

การเตรียมเอกสารอัตโนมัติสำหรับการสร้างต้นแบบรวดเร็ว

นายนภจร วัฒนกุล



จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)

เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย

The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2558

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

AUTOMATED DOCUMENTATION FOR RAPID PROTOTYPING

Mr. Napajorn Wattanagul



A Thesis Submitted in Partial Fulfillment of the Requirements  
for the Degree of Master of Science Program in Software Engineering

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2015

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การเตรียมเอกสารอัตโนมัติสำหรับการสร้างต้นแบบ รวดเร็ว
โดย	นายณภจร วัฒนกุล
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร. ญาใจ ลิมปิยะกรณ์

---

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้บัณฑิตวิทยานิพนธ์ฉบับนี้เป็นส่วน  
หนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์  
(รองศาสตราจารย์ ดร. สุพจน์ เตชวรสินสกุล)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ  
(ผู้ช่วยศาสตราจารย์ ดร. สุกวี สิ้นธุภิญญา)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก  
(รองศาสตราจารย์ ดร. ญาใจ ลิมปิยะกรณ์)

..... กรรมการภายนอกมหาวิทยาลัย  
(อาจารย์ ดร. ภาสกร อภิรักษ์วรพินิต)

นภจร วัฒนกุล : การเตรียมเอกสารอัตโนมัติสำหรับการสร้างต้นแบบรวดเร็ว (AUTOMATED DOCUMENTATION FOR RAPID PROTOTYPING) อ.ที่ปรึกษา  
วิทยานิพนธ์หลัก: รศ. ดร. ญาใจ ลีมปิยะภรณ์, 68 หน้า.

การสร้างต้นแบบ เป็นเทคนิคหนึ่งที่ใช้ระหว่างกระบวนการพัฒนาเชิงวิวัฒนาการเพื่อทำความเข้าใจความต้องการของลูกค้า และลดความเสี่ยงของการกำหนดความต้องการที่ไม่ถูกต้อง การเปลี่ยนแปลงที่เกิดขึ้นจากความเข้าใจที่คลาดเคลื่อนในระยะแรกของวงจรการพัฒนาจะส่งผลให้ลดต้นทุนค่าใช้จ่ายได้มาก นอกจากนี้ เอกสารประกอบระบบต้นแบบสุดท้ายสามารถใช้อ้างอิงสำหรับการพัฒนาในระยะต่อไปได้ อย่างไรก็ตาม การทำเอกสารเป็นงานที่น่าเบื่อและสิ้นเปลืองทรัพยากร งานวิจัยนี้จึงได้นำเสนอระเบียบวิธีการสร้างเอกสารอัตโนมัติสำหรับต้นแบบรวดเร็ว ซึ่งข้อดีของต้นแบบรวดเร็ว คือ การที่สามารถสร้างส่วนต่อประสานอย่างรวดเร็วเพื่อให้ผู้ใช้สามารถทดสอบว่าระบบถูกต้องตรงตามความคาดหวัง การทำให้เกิดผลของแนวทางที่เสนอในงานนี้มีส่วนทำให้เกิดการปรับปรุงกระบวนการซอฟต์แวร์



ภาควิชา วิศวกรรมคอมพิวเตอร์

ลายมือชื่อนิสิต .....

สาขาวิชา วิศวกรรมซอฟต์แวร์

ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

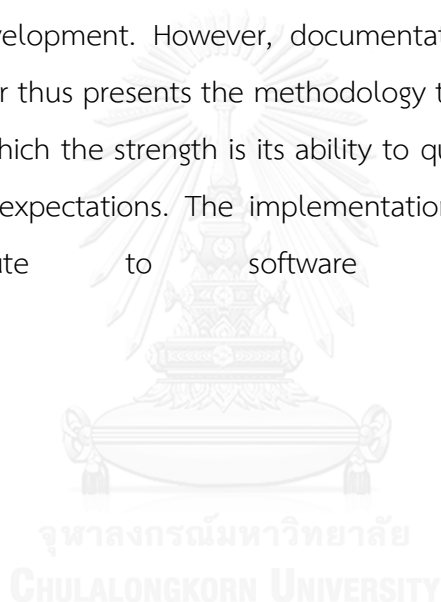
ปีการศึกษา 2558

# # 5770931821 : MAJOR SOFTWARE ENGINEERING

KEYWORDS: RAPID PROTOTYPING / DOCUMENTATION / DATA EXCHANGE / PROCESS IMPROVEMENT / MARKUP LANGUAGE

NAPAJORN WATTANAGUL: AUTOMATED DOCUMENTATION FOR RAPID PROTOTYPING. ADVISOR: ASSOC. PROF. YACHAI LIMPIYAKORN, Ph.D., 68 pp.

Prototyping is used as a technique during the evolutionary development process to understand the customer's requirements and reduce the risk of poorly defined requirements. Making changes early in the development lifecycle is extremely cost effective. The document accompanied with the final prototype will serve as the basis for further development. However, documentation is tedious and resource consuming. This paper thus presents the methodology to automate documenting the Rapid prototype of which the strength is its ability to quickly construct interfaces the users can test their expectations. The implementation of the proposed approach would contribute to software process improvement.



Department: Computer Engineering      Student's Signature .....

Field of Study: Software Engineering      Advisor's Signature .....

Academic Year: 2015

## กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความกรุณาของอาจารย์ที่ปรึกษาวิทยานิพนธ์ รองศาสตราจารย์ ดร.ญาใจ ลิ้มปิยะกรณ ได้สละเวลาให้ความรู้ คำปรึกษา คำแนะนำและตรวจทานแก้ไขข้อผิดพลาดต่างๆ ตลอดจนการกำกับดูแลและคอยติดตามความก้าวหน้า ทำให้การวิจัยนี้สำเร็จลุล่วงด้วยดี ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ผู้วิจัยใคร่ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.สุกรี สิ้นธุภิณู และ อาจารย์ ดร.ภาสกร อภิรักษ์วรพินิต กรรมการสอบวิทยานิพนธ์ ที่กรุณาเสียสละเวลาให้คำแนะนำ ตรวจสอบและแก้ไขวิทยานิพนธ์ฉบับนี้

ขอขอบพระคุณบิดา มารดา และญาติพี่น้องที่ให้การสนับสนุนและเป็นกำลังใจที่ดีให้เสมอมาและสนับสนุนด้านทุนทรัพย์ในการศึกษารวมไปถึงทุกท่านที่มีส่วนช่วยเหลือในการทำวิทยานิพนธ์ครั้งนี้ ซึ่งมีได้กล่าวนามในที่นี้

ท้ายที่สุด ผู้วิจัยขอขอบพระคุณเพื่อนร่วมงานทุกคน ที่คอยติดตามและให้กำลังใจ รวมถึงท่านอื่นๆ ที่มีได้กล่าวลงนามไว้ ณ ที่นี้ที่มีส่วนทำให้วิทยานิพนธ์สำเร็จลุล่วงไปได้ด้วยดี ผู้วิจัยหวังเป็นอย่างยิ่งว่าวิทยานิพนธ์ฉบับนี้จะเป็นประโยชน์บ้างไม่มากก็น้อยสำหรับผู้สนใจจะศึกษารายละเอียดต่อไป

## สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ญ
สารบัญรูป.....	ฎ
บทที่ 1 บทนำ .....	1
1.1 ความเป็นมาและความสำคัญของปัญหา .....	1
1.2 วัตถุประสงค์ของงานวิจัย.....	2
1.3 ขอบเขตการวิจัย.....	2
1.4 ขั้นตอนการวิจัย.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	3
1.6 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์ .....	3
1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์.....	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง .....	4
2.1 ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง.....	4
2.1.1 เอกซ์เอ็มแอล (eXtensible Markup Language, XML).....	4
2.1.2 เจสัน (Java Script Object Notation, JSON).....	4
2.1.3 โหนดเจเอส (Node.js) .....	5
2.1.4 แองกูลาร์ (Angular).....	6
2.2 งานวิจัยที่เกี่ยวข้อง.....	8
1) Comparative Study of Prototype Model for Software Engineering with System Development Life Cycle [7].....	8

2) Effective Prototyping for Usability[8] .....	8
บทที่ 3 วิเคราะห์และออกแบบระบบ .....	10
3.1 การเตรียมต้นแบบรวดเร็ว .....	14
3.2 การส่งออกต้นแบบรวดเร็ว .....	21
3.3 การอัปโหลดไปเว็บแอปพลิเคชัน .....	22
3.4 การสกัดข้อมูล .....	23
3.5 การสร้างเอกสารอธิบายต้นแบบ .....	25
บทที่ 4 การพัฒนาระบบ .....	28
4.1 เครื่องมือที่ใช้ในการพัฒนาและสภาพแวดล้อม .....	28
4.1.1 ฮาร์ดแวร์ .....	28
4.1.2 ซอฟต์แวร์ .....	28
4.1.3 ขั้นตอนการติดตั้งซอฟต์แวร์ .....	29
4.2 ขั้นตอนการพัฒนาระบบ .....	29
4.2.1 การเตรียมต้นแบบรวดเร็วเพื่อนำเข้า .....	30
4.2.2 การพัฒนาขั้นตอนวิธีการเตรียมข้อมูลเพื่อนำเข้า .....	31
4.2.3 การสกัดข้อมูลจากต้นแบบรวดเร็ว .....	34
4.2.4 การกำหนดคุณลักษณะ .....	37
4.2.5 การสร้างเอกสารอัตโนมัติ .....	42
บทที่ 5 การทดสอบและวิเคราะห์ผล .....	49
5.1 วัตถุประสงค์ของการทดสอบ .....	49
5.2 การทดสอบความถูกต้องของฟังก์ชันการทำงานของระบบ .....	49
5.3 สรุปผลการทดสอบ .....	56
บทที่ 6 สรุปผลการวิจัย .....	61



6.1 สรุปผลการวิจัย.....	61
6.2 ข้อจำกัดในงานวิจัย.....	61
6.3 แนวทางการวิจัยต่อในอนาคต.....	62
รายการอ้างอิง.....	63
ภาคผนวก การสร้างวินโดว์เนวิเกชันไดอะแกรม.....	65
ประวัติผู้เขียนวิทยานิพนธ์.....	68



## สารบัญตาราง

ตารางที่ 1 สัญลักษณ์ของเครื่องมือสแตนซิล สเกตซ์ จียูไอ เซพ .....	12
ตารางที่ 2 ส่วนประกอบหลักของหน้าล็อกอิน.....	14
ตารางที่ 3 ส่วนประกอบหลักของหน้าหลัก .....	15
ตารางที่ 4 ส่วนประกอบหลักที่สำคัญของหน้ารายละเอียดสินค้า .....	16
ตารางที่ 5 ส่วนประกอบหลักของหน้ารถเข็น .....	18
ตารางที่ 6 ส่วนประกอบหลักหน้าบัญชีผู้ใช้งาน.....	19
ตารางที่ 7 ตัวอย่างคุณลักษณะตามมาตรฐานของเอสทีเอ็มแอล .....	24
ตารางที่ 8 ตัวอย่างคุณลักษณะของกล่องข้อความที่ต้องการให้ผู้ใช้ใส่เพิ่มเติม.....	25
ตารางที่ 9 คำอธิบายคุณลักษณะของแม่แบบเอกสารอธิบายต้นแบบ .....	26
ตารางที่ 10 การแก้ไขเพิ่มข้อมูลพีเอ็นจีเพื่อเตรียมการนำเข้าข้อมูล.....	31
ตารางที่ 11 จำนวนส่วนประกอบในแต่ละเพจจากการส่งออกมาจากโปรแกรมไมโครซอฟต์วิซีโอ ...	35
ตารางที่ 12 ตารางอธิบายหน้าจอการเพิ่มคุณลักษณะ .....	39
ตารางที่ 13 ความหมายของการสร้างรูปแบบที่ใช้ในการสร้างแม่แบบ .....	43
ตารางที่ 14 ทดสอบการนำเข้าต้นแบบรวดเร็วในรูปแบบเพิ่มข้อมูลเอกซ์เอ็มแอล.....	50
ตารางที่ 15 ทดสอบการสกัดข้อมูลจากเพิ่มเอกซ์เอ็มแอล .....	50
ตารางที่ 16 ทดสอบการแปลงข้อมูลเป็นเจสัน.....	52
ตารางที่ 17 ทดสอบการกำหนดคุณลักษณะเพิ่มเติม .....	52
ตารางที่ 18 ทดสอบการสร้างเอกสารอัตโนมัติ .....	54
ตารางที่ 19 รายละเอียดสัดส่วนของส่วนประกอบของต้นแบบรวดเร็ว .....	56
ตารางที่ 20 ส่วนประกอบ Login Page .....	57
ตารางที่ 21 ส่วนประกอบ Home Page.....	57
ตารางที่ 22 ส่วนประกอบ ProductDetails Page.....	58

ตารางที่ 23 ส่วนประกอบ Cart Page..... 59

ตารางที่ 24 ส่วนประกอบ Account Page..... 60



## สารบัญรูป

รูปที่ 1	การทำงานของแบบจำลองขับเคลื่อนเหตุการณ์.....	6
รูปที่ 2	สถาปัตยกรรมของแองกูลาร์ 2 .....	7
รูปที่ 3	การพัฒนาต้นแบบรวดเร็วหรือต้นแบบโรอะเวย์ .....	10
รูปที่ 4	ขั้นตอนการจัดทำเอกสารจากต้นแบบรวดเร็วให้เป็นอัตโนมัติ.....	12
รูปที่ 5	หน้าจอล็อกอินที่สร้างจากเครื่องมือสเกตช์ จิยูไอ เชพ.....	14
รูปที่ 6	หน้าหลักที่สร้างจากเครื่องมือสเกตช์ จิยูไอ เชพ .....	15
รูปที่ 7	หน้ารายละเอียดของสินค้าที่สร้างจากเครื่องมือสเกตช์ จิยูไอ เชพ .....	17
รูปที่ 8	หน้ารถเข็นที่สร้างจากเครื่องมือสเกตช์ จิยูไอ เชพ .....	18
รูปที่ 9	หน้าบัญชีผู้ใช้งานที่สร้างจากเครื่องมือสเกตช์ จิยูไอ เชพ.....	19
รูปที่ 10	โครงสร้างเอกซ์เอ็มแอลของ Data.XML ไฟล์.....	21
รูปที่ 11	สถาปัตยกรรมของตัวสร้างคำอธิบายต้นแบบ .....	22
รูปที่ 12	แผนภาพกิจกรรมในการอัปโหลดและสกัดข้อมูล .....	23
รูปที่ 13	แผนภาพการสกัดข้อมูลจากไฟล์เอกซ์เอ็มแอล .....	24
รูปที่ 14	หน้าจอการกำหนดการตั้งค่าคุณลักษณะของกล่องข้อความ .....	25
รูปที่ 15	แม่แบบเอกสารอธิบายต้นแบบ .....	26
รูปที่ 16	รายละเอียดชื่อของเพจต้นแบบรวดเร็วจากโปรแกรมไมโครซอฟต์วิซีโอ .....	30
รูปที่ 17	การเตรียมข้อมูลเพื่อนำเข้าไปสกัด .....	30
รูปที่ 18	คำสั่งตรวจสอบการติดตั้งของโปรแกรม.....	31
รูปที่ 19	แพ็คเกจที่สำคัญที่ใช้ในการพัฒนาของแองกูลาร์ 2.....	32
รูปที่ 20	เพิ่มเอกสาร uploads ที่อยู่ภายใต้เพิ่มเอกสาร backend.....	32
รูปที่ 21	โค้ดในการพัฒนาฟังก์ชันอัปโหลด .....	33
รูปที่ 22	หน้าจอการอัปโหลดเพิ่มข้อมูลไปยังโหนดเจเอส.....	34

รูปที่ 23 โหนดเจเอสได้รับเพิ่มข้อมูลและทำการบันทึก .....	34
รูปที่ 24 ตัวอย่างโค้ดในการพัฒนาฟังก์ชันการสกัดข้อมูล.....	35
รูปที่ 25 ผลลัพธ์การทำงานของฟังก์ชันสกัดข้อมูล .....	36
รูปที่ 26 รูปแบบเจสันที่ผ่านฟังก์ชันการแปลง .....	37
รูปที่ 27 แอ่งกูลาร์คอมโพเนนท์ .....	37
รูปที่ 28 ตัวอย่างโค้ดในการพัฒนาคอมโพเนนท์ที่ใช้สำหรับเพิ่มคุณลักษณะ .....	38
รูปที่ 29 หน้าจอการกำหนดคุณลักษณะ .....	39
รูปที่ 30 กำหนดคุณลักษณะเพิ่มเติมให้กับส่วนประกอบ .....	40
รูปที่ 31 หน้าจอการกำหนดคุณลักษณะเมื่อทำการสำเร็จ .....	40
รูปที่ 32 ทำการบันทึกข้อมูลคุณลักษณะเพิ่มเติม .....	40
รูปที่ 33 การบันทึกข้อมูลเก็บลงในโลคอลสตอเรจ .....	41
รูปที่ 34 การตรวจสอบความถูกต้องของรูปแบบเจสันจาก Json Editor Online.....	41
รูปที่ 35 การกำหนดเอกสารแม่แบบด้วยโปรแกรมไมโครซอฟต์เวิร์ด.....	42
รูปที่ 36 ตัวอย่างโค้ดในการส่งข้อมูลไปทำการสร้างเอกสารต้นแบบ.....	44
รูปที่ 37 ข้อมูลที่แปลงเป็นเจสันเพื่อส่งไปสร้างเอกสารอธิบายต้นแบบ .....	45
รูปที่ 38 ตัวอย่างการกำหนดค่าและป้ายกำกับให้เอกสารแม่แบบ .....	46
รูปที่ 39 โค้ดในการพัฒนาฟังก์ชันการสร้างเอกสารอธิบายต้นแบบ .....	46
รูปที่ 40 เอกสารอธิบายต้นแบบถูกส่งมายังผู้ใช้งาน .....	47
รูปที่ 41 เอกสารอธิบายต้นแบบพร้อมข้อมูล .....	48
รูปที่ 42 กราฟสรุปรายละเอียดของส่วนประกอบ Login Page.....	56
รูปที่ 43 กราฟสรุปรายละเอียดของส่วนประกอบ Home Page .....	57
รูปที่ 44 กราฟสรุปรายละเอียดของส่วนประกอบ ProductDetails Page.....	58
รูปที่ 45 กราฟสรุปรายละเอียดของส่วนประกอบ Cart Page .....	58
รูปที่ 46 กราฟสรุปรายละเอียดของส่วนประกอบ Account Page .....	59

รูปที่ 47 โครงสร้างภาษาคอทที่ใช้ในการสร้างวินโดว์เนวิเกชัน .....	65
รูปที่ 48 คำสั่งในการแปลงเพิ่มข้อมูลคอทไปเป็นวินโดว์เนวิเกชันไดอะแกรม .....	65
รูปที่ 49 ระบบทำการสร้างวินโดว์เนวิเกชันไดอะแกรมโดยอัตโนมัติ .....	66
รูปที่ 50 วินโดว์เนวิเกชันไดอะแกรมแบบแนวตั้ง .....	66
รูปที่ 51 วินโดว์เนวิเกชันไดอะแกรมแบบแนวนอน .....	67



# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

การสร้างต้นแบบ (Prototyping) เป็นเทคนิคหนึ่งในการพัฒนาซอฟต์แวร์ในลักษณะพัฒนาเป็นวิวัฒนาการไปเรื่อยๆ เพื่อทำความเข้าใจถึงความต้องการของผู้ใช้ วิธีการดังกล่าวจะทำให้มีความเข้าใจมากยิ่งขึ้นในการกำหนดความต้องการระบบซอฟต์แวร์ อันเนื่องมาจากการตรวจสอบและข้อมูลป้อนกลับจากผู้ใช้ในแต่ละรอบการพัฒนาต้นแบบ เทคนิคดังกล่าวมีประสิทธิผลในการลดทรัพยากรงบประมาณและแรงงานที่ใช้ในโครงการ เนื่องจากสามารถแก้ไขข้อผิดพลาดตั้งแต่ช่วงระยะแรกๆของการพัฒนาซอฟต์แวร์ นอกจากนี้ ยังเพิ่มโอกาสที่ระบบจะสามารถนำไปใช้ได้จริงและสมเหตุสมผล

ต้นแบบ (Prototype) แบ่งออกเป็น 2 ประเภทหลักๆ คือ ต้นแบบที่มีวิวัฒนาการ (Evolutionary Prototype) และ ต้นแบบรวดเร็ว (Rapid Prototype) ความแตกต่างระหว่างต้นแบบที่มีวิวัฒนาการกับต้นแบบรวดเร็ว คือ ต้นแบบที่มีวิวัฒนาการเป็นระบบที่ทำงานได้จริง (Functional System) โดยทั่วไปส่วนมาก ผู้พัฒนาจะเริ่มต้นสร้างจากส่วนที่มีความเข้าใจชัดเจนหรือเป็นส่วนที่สำคัญก่อน แล้วจึงพัฒนาต่อแบบเพิ่ม (Incremental) ภายหลัง ในทางตรงกันข้ามวัตถุประสงค์การสร้างต้นแบบรวดเร็วจะไม่ใช้เพื่อเป็นระบบที่ทำงานได้จริง แต่ถูกสร้างขึ้นเพื่อทดสอบความเข้าใจที่ถูกต้องตรงกันระหว่างผู้ใช้กับทีมพัฒนา

โดยทั่วไป ขั้นตอนในการพัฒนาต้นแบบรวดเร็ว ประกอบด้วย 1) รวบรวมข้อกำหนดจากผู้ใช้งาน 2) วิเคราะห์ข้อกำหนดที่รวบรวมมา 3) พัฒนาต้นแบบ และ 4) ผู้ใช้ตรวจสอบต้นแบบ และให้ข้อมูลป้อนกลับถ้ามีส่วนที่ต้องการเพิ่มเติมหรือการเปลี่ยนแปลง ขั้นตอนที่ 3 และ 4 อาจถูกวนซ้ำจนกระทั่งได้ต้นแบบสุดท้าย ซึ่งจะถูกส่งต่อไปยังทีมพัฒนา เพื่อใช้เป็นพื้นฐานในการพัฒนาระยะต่อไป

อย่างไรก็ตาม การทำเอกสารจากต้นแบบรวดเร็วนั้น เป็นขั้นตอนที่สิ้นเปลืองทรัพยากรงานวิจัยนี้จึงนำเสนอวิธีการผลิตเอกสารอธิบายต้นแบบ (Prototype Description Document) ของต้นแบบรวดเร็วที่สร้างขึ้นโดยไมโครซอฟท์วิซิโอ (Microsoft Visio) [1] ทั้งนี้ เอกสารคำอธิบายต้นแบบสามารถใช้ประกอบเป็นส่วนหนึ่งของเอกสารข้อกำหนดความต้องการซอฟต์แวร์ (Software Requirements Specification— SRS) ที่ใช้อธิบายเหตุการณ์จำลองต่างๆ ที่จะเกิดขึ้นในแต่ละ

หน้าจอ ที่มีการจำลองว่ามีการนำเข้าสู่ข้อมูลอะไรบ้าง มีการนำข้อมูลที่ได้มาแสดงที่ไหนบ้าง มีการเชื่อมต่อไปยังที่ใดบ้าง เอกสารอธิบายต้นแบบยังมีประโยชน์ในอีกหลายๆ ด้าน เช่น ใช้ในการคิดคำนวณราคาซอฟต์แวร์โดยคิดจากหน้าจอที่ประกอบไปด้วยจำนวนอินพุตและเอาต์พุต ใช้อธิบายเหตุการณ์และสื่อสารกับทีมพัฒนา ในบางบริษัทเรียกว่า สตอรี (story)

## 1.2 วัตถุประสงค์ของงานวิจัย

เพื่อนำเสนอแนวทางและวิธีการสร้างเอกสารอัตโนมัติสำหรับต้นแบบรวดเร็ว สำหรับปรับปรุงกระบวนการซอฟต์แวร์

## 1.3 ขอบเขตการวิจัย

- 1) เครื่องมือที่ใช้ในการสร้างต้นแบบรวดเร็วต้องสร้างจากไมโครซอฟต์วิซิโอและเครื่องมือสแตนซิล สเกตซ์ จิยูไอ เซพ เท่านั้น
- 2) การนำเข้าสู่ต้นแบบจะต้องเป็นการส่งออกมาจากโปรแกรมไมโครซอฟต์วิซิโอในรูปแบบเอชทีเอ็มเท่านั้น
- 3) แฟ้มข้อมูลที่นำเข้าสู่จะต้องเป็นแฟ้มข้อมูลเอกซ์เอ็มแอลและแฟ้มข้อมูลพีเอ็นจีที่ได้มาจากข้อ 2 เท่านั้น
- 4) การนำเข้าสู่แฟ้มข้อมูลพีเอ็นจี จะต้องทำการเปลี่ยนชื่อหน้าจอให้ตรงกับต้นแบบที่ทำการสร้าง
- 5) แม่แบบที่ใช้ในการกำหนดเอกสารอธิบายต้นแบบจะต้องถูกกำหนดรูปแบบตามที่กำหนดและต้องเป็นไฟล์ .docx เท่านั้น
- 6) การระบุคุณลักษณะที่สำคัญเพิ่มไม่รองรับ HTML 5

## 1.4 ขั้นตอนการวิจัย

- 1) ศึกษาและทำความเข้าใจทฤษฎีและงานวิจัยที่เกี่ยวข้อง
- 2) ศึกษาการใช้เครื่องมือที่ใช้สำหรับการพัฒนาโปรแกรม
- 3) วิเคราะห์และกำหนดระเบียบวิธีวิจัย
- 4) พัฒนาระบบเว็บแอปพลิเคชัน



- 5) ทดสอบและประเมินผลงานวิจัย
- 6) สรุปผลงานวิจัย
- 7) ตีพิมพ์ผลงานทางวิชาการ
- 8) จัดทำวิทยานิพนธ์

### 1.5 ประโยชน์ที่คาดว่าจะได้รับ

ได้แนวทางอัตโนมัติและเครื่องมือสำหรับสร้างเอกสารจากต้นแบบรวดเร็ว

### 1.6 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์

เนื้อหาวิทยานิพนธ์ฉบับนี้แบ่งออกเป็น 6 บท โดยเริ่มจากบทที่ 1 บทนำจะกล่าวถึงความ เป็นมาและความสำคัญของปัญหา วัตถุประสงค์ของงานวิจัย ขอบเขตงานวิจัย ขั้นตอนการวิจัย และ ประโยชน์ที่คาดว่าจะได้รับ บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้องที่นำมาใช้ในงานวิจัยนี้ บทที่ 3 การ ออกแบบระบบ ขั้นตอนและวิธีการในการออกแบบระบบ บทที่ 4 การพัฒนาระบบ เครื่องมือที่ใช้ใน งานวิจัย การติดตั้งซอฟต์แวร์ และวิธีการพัฒนาระบบ บทที่ 5 การทดสอบระบบ ทดสอบการ ทำงานของระบบ และบทที่ 6 สรุปผลการวิจัย ข้อจำกัด และแนวทางสำหรับการทำวิจัยต่อในอนาคต

### 1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์

ส่วนหนึ่งของวิทยานิพนธ์ฉบับนี้ได้รับการตีพิมพ์เป็นบทความทางวิชาการในหัวข้อเรื่อง “AUTOMATED DOCUMENTATION FOR RAPID PROTOTYPING” โดย นภจร วัฒนกุล และ ฉญาใจ ลิ้มปิยะภรณ์ ในหนังสือรวมบทความการประชุมวิชาการนานาชาติ 2016 International Conference on Industrial Engineering, Management Science and Application (ICIMSA) ณ เกาะเชจู สาธารณรัฐเกาหลี วันที่ 25 พฤษภาคม 2559 หน้า 20 - 23

## บทที่ 2

### ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

#### 2.1 ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง

##### 2.1.1 เอกซ์เอ็มแอล (eXtensible Markup Language, XML)

เอกซ์เอ็มแอล เป็นตัวอธิบายการแสดงผลข้อมูลที่ถูกกำหนด เรียกว่า แฟ้มเอกสารเอกซ์เอ็มแอล (XML Document) และยังสามารถใช้อธิบายพฤติกรรมของโปรแกรมคอมพิวเตอร์ในการประมวลผล นอกจากนี้เอกซ์เอ็มแอลยังเป็น การประยุกต์ใช้โปรไฟล์ หรือ การจำกัดรูปแบบของเอสจีเอ็มแอล (Standard Generalized Markup Language, SGML) [2] โครงสร้างแฟ้มเอกสารเอกซ์เอ็มแอลจะถูกปรับเปลี่ยนให้เข้ากับเอกสารเอสจีเอ็มแอล โดยที่แฟ้มเอกสารเอกซ์เอ็มแอลสร้างขึ้นจากการจัดเก็บแต่ละส่วน เรียกว่า เอนทิตี (entities) ที่บรรจุไปด้วยข้อมูลที่ถูกระบุและไม่มีถูกระบุ ข้อมูลที่ถูกระบุประกอบไปด้วยคาแรคเตอร์ (character) ซึ่งบางส่วนในนั้นประกอบเป็นข้อมูลคุณลักษณะ และบางส่วนก็ประกอบเป็นมาร์กอัพ (markup) โดยมาร์กอัพจะเปลี่ยนคำอธิบายของรูปแบบเอกสารที่ถูกเก็บและโครงสร้างที่มีเหตุผล ดังนั้นเอกซ์เอ็มแอลจะจัดเตรียมกลไกเพื่อที่จะกำหนดข้อจำกัดในรูปแบบการจัดเก็บและโครงสร้างที่มีเหตุผล [3]

##### 2.1.2 เจสัน (Java Script Object Notation, JSON)

เจสัน คือ รูปแบบข้อมูลที่อยู่ในรูปของอ็อบเจกต์ (object) ที่มนุษย์สามารถอ่านเข้าใจและ (แก้ไขได้ง่าย เจสันถูกออกแบบมาให้มีขนาดเล็กเพื่อใช้แลกเปลี่ยนข้อมูลกับทางเซิร์ฟเวอร์ได้ง่ายและรวดเร็ว การใช้งานเจสันสามารถสร้างได้ 2 รูปแบบคือ 1) ชุดของข้อมูลที่เป็นการจัดเก็บของชื่อกับค่าข้อมูล ตัวอย่างรูปแบบของการจับคู่ { "name": "value" } เครื่องหมายปีกกาเปิด เป็นการบอกถึงตำแหน่งเริ่มต้นและจบด้วยปีกกาปิด ด้านในปีกกา คือ การจับคู่กันของชุดข้อมูล ชื่อ "name" และ ค่าข้อมูลคือ "value" โดยคั่นด้วยเครื่องหมายทวิภาค (:)

2) ค่าของข้อมูลสามารถใส่เป็นค่าอะเรย์ (array) ได้ รูปแบบการระบุค่าอะเรย์ คือ ใช้เครื่องหมายวงเล็บปีกกา ( [ ] ) ตัวอย่างการระบุค่าอะเรย์ให้กับค่าข้อมูล

```
{ "name": [ "value1", "value2", "value3" ] }
```

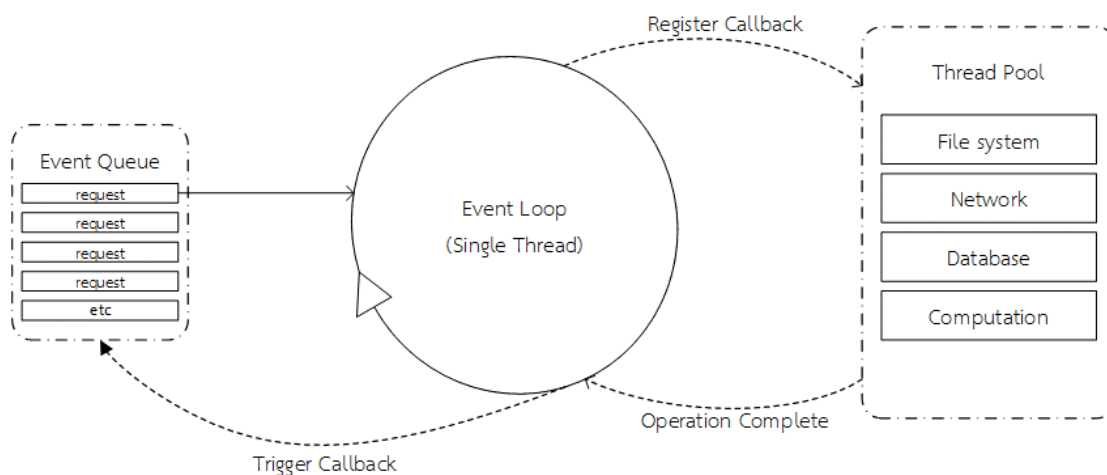
อีกทั้งยังสามารถใส่อ็อบเจกต์เข้าไปในค่าของอะเรย์ได้อีกด้วย ตัวอย่างการเพิ่มอ็อบเจกต์ในอะเรย์

```
{
  "name": [
    {"name1": "value1"},
    {"name2": "value2"},
    {"name3": "value3"}
  ]
}
```

จากตัวอย่างข้างบนด้านในวงเล็บปีกกาจะเก็บเป็นอะเรย์ที่ประกอบไปด้วยอ็อบเจกต์ “name1” “name2” และ “name3” การอ้างอิงถึงตำแหน่งของอะเรย์ก็สามารถที่จะอ้างอิงผ่านจากลำดับได้เลย ตัวอย่างการอ้างอิงถึงลำดับอะเรย์ name[0].name1 ดังนั้น เจสันจึงเป็นการรับส่งข้อมูลที่สะดวกและอ่านเข้าใจง่ายเหมาะจะนำมาใช้งานในการแลกเปลี่ยนข้อมูลกับทางเซิร์ฟเวอร์ได้เป็นอย่างดี

### 2.1.3 โหนดเจเอส (Node.js)

โหนดเจเอส คือ โปรแกรมเปิด (Open Source) ที่สามารถใช้ได้บนระบบปฏิบัติการ ลินุกซ์ (Linux), แมคโอเอส (Mac OS) และวินโดวส์ (Windows) โหนดเจเอสสร้างขึ้นโดยมี Non-Blocking I/O เป็นจุดเด่น การที่โหนดทำเช่นนี้ได้ ก็เพราะใช้แบบจำลองขับเคลื่อนเหตุการณ์ (Event-Driven Model) แบบอสมวาร (Asynchronous) คือการบอกให้ฟังก์ชันทำงาน เมื่อทำเสร็จแล้วค่อยมาบอกผล ทำให้หน่วยประมวลผล สามารถเอาเวลาไปทำงานให้ฟังก์ชันอื่นก่อนได้ จึงทำให้โหนดเจเอสคล่องตัวและมีประสิทธิภาพ [4]โหนดเจเอสเป็นหนึ่งในการพัฒนาที่มีความน่าสนใจมากในปัจจุบันและได้รับความนิยมเพิ่มมากขึ้นในฝั่งเซิร์ฟเวอร์จาวาสคริปต์ [5] ที่ทำให้สามารถเขียนจาวาสคริปต์ให้ทำงานบนฝั่งเซิร์ฟเวอร์ ได้และช่วยในการพัฒนาเว็บแอปพลิเคชัน ได้รวดเร็วขึ้นเนื่องจากมีแพ็คเกจให้เลือกใช้มากมายโดยไม่เสียค่าใช้จ่าย



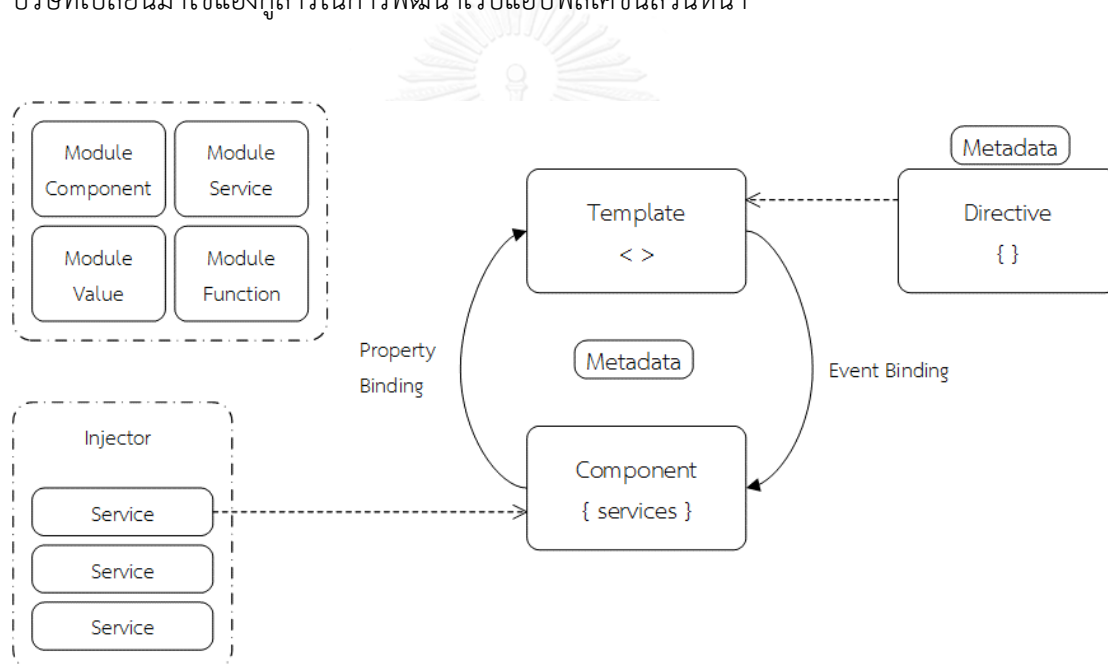
รูปที่ 1 การทำงานของแบบจำลองขับเคลื่อนเหตุการณ์

เริ่มจากการกระบวนการส่งการร้องขอ (Request) ออกไปยังเซิร์ฟเวอร์ ในอีเวนต์ลูป (Event Loop) จะมีซิงเกิลเธรด (Single Thread) คอยรับการร้องขอที่ถูกส่งเข้ามา โดยจะนำการร้องขอที่เข้ามานำไปลงทะเบียนฟังก์ชันการเรียกกลับ (Callback) ในเธรดพูล (Thread Pool) หลังจากลงทะเบียนเรียบร้อย ซิงเกิลเธรดจะไม่รอคอยผลลัพธ์แต่จะวนกลับไปรับการร้องขอถัดไปเพื่อไปนำการร้องขอนั้นมาลงทะเบียนฟังก์ชันเรียกกลับที่เธรดพูลใหม่ ซิงเกิลเธรดจะทำซ้ำแบบนี้วนไปเรื่อยๆ ในฝั่งเธรดพูลเมื่อปฏิบัติตามคำสั่งร้องขอเสร็จเรียบร้อยแล้วก็จะส่งสัญญาณกลับ (Trigger Callback) เพื่อจะส่งผลลัพธ์คืนกลับไปยังเจ้าของที่เป็นผู้ร้องขอนั้นมา ด้วยการทำงานที่ซิงเกิลเธรดไม่รอคอยผลลัพธ์จนเสร็จก่อนแต่กลับวิ่งวนซ้ำเพื่อไปรับการร้องขอใหม่แทน ทำให้ไม่เกิดการรอคอยในการร้องขอ ในทางกลับกันถ้าซิงเกิลเธรดรอคอยผลลัพธ์ บางผลลัพธ์ที่ใช้เวลาในการประมวลผลนานกว่าจะได้ผลลัพธ์ ระหว่างที่ซิงเกิลเธรดรอทำให้เกิดการรอกันของการร้องขอ เพราะการร้องขอถัดไปต้องรอการร้องขอก่อนหน้าเสร็จก่อน ทำให้การส่งผลลัพธ์จะเกิดความล่าช้าขึ้นในระบบ

#### 2.1.4 แองกูลาร์ (Angular)

แองกูลาร์ เป็นหนึ่งในกรอบงานจาวาสคริปต์ (JavaScript Framework) ที่ได้รับความนิยมในการใช้พัฒนาส่วนหน้า (Front-End Development) ซึ่งถูกพัฒนาโดยกูเกิ้ล (Google) แองกูลาร์จึงเป็นชุดเครื่องมือสำหรับกำหนดกรอบการทำงาน โดยเหมาะกับการพัฒนาเว็บแอปพลิเคชัน [6] นอกจากนี้ ยังจะช่วยลดปริมาณโค้ดจากจาวาสคริปต์ได้อย่างมาก ได้โค้ดที่ชัดเจนและ คงสภาพง่าย โดยสรุปแองกูลาร์เจเอส จะทำให้สามารถสร้างเว็บแอปพลิเคชันที่ซับซ้อนได้ ด้วยโค้ดที่น้อยและใช้เวลาพัฒนาน้อยลง ปัจจุบัน แองกูลาร์ได้พัฒนามาถึงเวอร์ชัน 2 โดยมีการพัฒนากรอบการทำงานที่เพิ่ม

ประสิทธิภาพมากยิ่งขึ้นเข้ามา หนึ่งในเทคโนโลยีหลักที่แองกูลาร์ 2 เพิ่มเข้ามาคือ เว็บ คอมโพเนนท์ (Web Component) ที่จะมองส่วนประกอบต่างๆ ในหน้าเว็บเพจเป็นคอมโพเนนท์ และหน้าเว็บเพจจะประกอบไปด้วยหลายๆ คอมโพเนนท์มาเรียงต่อกัน ในหนึ่งคอมโพเนนท์จะประกอบไปด้วยข้อมูลจำเพาะ 4 อย่างคือ 1) การกำหนดป้ายกำกับใหม่ (Custom Elements) คือ การสร้างป้ายกำกับใหม่ให้กับส่วนประกอบ 2) การนำเข้าของไฟล์เอชทีเอ็มแอล (HTML Imports) 3) การกำหนดแม่แบบ (Templates) เป็นการกำหนดป้ายกำกับข้อความที่จะใช้เป็นแม่แบบ 4) ซาโดว์ ดอม (Shadow DOM) คือ การกำหนดสไตล์ในเว็บเพจนั้นจะไม่ส่งผลกระทบต่อองค์ประกอบ ทำให้แน่ใจว่าคอมโพเนนท์จะไม่ถูกแก้ไขตัดแปลงจาก สไตล์ ของเว็บเพจ จากองค์ประกอบทั้ง 4 ส่วนของเว็บคอมโพเนนท์ทำให้แองกูลาร์ 2 เปลี่ยนแปลงการพัฒนาเว็บให้มีประสิทธิภาพมากขึ้น จึงทำให้มีหลายบริษัทเปลี่ยนมาใช้แองกูลาร์ในการพัฒนาเว็บแอปพลิเคชันส่วนหน้า



รูปที่ 2 สถาปัตยกรรมของแองกูลาร์ 2

รูปที่ 2 อธิบายการทำงานภาพรวมของแองกูลาร์ 2 เริ่มจากคอมโพเนนท์ที่ต้องการจะสร้าง โดยในคอมโพเนนท์จะประกอบไปด้วยส่วนหลักคือเทมเพลตและเซอร์วิส ซึ่งการพัฒนาเซอร์วิสนั้นคือการที่สามารถแบ่งให้คอมโพเนนท์อื่นสามารถเรียกใช้งานได้ โดยการฉีด (Inject) เข้าไปในคอมโพเนนท์ ก็จะทำให้คอมโพเนนท์นั้นสามารถเรียกใช้เซอร์วิสต่างๆ ได้ อีกส่วนคือเทมเพลตที่ถูกกำหนดให้กับคอมโพเนนท์นั้นๆ เทมเพลตจะถูกนำไปแสดงผลตามคำสั่งที่ระบุไว้ (Directive) และเมทาเดตา (Metadata) ที่ได้ระบุไว้ อีกทั้งคอมโพเนนท์มีการผูกค่าไว้กับเทมเพลต (Property Binding) เพื่อทำการแสดงผลของค่าตัวแปรหรือผลลัพธ์ที่ต้องการแสดง ในขณะที่ส่วนของเทมเพลตนั้นได้ผูกกับมายัง

คอมโพเนนต์ด้วยเหตุการณ์ (Event Binding) เช่น การกดปุ่มเพื่อเรียกใช้งานฟังก์ชันในคอมโพเนนต์เพื่อการคำนวณ ซึ่งฟังก์ชันการคำนวณที่อยู่ในคอมโพเนนต์อาจจะเป็นเซอร์วิสที่ถูกฉีดเข้ามาก็ได้ จากที่ได้กล่าวมา จะเห็นว่าแองกูลาร์พยายามแยกส่วนในการพัฒนาออกเป็นส่วนๆ แล้วนำมาประกอบกัน โดยแต่ละคอมโพเนนต์ที่นำมาต่อกันจะถูกห่อหุ้มเอาไว้ เปรียบเสมือนกล่องดำที่มีการโยนอินพุตเข้ามา แล้วรับเอาต์พุตกลับไป

## 2.2 งานวิจัยที่เกี่ยวข้อง

### 1) Comparative Study of Prototype Model for Software Engineering with System Development Life Cycle [7].

งานวิจัยนี้เน้นไปที่การเปรียบเทียบการวิเคราะห์วงจรการพัฒนาซอฟต์แวร์ (Software Development Life Cycle Models) ของแบบจำลองวอเตอร์ฟอล (Waterfall Model), แบบจำลองตัวต้นแบบ (Prototype Model) และแบบจำลองอาร์เอที (RAD Model) จากการวิเคราะห์พบว่าแต่ละแบบจำลองมีข้อดี ข้อเสียแตกต่างกัน โดยกล่าวถึงแบบจำลองตัวต้นแบบว่าให้ความสำคัญไปที่การออกแบบซอฟต์แวร์ มากกว่าการทำเอกสาร จึงอาจทำให้แบบจำลองชนิดนี้สามารถปล่อยซอฟต์แวร์ออกมาได้เร็วกว่าแบบจำลองอื่นๆ นอกจากนั้นแบบจำลองประเภทนี้ยังต้องอาศัยความร่วมมือจากผู้ใช้งานมากขึ้นในการมีส่วนร่วม โดยเฉพาะในส่วนของการตอบสนอง (Feedback) และข้อมูลจำเพาะ (Specification) เพื่อป้องกันการเกิดความเข้าใจผิดในขั้นตอนการออกแบบ เพื่อให้ในที่สุดได้ตัวต้นแบบสุดท้ายที่ผู้ใช้ต้องการไม่ว่าจะเป็นเรื่องรูปแบบ, ความรู้สึกและสมรรถนะการทำงาน และยิ่งเพิ่มเติมอีกว่าเมื่อมีการขยายขององค์กรทำให้กระบวนการทำงานในทุกขั้นตอนต้องการ ความเป็นอัตโนมัติมากขึ้น เพื่อให้เกิดการทำงานที่ง่ายยิ่งขึ้น ผู้วิจัยจึงมีแนวคิดในการนำแบบจำลองตัวต้นแบบมาใช้ในงานวิจัยฉบับนี้ เพื่อที่จะได้ตัวต้นแบบสุดท้ายที่ผู้ใช้ต้องการอย่างแท้จริง

### 2) Effective Prototyping for Usability[8] .

งานวิจัยนี้ได้อภิปรายและทบทวนการใช้เทคนิคของการสร้างตัวต้นแบบประเภทต่างๆ รวมถึงข้อดี ข้อเสียในการใช้งานเครื่องมือที่มีอยู่ จุดประสงค์ก็เพื่อดูว่าตัวต้นแบบใดสามารถเป็นตัวแทนและส่งผลกระทบต่อตอบสนองมากกว่ากัน, ประเมินการใช้งานของแต่ละอินเตอร์เฟซ (Interface) และเพื่อเป็นแนวทางในการเพิ่มความกระจำของความต้อการทางซอฟต์แวร์ที่แตกต่างกัน จากงานวิจัยนี้มีการนำโปรแกรมไมโครซอฟท์วิซีโอมาเป็นเครื่องมือในการทำต้นแบบ ผู้วิจัยจึงมีแนวคิดในการนำ

โปรแกรมไมโครซอฟท์วิซิโอมาเป็นเครื่องมือในการทำต้นแบบ เพื่อหวังผลในเรื่องของความหลากหลายของเมนูและเครื่องมือการทำงานจากโปรแกรมนี้

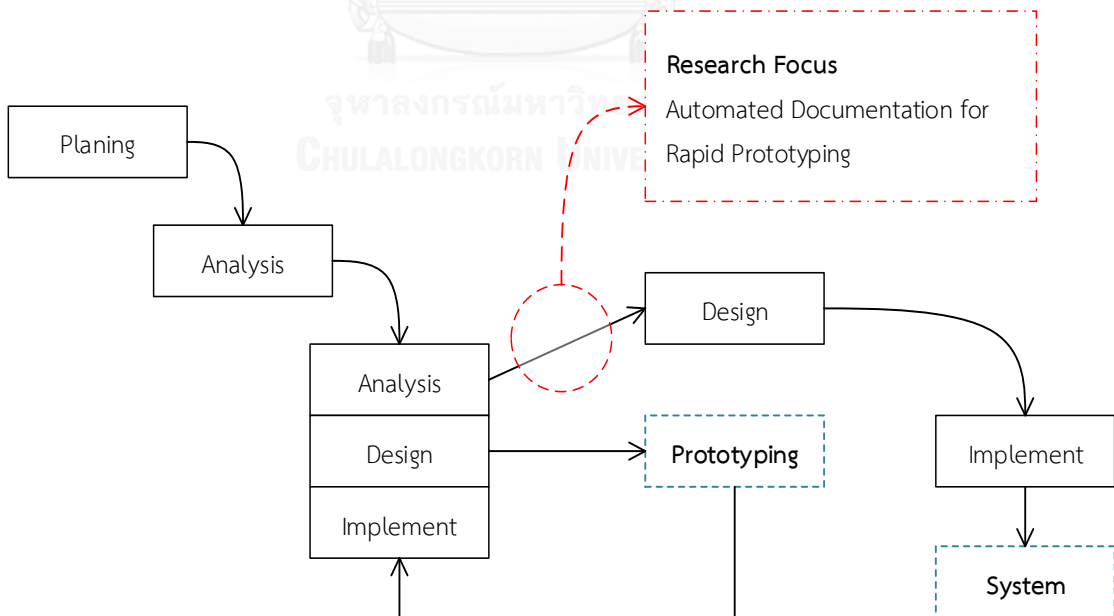
ผู้วิจัยได้ทำการศึกษางานวิจัยเพื่อให้ทราบถึงการพัฒนาตัวต้นแบบต่างๆ ที่มีประสิทธิภาพ กลับพบว่าขั้นตอนการพัฒนาต้นแบบทั้งหลายต้องการความรวดเร็วในการพัฒนาเพื่อที่จะได้ตรวจสอบคุณภาพของความต้องการได้ถูกต้อง ในขณะที่ตัวต้นแบบรวดเร็วนั้นสามารถพัฒนาได้เร็ว แต่กลับต้องเสียเวลาในการทำเอกสาร และตัวต้นแบบรวดเร็วไม่ได้ถูกนำมาใช้ประโยชน์ นอกเหนือจากตรวจสอบความต้องการ ผู้วิจัยจึงมีความคิดที่จะนำตัวต้นแบบรวดเร็วกลับมาใช้ในกระบวนการสร้างเอกสารอัตโนมัติจากต้นแบบรวดเร็ว เพื่อให้ลดภาระการทำงานเอกสารที่น่าเบื่อให้น้อยลง



### บทที่ 3

#### วิเคราะห์และออกแบบระบบ

งานวิจัยนี้นำเสนอแนวทางการสร้างเอกสารอัตโนมัติจากต้นแบบรวดเร็วหรือต้นแบบ  
โธระะเวย์ที่ถูกพัฒนาขึ้นโดยไมโครซอฟท์วิซิโอ ซึ่งโปรแกรมนี้เป็นหนึ่งในเครื่องมือที่ถูกสร้างขึ้นมาเพื่อ  
ช่วยในการสร้างโฟลชาร์ต หรือไดอะแกรม ไมโครซอฟท์วิซิโอจึงถูกนำมาใช้อย่างแพร่หลาย เพราะมี  
องค์ประกอบกราฟฟิกที่หลากหลายและยังมีการควบคุมการวางรูปแบบที่ละเอียด ไมโครซอฟท์วิซิโอ  
ยังมีการจัดเตรียมแม่แบบสัญลักษณ์ (Shape Template) หลากหลายรูปแบบให้เลือกใช้ในการวาง  
รูปแบบ นอกจากนี้ ยังมีอินเตอร์เฟซที่ควบคุมการนำเสนอได้หลากหลายส่วน เช่น วินโดว์ (Windows)  
และ เอสทีเอ็มแอล (HTML) ทำให้ผู้ใช้สามารถสร้างรูปภาพในรูปแบบที่ต้องการ รวมถึงการสร้างเว็บ  
เพจ (Web Pages) [9] จุดแข็งของไมโครซอฟท์วิซิโออีกข้อหนึ่งที่เป็นส่วนสำคัญในการถูกเลือกมาใช้  
ในงานวิจัยนี้ คือการง่ายต่อการนำไปใช้งานร่วมกับซอฟต์แวร์ไมโครซอฟท์ออฟฟิศ (Microsoft Office  
Software)



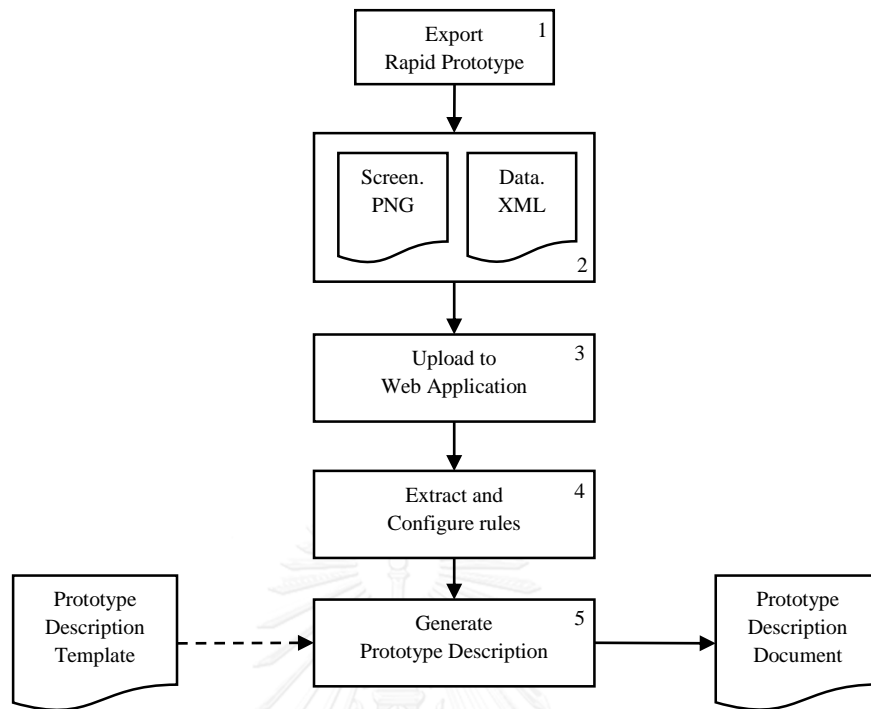
รูปที่ 3 การพัฒนาต้นแบบรวดเร็วหรือต้นแบบโธระะเวย์



แนวทางในการพัฒนาต้นแบบรวดเร็วจะมีวงจรในการพัฒนาดังรูปที่ 3 ที่จะเริ่มต้นจากการวางแผนงาน วิเคราะห์ความต้องการของผู้ใช้งาน จากนั้นจะเริ่มทำในรอบของการพัฒนาต้นแบบ โดยเริ่มจาก วิเคราะห์ ออกแบบ พัฒนาต้นแบบ นำต้นแบบไปตรวจสอบว่าตรงความต้องการของผู้ใช้งานหรือไม่ เมื่อได้รับผลป้อนกลับก็นำผลที่ได้มาปรับปรุง แล้วก็เริ่มวนซ้ำไปเรื่อยๆ จนกระทั่งได้ต้นแบบสุดท้ายที่พร้อมจะส่งมอบงานต่อให้ทีมอื่นเพื่อพัฒนาต่อไป งานวิจัยนี้จะมุ่งเน้นตั้งแต่ขั้นตอนที่ได้ต้นแบบสุดท้ายมา เพื่อที่จะนำต้นแบบสุดท้ายไปเตรียมการสร้างเอกสารอัตโนมัติสำหรับต้นแบบรวดเร็วต่อไป

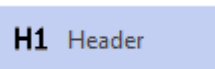
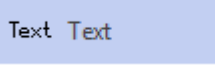
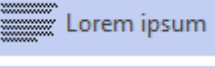
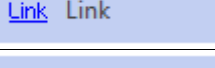
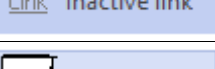
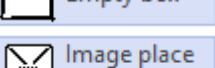
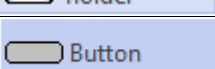

รูปที่ 4 แสดงถึง 5 ขั้นตอนที่น่าเสนอการสร้างเอกสารอัตโนมัติจากต้นแบบรวดเร็ว เริ่มต้นจากการขั้นตอนที่ 1) เตรียมต้นแบบสุดท้ายที่จะนำมาทำการสกัดข้อมูล 2) ผลที่ได้จากการส่งออกต้นแบบรวดเร็วจะได้ออกมาเป็นแฟ้มข้อมูล 2 รูปแบบ คือ แฟ้มข้อมูลเอกซ์เอ็มแอล (XML) และแฟ้มข้อมูลพีเอ็นจี (PNG) โดยไฟล์พีเอ็นจีจะให้ข้อมูลเกี่ยวกับการออกแบบหน้าจอ (Screen Layouts) ในขณะที่แฟ้มข้อมูลเอกซ์เอ็มแอล จะให้ข้อมูลเกี่ยวกับองค์ประกอบของหน้าจอ (Screen Elements) ยกตัวอย่างเช่น กล่องข้อความ (Textbox) ตาราง (Table) ป้าย (Label) ปุ่ม (Button) รูปภาพ (Image) และการเชื่อมโยง (Hyperlink) 3) พัฒนาระบบในการอัปโหลดแฟ้มข้อมูลที่ได้มาจากข้อ 2 เพื่อนำไปสกัดข้อมูล แต่เนื่องจากข้อมูลเกี่ยวกับองค์ประกอบของหน้าจอยังมีไม่เพียงพอ จึงทำให้ต้องมีการพัฒนาเว็บแอปพลิเคชัน สำหรับการสกัดและการเตรียมข้อมูลที่จะนำไปจัดทำเอกสารแบบอัตโนมัติ 4) ทำการสกัดป้ายกำกับข้อมูลออกมาจากแฟ้มข้อมูลเอกซ์เอ็มแอล หลังจากสกัดข้อมูลระบบจะอนุญาตให้ใส่คุณลักษณะเพิ่มเติม (Additional Attributes) ที่ใช้สำคัญสำหรับใช้ในการสื่อสาร ซึ่งอ้างอิงจากคุณลักษณะมาตรฐานของเอชทีเอ็มแอล 5) ระบบทำการนำข้อมูลที่ผู้ใช้ป้อนเข้านำมาเขียนลงในเอกสารแม่แบบ (Template) เพื่อทำการสร้างเอกสารอธิบายต้นแบบที่พร้อมส่งให้ผู้ใช้งานได้นำไปใช้งาน




ซอฟต์แวร์ที่ใช้สร้างต้นแบบรวดเร็วในงานวิจัยนี้ คือ โมโครซอฟท์วิซิโอ และเครื่องมือสแตนซิล (Stencil Tool) ชื่อว่า “สเกตช์ จียูไอ शेพ” (Sketch GUI Shapes) สามารถดูรูปเครื่องมือนี้ได้ดังแสดงในตารางที่ 1 และสามารถดูตัวอย่างของต้นแบบรวดเร็วที่สร้างได้จาก รูปที่ 4



รูปที่ 4 ขั้นตอนการจัดทำเอกสารจากต้นแบบรวดเร็วให้เป็นอัตโนมัติ

ตารางที่ 1 สัญลักษณ์ของเครื่องมือสแตนด์ลิต สเกตซ์ จียูไอ เซพ

#	Sketch GUI Shapes	คำอธิบาย
1	 H1 Header	แสดงหัวข้อของเนื้อหา
2	 Text Text	แสดงข้อความหรือใช้เพื่อติดป้ายกำกับ
3	 Lorem ipsum	แสดงตัวอย่างของเนื้อหา
4	 Link Link	ใช้ในการเชื่อมต่อไปยังเพจอื่น หรือเพื่อบอกว่าการเชื่อมต่อ
5	 Link Inactive link	ใช้เพื่อบอกว่าการเชื่อมต่อนี้ไม่สามารถใช้งานได้
6	 Empty box	ใช้ติกรอบเพื่อบอกการจัดกลุ่ม
7	 Image place holder	แสดงผลรูปภาพ
8	 Button	ใช้รองรับเหตุการณ์จากผู้ใช้งาน

9	 Check box	ใช้แสดงรายการแบบสามารถเลือกได้มากกว่า 1
10	 Check box selected	ใช้แสดงผลลัพธ์เพื่อบอกว่าสิ่งนี้ได้ถูกเลือก
11	 Radio button	ใช้แสดงรายการแบบเลือกได้เพียง 1
12	 Radio selected	ใช้แสดงผลลัพธ์เพื่อบอกว่าสิ่งนี้ได้ถูกเลือก
13	 Tab	ใช้แสดงผลที่ถูกแบ่งออกเป็นแต่ละส่วน
14	 Tab Selected	ใช้เพื่อบอกว่าการแสดงผลปัจจุบันอยู่ที่ส่วนนี้
15	 Horizontal scroll	ใช้เพื่อบอกว่ามีข้อมูลต่อเนื่องไปยังด้านขวา
16	 Vertical Scroll	แสดงให้รู้ว่ามีข้อมูลต่อเนื่องไปยังด้านล่าง
17	 Text field	ใช้เพื่อต้องการรับข้อมูลจากผู้ใช้งาน
18	 Combo box	ใช้เพื่อแสดงรายการที่สามารถเลือกได้
19	 Menu/Com... box drop-d...	แสดงรายการที่สามารถเลือกได้
20	 Menu item	แสดงรายการที่ให้บริการ
21	 Browser 800x600	ขนาดกรอบของเว็บแอปพลิเคชัน
22	 Portlet	ใช้แสดงผลต่างๆ
23	 Callout	ใช้แสดงข้อคิดเห็น
24	 Grid column	ใช้สร้างเป็นตารางเพื่อแสดงผล

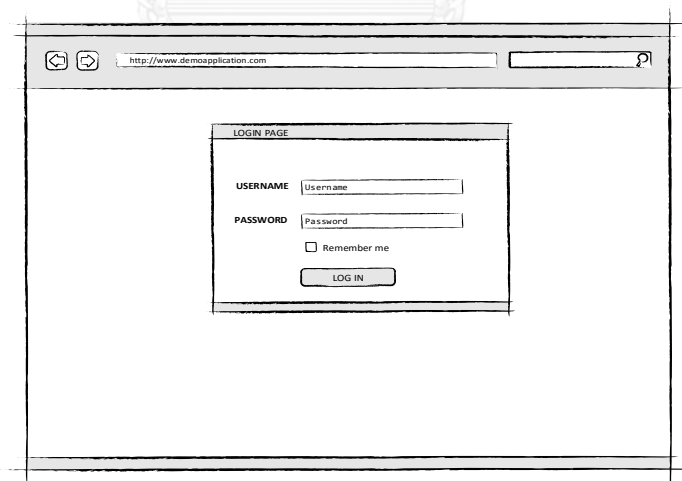
ตารางที่ 1 แสดงเครื่องมือทั้งหมดที่ใช้ในการพัฒนาต้นแบบ จากกลุ่มของชุดเครื่องมือสเกตซ์ จียูไอ เซพ สามารถแบ่งออกได้ 2 กลุ่มใหญ่ คือ กลุ่มที่หนึ่ง คือกลุ่มที่รับอินพุต กับกลุ่มที่สอง คือกลุ่มที่แสดงค่าข้อมูลต่างๆ เครื่องมือจากทั้งสองกลุ่มใหญ่ดังที่กล่าวมานับว่าเพียงพอต่อการใช้ในการพัฒนาต้นแบบรวดเร็วเพื่อสื่อสารกับทีมพัฒนา หรือการคิดคำนวณจากจำนวนอินพุตและเอาต์พุตในหนึ่งหน้าจอ ดังนั้นงานวิจัยชิ้นนี้จะมุ่งเน้นการเพิ่มคุณลักษณะที่จะใช้งานไปยังสองกลุ่มใหญ่เป็นหลัก โดยจะละเว้นคุณลักษณะของเครื่องมือประเภทตกแต่งให้สวยงาม เช่น กรอบหน้าจอ เส้นแบ่ง แนวนอน แถบเลื่อนหน้าจอ เป็นต้น

### 3.1 การเตรียมต้นแบบรวดเร็ว

ต้นแบบรวดเร็วจะถูกพัฒนานำเข้าตามกระบวนการที่อธิบายไว้แล้วข้างต้น จากการพัฒนาซ้ำหลายครั้งจะทำให้สามารถรวบรวมข้อมูลข้อกำหนดจากผู้ใช้ วิเคราะห์ สร้างตัวต้นแบบ ได้เป็นต้นแบบสุดท้าย และจะได้ออกมาเป็นแฟ้มข้อมูลผลผลิต 2 ชุดข้อมูลด้วยกัน ที่จะถูกส่งออกมาเพื่อสร้างเป็นเอกสารคำอธิบาย

จากวิธีวิจัยที่ได้นำเสนอไปแล้วนั้น มีการกำหนดข้อจำกัดในชนิดของเครื่องมือสแตนด์ลิตที่เลือกมาใช้ต้องอยู่ในส่วนของเครื่องมือสเกตช์ จียูไอ เซพ เท่านั้น เครื่องมือที่ถูกนำมาใช้ในงานวิจัยชิ้นนี้จะสนับสนุนการออกเอกสารอธิบายต้นแบบ โดยใช้ซินแทกซ์เฉพาะ (Specific Syntax) ของแฟ้มข้อมูลคอนฟิกเกอร์เช่นใน Data.XML เหตุผลที่เครื่องมือสเกตช์ จียูไอ เซพ ถูกเลือกมาใช้ เนื่องจากเครื่องมือนี้มีรูปแบบโครงสร้างที่หลากหลายที่จำเป็นต่อการพัฒนาเว็บแอปพลิเคชันและไวร์เฟรม (Wireframe)

งานวิจัยนี้ทำการจำลองสร้างแอปพลิเคชันต้นแบบรวดเร็วจากโปรแกรมไมโครซอฟต์วิซีโอ โดยประกอบไปด้วยหน้าล็อกอิน (Login page) หน้าเพจหลัก (Home Page) หน้ารายละเอียดสินค้า (Product Details Page) หน้ารถเข็น (Cart Page) และหน้าบัญชีผู้ใช้งาน (Account Page)



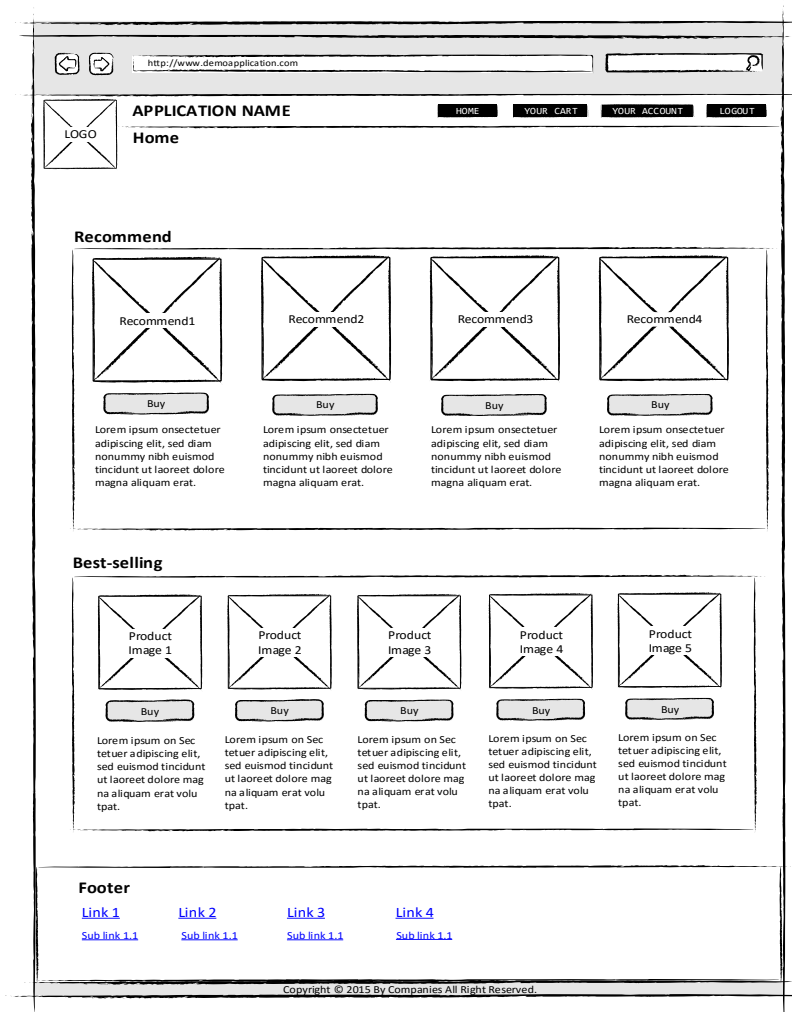
รูปที่ 5 หน้าจอล็อกอินที่สร้างจากเครื่องมือสเกตช์ จียูไอ เซพ

ตารางที่ 2 ส่วนประกอบหลักของหน้าล็อกอิน

ชื่อ	ประเภท	คำอธิบาย
Username	Text field	กล่องข้อความรองรับชื่อผู้ใช้งาน

Password	Text field	กล่องข้อความรองรับรหัสผ่าน
Remember	Check box	ฟังก์ชันจดจำการเข้าใช้งาน
Login	Button	ปุ่มกดทำการล็อกอินเข้าสู่ระบบ

หน้าล็อกอินดังรูปที่ 5 และส่วนประกอบหลักของหน้าล็อกอินตามตารางที่ 2 จะประกอบไปด้วย ป้ายกำกับข้อความ และมีกล่องข้อความ 2 กล่องเพื่อรองรับการอินพุตชื่อและรหัสผ่านจากผู้ใช้ และปุ่มล็อกอิน ที่จะทำการส่งข้อมูลอินพุตไปยังเซิร์ฟเวอร์เพื่อยืนยันตัวตนในการเข้าสู่ระบบ



รูปที่ 6 หน้าหลักที่สร้างจากเครื่องมือสเกตซ์ จียูไอ เชพ

ตารางที่ 3 ส่วนประกอบหลักของหน้าหลัก

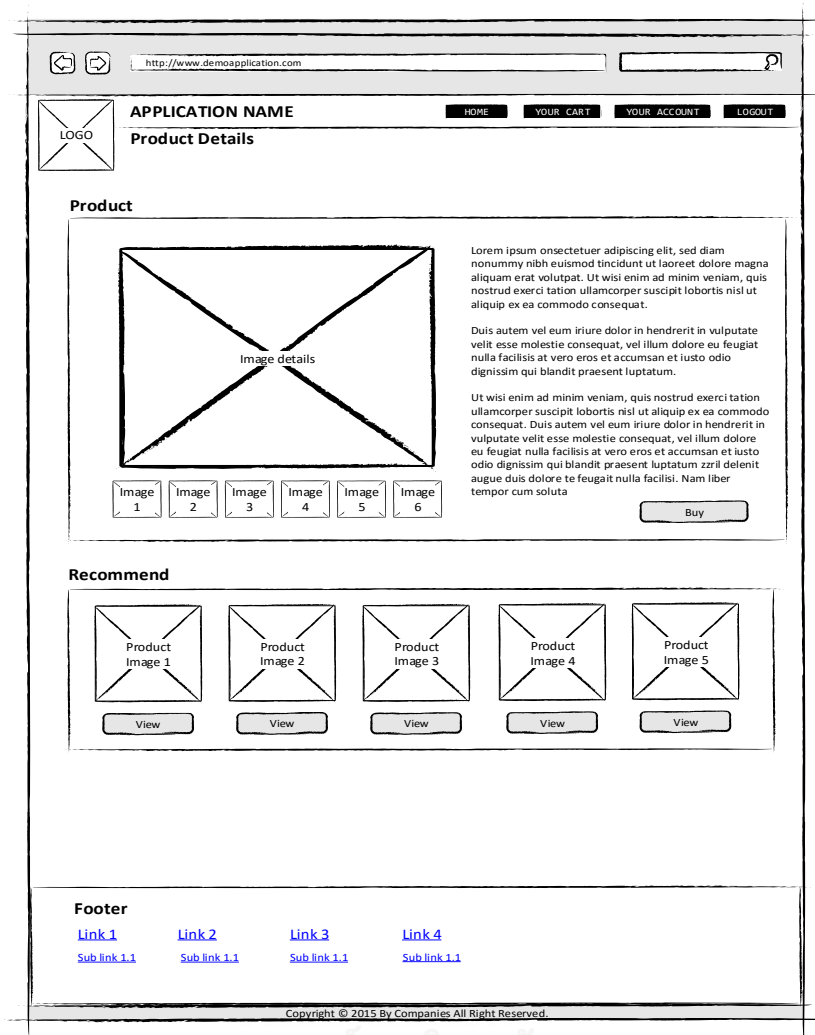
ชื่อ	ประเภท	คำอธิบาย
Logo	Image	แสดงรูปภาพ

Application Name	Text	แสดงข้อความ
Menu Home	Menu Item	เชื่อมต่อไปยังหน้าหลัก
Menu Cart	Menu Item	เชื่อมต่อไปยังหน้ารถเข็น
Menu Account	Menu Item	เชื่อมต่อไปยังหน้าบัญชีผู้ใช้
Menu Logout	Menu Item	ออกจากระบบ พร้อมกลับไปสู่หน้าล็อกอิน
Recommend Image	Image	แสดงรูปภาพ
Buy Recommend	Button	ปุ่มกดสั่งซื้อสินค้า
Recommend Details	Lorem ipsum	แสดงเนื้อหาที่เกี่ยวข้องกับรูปภาพ
Best Product Image	Image	แสดงรูปภาพ
Buy Best Product	Button	ปุ่มกดสั่งซื้อสินค้า
Best Product Details	Lorem ipsum	แสดงเนื้อหาที่เกี่ยวข้องกับรูปภาพ
Link Dummy	Link	จำลองการเชื่อมต่อไปยังภายนอก
Sub Link Dummy	Link	จำลองการเชื่อมต่อไปยังภายนอก

รูปที่ 6 หน้าหลักของแอปพลิเคชัน และตารางที่ 3 แสดงส่วนประกอบที่สำคัญของหน้าหลัก มีการแสดงรูปภาพของแอปพลิเคชันทางด้านบนมุมซ้าย รายการเมนูทางขวาบน โฉวรายการสินค้า รายการสินค้าที่แนะนำพร้อมรายละเอียดและปุ่มกดเพื่อซื้อ รายการสินค้าที่ขายดีและรายละเอียด พร้อมปุ่มกดเพื่อซื้อ และด้านล่างจะแสดงการเชื่อมต่อไปยังหน้าเพจอื่น

ตารางที่ 4 ส่วนประกอบหลักที่สำคัญของหน้ารายละเอียดสินค้า

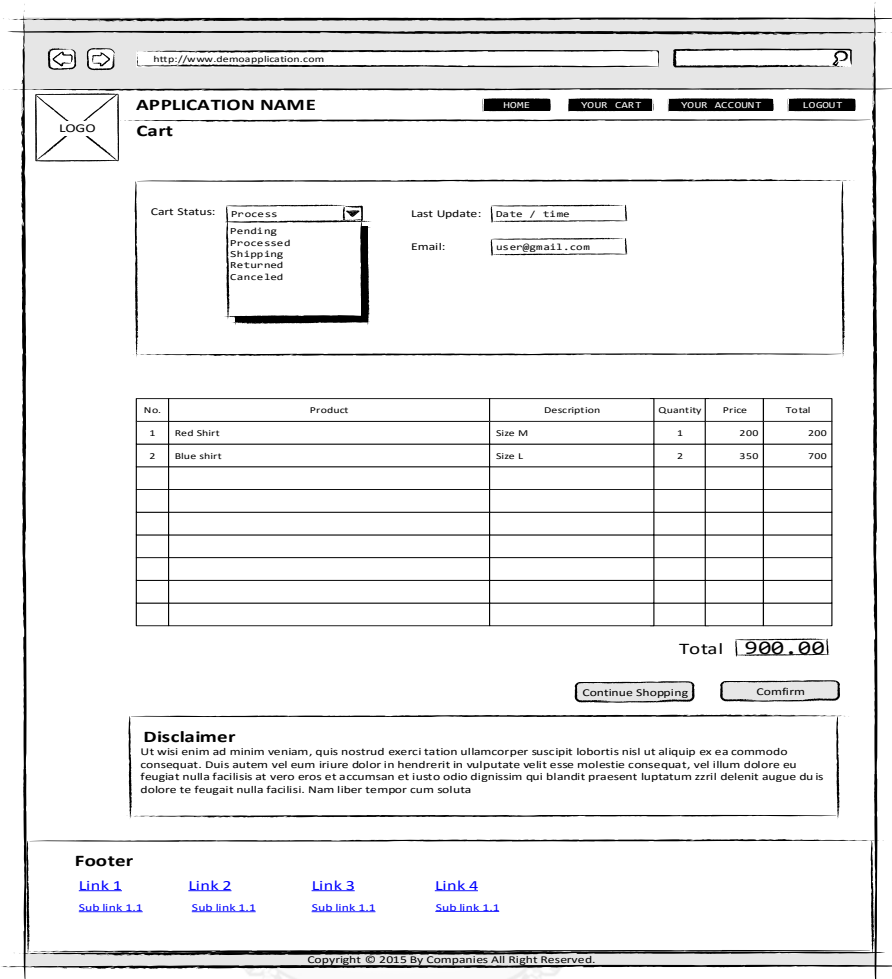
ชื่อ	ประเภท	คำอธิบาย
Product	Image	แสดงรูปภาพหลัก
Product Thumbnail	Image	แสดงรูปภาพรอง
Product details	Lorem ipsum	แสดงเนื้อหาที่เกี่ยวข้องกับรูปภาพ
Buy Product	Button	ปุ่มกดสั่งซื้อสินค้า
Product Recommend	Image	แสดงรูปภาพที่ได้รับการแนะนำ
View	Button	ปุ่มแสดงรายละเอียดสินค้า นำไปโชว์เป็นรูปภาพหลัก



รูปที่ 7 หน้ารายละเอียดของสินค้าที่สร้างจากเครื่องมือสเกตซ์ จียูไอ เซฟ

รูปที่ 7 และตารางที่ 4 แสดงหน้ารายละเอียดสินค้าจะแสดงรายละเอียดเนื้อหาของสินค้า และรูปภาพประกอบที่เยอะกว่าหน้าหลัก และมีปุ่มกดเพื่อทำรายการสั่งซื้อ และมีรายการสินค้าแนะนำไว้ที่ข้างล่างของรายการพร้อมปุ่มแสดงรายละเอียดเพิ่มเติม

รูปที่ 8 และตารางที่ 5 แสดงรายละเอียดสินค้าและส่วนประกอบหลัก ที่ถูกเลือกมาจากการกดปุ่มซื้อ โดยจะมีรายละเอียดของสินค้า จำนวนสินค้า ราคารวมของสินค้า และปุ่มยืนยันดำเนินการสั่งซื้อสินค้า หรือทำการเลือกสินค้าต่อไป



รูปที่ 8 หน้ารถเข็นที่สร้างจากเครื่องมือสเกตซ์ จียูไอ เชพ

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

ตารางที่ 5 ส่วนประกอบหลักของหน้ารถเข็น

ชื่อ	ประเภท	คำอธิบาย
Cart Status	Combo Box	แสดงรูปภาพหลัก
Last Update	Image	แสดงรูปภาพรอง
Email	Text field	แสดงเนื้อหาที่เกี่ยวข้องกับรูปภาพ
No.	Grid.1	ตารางที่ 1 แสดงหมายเลข
Product	Grid.2	ตารางที่ 2 แสดงชื่อสินค้า
Description	Grid.3	ตารางที่ 3 แสดงรายละเอียดสินค้า
Quantity	Grid.4	ตารางที่ 4 แสดงจำนวน
Price	Grid.5	ตารางที่ 5 แสดงราคา
Total	Grid.6	ตารางที่ 6 แสดงยอดรวม



Grand Total	Text	แสดงยอดรวมทั้งหมด
Continue Shopping	Button	เชื่อมโยงกลับไปหน้าแสดงสินค้า
Confirm	Button	ยืนยันการซื้อสินค้า
Disclaimer	Lorem ipsum	แสดงเนื้อหาที่ไม่รองรับการสั่งซื้อสินค้า

รูปที่ 9 หน้าบัญชีผู้ใช้งานที่สร้างจากเครื่องมือสเกตซ์ จียูไอ เซพ

ตารางที่ 6 ส่วนประกอบหลักหน้าบัญชีผู้ใช้งาน

ชื่อ	ประเภท	คำอธิบาย
Image Profile	Image	แสดงรูปภาพผู้ใช้งาน

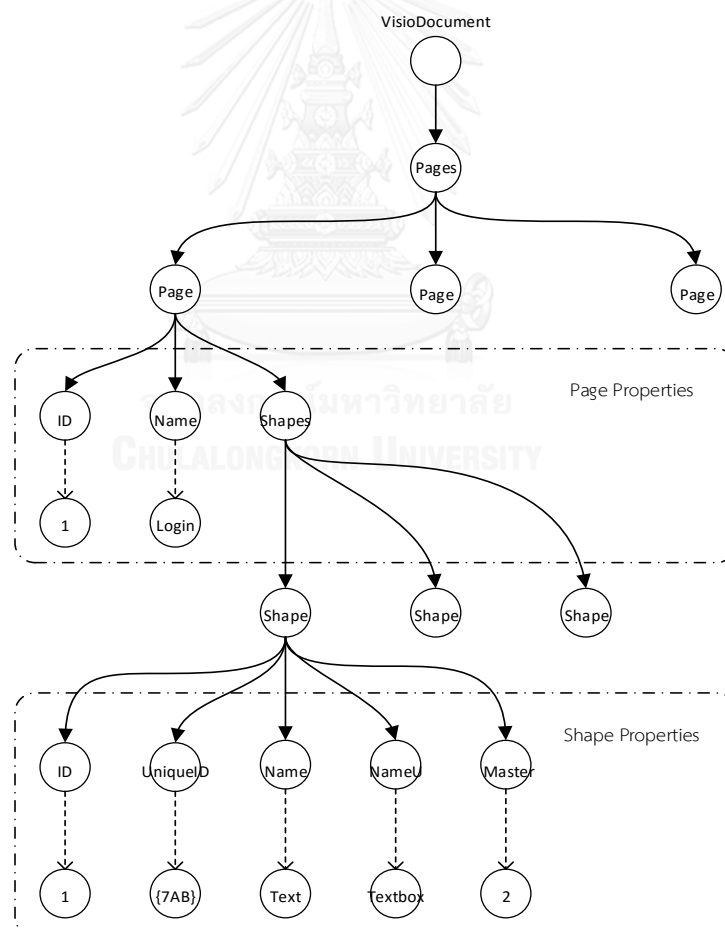
Name	Text field	แสดงชื่อผู้ใช้งาน
Gender	Combo Box	แสดงเพศ
Nationality	Combo Box	แสดงสัญชาติ
Age	Text	แสดงอายุ
Day	Combo Box	แสดงวันที่เกิด
Month	Combo Box	แสดงเดือนที่เกิด
Year	Combo Box	แสดงปีที่เกิด
Email	Text field	แสดงอีเมล
Mobile	Text field	แสดงเบอร์มือถือ
Telephone	Text field	แสดงเบอร์โทรศัพท์
Address 1	Text field	แสดงที่อยู่
Address 2	Text field	แสดงที่อยู่
City	Combo Box	แสดงเมือง
Country	Combo Box	แสดงประเทศ
Zip Code	Text field	แสดงรหัสไปรษณีย์
ID Card	Text field	แสดงรหัสบัตร
Visa / Master	Combo Box	ประเภทของบัตร
Expire date	Text field	วันหมดอายุของบัตร
Purchase History Date / Time	Text	ประวัติ วันที่และเวลาการซื้อสินค้า
Order id	Text	รหัสสินค้า
Product	Text	ชื่อสินค้า
Status	Text	สถานะสินค้า
Quantity	Text	จำนวน
Total Price	Text	ราคาทั้งหมด

รูปที่ 9 และ ตารางที่ 6 แสดงหน้าบัญชีผู้ใช้งานและส่วนประกอบหลัก ที่จะแสดงรายละเอียด ส่วนตัวของผู้ใช้งาน เช่น ข้อมูลพื้นฐานของผู้ใช้งาน ข้อมูลที่อยู่ที่ใช้ในการติดต่อ ข้อมูลในการชำระ สินค้าและที่อยู่ในการจัดส่งสินค้า ประวัติการสั่งซื้อสินค้า จากแอปพลิเคชันต้นแบบรวดเร็วที่ได้สร้าง

ขึ้นมา ผู้วิจัยจะสรุปว่าต้นแบบรวดเร็วที่ได้กล่าวมานี้เป็นต้นแบบสุดท้ายที่พร้อมจะนำไปอัปโหลดเข้าสู่ระบบที่พัฒนาขึ้นมาเพื่อสกัดข้อมูลและนำไปสร้างเอกสารคำอธิบายต้นแบบในขั้นต่อไป

### 3.2 การส่งออกต้นแบบรวดเร็ว

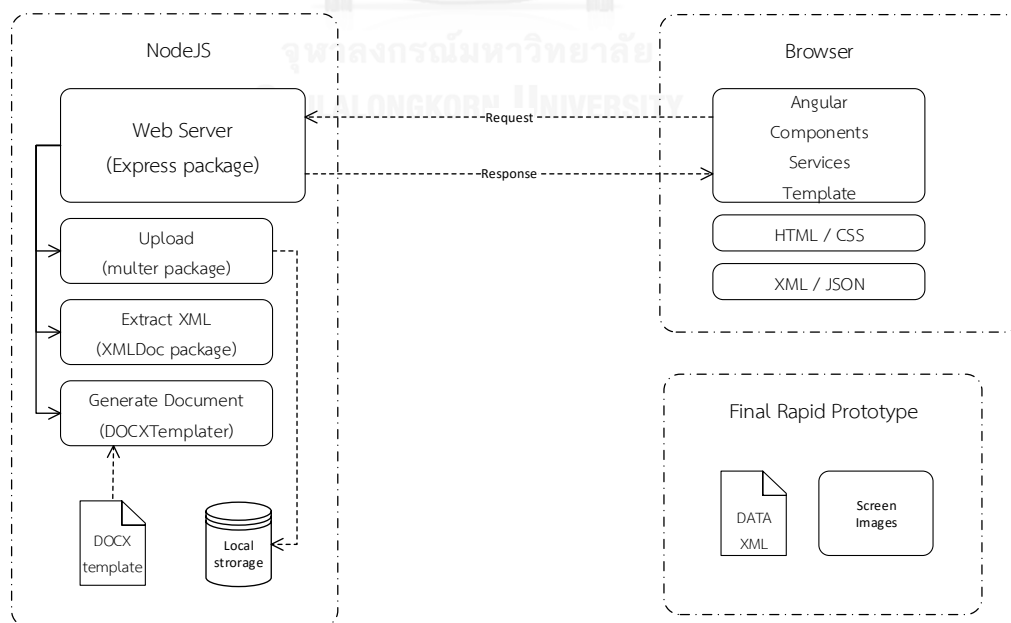
หลังจากที่ได้ต้นแบบรวดเร็วมาแล้ว ต้องทำการส่งออกต้นแบบรวดเร็วเพื่อนำไปสกัดข้อมูล โดยจะได้เพิ่มข้อมูลมา 2 รูปแบบดังที่ได้กล่าวมาแล้วข้างต้น รูปที่ 10 แสดงถึงโครงสร้างภายในของ Data.XML โดย รากของโหนดเป็นเอกสารวิซีโอ และมีโหนดลูก คือ Pages ใต้ป้ายกำกับ Pages จะประกอบไปด้วย Page ย่อยๆ ซึ่งจะมีคุณสมบัติที่จะสกัด คือ ID Name และ Shapes โดย Page จะประกอบไปด้วย Shape หลายๆ อัน และภายใต้ป้ายกำกับ Shapes จะประกอบไปด้วย Shape ที่เป็นส่วนประกอบใน Page นั้นๆ อีก โดยในแต่ละ Shape จะมีคุณสมบัติประกอบด้วย ID UniqueID Name NameU และ Master ซึ่งทั้งหมดนี้คือ คุณสมบัติที่จะสกัดออกมา



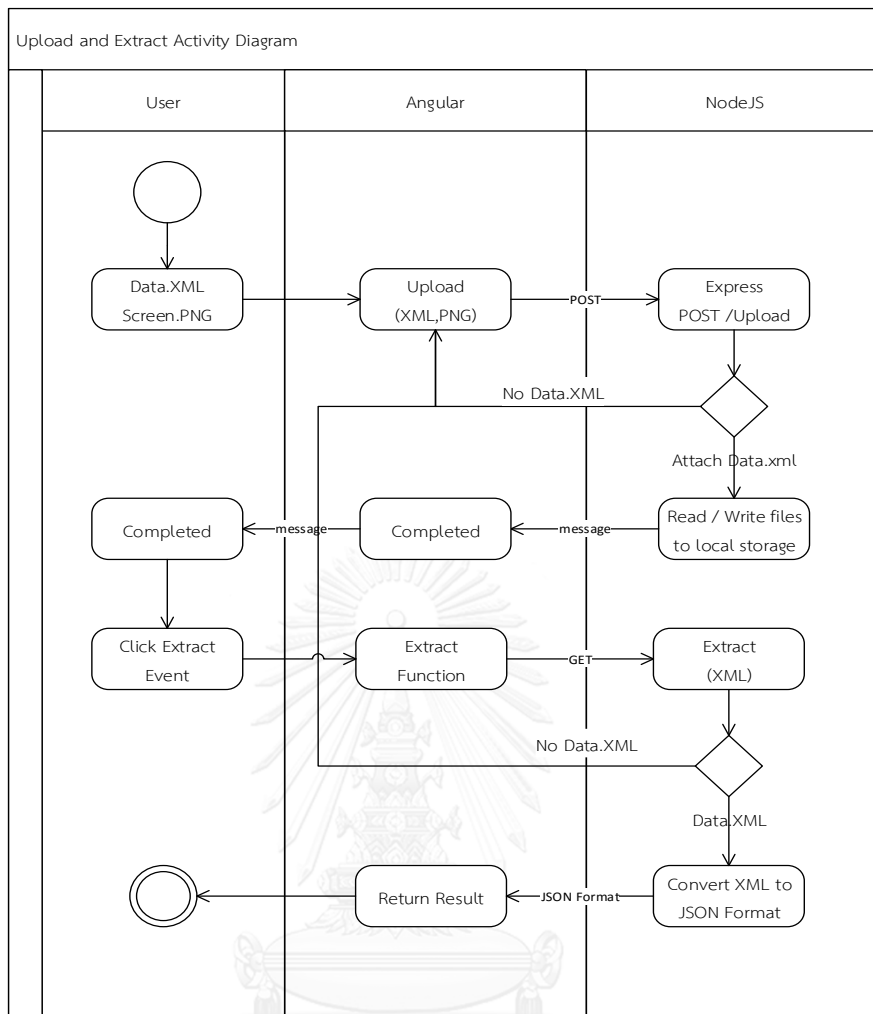
รูปที่ 10 โครงสร้างเอกซ์เอ็มแอลของ Data.XML ไฟล์

### 3.3 การอัปโหลดไปเว็บแอปพลิเคชัน

ในงานวิจัยนี้ได้พัฒนาเครื่องมือเป็นเว็บแอปพลิเคชันที่จะใช้ในการสร้างเอกสารอัตโนมัติ สถาปัตยกรรมของระบบแสดงในรูปที่ 11 ผู้วิจัยได้เลือกจาวาสคริปต์เทคโนโลยีมาใช้ในการพัฒนาทั้งในส่วนหน้าบ้านและส่วนหลังบ้าน (Full-Stack JavaScript) ระบบจะถูกพัฒนาด้วยโหนดเจเอสและแองกูลาร์เป็นหลัก โดยที่แองกูลาร์จะถูกใช้พัฒนาในส่วนหน้า ซึ่งทำหน้าที่ติดต่อกับผู้ใช้งาน และโหนดเจเอสจะถูกใช้พัฒนาการช่วยประมวลผลหลังบ้าน นอกจากนี้แองกูลาร์เป็นส่วนหน้าที่ทำงานภายใต้เบราว์เซอร์ (Browser) รูปที่ 12 แสดงแผนภาพกิจกรรมของการอัปโหลดและสกัดข้อมูล เริ่มจากผู้ใช้อัปโหลดเพิ่มข้อมูลหน้าจอฟีเอ็นจี และเพิ่มข้อมูลเอกซ์เอ็มแอล ผ่านฟังก์ชันอัปโหลด จากนั้นจะส่งต่อไปยังโหนดเจเอส ที่ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์ โหนดเจเอสจะติดตั้งแพ็คเกจ multer เพื่อรองรับการส่งข้อมูลมาจากฟังก์ชันอัปโหลด แต่ถ้าไม่มีเพิ่มข้อมูลอัปโหลดเข้ามาโหนดเจเอสจะส่งข้อความไปเตือนผู้ใช้งานให้ทำการอัปโหลดเพิ่มข้อมูลมาก่อน เมื่อโหนดเจเอสได้รับเพิ่มข้อมูลจึงจะเรียกใช้แพ็คเกจอ่าน เขียนเพิ่มข้อมูลเพื่อทำการบันทึกเพิ่มข้อมูล จากนั้นก็จะทำการสกัดเพิ่มข้อมูลโดยใช้แพ็คเกจ XMLDoc ในการหาตำแหน่งป้ายกำกับและสกัดข้อมูลจากออกมาจากเพิ่มข้อมูล Data.XML ข้อมูลที่สกัดมาได้จะถูกแปลงให้อยู่ในรูปแบบเจสันก่อน แล้วจึงส่งกลับไปยังเบราว์เซอร์ เพื่อให้ผู้ใช้ได้ทำการกำหนดคุณสมบัติต่อไป



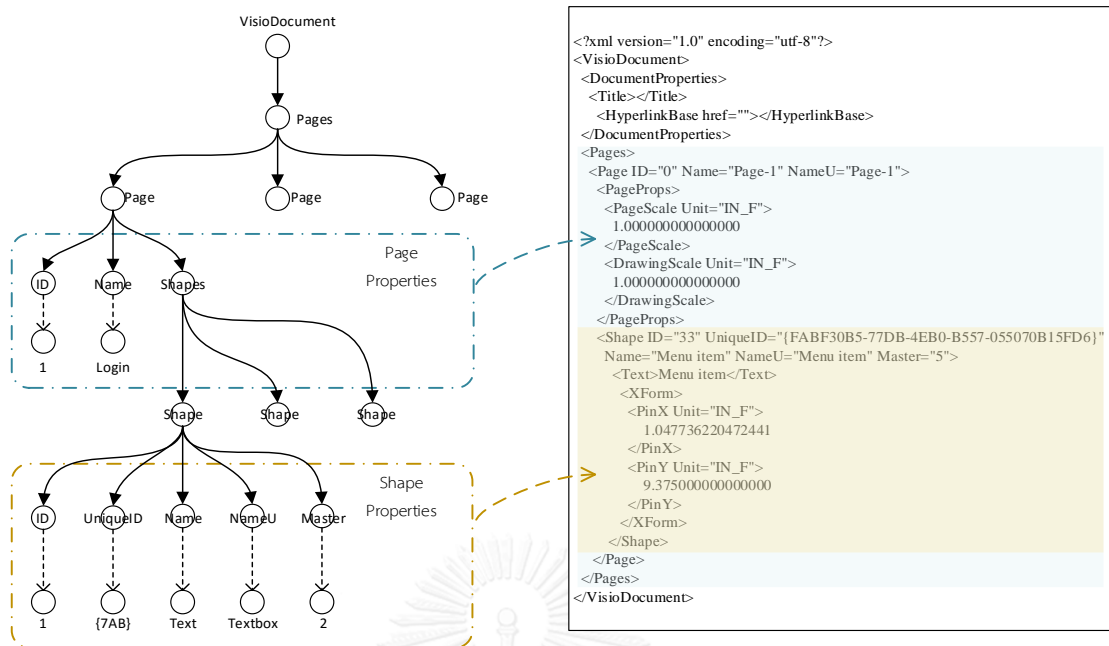
รูปที่ 11 สถาปัตยกรรมของตัวสร้างคำอธิบายต้นแบบ



รูปที่ 12 แผนภาพกิจกรรมในการอัปโหลดและสกัดข้อมูล

### 3.4 การสกัดข้อมูล

การสกัดข้อมูลที่ได้จากการนำเข้าแฟ้มข้อมูลเอกซ์เอ็มแอล โครงสร้างแฟ้มข้อมูลจะประกอบไปด้วยป้ายกำกับต่างๆ ดังที่ได้อธิบายไว้ข้างแล้วที่ด้านบน รูปที่ 13 ในกรอบสี่ฟากกับสี่เหลี่ยมคือข้อมูลในแฟ้มข้อมูลเอกซ์เอ็มแอล ที่ประกอบไปด้วย 2 ส่วนหลักที่จะสกัดออกมาคือ 1) คุณสมบัติเพจ (Page Property) ที่จะบอกข้อมูลว่าตัวต้นแบบประกอบไปด้วยจำนวนหน้าเท่าไร และแต่ละหน้ามีชื่ออะไรบ้าง 2) คุณสมบัติรูปแบบ (Shape Property) ที่จะบอกข้อมูลว่าในแต่ละหน้านั้นประกอบไปด้วยอะไรบ้าง




รูปที่ 13 แผนภาพการสกัดข้อมูลจากไฟล์เอกซ์เอ็มแอล

จากมาตรฐานของเอชทีเอ็มแอล คุณลักษณะของอินพุต (HTML input attributes) ผู้พัฒนาซอฟต์แวร์จะต้องเพิ่มคุณลักษณะสำคัญบางอันเข้าไปโดยอ้างอิงจาก w3school [10] แสดงในตารางที่ 3 หน้าจอพร้อมด้วยคุณลักษณะของกล่องข้อความที่อ้างอิงจากสเกตซ์ จียูไอ เซพ แสดงในรูปที่ 13 และคุณลักษณะที่ต้องการเพิ่ม ดังแสดงในตารางที่ 3

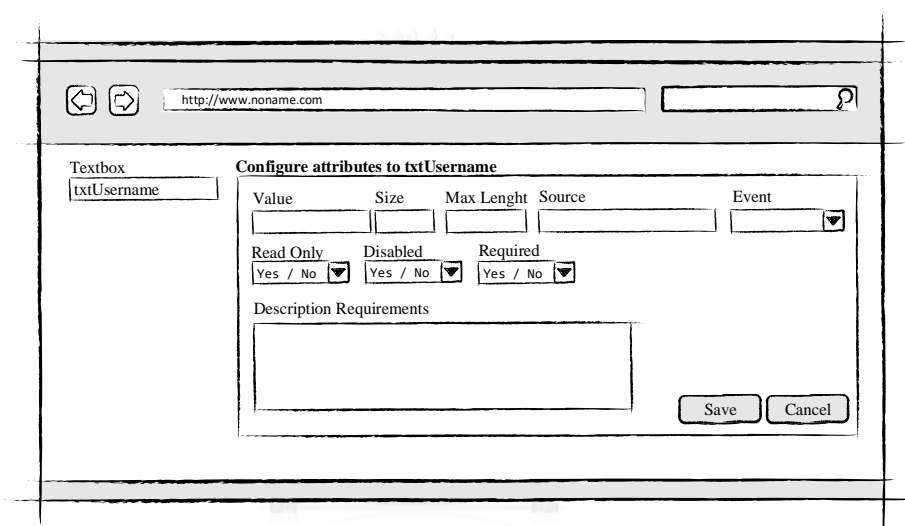
ตารางที่ 7 ตัวอย่างคุณลักษณะตามมาตรฐานของเอชทีเอ็มแอล

คุณลักษณะ	คำอธิบาย
value	ค่าของข้อมูล
readonly	การระบุค่าให้กับกลุ่มของการรับอินพุตให้สามารถอ่านได้อย่างเดียว ไม่สามารถเปลี่ยนแปลงแก้ไขได้
disabled	การระบุค่าให้กับกลุ่มของการรับอินพุตให้ไม่สามารถใช้งานได้ เช่น ไม่สามารถกดได้
maxlength	การระบุค่าให้กับกลุ่มของการรับอินพุตให้มีข้อจำกัดในการรับค่าโดยไม่เกินค่าที่ได้รับระบุไว้
size	การระบุขนาด
required	การระบุค่าให้กับกลุ่มของการรับอินพุตที่ผู้ใช้งานจำเป็นต้องอินพุตข้อมูล
placeholder	แสดงค่าเบื้องต้นเพื่อแนะนำให้กับผู้ใช้งานก่อนจะอินพุตเข้ามา

ตารางที่ 8 ตัวอย่างคุณลักษณะของกล่องข้อความที่ต้องการให้ผู้ใช้ใส่เพิ่มเติม

สเกตซ์ จียูไอ เชพ	คำอธิบาย
 Text field	value, readonly, disabled, maxleangh, size, required, placeholder

เมื่อข้อมูลทั้งหมดถูกรวบรวมเรียบร้อยแล้ว ระบบก็จะเตรียมข้อมูลและส่งกลับไปทีไหนดเจเอส เพื่อสร้างเป็นเอกสารอธิบายต้นแบบ



รูปที่ 14 หน้าจอการกำหนดการตั้งค่าคุณลักษณะของกล่องข้อความ

### 3.5 การสร้างเอกสารอธิบายต้นแบบ

แม่แบบเอกสารอธิบายต้นแบบซึ่งจะระบุตำแหน่งและเนื้อหาในแต่ละส่วนประกอบของหน้าจอ จะใส่อยู่ในปีกกา {...} ดังแสดงในรูปที่ 15 ตัวแม่แบบจะมีรายละเอียดของส่วนประกอบที่สกัดออกมาจากตัวต้นแบบสุดท้ายที่อยู่ในแฟ้มข้อมูลเอกซ์เอ็มแอล รวมกับคุณลักษณะที่เพิ่มเติมโดยผู้พัฒนาซอฟต์แวร์ ดังแสดงในรูปที่ 14 และตารางที่ 4 จะแสดงคำอธิบายถึงส่วนประกอบต่างๆ ที่อยู่ในแม่แบบเอกสารอธิบายต้นแบบ

Prototype Description Document					{DATE / TIME}
{SCREEN NAME}					
{Screen Image}					
{SHAPE ID}	{VALUE}	{SIZE}	{MAXLENGHT}	{REQUIRED}	
{SOURCE}			{EVENT}		
{DESCRIPTION REQUIREMENTS}					
<b>List of Attributes</b>					
{ATTRIBUTE}	{ATTRIBUTE}	{ATTRIBUTE}	{ATTRIBUTE}		
{ATTRIBUTE}	{ATTRIBUTE}	{ATTRIBUTE}	{ATTRIBUTE}		
<hr style="width: 50%; margin-left: 0;"/> Approval Design    Date					

รูปที่ 15 แม่แบบเอกสารอธิบายต้นแบบ

ตารางที่ 9 คำอธิบายคุณลักษณะของแม่แบบเอกสารอธิบายต้นแบบ

Element	Description
Screen name	ชื่อหน้าจอ
Screen Image	แสดงรูปภาพหน้าจอ
Shape ID	รหัสของส่วนประกอบ
Value	ค่าข้อมูลของส่วนประกอบ
Size	ขนาดของส่วนประกอบ
Source	ระบุแหล่งที่มาของข้อมูลของส่วนประกอบ
Event	เหตุการณ์เชื่อมโยงไปยังหน้าจออื่น
Description Requirements	อธิบายรายละเอียดเพิ่มเติม



ต้นแบบรวดเร็วจะมุ่งสร้างตัวต้นแบบอย่างรวดเร็วเพื่อทำการทดลองกับความต้องการของผู้ใช้ที่ยังไม่มีความเข้าใจที่ตีพอ [11] และนำผลตอบรับที่ได้มาปรับปรุงกระบวนการพัฒนาจนกระทั่งได้ต้นแบบสุดท้ายที่เป็นส่วนสำคัญสำหรับการนำไปพัฒนาซอฟต์แวร์ต่อ อย่างไรก็ตาม ในการพัฒนาต้นแบบรวดเร็วกลับต้องสิ้นเปลืองทรัพยากรและเวลาไปในการสร้างเอกสารสนับสนุนและอธิบายต้นแบบรวดเร็ว

งานวิจัยนี้จึงได้นำเสนอแนวทางและวิธีการสร้างเอกสารอัตโนมัติจากต้นแบบรวดเร็ว ด้วยเว็บแอปพลิเคชันที่พัฒนาจากแองกูลาร์และโหนดเจเอสที่จะทำงานร่วมกับเบราวเซอร์ อีกทั้งระบบได้ถูกออกแบบมาให้รองรับการสร้างต้นแบบที่สร้างขึ้นด้วยไมโครซอฟท์วิซีโอกับสเกตช์ จียูไอ เชฟ เพื่อสร้างเอกสารอัตโนมัติที่จะช่วยเพิ่มผลผลิตของโครงการและมีส่วนในการปรับปรุงกระบวนการให้มีประสิทธิภาพมากขึ้น



## บทที่ 4

### การพัฒนาระบบ

#### 4.1 เครื่องมือที่ใช้ในการพัฒนาและสภาพแวดล้อม

รายการเครื่องมือที่ใช้ในการพัฒนาระบบการสร้างเอกสารอัตโนมัติจากต้นแบบรวดเร็วและแฟ้มเอกสารที่ใช้ในการพัฒนาระบบ มีดังต่อไปนี้

##### 4.1.1 ฮาร์ดแวร์

- 1) หน่วยประมวลผลกลาง อินเทล คอร์ ไอ เซเว่น โมเดล 2620M 2.70 กิกะเฮิร์ต (CPU Intel Core i7-2620M 2.70 GHz)
- 2) หน่วยความจำ 8 กิกะไบต์ (8 GB RAM)
- 3) โซลิดสเตท ไดรฟ์ ความจุ 256 กิกะไบต์ (Solid State Drive 256 GB)

##### 4.1.2 ซอฟต์แวร์

- 1) ระบบปฏิบัติการไมโครซอฟต์ วินโดวส์ 10 โพร (Microsoft Windows 10 Pro)
- 2) โปรแกรมไมโครซอฟต์เวิร์ด 2016 (Microsoft Word 2016)
- 3) โปรแกรมไมโครซอฟต์วิซิโอ 2013 (Microsoft Visio 2013)
- 4) เครื่องมือสแตนด์ลิ่ง สเกตช์ จียูไอ เชพ (Sketch GUI Shapes)
- 5) โปรแกรม โหนดเจเอส เวอร์ชัน 5.3.0 (Node.js)
- 6) เครื่องมือ วิวาล สตูดิโอ โค้ด เวอร์ชัน 1.1 (Visual Studio Code V.1.1)
- 7) โหนด แพ็กเกจ เมเนเจอร์ เวอร์ชัน 3.8.3 (Node Package Manager Version 3.8.3)
- 8) แฟ้มเอกสารแองกูลาร์ 2 เวอร์ชัน 2.0.0 (Angular 2 Version 2.0.0)
- 9) แฟ้มเอกสารเอ็กซ์เพรส เวอร์ชัน 4.13.4 (Express Version 4.13.4)
- 10) แฟ้มเอกสารบอดี้พาร์เซอร์ (Body-Parser Version 1.15.0)
- 11) แฟ้มเอกสารมัลเตอร์ เวอร์ชัน 1.1.0 (Multer Version 1.1.0)
- 12) แฟ้มเอกสารเอ็กซ์เอ็มแอลด็อก เวอร์ชัน 0.4.0 (XMLDoc Version 0.4.0)
- 13) แฟ้มเอกสารด็อกเทมเพลตเตอร์ เวอร์ชัน 1.2.1 (Docxtemplater Version 1.2.1)

- 14) เบราว์เซอร์ กูเกิ้ล โครม เวอร์ชัน 51.0.2704.84 m (Google Chrome Version 51.0.2704.84)

#### 4.1.3 ขั้นตอนการติดตั้งซอฟต์แวร์

เครื่องมือและซอฟต์แวร์จะต้องถูกติดตั้งให้พร้อมก่อนจะเริ่มพัฒนาระบบ โดยลำดับการติดตั้งซอฟต์แวร์มีดังต่อไปนี้

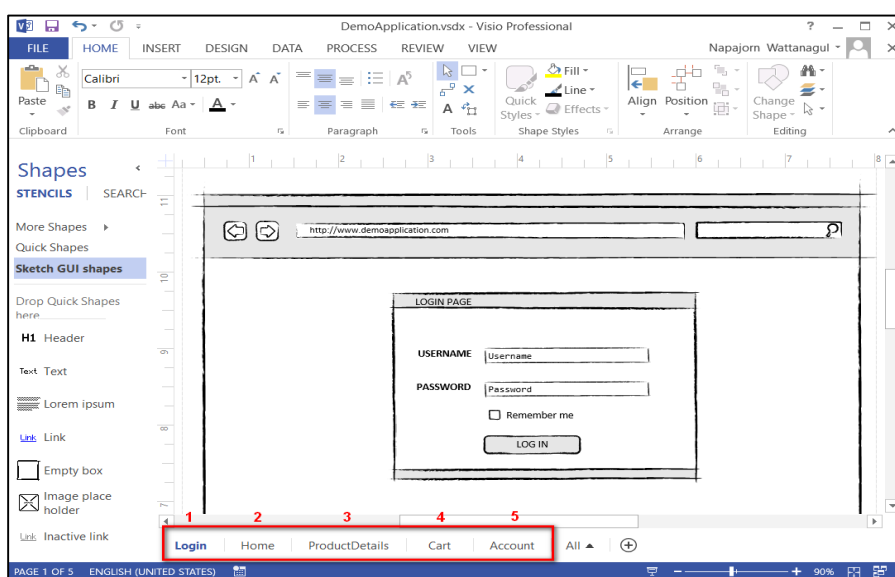
- 1) ติดตั้งระบบปฏิบัติการไมโครซอฟท์วินโดวส์ 10 โพร
- 2) ติดตั้งโปรแกรมไมโครซอฟท์เวิร์ด 2016
- 3) ติดตั้งโปรแกรมไมโครซอฟท์วิซิโอ 2013
- 4) ติดตั้งเครื่องมือสแตนซิล สเกตซ์ จิยูไอ เซพ
- 5) ติดตั้งโปรแกรมโนนดเจเอส 5.3.0
- 6) ติดตั้งเครื่องมือ วิชวล สตูดิโอ โค้ด
- 7) ติดตั้งเครื่องมือ โนนด แพ็กเกจ เมเนเจอร์ 3.8.3
- 8) นำเข้าแพ็กเกจแอนกูลาร์ 2 2.0.0
- 9) นำเข้าแพ็กเกจเอ็กซ์เพรส 4.13.4
- 10) นำเข้าแพ็กเกจบอดีฟาเซอร์ 1.15.0
- 11) นำเข้าแพ็กเกจมัลเตอร์ 1.1.0
- 12) นำเข้าแพ็กเกจเอ็กซ์เอ็มแอลด็อก 0.4.0
- 13) นำเข้าแพ็กเกจด็อกเอ็กซ์เทมเพลตเตอร์ 1.2.1

#### 4.2 ขั้นตอนการพัฒนาระบบ

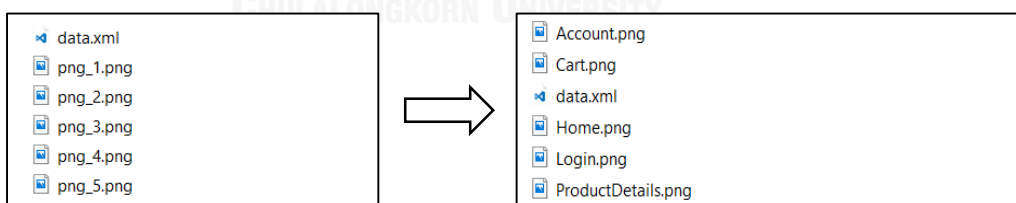
การพัฒนาระบบจะถูกแบ่งออกเป็น 2 ส่วนหลัก คือ ฝั่งเซิร์ฟเวอร์ และ ฝั่งไคลเอนท์ ทางฝั่งเซิร์ฟเวอร์จะถูกพัฒนาด้วยโนนดเจเอสโดยจะจำลองการสร้างเว็บเซิร์ฟเวอร์ด้วยแพ็กเกจ เอ็กซ์เพรส ขึ้นมาเพื่อรองรับการร้องขอจากฝั่งไคลเอนท์ และทางฝั่งไคลเอนท์จะถูกพัฒนาด้วยแอนกูลาร์ ที่ทำหน้าที่ติดต่อกับผู้ใช้งาน งานวิจัยนี้ได้ทำการจำลองสร้างต้นแบบรวดเร็วจากเครื่องมือสแตนซิล สเกตซ์ จิยูไอ เซพ ผ่านทางโปรแกรมวิซิโอ เพื่อใช้เป็นข้อมูลในการนำไปสกัดเพื่อสร้างเอกสารอัตโนมัติ

#### 4.2.1 การเตรียมต้นแบบรวดเร็วเพื่อนำเข้า

เริ่มจากการจำลองสร้างต้นแบบรวดเร็วผ่านทางโปรแกรมไมโครซอฟต์วิซิโอ ดังที่ได้กล่าวไว้ในบทที่ 3 จากนั้นทำการส่งออกโดยเลือกรูปแบบการส่งออกเป็น Web Page (\*.htm) หลังจากการส่งออก จะได้เพิ่มข้อมูลที่ประกอบไปด้วยเพิ่มข้อมูลเอกซ์เอ็มแอล และเพิ่มข้อมูลพีเอ็นจี ดังรูปที่ 16 แสดงรายละเอียดชื่อของแต่ละเพจที่จะถูกส่งออกมา เรียงลำดับจากซ้ายไปขวา และจะต้องนำมาทำการแก้ไขชื่อไฟล์พีเอ็นจี



รูปที่ 16 รายละเอียดชื่อของเพจต้นแบบรวดเร็วจากโปรแกรมไมโครซอฟต์วิซิโอ



รูปที่ 17 การเตรียมข้อมูลเพื่อนำเข้าไปสกัด

รูปที่ 17 ด้านซ้ายคือเพิ่มข้อมูลที่ถูส่งออกมาจากโปรแกรมไมโครซอฟต์วิซิโอ โดยชื่อเพิ่มข้อมูลพีเอ็นจีจะเป็นชื่อที่ไม่สื่อความหมาย จึงต้องทำการแก้ไขให้เป็นชื่อของเพจตามที่ได้ทำการออกแบบไว้ โดยอ้างอิงการแก้ไขตามตารางที่ 10

ตารางที่ 10 การแก้ไขแฟ้มข้อมูลพีเอ็นจีเพื่อเตรียมการนำเข้าข้อมูล

ลำดับ	ชื่อต้นฉบับ	เปลี่ยนแปลง
1	png_1.png	Login.png
2	png_2.png	Home.png
3	png_3.png	ProductDetails.png
4	png_4.png	Cart.png
5	png_5.png	Account.png

#### 4.2.2 การพัฒนาขั้นตอนวิธีการเตรียมข้อมูลเพื่อนำเข้า

เริ่มจากการติดตั้งซอฟต์แวร์ที่จะใช้พัฒนา โดยจะเริ่มพัฒนาจากระบบเพื่อรองรับการนำเข้าไฟล์ การติดตั้งโปรแกรมหรือแพ็คเกจสามารถตรวจสอบเช็คการติดตั้งได้ ดังรูปที่ 18 ใช้คำสั่งในการตรวจสอบเวอร์ชันของโปรแกรมโนนดเจเอส ส่วนการติดตั้งแพ็คเกจอื่นโดย โหนด แพ็คเกจเมนเจอร์ สามารถใช้คำสั่ง “npm install package” ให้เปลี่ยนชื่อแพ็คเกจเป็นชื่อที่ต้องการติดตั้ง

```
C:\Users\Napajorn.w>node -v
v5.3.0
```

รูปที่ 18 คำสั่งตรวจสอบการติดตั้งของโปรแกรม

การติดตั้งแองกูลาร์ 2 ทำได้โดยการดาวน์โหลดจากเว็บไซต์แองกูลาร์ ทำการตั้งชื่อแฟ้มเอกสารใหม่เป็น “automated-documentation” ในแฟ้มเอกสารนั้นจะประกอบไปด้วย กรอบการทำงานของแองกูลาร์ 2 จะมีแฟ้มข้อมูล package.json ที่จะระบุแพ็คเกจที่สำคัญในการใช้พัฒนาแองกูลาร์ 2 ให้ทำการติดตั้งตามแฟ้มข้อมูล package.json การติดตั้งทำได้โดยการใช้คำสั่ง “npm install” ที่ภายใต้แฟ้มเอกสาร “automated-documentation”

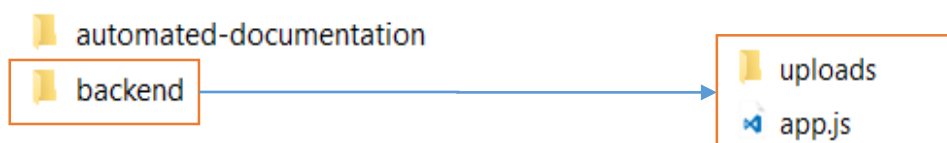
```

package.json C:\Users\Napajorn.W\Google Drive\Angular Projects\automated-documentation
1  {
2    "name": "automated-documentation",
3    "version": "1.0.0",
4    "scripts": {
5      "lite": "lite-server",
6      "gulp": "gulp",
7      "start": "concurrent \"npm run gulp\" \"npm run lite\" "
8    },
9    "license": "ISC",
10   "dependencies": {
11     "angular2": "2.0.0-beta.14",
12     "systemjs": "0.19.25",
13     "es6-shim": "^0.35.0",
14     "reflect-metadata": "0.1.2",
15     "rxjs": "5.0.0-beta.2",
16     "zone.js": "0.6.6"
17   },
18   "devDependencies": {
19     "autoprefixer": "^6.3.5",
20     "cssnano": "^3.4.0",
21     "concurrently": "^2.0.0",
22     "gulp": "^3.9.0",
23     "gulp-ext-replace": "^0.2.0",
24     "gulp-imagemin": "^2.4.0",
25     "gulp-postcss": "^6.0.1",
26     "gulp-sourcemaps": "^1.6.0",
27     "gulp-typescript": "^2.10.0",
28     "gulp-uglify": "^1.5.1",
29     "lite-server": "^2.1.0",
30     "postcss": "^5.0.13",
31     "postcss-scss": "^0.1.3",
32     "precss": "^1.3.0"
33   }
34 }

```

รูปที่ 19 แพ็กเกจที่สำคัญที่ใช้ในการพัฒนาของแองกูลาร์ 2

จากการติดตั้งโหนดเจเอสเรียบร้อยแล้ว ให้ทำการสร้างแฟ้มเอกสารชื่อ “backend” แฟ้มเอกสารนี้จะทำการเก็บแฟ้มข้อมูล app.js ที่เป็นแฟ้มข้อมูลที่จะใช้พัฒนาเป็นเว็บเซิร์ฟเวอร์ในการรับการร้องขอจากฝั่งไคลเอนท์ และสร้างแฟ้มเอกสารอีกอันชื่อ uploads ที่จะนำเอาแฟ้มข้อมูลที่ถูกอัปโหลดนำมาเก็บไว้ในแฟ้มเอกสารนี้



รูปที่ 20 แฟ้มเอกสาร uploads ที่อยู่ภายใต้แฟ้มเอกสาร backend

ภายใต้แฟ้มข้อมูล backend เริ่มพัฒนา app.js ฟังก์ชันการอัปโหลดที่จะใช้ในการรับการร้องขอมาจากไคลเอนท์

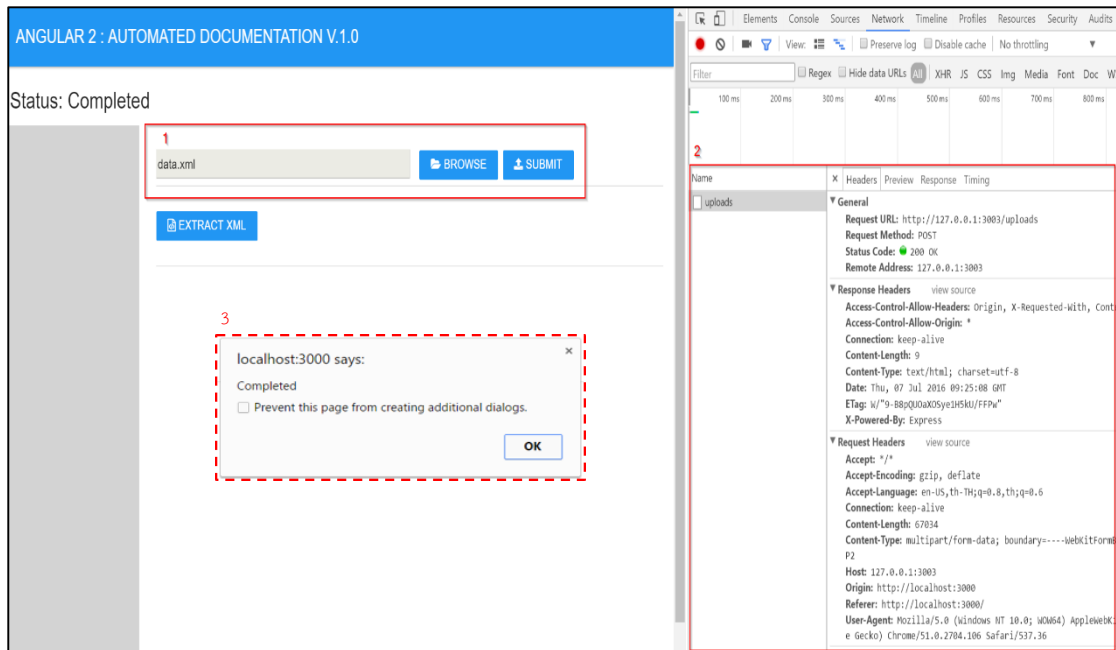
```

• app.js C:\Users\Napajorn.w\Google Drive\Angular Projects\backend
22
23 var type = upload.single('uploads');
24
25 app.post('/uploads', type, function (req,res) {
26   var tmp_path = req.file.path;
27
28   console.log(tmp_path);
29   var target_path = 'uploads/' + req.file.originalname;
30   console.log(target_path);
31   var readable = fs.createReadStream(tmp_path);
32   var writable = fs.createWriteStream(target_path);
33   readable.pipe(writable, { end: false });
34   readable.on('end', () => {
35     writable.end();
36     res.send('Completed');
37   });
38
39   setTimeout(() => {
40     console.log('stop writing to ' + target_path);
41     readable.unpipe(writable);
42     console.log('manually close the file stream');
43     writable.end();
44   }, 1000);
45
46   fs.readdir('uploads/', (err, data) => {
47     if (!err) {
48       for(i=0; i <= data.length; i++) {
49         if (data[i] === 'data.xml') {
50           console.log(data[i]);
51         }
52       }
53     }
54   });

```

รูปที่ 21 โค้ดในการพัฒนาฟังก์ชันอัปโหลด

ฟังก์ชันอัปโหลดจะรอรับการส่งเพิ่มข้อมูลมาด้วยวิธี POST จาก ยูอาร์แอล (URL) ที่ระบุไว้ในที่นี้ได้ระบุยูอาร์แอลไว้ที่ “/uploads” จากที่ได้รับเพิ่มข้อมูลมาแล้วก็จะนำไปบันทึกเก็บไว้ในแฟ้มเอกสาร “uploads/” ส่วนสุดท้ายทางระบบจะมีการเช็คว่ามีเพิ่มข้อมูล Data.XML ในระบบหรือไม่ เพื่อที่จะได้ส่งข้อความกลับไปบอกผู้ใช้งานว่ามีเพิ่มข้อมูลมีหรือไม่มีอยู่ในระบบ



รูปที่ 22 หน้าจอการอัปโหลดเพิ่มข้อมูลไปยังโหนดเจเอส

จากรูปที่ 22 ขั้นตอนที่ 1 ผู้ใช้งานได้ทำการอัปโหลดเพิ่มข้อมูลไปยังโหนดเจเอสจาก จากนั้น ผู้ใช้งานกดปุ่ม Submit เพื่อเรียกใช้งานฟังก์ชันการอัปโหลด แอ็กกูลาร์จะส่งการร้องขอเพื่อทำการส่ง เพิ่มข้อมูลไปให้โหนดเจเอสในขั้นตอนที่ 2 และขั้นตอนที่ 3 กล้องข้อความแสดงบอกว่าการร้องขอนั้น ทำงานเสร็จสิ้นแล้ว รูปที่ 23 แสดงข้อความว่าโหนดเจเอสนั้นได้รับเพิ่มข้อมูลเอกซ์เอ็มแอลและทำ การบันทึกเพิ่มข้อมูลไปยังแฟ้มเอกสารนั้นแล้ว

```
C:\Users\Napajorn.w\Google Drive\Angular Projects\backend>node app.js
Working on port 3003
tmp path:uploads\c5db69cd8346069ffcee926b9e7f3ce6
target path:uploads/data.xml
data.xml
```

รูปที่ 23 โหนดเจเอสได้รับเพิ่มข้อมูลและทำการบันทึก

#### 4.2.3 การสกัดข้อมูลจากต้นแบบรวดเร็ว

พัฒนาโหนดเจเอสในส่วนการร้องขอสำหรับการสกัดข้อมูลจากเพิ่มข้อมูลเอกซ์เอ็มแอล โดยโครงสร้างข้อมูลเอกซ์เอ็มแอลจะเป็นดังรูปที่ 13 ที่อยู่ในบทที่ 3 จากนั้นทำการพัฒนาฟังก์ชันการสกัดเอกซ์เอ็มแอลแล้วแปลงข้อมูลที่สกัดออกมาเป็นรูปแบบของเจสัน เพื่อที่จะส่งกลับไปยังผู้ใช้งาน



```

app.js C:\Users\Napajom.w\Google Drive\Angular Projects\backend
77 app.use('/extract', function(req, res, next){
78
79   var contents = fs.readFileSync(__dirname + "/uploads/data.xml","binary");
80   if(fs.readFileSync(__dirname + "/uploads/data.xml")) {
81     var results = new xmldoc.XmlDocument(contents);
82     var pages = results.childNamed("Pages");
83     //console.log(pages);
84     var Pages = '{ "pages" : [';
85     var jsonPages = '';
86     var k = 1;
87     var shapeID, shapeName, shapeValue, shapeProperties, propLabel, propValue, properties;
88     pages.eachChild(function (shapes) {
89       var j = 0;
90       var z = 0;
91       var i = 1;
92       console.log('Page ID: ' + shapes.attr.ID + ' Page Name: ' + shapes.attr.Name);
93       Pages += '{pid:"'+ shapes.attr.ID + '", pname:"'+shapes.attr.Name + '", shapes: [';
94       shapes.eachChild(function (shape) {
95
96         shape.eachChild(function (s) {
97
98           if (s.attr.ID != undefined) {
99             j++;
100            shapeID = s.attr.ID;
101            shapeName = s.attr.Name
102            shapeValue = (s.valueWithPath("Text") == undefined ? '' : s.valueWithPath("Text")
103
104            console.log(i++
105            + (s.attr.ID == undefined ? "" : ' ID: ' + s.attr.ID)
106            + (s.attr.Name == undefined ? "" : ' Name: ' + s.attr.Name)
107            + (s.valueWithPath("Text") == undefined ? '' : ' -> ' + s.valueWithPath("Text")
108            );
109            properties = "";
110            if (s.valueWithPath("Prop@ID")) {
111              z=1;
112              shapeProperties = "";
113              s.eachChild(function (prop) {
114
115                (prop.attr.ID === undefined ? '' :

```

รูปที่ 24 ตัวอย่างโค้ดในการพัฒนาฟังก์ชันการสกัดข้อมูล

ฟังก์ชันการการสกัดข้อมูลจะเริ่มทำการอ่านแฟ้มข้อมูล Data.XML ที่อัปโหลดเข้ามาจากขั้นตอนด้านบน จากนั้นจะเริ่มสกัดและแปลงข้อมูลจากเอกซ์เอ็มแอลไปเป็นรูปแบบเจสันแล้วส่งกลับไปยังผู้ใช้งาน ตารางที่ 11 แสดงข้อมูลส่วนประกอบของแต่ละเพจที่จะต้องสกัดออกมา

ตารางที่ 11 จำนวนส่วนประกอบในแต่ละเพจจากการส่งออกมาจากโปรแกรมไมโครซอฟต์วิซีโอ

No.	Pages	Shapes
1	Login	8
2	Home	51

3	ProductDetails	45
4	Cart	41
5	Account	73

รูปที่ 25 ผลลัพธ์การทำงานของฟังก์ชันสกัดข้อมูล

รูปที่ 25 ผู้ใช้งานทำการสกัดข้อมูลเอกซ์เอ็มแอลโดยการกดปุ่ม “Extract XML” ระบบจะทำการส่งการร้องขอไปยังโหนดเจเอสทียูอาร์แอล “/extract” ตามที่ได้พัฒนาไว้ดังรูปที่ 24 หลังจากทำการแปลงเอกซ์เอ็มแอลเป็นรูปแบบของเจสันเสร็จสิ้น จะส่งข้อมูลเจสันกลับมาให้ผู้ใช้งาน จากนั้นจะนำมาแสดงที่ด้านซ้ายมือของผู้ใช้งาน ที่จะแสดงรายละเอียดของเพจและส่วนประกอบภายใต้เพจนั้น รูปที่ 26 แสดงรูปแบบเจสันที่ได้รับค่ามาจากโหนดเจเอสเรียกใช้ฟังก์ชันการสกัดข้อมูล

The screenshot shows a web browser's developer tools interface. The 'Name Path' tab is active, displaying 'extract' with a path of '127.0.0.1'. The 'Preview' tab shows a JSON response structure for a login page. The structure is as follows:

```

{
  pages: [
    { pid: "0", pname: "Login", ... },
    { pid: "4", pname: "Home", ... },
    { pid: "5", pname: "ProductD", ... }
  ]
}

```

The 'Login' page (pid: "0") contains the following shapes:

- Browser 800x600 (value: "http://www.demoapplication.com")
- Portlet (value: "LOGIN PAGE")
- Text field (value: "Username")
- Text field.77 (value: "Password")
- Text (value: "USERNAME")
- Text.84 (value: "PASSWORD")
- Button (value: "LOG IN")
- Check box (value: "Remember me")

The bottom of the screenshot shows '1 requests | 31.2 KB transferred'.

รูปที่ 26 รูปแบบเจสันที่ผ่านฟังก์ชันการแปลง

#### 4.2.4 การกำหนดคุณลักษณะ

การพัฒนาในส่วนกำหนดคุณลักษณะเพิ่มเติมนั้นจะถูกพัฒนาในแองกูลาร์เป็นส่วนใหญ่ เริ่มจากการพัฒนาคอมโพเนนท์ที่จะใช้ในการรับค่าของอ็อบเจกต์นั้น เพื่อที่จะทำการกำหนดคุณลักษณะเพิ่มเติมให้กับอ็อบเจกต์นั้น

The screenshot displays the Angular 2 Automated Documentation V1.0 interface. The status is 'Extract Data.xml to JSON format'. The interface is divided into several components:

- App Component:** The top right section, containing a 'Main Component' and a 'Chart Component'.
- Main Component:** A pie chart showing the distribution of elements on the page, including Browser 800x600, Button, Check box, Portlet, and Text.
- Chart Component:** A bar chart showing the count of elements for each category: Login (1), Home (1), ProductDetails (1), Cart (1), and Account (1).
- Shape Component:** A detailed view of a 'portlet login page' shape, showing its value ('login page'), size, source, and read-only status.
- Screen Image:** A screenshot of the login page being analyzed.

The left sidebar shows a tree view of the application's components, including 'Login' and 'Home' pages with their respective elements and shapes.

รูปที่ 27 แองกูลาร์คอมโพเนนท์

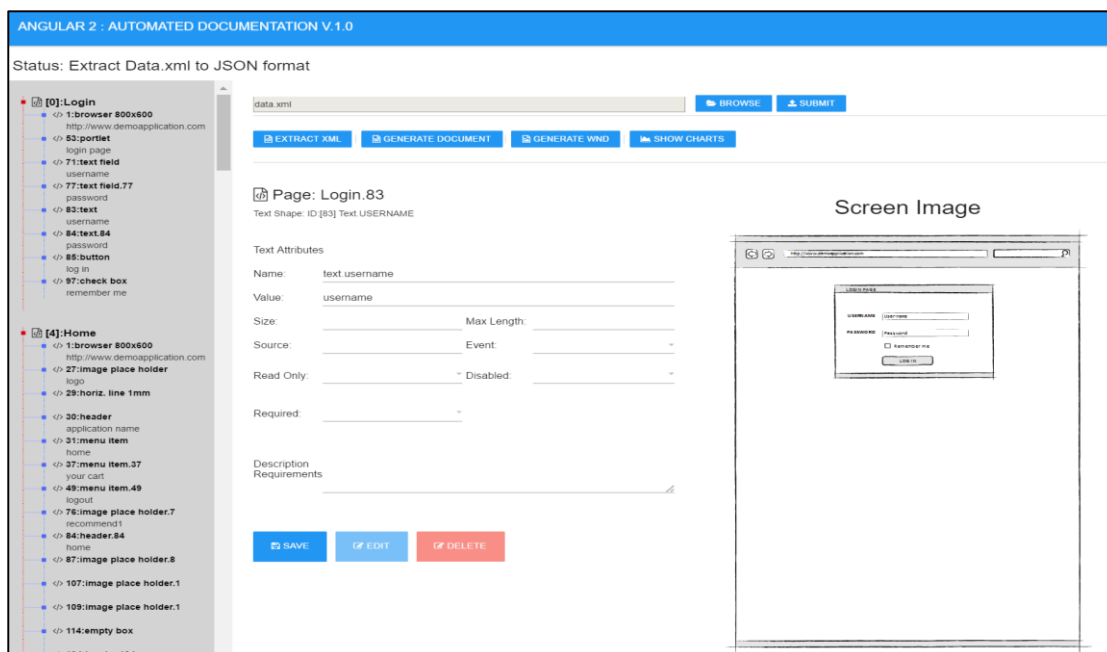
รูปที่ 27 แสดงหน้าจอที่ประกอบไปด้วยคอมโพเนนต์ต่างๆ ในแองกูลาร์ โดยเริ่มจากกรอบสี่ด้านนอกสุด คือ ส่วนที่เป็นหลักของแอปพลิเคชันที่เกี่ยวข้องกับไลบรารีที่สำคัญในแองกูลาร์ กรอบสีน้ำเงิน คือ ส่วนหลักของโปรแกรมที่ทำหน้าที่ในการอัปเดตและแสดงผลข้อมูลที่ผ่านการสกัดมาแล้ว และสุดท้ายกรอบสีเขียว เป็นคอมโพเนนต์แสดงผลกราฟและรอรับค่าเพื่อที่จะนำมาทำการกำหนดคุณลักษณะเพิ่มเติม

```

● shape.component.ts dev
1  import {Component, Input, Output, EventEmitter} from 'angular2/core'
2
3  @Component({
4    selector: 'shape-component',
5    template: `...
100
101  })
102
103  export class ShapeComponent {
104    @Input() page;
105    @Input() shape;
106    @Input() data;
107
108    txtAreaContent = "";
109    config;
110    pageId;
111    shapeId;
112    result;
113    saveCompleted;
114
115    constructor(){
116      this.saveCompleted = false;
117    }
118
119
120    ngOnChanges(change) {
121      if(localStorage[this.page.pname+'.'+this.shape.id]){
122        this.config = true;
123      } else {
124        this.config = false;
125      }
126      console.log(change);
127    }
128
129
130
131    saveConfig(sname, svalue, size, maxlen, source, sevent, readonly, disabled, required, desp) {
132      this.pageId = this.page.pid;
133      this.shapeID = this.shape.id;
134      this.result = '{"page": "'+this.page.pname+'", "shape": "'+ this.shape.id + '", "shapeProperties": [';
135      this.result += '{"name": "'+sname.value+'"},';
136      this.result += '{"value": "'+svalue.value+'"},';
137      this.result += '], "config": [';
138      this.result += '{"shapeValue": "'+svalue.value+'"},';
139      this.result += '{"size": "'+size.value+'"},';
140      this.result += '{"maxLength": "'+maxlen.value+'"},';
141      this.result += '{"source": "'+source.value+'"},';
142      this.result += '{"event": "'+sevent.value+'"},';
143      this.result += '{"readonly": "'+readonly.value+'"},';

```

รูปที่ 28 ตัวอย่างโค้ดในการพัฒนาคอมโพเนนต์ที่ใช้สำหรับเพิ่มคุณลักษณะ



รูปที่ 29 หน้าจอการกำหนดคุณลักษณะ

ผู้ใช้งานทำการเลือกรายการส่วนประกอบด้านซ้ายเพื่อจะทำการกำหนดคุณลักษณะเพิ่มเติม หน้าจอก็จะมีรายการกำหนดคุณลักษณะขึ้นมาให้ผู้ใช้งานได้ทำการกำหนดคุณลักษณะ โดยทางด้านขวาจะแสดงรูปภาพหน้าจอที่เกี่ยวข้อง เพื่อความสะดวกของผู้ใช้งาน

ตารางที่ 12 ตารางอธิบายหน้าจอการเพิ่มคุณลักษณะ

ลำดับ	ชื่อคุณลักษณะ	คำอธิบาย
1	Name	ชื่อของส่วนประกอบ
2	Value	ค่าข้อมูลของส่วนประกอบ
3	Size	การกำหนดขนาดให้กับส่วนประกอบ
4	Max Length	จำนวนตัวอักษรที่สามารถรับได้
5	Source	บอกแหล่งที่มาของส่วนประกอบ
6	Event	เหตุการณ์ที่เชื่อมโยงไปยังหน้าเพจอื่น
7	Read Only	ไม่อนุญาตให้แก้ไขของส่วนประกอบ
8	Disabled	ไม่อนุญาตให้ใช้งานของส่วนประกอบ
9	Required	ไม่อนุญาตให้เป็นค่าว่าง
10	Description Requirement	ใส่คำอธิบายเพิ่มเติมให้ส่วนประกอบ

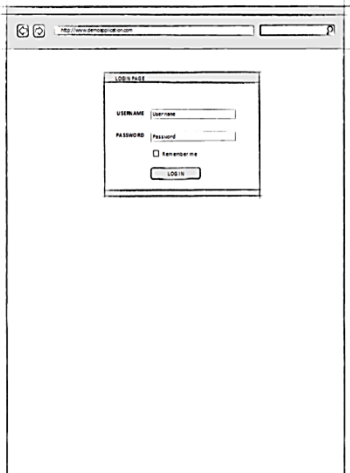
**Page: Login.83**  
Text Shape: ID [83] Text.USERNAME

**Text Attributes**

Name: text.username  
Value: username  
Size: 50 Max Length: 10  
Source: table.field.username Event:   
Read Only: Disabled:   
Required: Yes   
Description Requirements:   
input username for authentication

SAVE EDIT DELETE

**Screen Image**



รูปที่ 30 กำหนดคุณลักษณะเพิ่มเติมให้กับส่วนประกอบ

**[0]: Login**

- </> 1:browser 800x600
- </> 53:portlet
- </> 71:text field
- </> 77:text field.77
- </> 83:text
- </> 84:text.84
- </> 85:button
- </> 97:check box

**[4]: Home**

- </> 1:browser 800x600
- </> 27:image place holder
- </> 29:horiz. line 1mm
- </> 30:header
- </> 31:menu item
- </> 37:menu item.37
- </> 49:menu item.49
- </> 76:image place holder.7
- </> 84:header.84
- </> 87:image place holder.8

data.xml

EXTRACT XML GENERATE DOCUMENT GENERATE WND SHOW CHARTS

**Page: Login.83**  
Text Shape: ID [83] Text.USERNAME

✔ This shape already has been configured.

**Text Attributes**

Name: text.username  
Value:   
Size: Max Length:   
Source: Event:   
Read Only: Disabled:   
Required:   
Description Requirements:

SAVE EDIT DELETE

รูปที่ 31 หน้าจอการกำหนดคุณลักษณะเมื่อทำการสำเร็จ

Console

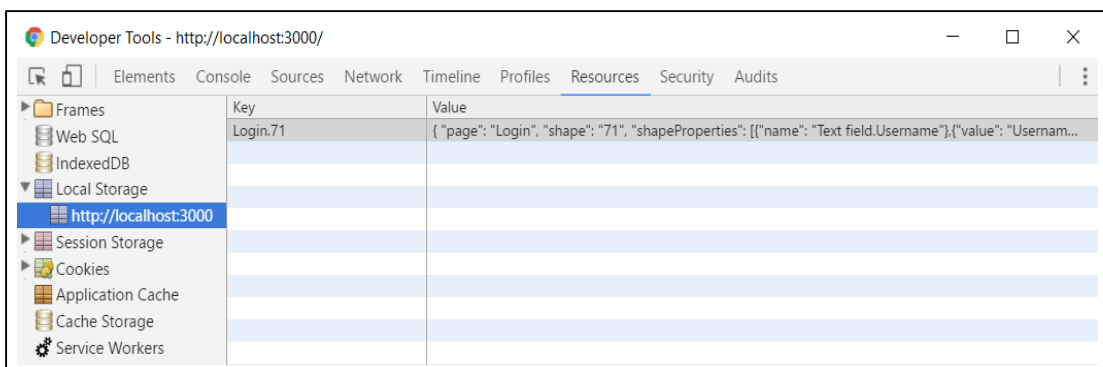
top Preserve log

```

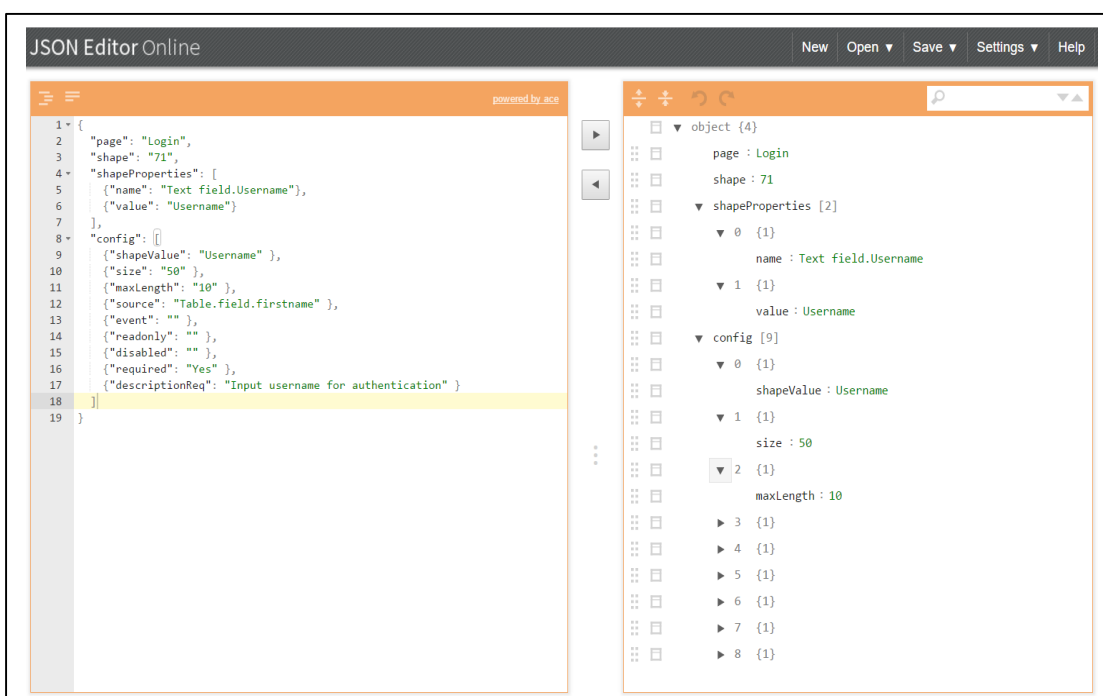
clicked:[object MouseEvent] main.component.ts:152
clicked:[object MouseEvent] main.component.ts:152
{ "page": "Login", "shape": "71", "shapeProperties": [{"name": "Text field.Username"}, {"value": "shape.component.ts:151", "shapeValue": "Username"}], "config": [{"shapeValue": "Username"}, {"size": "50"}, {"maxLength": "10"}, {"source": "Table.field.firstname"}, {"event": ""}, {"readOnly": ""}, {"disabled": ""}, {"required": "Yes"}, {"descriptionReq": "Input username for authentication"}]}
clicked:[object MouseEvent] main.component.ts:152

```

รูปที่ 32 ทำการบันทึกข้อมูลคุณลักษณะเพิ่มเติม



รูปที่ 33 การบันทึกข้อมูลเก็บลงในโลคอลสตอเรจ



รูปที่ 34 การตรวจสอบความถูกต้องของรูปแบบเจสันจาก Json Editor Online

เมื่อผู้ใช้ทำการกำหนดคุณลักษณะเรียบร้อยแล้ว และกดปุ่มเพื่อทำการบันทึก ข้อมูลจะถูกแปลงเป็นเจสันและจะถูกนำไปเก็บไว้ใน โลคอล สตอเรจ (Local Storage) โดยจะกำหนดคีย์ที่จะใช้เข้าถึงข้อมูล คือ ชื่อเพจ ตามด้วยมหัพภาค และรหัสของส่วนประกอบ จากรูป 33 จะได้คีย์เป็น “Login.71” หมายถึง เพจ Login รหัสส่วนประกอบ 71 เมื่อส่วนประกอบนั้นได้ถูกกำหนดคุณลักษณะแล้วจะมีรูปเครื่องหมายถูกมาเพื่อแจ้งให้ผู้ใช้ทราบว่า ส่วนประกอบนี้ได้ถูกกำหนดคุณลักษณะไปแล้ว

#### 4.2.5 การสร้างเอกสารอัตโนมัติ

การจะสร้างเอกสารอัตโนมัติจำเป็นต้องใช้แม่แบบในการระบุป้ายกำกับเพื่อที่จะนำข้อมูลที่ได้ไปใส่ในเอกสารแม่แบบได้อย่างถูกต้อง งานวิจัยนี้ได้สร้างเอกสารแม่แบบจากโปรแกรมไมโครซอฟต์เวิร์ด รูปที่ 35 ทำการกำหนดแม่แบบเอกสาร โดยการใช้ปีกกาในการบอกโปรแกรมว่าส่วนนี้จะมีการแทนค่าในปีกกา ส่วนเครื่องหมาย “#” ที่อยู่ภายใต้ปีกกาจะบอกโปรแกรมว่าให้วนซ้ำ โดยจะใช้คู่กับเครื่องหมาย “/” ตัวอย่าง {#shape} {/shape} หมายถึงมีการวนซ้ำภายใต้ปีกกา นี้จนกว่าข้อมูลจะสิ้นสุดถึงจะจบการวนซ้ำ จากรูปที่ 35 จะมีการใช้การวนซ้ำในตาราง หมายถึง ตารางนี้จะมีการวนซ้ำเพื่อแสดงค่าส่วนประกอบที่มีการกำหนดคุณลักษณะเพิ่มเติมเข้ามา

Prototype Description Document 12/06/2016 17:32

{|screen\_name1} Screen  
{%image1}

{#pages1}Page ID	Page Name	Shapes
{page_id}	{page_name}	{shapes}/pages1

{#shapes1}ID	Shape	Size	Max Length
{shape_id}	{shape_name}; {shapeValue}	{size}	{max_length}
Source:	{source}	Event:	{event}
Read Only	{readonly}	Disabled:	{disabled}
		Required:	{required}
Description			
{description_req}/shapes1			

รูปที่ 35 การกำหนดเอกสารแม่แบบด้วยโปรแกรมไมโครซอฟต์เวิร์ด

การพัฒนาในส่วนรองรับการร้องขอเพื่อสร้างเอกสารอธิบายต้นแบบ จะเพิ่มเติมในโหมดเจเอสทีทีที่รองรับการเรียกผ่านทางยูอาร์แอล “/send” โดยที่แองกูลาร์จะพัฒนาฟังก์ชันในการรวบรวมโลคอล สตอเรจ ที่ผู้ใช้ได้ทำการกำหนด จากนั้นจะแปลงข้อมูลให้อยู่ในรูปแบบที่จะนำไปใช้งาน แล้วจึงส่งออกไปให้โหมดเจเอสทีทีเพื่อนำไปสร้างเอกสารอธิบายต้นแบบ



ตารางที่ 13 ความหมายของการสร้างรูปแบบที่ใช้ในการสร้างแม่แบบ

ลำดับ	รูปแบบ	คำอธิบาย
1	{screen_name}	ป้ายกำกับที่จะนำชื่อหน้าจามาใส่ในป้ายกำกับ
2	{image}	ป้ายกำกับที่จะนำรูปภาพหน้าจามาใส่
3	{#page}	เริ่มต้นการวนซ้ำ
4	{page_name}	แสดงชื่อเพจ
5	{shapes}	จำนวนส่วนประกอบในเพจ
6	{/pages}	ปิดการวนซ้ำ
7	{shape}	เริ่มต้นการวนซ้ำของส่วนประกอบ
8	{shape_id}	แสดงรหัสของส่วนประกอบ
9	{shape_name}	แสดงชื่อส่วนประกอบ
10	{shape_value}	แสดงค่าของส่วนประกอบ
11	{size}	แสดงค่าขนาดของส่วนประกอบ
12	{max_length}	แสดงค่าสูงสุดในการรับข้อมูลของส่วนประกอบ
13	{source}	แสดงแหล่งที่มาของส่วนประกอบ
14	{event}	แสดงเหตุการณ์ของส่วนประกอบ
15	{readonly}	แสดงค่าคุณลักษณะห้ามแก้ไข
16	{disabled}	แสดงค่าคุณลักษณะห้ามใช้งาน
17	{required}	แสดงค่าคุณลักษณะนี้ต้องการ
18	{description_req}	แสดงค่าคำอธิบาย
19	{/shape}	ปิดการวนซ้ำของส่วนประกอบ

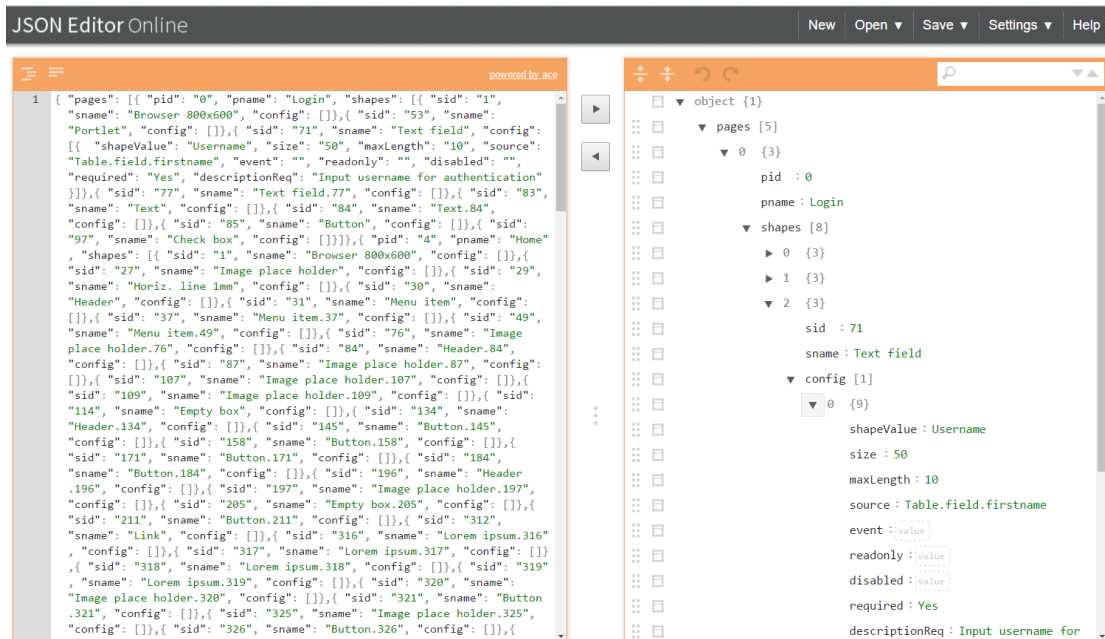
```

main.component.ts dev
211
212   sendJson(url: string) {
213     var data;
214     var pages = this.dataJson.pages;
215     var result;
216
217     //console.log(this.dataJson);
218
219     //gathering
220     result = '{ "pages": [';
221     for (var i = 0; i < pages.length; i++){
222       var p = pages[i].pname;
223       var s = pages[i].shapes;
224       //page json
225       result += '{ "pid": "'+pages[i].pid+'", "pname": "'+p+'", "shapes": [';
226       for(var j =0; j < s.length; j++){
227         //shape json
228         result += '{ "sid": "'+s[j].id+'", "sname": "'+s[j].name+'", "config": [';
229         if (localStorage[p+'.'+s[j].id]){
230
231           console.log("Has configured "+p+"."+s[j].id);
232           //console.log(localStorage[p+'.'+s[j].id]);
233           data = JSON.parse(localStorage[p+'.'+s[j].id]);
234           //has configured
235           if(data.config.length > 0) {
236             result += "{ "
237             for(var k = 0; k < data.config.length; k++) {
238               switch (k) {
239                 case 0:
240                   result += ' "shapeValue": "'+data.config[k].shapeValue+'";
241                   break;
242                 case 1:
243                   result += ' "size": "'+data.config[k].size+'";
244                   break;
245                 case 2:
246                   result += ' "maxLength": "'+data.config[k].maxLength+'";
247                   break;
248                 case 3:
249                   result += ' "source": "'+data.config[k].source+'";
250                   break;
251                 case 4:
252                   result += ' "event": "'+data.config[k].event+'";
253                   break;
254                 case 5:
255                   result += ' "readonly": "'+data.config[k].readonly+'";
256                   break;
257                 case 6:
258                   result += ' "disabled": "'+data.config[k].disabled+'";
259                   break;
260                 case 7:

```

รูปที่ 36 ตัวอย่างโค้ดในการส่งข้อมูลไปทำการสร้างเอกสารต้นแบบ

รูปที่ 36 พัฒนาฟังก์ชันในการส่งข้อมูลที่อยู่ในโลคอล สตอเรจทั้งหมดที่เก็บค่าการเพิ่มคุณลักษณะของแต่ละส่วนประกอบ โดยการอ่านค่าในโลคอล สตอเรจออกมาทั้งหมดแล้วทำการแปลงเป็นข้อมูลเจสันเพื่อที่จะส่งต่อไปยังโหนดเจเอสผ่านทางยูอาร์แอล “/send” ด้วยวิธีการ POST กล่าวคือ การผ่านข้อมูลเจสันแบบอ็อบเจกต์เจอไปยังโหนดเจเอส เพื่อทำการสร้างเอกสารอธิบายต้นแบบ รูปที่ 37 รูปแบบเจสันที่ส่งไปยังโหนดเจเอส โดยจะเพิ่มข้อมูลในส่วนของคุณลักษณะที่ได้ทำการเพิ่มเข้าไปด้วย



รูปที่ 37 ข้อมูลที่แปลงเป็นเจสันเพื่อส่งไปสร้างเอกสารอธิบายต้นแบบ

พัฒนาโหนดเจเอสเพื่อรับการร้องขอในส่วนของการสร้างเอกสาร เมื่อโหนดเจเอสได้รับอ็อบเจกต์เจสันที่ส่งเข้ามาแล้วก็จะทำการแปลงเจสันให้อยู่ในรูปของการสร้างเอกสาร โดยมีรูปแบบดังรูปที่ 38 เป็นตัวอย่างในการระบุป้ายกำกับและระบุค่าให้กับตัวแปรชื่อ config เริ่มจากการระบุป้ายกำกับ image ซึ่งจะตรงในเอกสารแม่แบบที่ป้ายกำกับ {image} ถัดมาคือการระบุป้ายกำกับ screen\_name ซึ่งก็จะไปตรงในเอกสารแม่แบบที่ป้ายกำกับ {screen\_name} จากนั้นจะเข้าไปในส่วนที่เป็นการวนซ้ำ ซึ่งจะต้องระบุเป็นค่าอะไรก็ได้ จากตัวอย่างระบุเป็นอะไรก็ได้ pages ที่ประกอบไปด้วยป้ายกำกับ page\_id page\_name และ shapes เช่นเดียวกันกับอะไรก็ได้ shapes ภายใต้ป้ายกำกับวงเล็บปีกกา สามารถระบุค่าลงไปได้มากกว่าหนึ่งค่าโดยใช้เครื่องหมายจุลภาคในการแบ่ง ตัวอย่าง [ { value1: "1"}, { value2: "2"} ] ก็จะได้อะไรก็ได้เก็บค่าชุดข้อมูล 2 ชุด

```

196   var config = {}
197   config = {
198       image:"Screen.png",
199       screen_name:"Login",
200       "pages": [{
201           "page_id":"0",
202           "page_name":"Login",
203           "shapes":"47"
204       }],
205       "shapes":[{"
206           "shape_id":"72",
207           "shape_name":"Doe",
208           "shape_value":"Doe",|
209           "size": "",
210           "max_length": "",
211           "source": "",
212           "event": "",
213           "readonly": "",
214           "disabled": "",
215           "required": "",
216           "description_req": ""
217       }]}
218   }
219

```

รูปที่ 38 ตัวอย่างการกำหนดค่าและป้ายกำกับให้เอกสารแม่แบบ

```

• app.js C:\Users\Napajorn.w\Google Drive\Angular Projects\backend
302 app.post('/send', function(req, res, next) {
303   var dataJson = req.body.pages;
304   var result = "";
305   var content = fs
306     .readFileSync(__dirname + "\\Template-Example.docx", "binary");
307   var docx = new Docxtemplater(content);
308   var opts = {}
309   opts.centered = true;
310   opts.getImage=function(tagValue, tagName) {
311     return fs.readFileSync(tagValue,'binary');
312   }
313   opts.getSize=function(img, tagValue, tagName) {
314     return [420,550];
315   }
316   var imageModule=new ImageModule(opts);
317   for (var i = 0; i < dataJson.length; i++){
318     var p = dataJson[i].pname;
319     var s = dataJson[i].shapes;
320     count=0;
321     result += ""image'+(i + 1)+'": "uploads/'+p+'.png",';
322     result += ""screen_name'+(i + 1)+'": "'+p+'",';
323     result += ""pages'+(i + 1)+'": [{ ";
324     result += "   "page_id": '"+dataJson[i].pid+'",';
325     result += "   "page_name": "'+p+'",';
326     result += "   "shapes": '"+dataJson[i].shapes.length+'";
327     result += '}],';
328     result += ""shapes'+(i + 1)+'": [ ";
329
330     for(var j = 0; j < s.length; j++){
331       var c = s[j].config;
332       if(c.length > 0) {
333         result += '{ ';
334         result += "  "shape_id": '"+s[j].sid+'",';
335         (c.length > 0 ? result += "  "shape_name": '"+s[j].sname+'", " : result += "  "shape_name": '"+s[j]
336           for(var k = 0; k < c.length; k++){
337             result += "    " : '"+s[j].config[k].shapeValue+'",';
338             result += "    "size": '"+s[j].config[k].size+'",';
339             result += "    "max_length": '"+s[j].config[k].maxLength+'",';
340             result += "    "source": '"+s[j].config[k].source+'",';
341             result += "    "event": '"+s[j].config[k].event+'",';
342             result += "    "readonly": '"+s[j].config[k].readonly+'",';
343             result += "    "disabled": '"+s[j].config[k].disabled+'",';
344             result += "    "required": '"+s[j].config[k].required+'",';
345             result += "    "description_req": '"+s[j].config[k].descriptionReq+'";
346             count++;
347           }
348           (j == count ? result += "}" : result += "},");
349         }
350       }

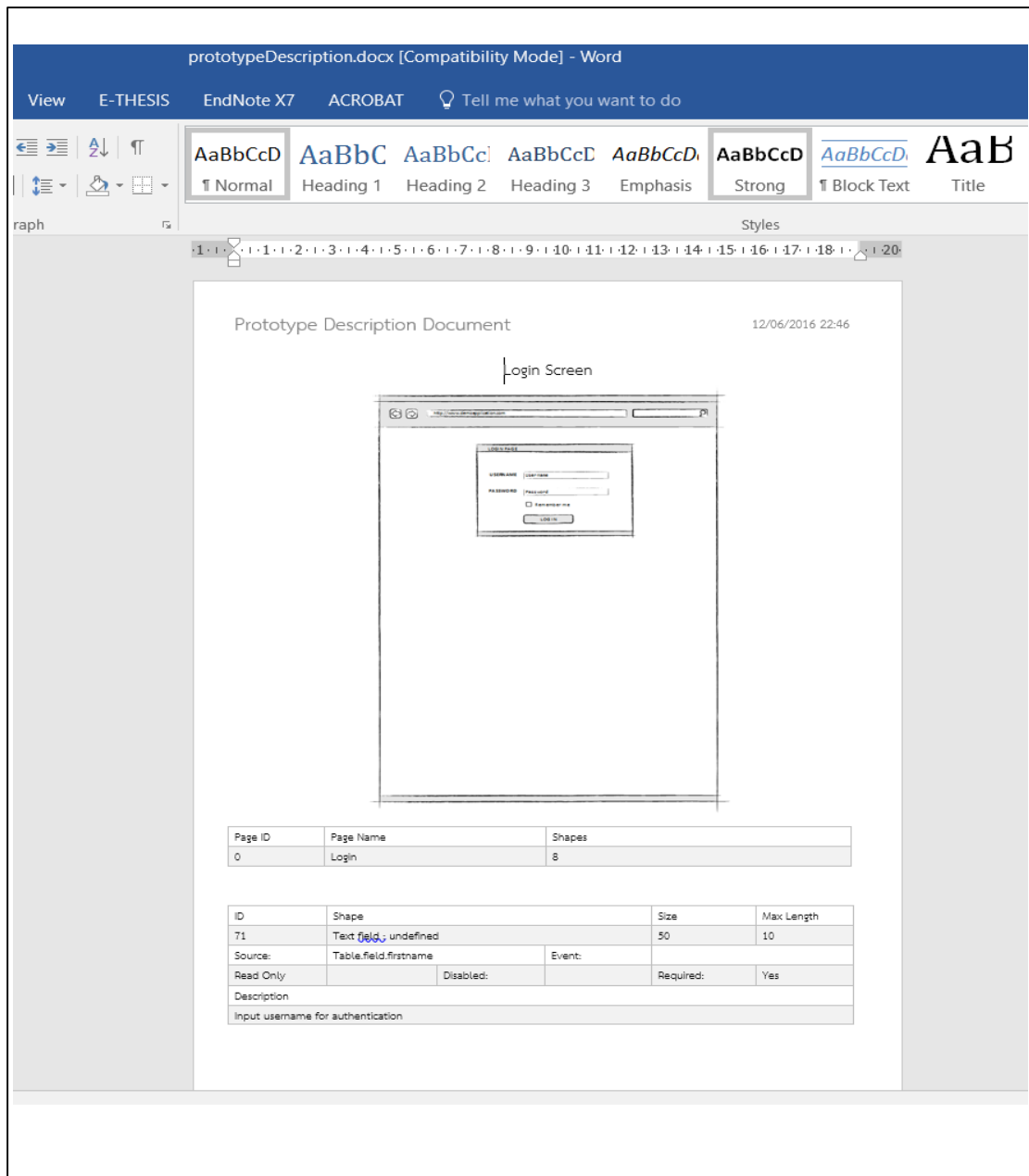
```

รูปที่ 39 โค้ดในการพัฒนาฟังก์ชันการสร้างเอกสารอธิบายต้นแบบ

รูปที่ 39 ฟังก์ชันการสร้างเอกสารอธิบายต้นแบบ จะเริ่มจากการรับข้อมูลที่แปลงเป็นเจสันมาแล้ว (รูปที่ 37) ทำการแปลงข้อมูลเจสันโดยการจับคู่ให้เข้ากับป้ายกำกับที่ระบุไว้ในเอกสารแม่แบบ จากนั้นทำการอ่านเพิ่มข้อมูลเอกสารแม่แบบขึ้นมาเพื่อนำค่าที่จับคู่ตามป้ายกำกับไปใส่ในเอกสารแม่แบบ เมื่อนำข้อมูลไปใส่ครบหมดแล้วก็ทำการเขียนเพิ่มข้อมูลใหม่โดยเปลี่ยนชื่อเป็น “PrototypeDescription.docx” แล้วก็ส่งเพิ่มข้อมูลกลับไปยังการร้องขอที่เรียกเข้ามาตามรูปที่ 40

รูปที่ 40 เอกสารอธิบายต้นแบบถูกส่งมายังผู้ใช้งาน

รูปที่ 40 แสดงการส่งเอกสารอธิบายต้นแบบมายังผู้ใช้งานโดยการดาวน์โหลดผ่านทางเว็บแอปพลิเคชัน ผู้ใช้งานสามารถที่จะเพิ่มคุณลักษณะเพิ่มแล้วกดปุ่ม generate ใหม่ได้เรื่อยๆตามความต้องการ รูปที่ 41 แสดงเอกสารอธิบายต้นแบบที่ได้นำข้อมูลที่สกัดและเพิ่มเติมคุณลักษณะ นำมาใส่เรียบร้อยแล้วทั้งรูปภาพหน้าจอแต่ละหน้าจอได้ถูกต้อง



รูปที่ 41 เอกสารอธิบายต้นแบบพร้อมข้อมูล

## บทที่ 5

### การทดสอบและวิเคราะห์ผล

#### 5.1 วัตถุประสงค์ของการทดสอบ

เพื่อสนับสนุนแนวทางอัตโนมัติและเครื่องมือสำหรับสร้างเอกสารจากต้นแบบรวดเร็ว ที่ได้พัฒนาในบทที่ 4 โดยการทดสอบเริ่มจากนำต้นแบบรวดเร็วที่สร้างผ่านทางโปรแกรมไมโครซอฟต์วิซิโอกับเครื่องมือสแตนซิล สเกตซ์ จียูไอ เซพ นำไปสกัดข้อมูลจากต้นแบบออกมาเพื่อให้ผู้ใช้งานได้ทำการเพิ่มคุณลักษณะที่ต้องการเพื่อนำไปสร้างเอกสารอธิบายต้นแบบ สุดท้ายผลทดสอบสำเร็จก็คือการได้รับเอกสารอธิบายต้นแบบ ที่มีข้อมูลที่ได้เพิ่มคุณลักษณะถูกต้องครบถ้วนสมบูรณ์

#### 5.2 การทดสอบความถูกต้องของฟังก์ชันการทำงานของระบบ

การทดสอบระบบจะกระทำการทดสอบแบบ แบล็ก บ็อก (Black Box Testing) ซึ่งจะเป็นการทดสอบที่จะดูว่าผลลัพธ์ที่ได้ออกมานั้นถูกต้อง ตรงตามความต้องการ โดยจะทำการทดสอบแต่ละส่วนของระบบดังรายการต่อไปนี้

- 1) ทดสอบการนำเข้าต้นแบบรวดเร็วในรูปแบบแฟ้มข้อมูลเอกซ์เอ็มแอล

ทำการทดสอบการอัปโหลดแฟ้มข้อมูลเอกซ์เอ็มแอลและรูปภาพ เพื่อนำมาสกัดข้อมูลที่สำคัญที่จะนำไปกำหนดคุณลักษณะเพิ่มเติม และสุดท้ายข้อมูลจะถูกนำไปใส่ในเอกสารอธิบายต้นแบบ

- 2) ทดสอบการสกัดข้อมูลจากแฟ้มเอกซ์เอ็มแอล

ทำการทดสอบการสกัดข้อมูลที่ได้มาจากข้อ 1 วัตถุประสงค์เพื่อตรวจสอบว่าส่วนประกอบทุกส่วนได้ถูกสกัดมาถูกต้องและครบถ้วน เพื่อที่จะเตรียมนำไปแปลงให้อยู่ในรูปแบบของเจสันในขั้นตอนทดสอบถัดไป

- 3) ทดสอบการแปลงข้อมูลจากเอกซ์เอ็มแอลไปอยู่ในรูปแบบเจสัน

ทดสอบผลในการแปลงข้อมูลจากแฟ้มเอกซ์เอ็มแอลให้อยู่ในรูปแบบของเจสันที่ใช้รับส่งข้อมูลระหว่างโหนดเจเอสและแองกูลาร์

- 4) ทดสอบการกำหนดคุณลักษณะเพิ่มเติมให้กับส่วนประกอบ

ทดสอบผลการกำหนดคุณลักษณะเพิ่มเติม วัตถุประสงค์เพื่อให้ส่วนประกอบสามารถเพิ่มคุณลักษณะที่ต้องการเข้าไปได้ อีกทั้งสามารถแก้ไขค่า และสามารถส่งค่าข้อมูลเพื่อนำไปสร้างเอกสารอธิบายต้นแบบได้

## 5) ทดสอบการสร้างเอกสารอัตโนมัติ

ทดสอบการสร้างเอกสารอัตโนมัติโดยข้อมูลได้มาจากการสกดและที่ได้กำหนดคุณลักษณะเพิ่มเข้าไป วัตถุประสงค์ คือ เอกสารอธิบายต้นแบบจะต้องมีข้อมูลครบถ้วนและถูกต้องสมบูรณ์

วัตถุประสงค์ในการทดสอบเพื่อจะยืนยันความถูกต้องการทำงานของระบบ โดยการทดสอบจะกระทำการทดสอบตามรายการตารางด้านล่าง ดังต่อไปนี้

ตารางที่ 14 ทดสอบการนำเข้าต้นแบบรวดเร็วในรูปแบบเพิ่มข้อมูลเอกซ์เอ็มแอล

รหัส	การทดสอบ	ผลที่คาดหวัง	ผลลัพธ์
TC01	การนำเข้าต้นแบบรวดเร็วในรูปแบบเพิ่มข้อมูลเอกซ์เอ็มแอล	ระบบจะต้องนำเพิ่มข้อมูลเอกซ์เอ็มแอลไปเขียนบันทึกลงในฝั่งเซิร์ฟเวอร์	ถูกต้อง
TC02	การนำเข้ารูปภาพหน้าจอบนรูปแบบเพิ่มข้อมูลพีจีเอ็น	ระบบจะต้องนำเพิ่มข้อมูลพีเอ็นจีไปบันทึกเขียนลงในฝั่งเซิร์ฟเวอร์	ถูกต้อง

ตารางที่ 15 ทดสอบการสกดข้อมูลจากเพิ่มเอกซ์เอ็มแอล

รหัส	การทดสอบ	ผลที่คาดหวัง	ผลลัพธ์
TC03	ตรวจสอบว่ามีเพิ่มข้อมูลเอกซ์เอ็มแอลพร้อมสกดข้อมูล	ถ้าไม่พบเพิ่มข้อมูลเอกซ์เอ็มแอลจะต้องมีข้อความแจ้งเตือน	ถูกต้อง
TC04	สกดข้อมูลจากเพิ่มข้อมูลเอกซ์เอ็มแอลกับป้ายกำกับที่มีชื่อ Header	ระบบต้องสกดข้อมูลพร้อมแปลงเป็นเจสันเพื่อส่งกลับมาเพิ่มคุณลักษณะที่หน้าเบราเซอร์ได้อย่างถูกต้อง	ถูกต้อง
TC05	สกดข้อมูลจากเพิ่มข้อมูลเอกซ์เอ็มแอลกับป้ายกำกับที่มีชื่อ Text	ระบบต้องสกดข้อมูลพร้อมแปลงเป็นเจสันเพื่อส่งกลับมาเพิ่มคุณลักษณะที่หน้าเบราเซอร์ได้อย่างถูกต้อง	ถูกต้อง
TC06	สกดข้อมูลจากเพิ่มข้อมูลเอกซ์เอ็มแอลกับป้ายกำกับที่มีชื่อ Lorem Ipsum	ระบบต้องสกดข้อมูลพร้อมแปลงเป็นเจสันเพื่อส่งกลับมาเพิ่มคุณลักษณะที่หน้าเบราเซอร์ได้อย่างถูกต้อง	ถูกต้อง
TC07	สกดข้อมูลจากเพิ่มข้อมูลเอกซ์เอ็มแอลกับป้ายกำกับที่มีชื่อ Link	ระบบต้องสกดข้อมูลพร้อมแปลงเป็นเจสันเพื่อส่งกลับมาเพิ่มคุณลักษณะที่หน้าเบราเซอร์ได้อย่างถูกต้อง	ถูกต้อง





TC18	สกัดข้อมูลจากแฟ้มข้อมูลเอกซ์เอ็มแอลกับป้ายกำกับที่มีชื่อ Menu/Combo Box Drop-Down	ระบบต้องสกัดข้อมูลพร้อมแปลงเป็นเจสันเพื่อส่งกลับมาเพิ่มคุณลักษณะที่หน้าเบรราเซอร์ได้อย่างถูกต้อง	ถูกต้อง
TC19	สกัดข้อมูลจากแฟ้มข้อมูลเอกซ์เอ็มแอลกับป้ายกำกับที่มีชื่อ Browser 800x600	ระบบต้องสกัดข้อมูลพร้อมแปลงเป็นเจสันเพื่อส่งกลับมาเพิ่มคุณลักษณะที่หน้าเบรราเซอร์ได้อย่างถูกต้อง	ถูกต้อง
TC20	สกัดข้อมูลจากแฟ้มข้อมูลเอกซ์เอ็มแอลกับป้ายกำกับที่มีชื่อ Portlet	ระบบต้องสกัดข้อมูลพร้อมแปลงเป็นเจสันเพื่อส่งกลับมาเพิ่มคุณลักษณะที่หน้าเบรราเซอร์ได้อย่างถูกต้อง	ถูกต้อง
TC21	สกัดข้อมูลจากแฟ้มข้อมูลเอกซ์เอ็มแอลกับป้ายกำกับที่มีชื่อ Grid column	ระบบต้องสกัดข้อมูลพร้อมแปลงเป็นเจสันเพื่อส่งกลับมาเพิ่มคุณลักษณะที่หน้าเบรราเซอร์ได้อย่างถูกต้อง	ถูกต้อง

ตารางที่ 16 ทดสอบการแปลงข้อมูลเป็นเจสัน

รหัส	การทดสอบ	ผลที่คาดหวัง	ผลลัพธ์
TC22	แสดงผลลัพธ์จากการสกัดทั้งหมดในรูปแบบเจสันเมื่อเรียกใช้งานผ่านทาง URL “/extract”	แสดงค่าที่ส่งกลับคืนมาในรูปแบบของเจสัน	ถูกต้อง

ตารางที่ 17 ทดสอบการกำหนดคุณลักษณะเพิ่มเติม

รหัส	การทดสอบ	ผลที่คาดหวัง	ผลลัพธ์
TC23	แสดงผลเมื่อมีการเลือกส่วนประกอบชื่อ Header เพื่อนำมาเพิ่มคุณลักษณะ	ระบบต้องแสดงผลและบันทึกข้อมูลการเพิ่มคุณลักษณะให้ส่วนประกอบที่ชื่อ Header ได้ถูกต้อง	ถูกต้อง
TC24	แสดงผลเมื่อมีการเลือกส่วนประกอบชื่อ Text เพื่อนำมาเพิ่มคุณลักษณะ	ระบบต้องแสดงผลและบันทึกข้อมูลการเพิ่มคุณลักษณะให้ส่วนประกอบที่ชื่อ Text ได้ถูกต้อง	ถูกต้อง
TC25	แสดงผลเมื่อมีการเลือกส่วนประกอบชื่อ Lorem Ipsum เพื่อนำมาเพิ่มคุณลักษณะ	ระบบต้องแสดงผลและบันทึกข้อมูลการเพิ่มคุณลักษณะให้ส่วนประกอบที่ชื่อ Lorem ipsum ได้ถูกต้อง	ถูกต้อง



TC35	แสดงผลเมื่อมีการเลือกส่วนประกอบ ชื่อ Text Field เพื่อนำมาเพิ่มคุณลักษณะ	ระบบต้องแสดงผลและบันทึกข้อมูล การเพิ่มคุณลักษณะให้ส่วนประกอบที่ ชื่อ Text Field ได้ถูกต้อง	ถูกต้อง
TC36	แสดงผลเมื่อมีการเลือกส่วนประกอบ ชื่อ Combo Box เพื่อนำมาเพิ่มคุณลักษณะ	ระบบต้องแสดงผลและบันทึกข้อมูล การเพิ่มคุณลักษณะให้ส่วนประกอบที่ ชื่อ Combo Box ได้ถูกต้อง	ถูกต้อง
TC37	แสดงผลเมื่อมีการเลือกส่วนประกอบ ชื่อ Menu / Combo Box Drop-Down เพื่อนำมาเพิ่มคุณลักษณะ	ระบบต้องแสดงผลและบันทึกข้อมูล การเพิ่มคุณลักษณะให้ส่วนประกอบที่ ชื่อ Menu / Combo Box Drop-Down ได้ถูกต้อง	ถูกต้อง
TC38	แสดงผลเมื่อมีการเลือกส่วนประกอบ ชื่อ Browser 800x600 เพื่อนำมาเพิ่มคุณลักษณะ	ระบบต้องแสดงผลและบันทึกข้อมูล การเพิ่มคุณลักษณะให้ส่วนประกอบที่ ชื่อ Browser 800x600 ได้ถูกต้อง	ถูกต้อง
TC39	แสดงผลเมื่อมีการเลือกส่วนประกอบ ชื่อ Portlet เพื่อนำมาเพิ่มคุณลักษณะ	ระบบต้องแสดงผลและบันทึกข้อมูล การเพิ่มคุณลักษณะให้ส่วนประกอบที่ ชื่อ Portlet ได้ถูกต้อง	ถูกต้อง
TC40	แสดงผลเมื่อมีการเลือกส่วนประกอบ ชื่อ Grid Column เพื่อนำมาเพิ่มคุณลักษณะ	ระบบต้องแสดงผลและบันทึกข้อมูล การเพิ่มคุณลักษณะให้ส่วนประกอบที่ ชื่อ Grid Column ได้ถูกต้อง	ถูกต้อง
TC41	ระบบแสดงรูปภาพหน้าจอได้ถูกต้อง	เมื่อเลือกส่วนประกอบระบบจะต้องแสดงหน้าจอของส่วนประกอบนั้น	ถูกต้อง
TC42	การเรียกดูรูปภาพหน้าจอผ่านทาง URL “/image/screen.png”	สามารถแสดงรูปภาพหน้าจอผ่านทาง URL “/image/screen.png”	ถูกต้อง

ตารางที่ 18 ทดสอบการสร้างเอกสารอัตโนมัติ

รหัส	การทดสอบ	ผลที่คาดหวัง	ผลลัพธ์
TC43	ทำการส่งข้อมูลที่มีการกำหนดคุณลักษณะชื่อ Name กลับไปสร้างเอกสาร	ส่งผลการกำหนดคุณลักษณะชื่อ Name ในรูปแบบเจสันไปสร้างเอกสารได้ถูกต้อง	ถูกต้อง

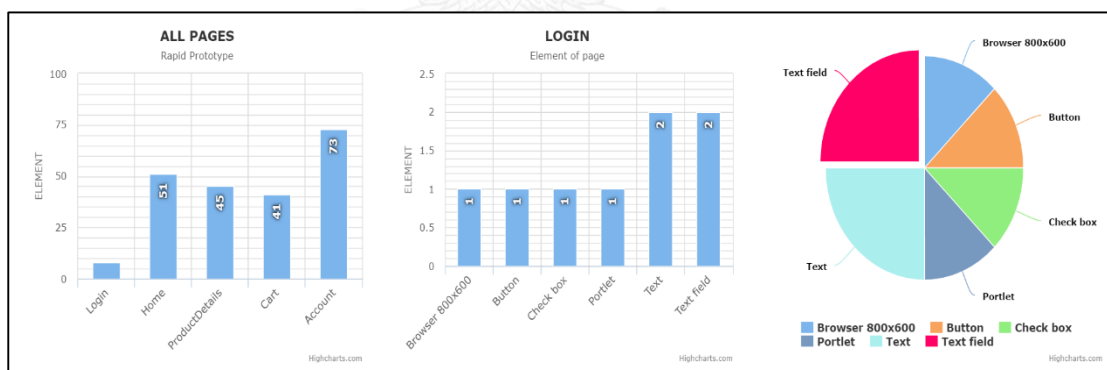
TC44	ทำการส่งข้อมูลที่มีการกำหนด คุณลักษณะชื่อ Value กลับไปสร้าง เอกสาร	ส่งผลการกำหนดคุณลักษณะชื่อ Value ในรูปแบบเจสันไปสร้างเอกสาร ได้ถูกต้อง	ถูกต้อง
TC45	ทำการส่งข้อมูลที่มีการกำหนด คุณลักษณะชื่อ Size กลับไปสร้าง เอกสาร	ส่งผลการกำหนดคุณลักษณะชื่อ Size ในรูปแบบเจสันไปสร้างเอกสารได้ ถูกต้อง	ถูกต้อง
TC46	ทำการส่งข้อมูลที่มีการกำหนด คุณลักษณะชื่อ Max Length กลับไปสร้างเอกสาร	ส่งผลการกำหนดคุณลักษณะชื่อ Max Length ในรูปแบบเจสันไปสร้าง เอกสารได้ถูกต้อง	ถูกต้อง
TC47	ทำการส่งข้อมูลที่มีการกำหนด คุณลักษณะชื่อ Source กลับไปสร้าง เอกสาร	ส่งผลการกำหนดคุณลักษณะชื่อ Source ในรูปแบบเจสันไปสร้าง เอกสารได้ถูกต้อง	ถูกต้อง
TC48	ทำการส่งข้อมูลที่มีการกำหนด คุณลักษณะชื่อ Event กลับไปสร้าง เอกสาร	ส่งผลการกำหนดคุณลักษณะชื่อ Event ในรูปแบบเจสันไปสร้างเอกสาร ได้ถูกต้อง	ถูกต้อง
TC49	ทำการส่งข้อมูลที่มีการกำหนด คุณลักษณะชื่อ Read Only กลับไป สร้างเอกสาร	ส่งผลการกำหนดคุณลักษณะชื่อ Read Only ในรูปแบบเจสันไปสร้างเอกสาร ได้ถูกต้อง	ถูกต้อง
TC50	ทำการส่งข้อมูลที่มีการกำหนด คุณลักษณะชื่อ Disabled กลับไป สร้างเอกสาร	ส่งผลการกำหนดคุณลักษณะชื่อ Disabled ในรูปแบบเจสันไปสร้าง เอกสารได้ถูกต้อง	ถูกต้อง
TC51	ทำการส่งข้อมูลที่มีการกำหนด คุณลักษณะชื่อ Required กลับไป สร้างเอกสาร	ส่งผลการกำหนดคุณลักษณะชื่อ Required ในรูปแบบเจสันไปสร้าง เอกสารได้ถูกต้อง	ถูกต้อง
TC52	ทำการส่งข้อมูลที่มีการกำหนด คุณลักษณะชื่อ Description Requirements กลับไปสร้าง เอกสาร	ส่งผลการกำหนดคุณลักษณะชื่อ Description Requirements ใน รูปแบบเจสันไปสร้างเอกสารได้ถูกต้อง	ถูกต้อง
TC53	ระบบส่งเอกสารอธิบายต้นแบบมาให้ ผู้ใช้งานได้ถูกต้อง	ได้รับเอกสารอธิบายต้นแบบพร้อม ข้อมูลถูกต้อง	ถูกต้อง

### 5.3 สรุปผลการทดสอบ

การทดสอบตามข้อ 5.2 ผู้วิจัยขอสรุปผลการทดสอบดังต่อไปนี้

- 1) การนำเข้าต้นแบบรวดเร็วในรูปแบบแฟ้มข้อมูลเอกซ์เอ็มแอลและรูปภาพหน้าจอสามารถนำเข้ามาและบันทึกได้อย่างถูกต้อง ทำให้สามารถนำแฟ้มข้อมูลเอกซ์เอ็มแอลไปใช้ในการสกัดข้อมูลได้ และรูปภาพหน้าจอใช้เป็นส่วนหนึ่งในการสร้างเอกสาร
- 2) การสกัดข้อมูลจากแฟ้มเอกซ์เอ็มแอล สามารถสกัดข้อมูลป้ายกำกับได้ถูกต้องครบถ้วน
- 3) การแปลงข้อมูลจากแฟ้มข้อมูลเอกซ์เอ็มแอลไปอยู่ในรูปแบบเจสันได้ถูกต้อง ทำให้สามารถใช้ในการสื่อสารกับส่วนอื่นได้เป็นอย่างดี
- 4) การกำหนดคุณลักษณะเพิ่มเติมให้กับส่วนประกอบ สามารถทำการเพิ่มคุณลักษณะเพิ่มเติมได้อย่างถูกต้อง ซึ่งทำให้ส่วนประกอบสามารถเก็บคุณลักษณะที่มีประโยชน์ที่จะนำไปใช้ในการสร้างเอกสารอธิบายต้นแบบได้เป็นอย่างดี
- 5) การสร้างเอกสารอัตโนมัติ ข้อมูลทั้งหมดถูกส่งมาอย่างถูกต้องและครบถ้วน สามารถนำไปใส่ในเอกสารอธิบายต้นแบบได้ตรงตามที่กำหนดมา

จากการทำการทดสอบโปรแกรม โปรแกรมสามารถทำงานได้ถูกต้องครบถ้วนสมบูรณ์ โดยผู้วิจัยจะทำการสรุปภาพรวมของโปรแกรมในรูปแบบกราฟ ดังต่อไปนี้



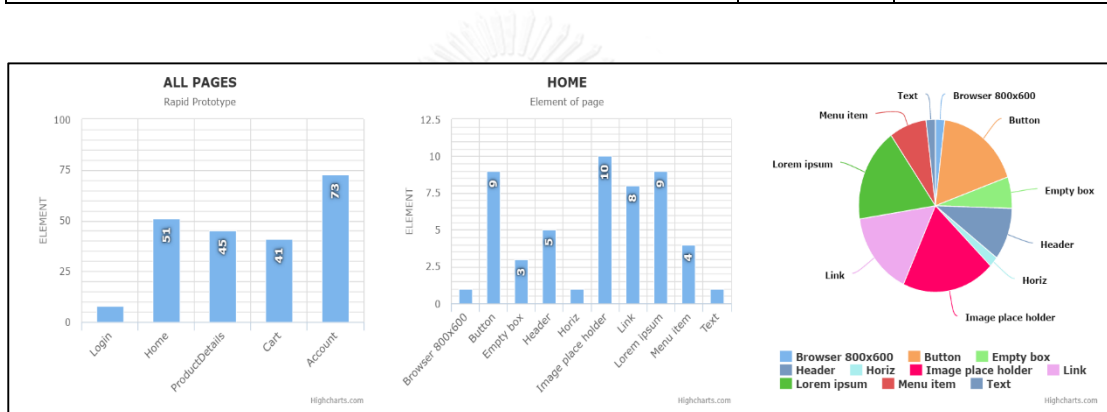
รูปที่ 42 กราฟสรุปรายละเอียดของส่วนประกอบ Login Page

ตารางที่ 19 รายละเอียดสัดส่วนของส่วนประกอบของต้นแบบรวดเร็ว

Login Page	ประกอบไปด้วย 8 ส่วนประกอบ
Home Page	ประกอบไปด้วย 51 ส่วนประกอบ
ProductDetails Page	ประกอบไปด้วย 45 ส่วนประกอบ
Cart Page	ประกอบไปด้วย 41 ส่วนประกอบ
Account Page	ประกอบไปด้วย 73 ส่วนประกอบ

ตารางที่ 20 ส่วนประกอบ Login Page

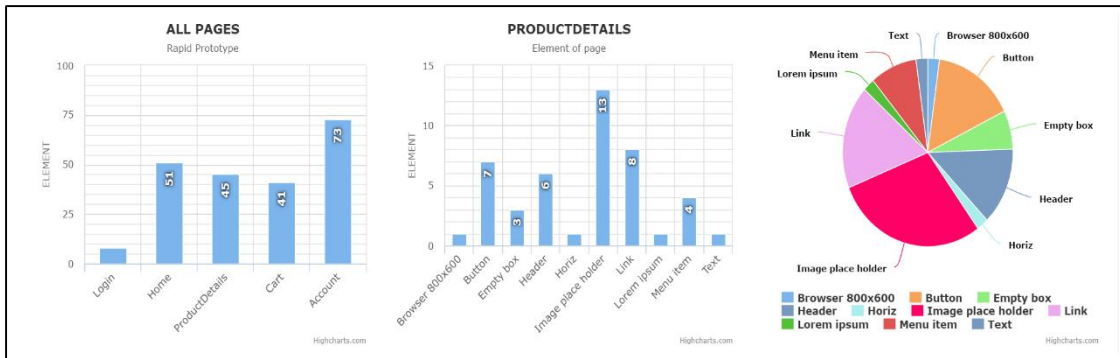
ส่วนประกอบ	จำนวน	คิดเป็นสัดส่วน
Browser 800x600	1	12.5%
Button	1	12.5%
Check box	1	12.5%
Portlet	1	12.5%
Text	2	25%
Text field	2	25%
รวมทั้งหมด	8	100%



รูปที่ 43 กราฟสรุปรายละเอียดของส่วนประกอบ Home Page

ตารางที่ 21 ส่วนประกอบ Home Page

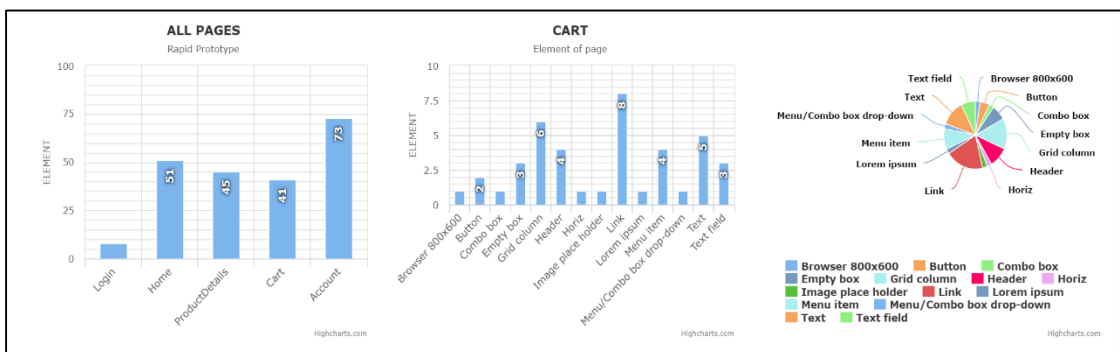
ส่วนประกอบ	จำนวน	คิดเป็นสัดส่วน
Browser 800x600	1	2%
Button	9	17.6%
Empty box	3	5.9%
Header	5	9.8%
Horiz	1	2%
Image place holder	10	19.6%
Link	8	15.7%
Lorem ipsum	9	17.6%
Menu item	4	7.8%
Text	1	2%
รวมทั้งหมด	51	100%



รูปที่ 44 กราฟสรุปรายละเอียดของส่วนประกอบ ProductDetails Page

ตารางที่ 22 ส่วนประกอบ ProductDetails Page

ส่วนประกอบ	จำนวน	คิดเป็นสัดส่วน
Browser 800x600	1	2.2%
Button	7	15.6%
Empty box	3	6.7%
Header	6	13.3%
Horiz	1	2.2%
Image place holder	13	28.9%
Link	8	17.8%
Lorem ipsum	1	2.2%
Menu item	4	8.9%
Text	1	2.2%
รวมทั้งหมด	45	100%

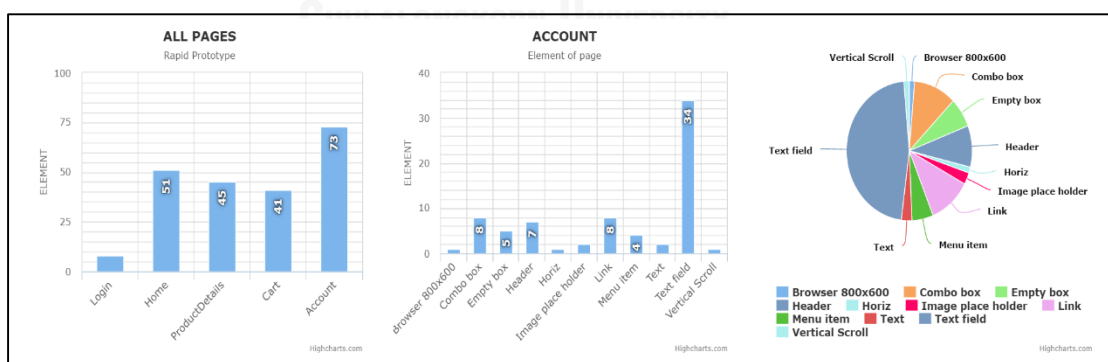


รูปที่ 45 กราฟสรุปรายละเอียดของส่วนประกอบ Cart Page



ตารางที่ 23 ส่วนประกอบ Cart Page

ส่วนประกอบ	จำนวน	คิดเป็นสัดส่วน
Browser 800x600	1	2.4%
Button	2	4.9%
Combo box	1	2.4%
Empty box	3	7.3%
Grid column	6	14.6%
Header	4	9.8%
Horiz	1	2.4%
Image place holder	1	2.4%
Link	8	19.5%
Lorem ipsum	1	2.4%
Menu item	4	9.8%
Menu/Combo box drop-down	1	2.4%
Text	5	12.2%
Text field	3	7.3%
รวมทั้งหมด	41	100%



รูปที่ 46 กราฟสรุปรายละเอียดของส่วนประกอบ Account Page

ตารางที่ 24 ส่วนประกอบ Account Page

ส่วนประกอบ	จำนวน	คิดเป็นสัดส่วน
Browser 800x600	1	1.4%
Combo box	8	11%
Empty box	5	6.8%
Header	7	9.6%
Horiz	1	1.4%
Image place holder	2	2.7%
Link	8	11%
Menu item	4	5.5%
Text	2	2.7%
Text field	34	46.6%
Vertical scroll	1	1.4%
รวมทั้งหมด	73	100%

จากต้นแบบรวดเร็วที่ออกแบบเมื่อนำมาสร้างเอกสารอัตโนมัติโดยทำการกำหนดคุณสมบัติให้กับส่วนประกอบทั้งหมดจำนวน 218 ส่วนประกอบจาก 5 เพจ จะสามารถสร้างเอกสารอธิบายต้นแบบได้ประมาณ 51 หน้า โดยใช้เวลาประมาณ 30 นาทีในการสร้างผ่านโปรแกรมและแนวทางที่ได้นำเสนอไป จะเห็นได้ว่าแนวทางนี้ได้ช่วยลดเวลาในการสร้างเอกสารอธิบายต้นแบบลดลง

## บทที่ 6

### สรุปผลการวิจัย

บทที่ 6 จะกล่าวถึงการสรุปผลวิจัยรวมถึงข้อจำกัดในงานวิจัยชิ้นนี้ และแนวทางสำหรับการวิจัยต่อไปในอนาคต โดยจะกล่าวดังต่อไปนี้

#### 6.1 สรุปผลการวิจัย

การสร้างต้นแบบรวดเร็วเป็นวิธีการที่มีประสิทธิภาพในการตรวจสอบความต้องการซอฟต์แวร์ ทำให้รู้ถึงความต้องการซอฟต์แวร์ชัดเจนมากยิ่งขึ้น แต่หลังจากการสร้างต้นแบบรวดเร็วแล้วต้นแบบเร็วนั้นก็กลับไม่ได้ถูกนำมาใช้ประโยชน์ต่อ และการทำเอกสารจากต้นแบบเร็วนี้เป็นขั้นตอนที่สิ้นเปลืองทรัพยากร งานวิจัยนี้จึงเสนอการเตรียมเอกสารอัตโนมัติสำหรับต้นแบบรวดเร็ว ที่ประกอบไปด้วยการสร้างต้นแบบรวดเร็วด้วยเครื่องมือไมโครซอฟต์แวร์วิซิโอและสแตนด์ลิสเกตซ์ จียูไอ เซพ จากนั้นนำเข้าสู่การสกัดข้อมูลออกมาจากต้นแบบ โดยผู้วิจัยได้ออกแบบและพัฒนาระบบขึ้นมาเพื่อสกัดและเพิ่มคุณลักษณะให้กับตัวต้นแบบรวดเร็วเพื่อให้สามารถนำไปสร้างเอกสารอธิบายต้นแบบได้ อีกทั้งระบบที่ถูกพัฒนาขึ้นมาได้เลือกใช้เทคโนโลยีที่เหมาะสมและมีความสามารถในการรองรับความหลากหลายของอุปกรณ์ เช่น โทรศัพท์มือถือที่มีขนาดหน้าจอที่หลากหลาย เป็นต้น อีกทั้งในด้านการพัฒนาผู้วิจัยก็ได้เลือกใช้แองกูลาร์ เวอร์ชัน 2 ในการพัฒนา โดยแองกูลาร์ 2 ใช้หลักการเว็บคอมโพเนนท์ ที่จะช่วยให้การพัฒนาซอฟต์แวร์ถูกแยกออกเป็นส่วนชัดเจน ทำให้ง่ายต่อการพัฒนาและดูแล จากเครื่องมือที่ได้กล่าวมาผู้วิจัยคาดหวังว่าเครื่องมือและวิธีการในการสร้างเอกสารอธิบายต้นแบบจะช่วยในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ ลดการใช้เวลาในการสร้าง ที่จะส่งผลให้ภาพรวมของโครงการพัฒนาซอฟต์แวร์ไปในทางที่ดีขึ้น

จากการออกแบบระบบในบทที่ 3 นำไปสู่การพัฒนาในบทที่ 4 และการทดสอบระบบในบทที่ 5 ตามที่ปรากฏให้เห็นในแนวทางการงานวิจัยนี้ จะแสดงให้เห็นการทำงานของระบบในแต่ละส่วนประกอบได้ให้ผลลัพธ์ตามที่คาดหวังได้ถูกต้องครบถ้วน สมบูรณ์ รวมทั้งสามารถสร้างเอกสารอธิบายต้นแบบพร้อมข้อมูลได้อย่างถูกต้อง

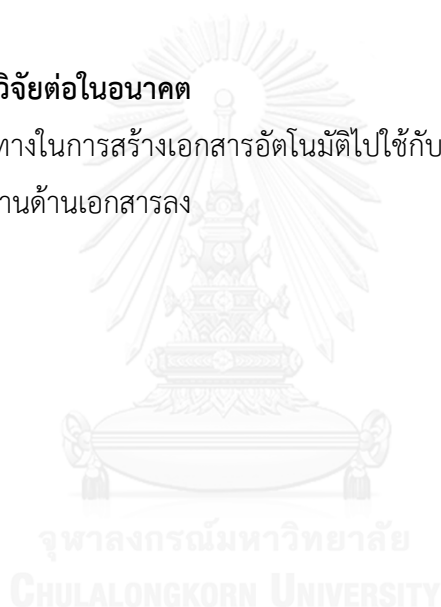
#### 6.2 ข้อจำกัดในงานวิจัย

- 1) เครื่องมือที่ใช้ในการสร้างต้นแบบรวดเร็วต้องสร้างจากไมโครซอฟต์แวร์วิซิโอและเครื่องมือสแตนด์ลิสเกตซ์ จียูไอ เซพ เท่านั้น

- 2) การนำเข้าต้นแบบจะต้องเป็นการส่งออกมาจากโปรแกรมไมโครซอฟต์วิชีโอในรูปแบบเอชทีเอ็มเท่านั้น
- 3) เพิ่มข้อมูลที่น่าเข้าจะต้องเป็นเพิ่มข้อมูลเอกซ์เอ็มแอลและเพิ่มข้อมูลพีเอ็นจีที่ได้มาจากข้อ 2 เท่านั้น
- 4) การนำเข้าเพิ่มข้อมูลพีเอ็นจี จะต้องทำการเปลี่ยนชื่อหน้าจอให้ตรงกับต้นแบบที่ทำการสร้าง
- 5) แม่แบบที่ใช้ในการกำหนดเอกสารอธิบายต้นแบบจะต้องถูกกำหนดรูปแบบตามที่กำหนดและต้องเป็นไฟล์ .docx เท่านั้น
- 6) การระบุคุณลักษณะที่สำคัญเพิ่มไม่รองรับ HTML 5

### 6.3 แนวทางสำหรับการวิจัยต่อในอนาคต

- 1) นำแนวทางในการสร้างเอกสารอัตโนมัติไปใช้กับต้นแบบอื่นๆ เพื่อประหยัดเวลาในการทำงานด้านเอกสารลง



## รายการอ้างอิง

- [1] (2016, January 28). *Microsoft Visio*. Available: <https://products.office.com/en/Visio>
- [2] I. O. Standardization, *Information Processing: Text and Office Systems: Standard Generalized Markup Language (SGML)*: ISO, 1986.
- [3] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, and F. Yergeau, "Extensible markup language (XML)," *World Wide Web Consortium Recommendation REC-xml-19980210*. <http://www.w3.org/TR/1998/REC-xml-19980210>, vol. 16, 1998.
- [4] (2016, January 22). *Node.js*. Available: <https://nodejs.org>
- [5] S. Tilkov and S. Vinoski, "Node. js: Using javascript to build high-performance network programs," *IEEE Internet Computing*, vol. 14, p. 80, 2010.
- [6] (2016, January 22). *AngularJS*. Available: <https://angular.io>
- [7] R. G. Sabale and A. Dani, "Comparative study of prototype model for software engineering with system development life cycle," *IOSR Journal of Engineering*, vol. 2, pp. 21-24, 2012.
- [8] J. Hakim and T. Spitzer, "Effective prototyping for usability," in *Proceedings of IEEE professional communication society international professional communication conference and Proceedings of the 18th annual ACM international conference on Computer documentation: technology & teamwork*, 2000, pp. 47-54.
- [9] (2016, January 16). *Prototyping*. Available: [http://www.teach-ict.com/as\\_a2\\_ict\\_new/ocr/A2\\_G063/331\\_systems\\_cycle/prototyping\\_RAD/miniview/pg2.htm](http://www.teach-ict.com/as_a2_ict_new/ocr/A2_G063/331_systems_cycle/prototyping_RAD/miniview/pg2.htm)
- [10] (2016, January 28). *W3School HTML Reference*. Available: <http://www.w3schools.com>
- [11] I. Sommerville, *Software Engineering, Eighth Edition*: Pearson Education Limited 2007.



ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

## ภาคผนวก การสร้างวินโดว์เนวิเกชันไดอะแกรม

การสร้างวินโดว์เนวิชันไดอะแกรม (Window Navigation Diagram) ด้วยภาษา ดอท (DOT) และแพ็คเกจกราฟวิซ (GraphViz) ที่เป็นส่วนของการแสดงแผนภาพสำหรับใช้จำลองเหตุการณ์แต่ละเพจว่ามีการเชื่อมโยงอย่างไรบ้าง โดยการพัฒนาเริ่มจากรายการดังต่อไปนี้

- 1) ติดตั้งโปรแกรม กราฟวิซ เวอร์ชัน 2.38
- 2) พัฒนาโค้ดสำหรับการสร้างวินโดว์เนวิเกชันไดอะแกรมดังแสดงในรูป 47

```
graph.dot C:\Users\Napajorn.w\Google Drive\Angular Projects\backend
1 digraph G {
2   graph [fontsize=10 fontname="Verdana" compound=true];
3   node [shape=record fontsize=9 fontname="Verdana"];
4   subgraph cluster0 {
5     node [style=filled];
6     label = "<<window>>"
7     "<<Login>>"
8     color=blue;
9   }
10
11   subgraph cluster1 {
12     node [style=filled];
13     label = "<<window>>"
14     "<<Home>>"
15     color=blue;
16   }
17
18   subgraph cluster2 {
19     node [style=filled];
20     label = "<<window>>"
21     "<<ProductDetails>>"
22     color=blue;
23   }
24
25   subgraph cluster3 {
26     node [style=filled];
27     label = "<<window>>"
28     "<<Cart>>"
29     color=blue;
30   }
31
32   subgraph cluster4 {
33     node [style=filled];
34     label = "<<window>>"
35     "<<Account>>"
36     color=blue;
37   }
38 }
```

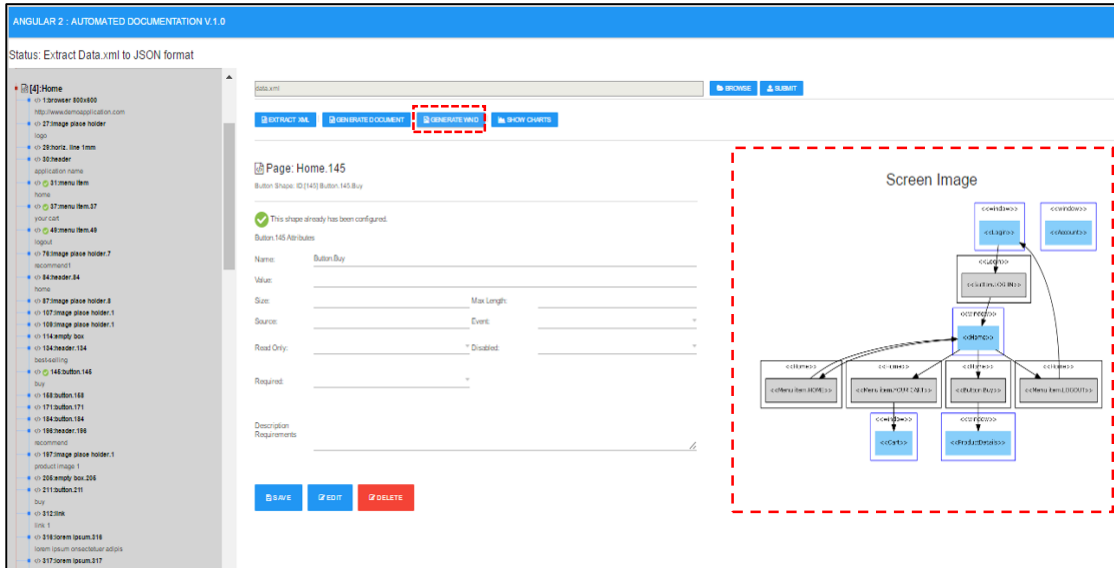
รูปที่ 47 โครงสร้างภาษาดอทที่ใช้ในการสร้างวินโดว์เนวิเกชัน

- 3) ถ้าโค้ดในส่วนการสร้างเสร็จ สามารถรันได้ด้วยคำสั่งดังรูปที่ 48

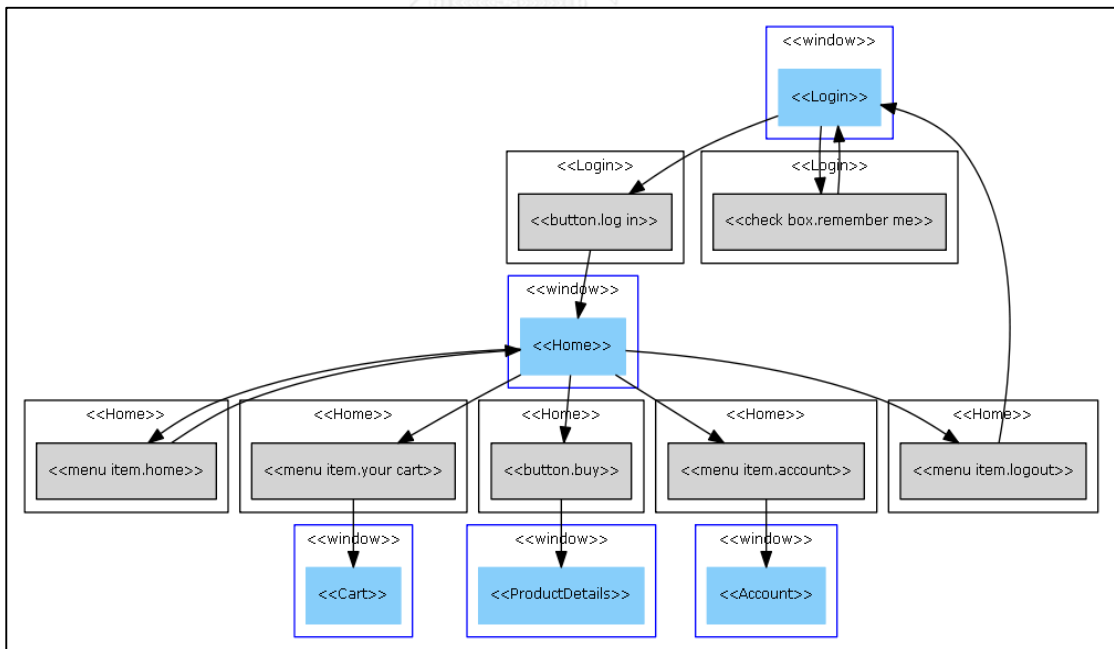
```
C:\Users\Napajorn.w\Google Drive\Angular Projects\backend>dot "graph.dot"
-Tpng -o wnd.png
```

รูปที่ 48 คำสั่งในการแปลงเพิ่มข้อมูลดอทไปเป็นวินโดว์เนวิเกชันไดอะแกรม

- 4) การจากพัฒนาในส่วนวินโดว์เนวิกชันไดอะแกรมเพิ่มเติม ระบบจะทำการจับเหตุการณ์ที่ส่งออกของแต่ละเพจเพื่อมาสร้างวินโดว์เนวิกชันไดอะแกรมใหม่อัตโนมัติ

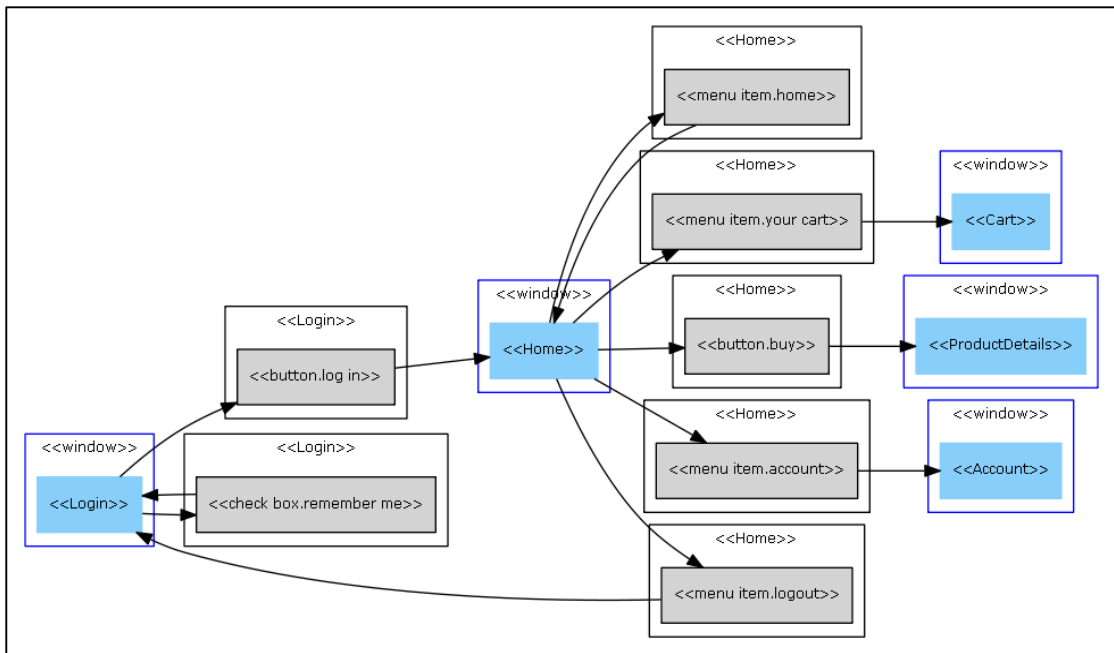


รูปที่ 49 ระบบทำการสร้างวินโดว์เนวิกชันไดอะแกรมโดยอัตโนมัติ



รูปที่ 50 วินโดว์เนวิกชันไดอะแกรมแบบแนวตั้ง





รูปที่ 51 วินโดว์เนวิเกชันไดอะแกรมแบบแนวนอน

จากรูปที่ 49 ระบบทำการจับเหตุการณ์ที่ถูกผู้ใช้งานกำหนดเหตุการณ์ต่างๆ เข้ามาแล้วนำมาสร้างวินโดว์เนวิเกชันไดอะแกรม ดังแสดงในรูปที่ 50 และ 51 วินโดว์เนวิเกชันไดอะแกรมช่วยให้ผู้ใช้งานสามารถมองภาพรวมของระบบได้ รวมถึงเหตุการณ์เชื่อมโยงในแต่ละเพจ

## ประวัติผู้เขียนวิทยานิพนธ์

นาย นภจร วัฒนกุล เกิดเมื่อวันที่ 17 มีนาคม พ.ศ. 2523 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีหลักสูตรวิทยาศาสตรบัณฑิต (วท.บ.) สาขาวิชาเทคโนโลยีสารสนเทศ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเซนต์จอร์ห์น ในปีการศึกษา 2546 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2557

