = 123456 3. Username = Incorrect Password = Incorrect	- ไม่สามารถเข้าสู่ระบบได้ 1. แจ้งเตือน Password is required. 2. แจ้งเตือน Username is required. 3. แจ้งเตือน the username or password is incorrect!

ภาพที่ 4.22 ตัวอย่างเอกสาร Unit Test

4.1.3 ศึกษาวิธีการทำ Automate Test สำหรับ Ionic Framework

งานที่ได้รับมอบหมายต่อไปก็จะเป็นงานศึกษาเกี่ยวกับเรื่องการทำทดสอบโดยใช้เครื่องมือ หรือเรียกว่าการทำ Automate Test

การทำทดสอบจะถูกแบ่งออกเป็น 3 ประเภท

- 1) GUI Test
- 2) Integration Test
- 3) Unit Test

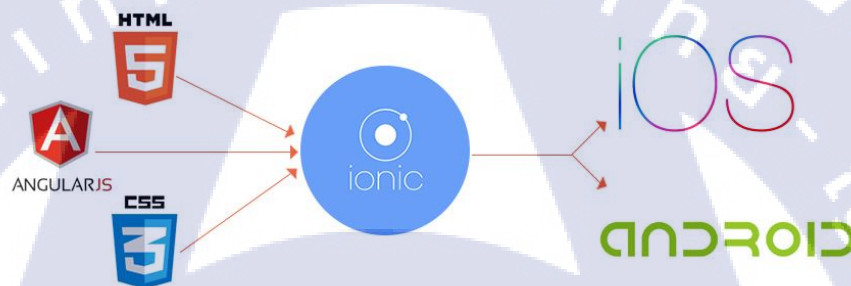
GUI Test จะทดสอบแอปพลิเคชันเสมือนเป็นผู้ใช้งานคนหนึ่ง ซึ่งจะทดสอบการทำงานของทั้งระบบของ แอปพลิเคชันว่าสามารถทำงานตาม Flow การใช้งานได้หรือไม่

Integration Test จะทดสอบการเชื่อมต่อว่าเป็นไปตามรูปแบบที่คาดหวังไว้หรือไม่ เน้นไปที่การเชื่อมต่อเป็นส่วน ๆ เช่น การเชื่อมต่อของ API และ Database

Unit Test จะทดสอบในส่วนเป็น Function ภายในแอปพลิเคชันว่าสามารถทำงานได้ตามที่คาดหวังหรือไม่ และผลลัพธ์ที่ออกมาตรงตามความต้องการหรือไม่

เมื่อได้รู้แล้วว่าการทดสอบแบ่งออกเป็นกี่ประเภท พนักงานที่ปรึกษาาก็ได้มอบหมายงานให้ทำการศึกษาหาเครื่องมือที่ใช้ทำทดสอบ Automate Test สำหรับ Ionic Framework ประเภท GUI Test และ Unit Test และเมื่อได้รับมอบหมายงานมาก็ได้ดำเนินการศึกษาเรื่อง ข้อจำกัดของ Ionic Framework

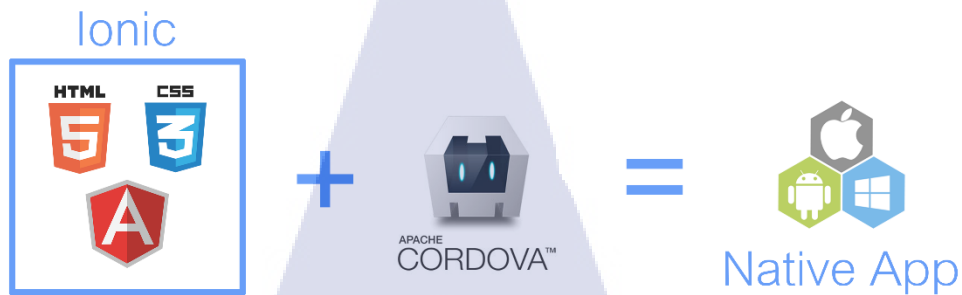
Ionic Framework เป็นเครื่องมือที่ใช้ในการพัฒนา Mobile Application แบบ Hybrid สามารถพัฒนาแอปพลิเคชันภายในครั้งเดียวแล้วสามารถใช้งานได้หลาย Platform เช่น Android , iOS ที่ได้รับความนิยมในปัจจุบัน



ภาพที่ 4.23 รูปแสดงการทำงานของ Ionic Framework

Ionic Framework จะใช้เทคโนโลยีในการพัฒนาคือ HTML5, CSS3 และ JAVA Script ตัวการพัฒนาหลักๆเลยที่จะต้องใช้คือ JAVA Script เพราะว่า Ionic framework ใช้ Core หลักเป็น Angular JS

ข้อดีที่ของ Ionic Framework คือมีความใกล้เคียงกับ Native Application มาก ทั้งหน้า User Interface การใช้งาน แต่ก็ไม่ถึงกับเหมือนกับ Native App และ Ionic Framework สามารถพัฒนาให้ติดต่อกับ Hardware ของอุปกรณ์ได้ เช่น กล้อง , ไมโครโฟน การตรวจสอบการเชื่อมต่ออินเทอร์เน็ต เป็นต้น โดยใช้งานร่วมกับ PhoneGap/Cordova



ภาพที่ 4.24 รูปแสดงการทำงานของ Cordova

PhoneGap/Cordova เป็นเครื่องมือตัวหนึ่งที่มีหน้าที่ในการสร้างแอปพลิเคชันจากการพัฒนาการใช้เทคโนโลยี HTML, CSS, JavaScript

จะเห็นได้ว่า Ionic Framework คือ Framework ที่จะช่วยในการพัฒนา Web Application ให้ง่ายขึ้นโดยที่สามารถทำงานร่วมกับ Cordova เพื่อเปลี่ยน จาก Web Application เป็น Native App ด้วยเหตุผลนี้ทำให้สามารถหาเครื่องมือในการทำ Automate Test ได้ง่ายขึ้นอีกและได้ทำการศึกษาในเรื่อง Automate Test คืออะไรทำไมถึงต้องทำ Automate Test

ปัจจุบันการทดสอบซอฟต์แวร์ถือเป็นเรื่องที่สำคัญมากสำหรับการส่งมอบซอฟต์แวร์ และเชื่อได้ว่าหลายๆ บริษัทต้องมีตำแหน่งงานที่คอยทำหน้าที่นี้ ซึ่งส่วนใหญ่ยังใช้วิธีการแบบ Manual Test อยู่ ซึ่งต้องทำซ้ำไปซ้ำมา ซึ่งการทำ Automate Test จะดีกว่าการทำ Manual Test คือ

- 1) ลดระยะเวลาในการทำทดสอบแอปพลิเคชัน
- 2) ลดความผิดพลาดที่จะเกิดขึ้นในระหว่างการทำทดสอบแอปพลิเคชัน
- 3) เพิ่มความมั่นใจให้กับผู้พัฒนาตลอดจนลูกค้า
- 4) ลดค่าใช้จ่าย

ซึ่งในการที่จะทำ Automate Test ก็ต้องทำเอกสาร Test Case มาก่อนเพื่อที่จะให้ Automate Test ทำงานทดสอบตามเอกสาร Test Case ได้

เมื่อมีความเข้าใจเกี่ยวกับการทำ Automate Test ก็ได้ค้นหาเครื่องมือที่เหมาะสมในการทำทดสอบซึ่งได้ค้นหาและนำมาใช้งานในเบื้องต้นเพื่อทำความเข้าใจกับหลักการในการทำงานของเครื่องมือ ทดสอบในระดับที่เป็น GUI Test ได้ใช้เครื่องมือ Protractor Library และ Jasmin Framework และในระดับที่เป็น Unit Test ได้ใช้เครื่องมือ Karma Library และ Jasmine Framework



Protractor

end to end testing for AngularJS

ภาพที่ 4.25 โลโก้ของ Protractor Library

Protractor Library คือ เครื่องมือที่มีไว้สำหรับทำทดสอบ Angular JS ซึ่งจะทำหน้าที่เลียนแบบเปรียบเสมือนผู้ใช้งานจริง ซึ่ง Protractor Library สามารถทำงานได้เร็ว และ โค้ดในการเรียกก็สั้น สามารถเข้าใจง่ายและสามารถทำงานร่วมกับ Framework ตัวอื่นได้อีกเช่น Jasmine, Mocha, Jenkin, Appium, Cucumber, BrowserStack และรองรับการทดสอบบน Browser ได้หลายตัวเช่น Firefox, Chrome, Safari, Internet explorer ซึ่ง Protractor Library จะทำหน้าที่ในการหา Element ต่างๆของเว็บเพจนั้นๆเพื่อที่จะสามารถทำตามพฤติกรรมของผู้ใช้งานได้เช่นการคลิกปุ่ม, การระบุข้อมูล, การตรวจสอบการแสดงผล.



Jasmine

ภาพที่ 4.26 โลโก้ของ Jasmine Framework

Jasmine Framework คือ Framework ที่ใช้สำหรับทดสอบ JavaScript ซึ่งมีไวยากรณ์ที่อ่านได้ง่าย โดยที่ Jasmine Framework รองรับการทดสอบแบบ Asynchronous ซึ่งสามารถทำให้การทดสอบเป็นไปได้อย่างรวดเร็ว



ภาพที่ 4.27 โลโก้ของ Karna Library

Karma Library เป็นเครื่องมือหลักที่จะสร้างเว็บเซิร์ฟเวอร์ที่ประมวลผล Test Case ในแต่ละเบราว์เซอร์ที่เชื่อมต่อ ผลลัพธ์ของการทดสอบแต่ละครั้งกับเบราว์เซอร์แต่ละตัวจะได้รับการตรวจสอบและแสดงผลผ่านทาง Command Prompt เพื่อให้สามารถดูว่าเบราว์เซอร์และการทดสอบใดผ่านหรือล้มเหลว

ขั้นตอนการติดตั้งและใช้งาน Protractor Library และ Jasmine Framework

Required

- 1) Node JS
- 2) Ionic Framework

Step 1

- 1) เปิด Command Prompt
- 2) สร้างโปรเจกต์ด้วยคำสั่ง ionic start automate-test tabs (ต้องติดตั้ง ionic framework ก่อน)
- 3) ติดตั้ง Protractor Library , Jasmine Framework และตัวออก Report ด้วยคำสั่ง
`npm install --save-dev jasmine jasmine-spec-reporter protractor protractor-beautiful-reporter`
- 4) ติดตั้ง ตัว Run Test Case บน Web Browser ด้วยคำสั่ง

`npm install -g webdriver-manager`

Step 2

5) สร้างไฟล์ protractor.conf.js

```
1. // Import packet ที่สามารถช่วยทำ Report ตอน Test เสร็จ
2. let SpecReporter = require('jasmine-spec-reporter').SpecReporter;
3. let HtmlReporter = require('protractor-beautiful-reporter');
4. exports.config = {
5.   capabilities: {
6.     // ตั้งค่าโดยใช้ Browser Chrome ในการทดสอบ
7.     'browserName': 'chrome'
8.   },
9.   // ตั้งค่าโดยใช้ jasmine เป็น Framework ในการทดสอบ Test Case
10.  framework: 'jasmine',
11.  jasmineNodeOpts: {
12.    showColors: true,
13.    defaultTimeoutInterval: 30000,
14.    print: function() {},
15.  },
16.  // ตั้งค่ากำหนดที่อยู่ของไฟล์ที่จะทำการ Test
17.  specs: ['./e2e/*.js'],
18.  useAllAngular2AppRoots: true,
19.  // ตั้งค่าการออก Report
20.  onPrepare: function(){
21.    jasmine.getEnv().addReporter(new SpecReporter({
22.      displayFailuresSummary: true,
23.      displayFailedSpec: true,
24.      displaySuiteNumber: true,
25.      displaySpecDuration: true
26.    }));
27.    jasmine.getEnv().addReporter(new HtmlReporter({
```

```

28.     baseDirectory: './tmp/screenshots'
29.   }).getJasmine2Reporter());
30. }
31. }

```

Step 3

6) สร้าง Folder e2e ใน Project เพื่อเก็บไฟล์ Test Case

e2e	9/18/2018 4:26 PM	File folder
node_modules	9/18/2018 4:07 PM	File folder
src	9/18/2018 4:06 PM	File folder
.editorconfig	9/7/2018 10:48 PM	EDITORCONFIG File
	9/7/2018 10:48 PM	Text Document
ionic.config.json	9/18/2018 4:06 PM	JSON File
ionic.starter.json	9/7/2018 10:48 PM	JSON File
package.json	9/18/2018 4:06 PM	JSON File
package-lock.json	9/18/2018 4:07 PM	JSON File
protractor.conf	9/18/2018 4:27 PM	JavaScript File
tsconfig.json	9/7/2018 10:48 PM	JSON File
tslint.json	9/7/2018 10:48 PM	JSON File

ภาพที่ 4.28 แสดงให้เห็นถึงไฟล์ภายในโปรเจกต์

7) สร้างไฟล์ Test Case ใน Folder e2e ชื่อ welcome.js เพื่อทดสอบว่าสามารถทดสอบได้

```

1. describe('Test Case Name',()=>{
2.   // ก่อนที่จะเริ่มทำ Test Case
3.   beforeEach(() => {
4.     browser.get('http://localhost:8100');
5.   });
6.   //Test Case
7.   it('h1 tobe Welcome', () => {
8.     // Selector หา Element โดยใช้ Proctor

```

```
9.     var welcome = element(by.xpath('//*[@id="tabpanel-t0-0"]/page-home/ion-
      content/div[2]/h2'));
10.    // Expect ผลลัพธ์ที่คาดหวัง โดยใช้ Jasmine
11.    expect(welcome.getText()).toBe('Welcome to Ionic!');
12.  });
13. });
```

Step 4

- 8) เปิด Command Prompt
- 9) ใช้คำสั่ง ionic serve เพื่อทำการจำลอง Web application ขึ้นมา
- 10) เปิด Command Prompt
- 11) ใช้คำสั่ง webdriver-manager update
- 12) ใช้คำสั่ง protractor เพื่อสั่งทำการทดสอบ Test Case

13) สามารถดู Report เพิ่มเติมได้ที่ Folder tmp

```
C:\Users\PK\Desktop\Example\automate-test>protractor
[16:50:13] I/launcher - Running 1 instances of WebDriver
[16:50:13] I/local - Starting selenium standalone server...
[16:50:14] I/local - Selenium standalone server started at http://172.20.10.4:53014/wd/hub
Jasmine started

Test Case Name
  ✓ h2 tobe Welcome

Executed 1 of 1 spec SUCCESS in 2 secs.
[16:50:19] I/local - Shutting down selenium standalone server.
[16:50:19] I/launcher - 0 instance(s) of WebDriver still running
[16:50:19] I/launcher - chrome #01 passed
```

ภาพที่ 4.29 ผลลัพธ์การทดสอบ

	.sourcemaps	9/18/2018 4:49 PM	File folder
	e2e	9/18/2018 4:33 PM	File folder
	node_modules	9/18/2018 4:32 PM	File folder
	src	9/18/2018 4:06 PM	File folder
	tmp	9/18/2018 4:50 PM	File folder
	www	9/18/2018 4:49 PM	File folder
	.editorconfig	9/7/2018 10:48 PM	EDITORCONFIG File
		9/7/2018 10:48 PM	Text Document
	ionic.config.json	9/18/2018 4:06 PM	JSON File

ภาพที่ 4.30 ที่อยู่ของรายงานผลลัพธ์การทดสอบ

TNI



The screenshot shows a 'Test Results' page with a green progress bar at the top. Below the progress bar is a search bar and filter buttons (ALL, PASSED, FAILED, PENDING, OR WITH LOG). A table lists test results with columns: Status, Time, Description, Browser, Session ID, OS, Message, Log, Stack, and Screen. One test case is shown: 'h2 to be Welcome' with status 'Passed' (green checkmark), time '1.4s', browser 'chrome 69.0.3497.100', session ID 'fd8cf647b970cfebe017209bc68a4515', OS 'Windows NT', and message 'Passed.'.

Status	Time	Description	Browser	Session ID	OS	Message	Log	Stack	Screen
✓ 1.4s		h2 to be Welcome	chrome 69.0.3497.100	fd8cf647b970cfebe017209bc68a4515	Windows NT	Passed.			

ภาพที่ 4.31 รายงานผลการทดสอบ

ขั้นตอนการติดตั้ง Karma Library & Jasmine Framework

Required

- 1) Node JS
- 2) Ionic Framework

Step 1

- 3) เปิด Command Prompt
- 4) สร้างโปรเจกต์ด้วยคำสั่ง ionic start automate-test tabs (ต้องติดตั้ง ionic framework

ก่อน)

- 5) ติดตั้ง Karma Library , Jasmine Framework และตัวออก Report ด้วยคำสั่ง

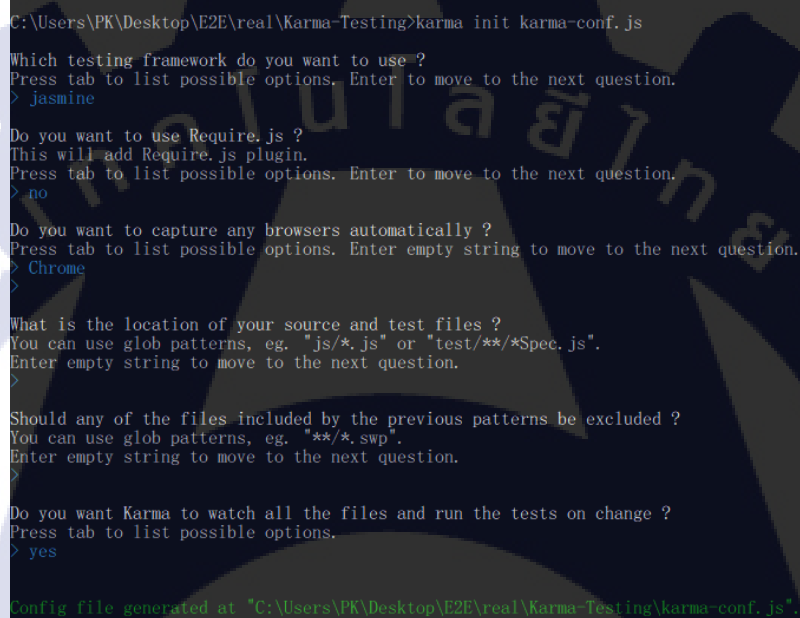
```
npm install @netbasal/spectator @types/jasmine angular2-template-loader html-loader jasmine
jasmine-core jasmine-spec-reporter karma karma-chrome-launcher karma-jasmine karma-jasmine-
html-reporter karma-sourcemap-loader karma-webpack null-loader ts-loader@3.5.0 --save-dev
```

Step 2

6) สร้าง Folder ชื่อ test-config

7) สร้างไฟล์ karma.conf.js โดยใช้คำสั่ง karma init karma.conf.js ไว้สำหรับการตั้งค่า

ต่างๆ ในการทดสอบเช่น กำหนดที่อยู่ของไฟล์ที่ทดสอบ กำหนดประเภทของ Report กำหนด Browser ที่ต้องการทดสอบและอื่นๆ



```
C:\Users\PK\Desktop\E2E\real\Karma-Testing>karma init karma.conf.js
Which testing framework do you want to use?
Press tab to list possible options. Enter to move to the next question.
> jasmine
Do you want to use Require.js?
This will add Require.js plugin.
Press tab to list possible options. Enter to move to the next question.
> no
Do you want to capture any browsers automatically?
Press tab to list possible options. Enter empty string to move to the next question.
> Chrome
What is the location of your source and test files?
You can use glob patterns, eg. "js/*.js" or "test/**/*.spec.js".
Enter empty string to move to the next question.
Should any of the files included by the previous patterns be excluded?
You can use glob patterns, eg. "**/*.swp".
Enter empty string to move to the next question.
Do you want Karma to watch all the files and run the tests on change?
Press tab to list possible options.
> yes
Config file generated at "C:\Users\PK\Desktop\E2E\real\Karma-Testing\karma.conf.js".
```

ภาพที่ 4.32 ขั้นตอนการสร้างไฟล์ Karma.conf.js

```
1.  var webpackConfig = require('./webpack.test.js');
2.  module.exports = function (config) {
3.    var _config = {
4.      // กำหนด Url link ที่จะให้เข้าทุกครั้งที่มีการทดสอบ
5.      basePath: "",
6.      // กำหนด Framework ที่ใช้
7.      frameworks: ['jasmine'],
8.      // กำหนดที่อยู่ของไฟล์ Test Case
9.      files: [
10.        { pattern: './karma-test-shim.js', watched: true }
11.      ],
12.      preprocessors: {
13.        './karma-test-shim.js': ['webpack', 'sourcemap'],
14.      },
15.      webpack: webpackConfig,
16.      webpackMiddleware: {
17.        stats: 'errors-only'
18.      },
19.      webpackServer: {
20.        noInfo: true
21.      },
22.      // กำหนดการออก Report
23.      coverageIstanbulReporter: {
24.        reports: ['html', 'lcovonly'],
25.        fixWebpackSourcePaths: true
26.      },
27.      // กำหนดการออก Report
28.      reporters: config.coverage ? ['kjhtml', 'dots', 'coverage-istanbul'] : ['kjhtml', 'dots'],
29.      port: 9876,
30.      colors: true,
```

```
31.   logLevel: config.LOG_INFO,  
32.   autoWatch: true,  
33.   browsers: ['Chrome'],  
34.   singleRun: false  
35. };  
36.  
37. config.set(_config);  
38. };
```

Step 3

8) สร้างไฟล์ karma-test-shim.js ไว้สำหรับระบบที่อยู่ของไฟล์ที่ต้องการทดสอบ

```
1. Error.stackTraceLimit = Infinity;  
2. require('core-js/es6');  
3. require('core-js/es7/reflect');  
4. require('zone.js/dist/zone');  
5. require('zone.js/dist/long-stack-trace-zone');  
6. require('zone.js/dist/proxy');  
7. require('zone.js/dist/sync-test');  
8. require('zone.js/dist/jasmine-patch');  
9. require('zone.js/dist/async-test');  
10. require('zone.js/dist/fake-async-test');  
11. // กำหนดที่อยู่ของไฟล์  
12. var appContext = require.context('./src', true, /\.spec\.ts/);  
13. appContext.keys().forEach(appContext);  
14. var testing = require('@angular/core/testing');  
15. var browser = require('@angular/platform-browser-dynamic/testing');  
16. testing.TestBed.initTestEnvironment(browser.BrowserDynamicTestingModule, browser.  
    platformBrowserDynamicTesting());
```

Step 4

9) สร้างไฟล์ webpack.test.js

```
1. var webpack = require('webpack');
2. var path = require('path');
3.
4. module.exports = {
5.   devtool: 'inline-source-map',
6.   resolve: {
7.     extensions: ['.ts', '.js']
8.   },
9.   module: {
10.    rules: [
11.      {
12.        test: /\.ts$/,
13.        loaders: [
14.          { loader: 'ts-loader' }, 'angular2-template-loader'
15.        ],
16.      },
17.      {
18.        test: /\.+\.ts$/,
19.        exclude: /(index.ts|mocks.ts|\.spec\.ts$)/,
20.        loader: 'istanbul-instrumenter-loader',
21.        enforce: 'post',
22.        query: {
23.          esModules: true
24.        },
25.      }
```

```

26.     test: /\.html$/,
27.     loader: 'html-loader?attrs=false'
28.   },
29.   {
30.     test: /\.(png|jpe?g|gif|svg|woff|woff2|ttf|eot|ico)$/,
31.     loader: 'null-loader'
32.   }
33. ],
34. plugins: [
35.   new webpack.ContextReplacementPlugin(
36.     // The (\\V) piece accounts for path separators in *nix and Windows
37.     /angular(\\V)core(\\V)(esm(\\V)src|src)(\\V)linker/,
38.     root('./src'), // location of your src
39.     {} // a map of your routes
40.   )
41. ];
42. function root(localPath) {
43.   return path.resolve(__dirname, localPath);
44. }
45.

```

10) เมื่อสร้างครบทุกไฟล์แล้วจะได้ไฟล์ทั้งหมดดังรูป

	karma.conf	9/18/2018 1:36 PM	JavaScript File	2 KB
	karma-test-shim	9/18/2018 1:38 PM	JavaScript File	1 KB
	webpack.test	9/18/2018 1:42 PM	JavaScript File	2 KB

ภาพที่ 4.33 ไฟล์ Config ทั้งหมด

Step 5

11) สร้าง Function Method เพื่อจำลองการทดสอบ

```
1. import { Component } from '@angular/core';
2. import { NavController } from 'ionic-angular';
3. @Component({
4.   selector: 'page-home',
5.   templateUrl: 'home.html'
6. })
7. export class HomePage {
8.   constructor(public navCtrl: NavController) {}
9.   cal(num1,num2,symbol) :number{
10.    if(symbol == "+")
11.    return parseInt(num1)+parseInt(num2);
12.    else if(symbol == "-")
13.    return parseInt(num1)-parseInt(num2);
14.    else if(symbol == "*")
15.    return parseInt(num1)*parseInt(num2);
16.    else if(symbol == "/")
17.    return parseInt(num1)/parseInt(num2);
18.  }
19. }
```

12) .สร้างไฟล์ Test Case ชื่อ **home.spec.ts** เพื่อทดสอบ Function , Method ให้อยู่ใน Folder เดียวกับ Class ที่ต้องการทดสอบ

```
1. import { NavController, IonicModule } from 'ionic-angular';
2. import { TestBed, ComponentFixture, async } from '@angular/core/testing';
```

```
3. import { HomePage } from './home';
4. describe('Create component & Test Method', () => {
5.   let comp: HomePage;
6.   let fixture: ComponentFixture<HomePage>;
7.   beforeEach(async () => {
8.     TestBed.configureTestingModule({
9.       // กำหนดชื่อ Class ที่จะทดสอบ
10.      declarations: [HomePage],
11.      // กำหนด Provider ให้เหมือน Class ที่จะทำการทดสอบ
12.      providers: [NavController],
13.      imports: [IonicModule.forRoot(HomePage)]
14.    }).compileComponents();
15.  });
16.  // สร้าง Component เพื่อง่ายต่อการเรียกใช้งาน
17.  beforeEach(() => {
18.    fixture = TestBed.createComponent(HomePage);
19.    comp = fixture.componentInstance;
20.  });
21.  // ตรวจสอบว่าการสร้าง Component สำเร็จหรือไม่
22.  it('should create component', () => {
23.    expect(comp instanceof HomePage).toBe(true);
24.  });
25.  // เรียก Class มาทำการทดสอบและทำการ expect เพื่อตรวจสอบผลลัพธ์
26.  it('Test Method 5 + 5 = 10', () => {
27.    expect(comp.cal(5, 5, "+")).toBe(10);
28.  });
29.  it('Test Method 10 - 5 = 5', () => {
30.    expect(comp.cal(10, 5, "-")).toBe(5);
31.  });
32.  it('Test Method 5 x 2 = 10', () => {
```

```

33.     expect(comp.cal(5,2,"*")).toBe(10);
34.   });
35.   it('Test Method 50 / 5 = 10', () => {
36.     expect(comp.cal(50,5,"/")).toBe(10);
37.   });
38. });

```

Step 6

13) แก้ไข file package.json โดยให้เพิ่ม Script

```

1. "test": "karma start ./test-config/karma.conf.js",
2. "test-ci": "karma start ./test-config/karma.conf.js --single-run",
3. "test-coverage": "karma start ./test-config/karma.conf.js --coverage"

```

14) เมื่อแก้ไขเสร็จจะได้ในดังรูปภาพ

```

"scripts": {
  "start": "ionic-app-scripts serve",
  "clean": "ionic-app-scripts clean",
  "build": "ionic-app-scripts build",
  "lint": "ionic-app-scripts lint",
  "test": "karma start ./test-config/karma.conf.js",
  "test-ci": "karma start ./test-config/karma.conf.js --single-run",
  "test-coverage": "karma start ./test-config/karma.conf.js --coverage",

```

ภาพที่ 4.34 ตัวอย่างที่แก้ไขในไฟล์ package.json

Step 7

- 15) พิมพ์คำสั่ง `npm run test` เพื่อเป็นการสั่งให้เริ่มทำการทดสอบ
- 16) เมื่อทดสอบเสร็จจะมีการแจ้งเตือนว่า Test Case ใดผ่านไม่ผ่าน

```
19 09 2018 11:09:49.135:INFO [karma]: Karma v3.0.0 server started at http://0.0.0.0:9876/
19 09 2018 11:09:49.135:INFO [launcher]: Launching browser Chrome with unlimited concurrency
19 09 2018 11:09:49.284:INFO [launcher]: Starting browser Chrome
19 09 2018 11:09:51.794:INFO [Chrome 69.0.3497 (Windows 10 0.0.0)]: Connected on socket V28D6c6PAovPEiMmAAAA with id 10090251
.....
Chrome 69.0.3497 (Windows 10 0.0.0): Executed 5 of 5 SUCCESS (1.035 secs / 2.814 secs)
TOTAL: 5 SUCCESS
```

ภาพที่ 4.35 ผลการทดสอบ

TNI

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

การพัฒนาโมบายแอปพลิเคชันการจองระบบขนส่งสินค้าที่พัฒนาด้วยไอโอไนคเฟรมเวิร์กเวอร์ชัน 2 เกิดมาจากการต้องการของทางบริษัท เอ-โฮสต์ ที่จะสร้างโมบายแอปพลิเคชันที่พัฒนาด้วยไอโอไนคเฟรมเวิร์ก เนื่องจากไอโอไนคเฟรมเวิร์กเวอร์ชัน 2 มีประสิทธิภาพที่ดีกว่า และรองรับเทคโนโลยีมากกว่า เพื่อช่วยลูกค้าได้รับประสบการณ์ที่ดีในการใช้งานแอปพลิเคชัน

ในการพัฒนาแอปพลิเคชันในตอนแรกนั้น จะต้องเริ่มจากการสร้างเทมเพลตโดยอ้างอิงจากแอปพลิเคชันที่ถูกพัฒนาด้วยไอโอไนคเฟรมเวิร์กเวอร์ชัน 2 ดังนั้นในช่วงแรกจึงต้องทำความรู้จักเกี่ยวกับระบบโครงสร้างต่างๆในการทำงาน เพื่อให้ง่ายต่อการพัฒนา รวดเร็วและมีประสิทธิภาพ โดยมีทีมที่ช่วยสอนและช่วยพัฒนาร่วมกันจนเสร็จ จากนั้นจึงเริ่มพัฒนาในส่วนของฟังก์ชันการทำงานต่างๆ เพื่อให้แอปพลิเคชันสามารถทำงานได้อย่างราบรื่น เช่น การเปิด-ปิดหน้าต่างๆ การร้องขอข้อมูลจากเซิร์ฟเวอร์เพื่อมาแสดง การสร้าง User Interface ของแอปพลิเคชัน นอกจากนี้ยังช่วยในการแก้ไขบั๊กของแอปพลิเคชันต่างๆที่ทำงานผิดพลาดหรือไม่สามารถทำงานได้ตามเป้าหมาย เช่น การกลับไปยังหน้าหลักเมื่อทำการสั่งซื้อสินค้าหรือบริการเสร็จ จะต้องกลับไปหน้าหลักเพื่อป้องกันความผิดพลาดที่อาจเกิดขึ้น ถ้าผู้ใช้งานกดย้อนกลับไปยังขั้นตอนการสั่งซื้อสินค้าและบริการ และยังมีอีกหลายๆฟังก์ชันและปัญหาต่างๆ ที่ได้ทำการแก้ไขเป็นที่เรียบร้อยแล้ว โดยระหว่างที่มีการพัฒนาแอปพลิเคชันนั้น ก็ได้เข้าร่วมประชุมอัปเดตความคืบหน้าของแอปพลิเคชันเมื่อพัฒนาแอปพลิเคชันเสร็จเรียบร้อยแล้วจึงได้รับมอบหมายให้จัดทำเอกสารสรุปฟังก์ชันการทำงานของแอปพลิเคชันทั้งหมด และได้รับมอบหมายให้ศึกษาเทคโนโลยีที่จะเข้ามาช่วยในการทำทดสอบแอปพลิเคชันสำหรับแอปพลิเคชันที่ถูกพัฒนาด้วยไอโอไนคเฟรมเวิร์กและจัดทำคู่มือโปรแกรมเมอร์ และคู่มือผู้ใช้งาน เพื่อให้ผู้ที่มาพัฒนาต่อสามารถเรียนรู้เกี่ยวกับโครงสร้างและการทำงานได้ง่ายและรวดเร็ว

5.2 แนวทางการแก้ไขปัญหา

ปัญหาที่พบระหว่างการศึกษาและนำเสนอเทคโนโลยีใหม่เข้ามาประยุกต์ใช้ในการพัฒนาระบบนั้น จะเป็นเรื่องของเนื้อหาใหม่ทั้งหมด ซึ่งต้องใช้เวลาในการทำความเข้าใจในการศึกษา เพราะเป็นเทคโนโลยีที่ยังไม่เคยลงศึกษาและยังไม่มีการใช้งานเกิดขึ้นกับคนในทีม อีกทั้งยังมีความเตรียมการและศึกษาในการเขียนแอปพลิเคชันโดยใช้ ionic เวอร์ชัน 2 น้อยเกินไป การพัฒนาแอปพลิเคชันการจองระบบขนส่งสินค้า จึงใช้เวลานานในการคิดค้นและพัฒนาแอปพลิเคชันตัวนี้ ซึ่งวิธีการแก้ปัญหาคือ เรียนรู้ผ่านเว็บไซต์ที่มีผู้พัฒนาได้มาเขียนไว้ก่อนหน้านี้ หรือศึกษาจากเอกสารหลักของเรื่องนั้น ๆ และในกรณีเขียนแอปพลิเคชัน จะปรึกษาพี่เลี้ยง และเพื่อนที่ชำนาญด้านนี้ในการคิด การแก้ไขปัญหาที่เกิดขึ้นต่าง ๆ

5.3 ข้อเสนอแนะจากการดำเนินงาน

- 1) ควรมีความรู้และทักษะในด้านการค้นคว้าเทคโนโลยีใหม่ๆ เพื่อลดระยะเวลาในการดำเนินงานในแต่ละขั้นตอน
- 2) ในการทำงานเป็นต้องหาข้อมูลเพื่อเตรียมความพร้อมก่อนลงมือและต้องวางแผนให้การทำงานเสมอ

TNI

เอกสารอ้างอิง

Angular Team Developer, AngularJS2 [Online], Available: <https://angularjs.org/>

[12 มิถุนายน 2561]

Ionic Team, Ionic Documentation[Online], Available: <http://ionicframework.com/>

[13 มิถุนายน 2561]

Leif Wells,2560, Testing in Ionic[Online], Available:

<https://leifwells.github.io/2017/08/27/testing-in-ionic-configure-existing-projects-for-testing/>

[10 กรกฎาคม 2561]

Suranart Niamcome,2557,วิธีใช้ Jasmine ทำ TDD[Online], Available:

<http://www.siamhtml.com/tdd-using-jasmine/>

[11 กรกฎาคม 2561]

Somkiat,2560, JavaScript Testing มันมีอะไรบ้าง[Online], Available:

<http://www.somkiat.cc/library-for-javascript-testing/>

[11 กรกฎาคม 2561]

ประวัติผู้จัดทำโครงการงาน



ชื่อ – สกุล

นายภาคย์ ผดุงพัฒนดิษฐ์

วัน เดือน ปีเกิด

30 เมษายน พ.ศ. 2540

ประวัติการศึกษา

ระดับประถมศึกษา

โรงเรียนชัยจิตต์วิทยา

ระดับมัธยมศึกษา

โรงเรียนสารสิทธิ์พิทยาลัย

ระดับอุดมศึกษา

สถาบันเทคโนโลยีไทย – ญี่ปุ่น
ปีการศึกษา 2558 – ปัจจุบัน

ทุนการศึกษา

- ไม่มี -

ประวัติการฝึกอบรม

- ไม่มี -

ผลงานที่ได้รับการตีพิมพ์

- ไม่มี -

TNI