

ໂປຣແກຣມສ່ວັງເວັບໄຟຕໍ່ສ່ວນໜ້າ

WEB FRONTEND BUILDER

ວັນນຸຮານ ແວຊູເຈິ້ງ
ອຮຣັນພ ທະຮຽມເທິ່ງທະຮຽມ
ເອກຣິນທີ່ ອອງອາຈ

ປະລຸງລູນນິພນ໌ນີ້ເປັນສ່ວນໜຶ່ງຂອງການສຶກໝາດາມຫລັກສູດປະລຸງລູນວິສວກຮົມສາສຕຣບັຄທິດ
ສາຂາວິຊາວິສວກຮົມຄອມພິວເຕອົ່ງ ຄະນະວິສວກຮົມສາສຕຣ
ສະຖັນເທິນໂນໂລຢີພະຈອມເກົ່າເຈົ້າຄຸນທ່ານລາດກະບັງ
ປີການສຶກໝາ 2566

ปริญญาบัตรปีการศึกษา 2566

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง โปรแกรมสร้างเว็บส่วนหน้า

WEB FRONTEND BUILDER

ผู้จัดทำ

- | | |
|----------------------------|-----------------------|
| 1. นายวันบรรหาร แวงเชิง | รหัสนักศึกษา 64015130 |
| 2. นายอรรถพ ธรรมเที่ยงธรรม | รหัสนักศึกษา 64015166 |
| 3. นายเอกรินทร์ องอาจ | รหัสนักศึกษา 64015172 |

อาจารย์ที่ปรึกษา

(ผศ. ธนา วงศ์สุวรรณ)

อาจารย์ที่ปรึกษาร่วม

(ผศ.ดร. ชมพนธุ์ เต็งเจริญ)

โปรแกรมสร้างเว็บส่วนหน้า

นายวันบรรจาน	แวงอชีง	64015130
นายอรรถพ	ธรรมเที่ยงธรรม	64015166
นายเอกนิทร์	องอาจ	64015172
พศ.ธนา	วงศ์สุวรรณ	อาจารย์ที่ปรึกษา
พศ.ดร.ชุมพูนุท	เต็งเจริญ	อาจารย์ที่ปรึกษาร่วม
ปีการศึกษา 2566		

บทคัดย่อ

ปริญญาบัตรนี้เป็นการพัฒนาเว็บแอปพลิเคชันสำหรับสร้างเว็บไซต์ส่วนหน้า จัดทำขึ้นเพื่อให้เป็นเว็บแอปพลิเคชันที่อำนวยความสะดวกแก่ผู้ที่ต้องการสร้างหน้าเว็บไซต์ และทดสอบข้อมูล API ที่ผู้ใช้มี เนื่องจากการทดสอบ API ที่สมบูรณ์ควรจะมีเว็บไซต์ส่วนหน้าในการแสดงผลผลลัพธ์ของ API อย่างมา เพื่อตรวจสอบความถูกต้องและความสมบูรณ์ของ API นั้น และยังช่วยให้ผู้พัฒนา API สามารถทำงานได้ตรงตามความต้องการของระบบนั้น ๆ จึงได้พัฒนาแอปพลิเคชันในรูปแบบการทำงานของ Low-code ซึ่งจะช่วยให้ผู้ใช้สามารถสร้างเว็บไซต์ได้ผ่านเครื่องมือต่างๆ ที่แอปพลิเคชันเตรียมไว้ให้ และยังสามารถปรับแต่ง CSS ได้ง่ายผ่าน Template CSS ที่มาจากการประยุกต์ไปด้วย Events ต่างๆ ที่แอปพลิเคชันเตรียมไว้ และแอปพลิเคชันนี้จะช่วยลดระยะเวลาในการทำงานของการสร้างเว็บไซต์ส่วนหน้า

Web Front Builder

Mr. Wanburhan	Wae-Useng	64015130
Mr. Unnop	Thamtiengham	64015166
Mr. Aekarin	Ongart	64015172
Asst.Prof. Thana	Hongsuwan	Advisor
Asst.Prof.Dr. Chompunuch	Tengcharoen	Co-Advisor

ABSTRACT

This thesis involves the development of a web application for creating frontend websites, designed to provide convenience to users who need to create web pages. It also serves as a tool for testing the API data that users have. Since comprehensive API testing requires a frontend website to display the results of the API, it helps to verify the accuracy and completeness of the API. Furthermore, it assists API developers in ensuring that the system meets their requirements. The application is developed in a Low-code paradigm, allowing users to create websites through various tools provided by the application. Users can easily customize CSS through predefined TailwindCSS templates. Additionally, they can connect to APIs through provided tools, including various events prepared by the application. This application aims to reduce the time required for frontend website development.

กิตติกรรมประกาศ

โครงการโปรแกรมสร้างเว็บล่าวนหน้า (Web Frontend Builder) สามารถดำเนินการได้ลุล่วงไปได้ด้วยดี โดยได้รับความช่วยเหลือจากอาจารย์ที่ปรึกษาสองท่าน คือ พศ.ธนฯ วงศ์สุวรรณ และ พศ.ดร.ชมพูนุท เติงเจริญ ที่กรุณายังความรู้และคำปรึกษาต่างๆ เกี่ยวกับแนวทางในการดำเนินงาน รวมถึงช่วยพิจารณาจุดบกพร่อง พร้อมแนะนำแนวทางการแก้ไขให้พร้อมทั้งกรรมการปริญญา นิพนธ์ ทุกท่านที่ร่วมให้คำแนะนำในการทำปริญญานิพนธ์ฉบับนี้ให้สมถุที ผลตลอดจน คณาจารย์ในภาควิชาวิศวกรรมคอมพิวเตอร์สถาบันเทคโนโลยีพระกั�้ำมหาวิทยาลัยบังทึก ท่านที่เคยอบรมให้ความรู้คำแนะนำ และความช่วยเหลืออันดีแก่คณะผู้จัดทำเสมอมา

วันบุรhan	แฉเชิง
อรรถพ	ธรรมเที่ยงธรรม
เอกринทร	องอาจ

สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูปภาพ	VI
สารบัญตาราง	XI
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของปัญหา	1
1.2 วัตถุประสงค์ของโครงการ	2
1.3 ขอบเขตของโครงการ	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	3
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	4
2.1 ทฤษฎีที่เกี่ยวข้อง	4
2.2 งานวิจัยที่เกี่ยวข้อง	14
บทที่ 3 การออกแบบและพัฒนา	15
3.1 การออกแบบสถาปัตยกรรมของระบบ (System Architecture)	15
3.2 ข้อกำหนดโครงการ (System Requirements Specification)	17
3.3 การออกแบบยูสเคส (Use case diagram)	20
3.4 BPMN (Business Process Modeling Notation)	26
3.5 การออกแบบฐานข้อมูล (Er-diagram)	28
3.6 การออกแบบส่วนติดต่อผู้ใช้ (User interface Design)	30

สารบัญ(ต่อ)

	หน้า
บทที่ 4 การทดลอง และผลการทดลอง	38
4.1 ทดสอบการใช้งานเว็บแอปพลิเคชัน	38
4.2 การทดลองใช้งานการสร้าง Elements	48
4.3 การทดลองใช้งานสีโทลี CSS จัดการรูปแบบ Elements.....	50
4.4 การทดลองใช้งาน API ร่วมกับ Elements	52
4.5 การทดลองใช้งาน Preview Page.....	56
4.6 การทดลองใช้งาน Manual JavaScript.....	58
4.7 การทดลองใช้งาน Generate Code.....	60
บทที่ 5 สรุปผลและข้อเสนอแนะ	62
5.1 สรุปผลที่ได้จากการพัฒนาโปรแกรม.....	62
5.2 ปัญหา และอุปสรรค.....	62
5.3 แนวทางการพัฒนาต่อ	62
บรรณานุกรม	63

สารบัญ

หัวข้อ	หน้า
รูป 2.1 โครงสร้างไฟล์ HTML	4
รูป 2.2 โครงสร้าง HTML Elements.....	5
รูป 2.3 โครงสร้าง HTML Attribute	5
รูป 2.4 โครงสร้าง DOM.....	6
รูป 2.5 Drag and Drop.....	7
รูป 2.6 โครงสร้าง CSS Property	8
รูป 2.7 MySQL Architecture.....	10
รูป 2.8 Jenkins Architecture.....	11
รูป 2.9 Docker Architecture	12
รูป 3.1 Three-Tier Architecture ของระบบเว็บไซต์ Front-end Builder	15
รูป 3.2 System architecture diagram ของระบบเว็บไซต์ Front-end Builder.....	16
รูป 3.3 แผนภาพ Use Case Diagram ของระบบเว็บไซต์ Front-end Builder	20
รูป 3.4 การออกแบบ (BPMN-Login) ของระบบเว็บไซต์ Front-end Builder.....	26
รูป 3.5 การออกแบบ (BPMN-Register) ของระบบเว็บไซต์ Front-end Builder.....	26
รูป 3.6 การออกแบบ (BPMN-Workspace) ของระบบเว็บไซต์ Front-end Builder	27
รูป 3.7 การออกแบบ (BPMN-Select Element) ของระบบเว็บไซต์ Front-end Builder	27
รูป 3.8 การออกแบบ (BPMN-Style Element) ของระบบเว็บไซต์ Front-end Builder.....	28
รูป 3.9 การออกแบบ (ER-Diagram) ของระบบเว็บไซต์ Front-end Builder	28
รูป 3.10 หน้าหลักเว็บไซต์ (Home)	30
รูป 3.11 หน้าเข้าสู่ระบบ (Login).....	30
รูป 3.12 หน้าเข้าสมัครสมาชิก (Sign Up).....	31
รูป 3.13 หน้าลืมรหัสผ่าน (Forgot Password)	31
3.14 หน้าลืมรหัสผ่าน (Password Reset)	32
รูป 3.15 หน้าตั้งค่ารหัสผ่านใหม่ (Set new password).....	32
รูป 3.16 หน้าบอกรุ่ดประสงค์เว็บไซต์ (Platform).....	33
รูป 3.17 หน้าสอนการใช้งานเว็บไซต์ (Tutorial)	34

สารบัญ (ต่อ)

รูป	หน้า
รูป 3.18 หน้าแสดงโปรเจกต์ที่มีทั้งหมด (Workspace)	34
รูป 3.19 หน้าป้อนอพาร์ต์เม้นต์ (Create project)	35
รูป 3.20 หน้าสร้างหน้าออกแบบ (Create new page).....	35
รูป 3.21 หน้าสร้างหน้าออกแบบ (Create new page).....	36
รูป 3.22 หน้าการทำงาน (Working).....	36
รูป 3.23 หน้าโค้ดออกแบบ (Code).....	37
รูป 3.24 หน้าการสร้าง API (Create API).....	37
รูป 4.1 ผลลัพธ์ของ TC01 Use case Sign up	39
รูป 4.2 กรอกข้อมูล First Name, Last Name, Email, Password, Confirm Password.....	40
รูป 4.3 สมัครสมาชิกเสร็จสมบูรณ์	40
รูป 4.4 ผลลัพธ์ของ TC02 Use case Forgot	41
รูป 4.5 ทำการกรอกข้อมูล Email.....	42
รูป 4.6 ส่งรหัส OTP ไปที่ Email ผู้ใช้	42
รูป 4.7 รหัส OTP	42
รูป 4.8 ทำการกรอกรหัส OTP	43
รูป 4.9 ทำการกรอกข้อมูล New Password, Confirm Password	43
รูป 4.10 เปลี่ยนรหัสผ่านเสร็จสิ้น	44
รูป 4.11 ผลลัพธ์ของ TC03 Use case Sign in	45
รูป 4.12 ทำการกรอกข้อมูล Email, Password	45
รูป 4.13 Login สำเร็จเข้าสู่หน้า Project	45
รูป 4.14 ผลลัพธ์ของ TC04 Use case Project	46
รูป 4.15 กรอกชื่อ Project	47
รูป 4.16 สร้าง Project สำเร็จ	47
รูป 4.17 หน้า Workspace	47
รูป 4.18 ผลลัพธ์ของ TC05 Use case Use Element	48
รูป 4.19 หน้าการทำงาน Workspace	49

สารบัญรูป (ต่อ)

รูป	หน้า
รูป 4.20 การจัดวาง Element	49
รูป 4.21 ผลลัพธ์ของ TC06 Use case Format CSS	51
รูป 4.22 เลือก Element ที่ปรับแต่ง Style	51
รูป 4.23 ผลลัพธ์การปรับแต่ง CSS สำเร็จ	51
รูป 4.24 การเชื่อมต่อ API แบบ GET	53
รูป 4.25 ผลลัพธ์การดึงข้อมูล API แบบ GET	53
รูป 4.26 การเชื่อมต่อ API แบบ POST	54
รูป 4.27 สร้าง Function สำหรับการยิง API	54
รูป 4.28 ตัวอย่างการใส่ข้อมูล	55
รูป 4.29 ผลลัพธ์ของข้อมูลที่ผ่านการส่งข้อมูล API	55
รูป 4.30 ผลลัพธ์ของ TC08 Use case Preview	56
รูป 4.31 กดปุ่มเพื่อแสดงผลลัพธ์เว็บไซต์	57
รูป 4.32 ผลลัพธ์การ Preview Page	57
รูป 4.33 หน้า Code HTML	59
รูป 4.34 แก้ไข Code HTML	59
รูป 4.35 ผลลัพธ์หลังจากเพิ่ม Code HTML	60
รูป 4.36 หน้า Project	61
รูป 4.37 ทำการ Download Code Project	61
รูป 4.38 Code ที่ Download	61

สารบัญตาราง

ตาราง	หน้า
ตาราง 3.1 รายการความสามารถของระบบ	17
ตาราง 3.2 รายละเอียด Use Case Sign up	21
ตาราง 3.3 รายละเอียด Use Case Forgot Password	21
ตาราง 3.4 รายละเอียด Use Case Sign in (Username, Password).....	22
ตาราง 3.5 รายละเอียด Use Case Create Project.....	22
ตาราง 3.6 รายละเอียด Use Case Create and Use Element.....	23
ตาราง 3.7 รายละเอียด Use Case Format CSS.....	23
ตาราง 3.8 รายละเอียด Use Case API	24
ตาราง 3.9 รายละเอียด Use Case Preview	24
ตาราง 3.10 รายละเอียด Use Case Manual JavaScript	25
ตาราง 3.11 รายละเอียด Use Case Generate Code.....	25
ตาราง 3.12 รายละเอียดฐานข้อมูลตาราง User	29
ตาราง 3.13 รายละเอียดฐานข้อมูลตาราง Workspace	29
ตาราง 3.14 รายละเอียดฐานข้อมูลตาราง Project.....	29
ตาราง 4.1 ผลการทดสอบ Use Case ของการ Sign up.....	39
ตาราง 4.2 ผลการทดสอบ Use Case ของการ Forgot Password.....	41
ตาราง 4.3 ผลการทดสอบ Use Case ของการ Sign in	44
ตาราง 4.4 ผลการทดสอบ Use Case ของการ Project.....	46
ตาราง 4.5 ผลการทดสอบ Use Case ของการ Element.....	48
ตาราง 4.6 ผลการทดสอบ Use Case ของการ Format CSS	50
ตาราง 4.7 ผลการทดสอบ Use Case ของการ API.....	52
ตาราง 4.8 ผลการทดสอบ Use Case ของการ Preview	56
ตาราง 4.9 ผลการทดสอบ Use Case ของการ Manual JavaScript	58
ตาราง 4.10 ผลการทดสอบ Use Case ของการ Generate Code	60

บทที่ 1

บทนำ

1.1 ความเป็นมาของปัญหา

การพัฒนาเว็บไซต์และแอปพลิเคชันในปัจจุบันได้รับการปฏิวัติอย่างมากด้วยแนวคิดใหม่ที่ชื่อว่า Front-end Builder หรือที่เรียก กันอีกชื่อหนึ่งว่า Low-Code Platform ซึ่งเป็นแนวคิดที่เน้นการทำงานได้ละเอียดกว่าโดยไม่ต้องมีความรู้ทางเทคนิคเฉพาะหรือความชำนาญในการเขียนโค้ด HTML และ CSS อย่างเช่นข้าม เป็นแนวคิดที่ทำให้งานสร้างเว็บไซต์เป็นเรื่องง่ายและไม่ซับซ้อน แนวคิดของ Low-Code Platform ช่วยให้นักพัฒนาและผู้สร้างแอปพลิเคชันสามารถสร้างแอปพลิเคชันได้อย่างรวดเร็วโดยไม่จำเป็นต้องมีระดับทักษะการเขียนโค้ดสูง ซึ่งการสร้างเว็บไซต์หรือแอปพลิเคชันด้วยแนวคิด Low-Code Platform เป็นวิธีใหม่ที่ทำให้ผู้ที่ไม่มีความเชี่ยวชาญในการเขียนโค้ดสามารถออกแบบและพัฒนาเว็บไซต์หรือแอปพลิเคชันได้อย่างง่ายๆ

ในทางกลับกัน การทดสอบ Application Programming Interface (API) เป็นกระบวนการสำคัญในการพัฒนาซอฟต์แวร์ในปัจจุบัน ไม่ว่าจะเป็นภายในองค์กรหรือสู่สาธารณะ การทดสอบ API ช่วยให้กันพบข้อผิดพลาดและปัญหาต่างๆ ก่อนที่ระบบจะเข้าใช้งานจริง ซึ่งเป็นวิธีที่สำคัญในการรักษาคุณภาพของซอฟต์แวร์และความมั่นใจในการทำงานของระบบ API เป็นอินเตอร์เฟซที่ช่วยให้แอปพลิเคชันและระบบต่างๆ สื่อสารกัน และการทดสอบ API ช่วยตรวจสอบว่า API ทำงานอย่างถูกต้อง และตรงตามความต้องการของระบบที่มา อย่างไรก็ตาม การทดสอบ API ยังคงเป็นกระบวนการที่ไม่มีการแสดงผลตามหน้าจอ ซึ่งทำให้การตรวจสอบความถูกต้องและประสิทธิภาพของ API เป็นเรื่องที่มีความซับซ้อน

ด้วยเหตุนี้เอง การพัฒนาเว็บแอปพลิเคชัน Front-end Builder จึงเกิดขึ้น เพื่อช่วยให้ผู้พัฒนาสามารถออกแบบหน้าเว็บไซต์ในรูปแบบที่ต้องการเพื่อทดสอบ API ได้ว่า API ที่สร้างขึ้นมานั้น เป็นไปตามวัตถุประสงค์ของผู้พัฒนาหรือไม่ ด้วย Front-end Builder ผู้พัฒนาสามารถทำการออกแบบหน้าเว็บไซต์ที่ต้องการให้สอดคล้องกับ API และทำการทดสอบได้ว่า API ทำงานอย่างถูกต้อง หรือไม่สอดคล้องกับความต้องการของระบบ ทั้งยังเป็นการทำงานที่ง่ายและมีประสิทธิภาพสูงสุดสำหรับผู้ใช้งาน ด้วยความสามารถในการทำงานที่มีประสิทธิภาพและความสะดวกในการใช้งาน เว็บแอปพลิเคชัน Front-end Builder แสดงถึงการผสมผสานระหว่างโลกของ Low-Code Platform และการทดสอบ API ให้กลายเป็นเครื่องมือที่สำคัญในการพัฒนาและทดสอบซอฟต์แวร์ในยุคปัจจุบันและอนาคต การเน้นความง่าย ความสะดวก และความเร็วในการพัฒนาจึงกลายเป็นจุดเด่นสำคัญที่ Front-end Builder และ Low-Code Platform

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อทดสอบการใช้งาน API ของตนเองในแพลตฟอร์ม Low-Code ผู้ใช้สามารถเข้ามือต่อและใช้งาน API จากแหล่งที่มาต่าง ๆ เพื่อเพิ่มฟังก์ชันและความสามารถของเว็บไซต์
- 1.2.2 เพื่อลดความซับซ้อนและประหยัดเวลาในการบูนการสร้างและปรับปรุงเว็บไซต์ ในด้าน Front-end ใช้แพลตฟอร์ม Low-code จะช่วยลดเวลาในสิ่งนี้ได้
- 1.2.3 เพื่อนำความรู้และทักษะที่ได้จากการศึกษามาใช้ในการปฏิบัติงานจริง
- 1.2.4 เพื่อฝึกฝนการสร้าง Front end หรือสร้าง Font end ได้รวดเร็ว

1.3 ขอบเขตของโครงการ

- 1.3.1 สามารถพัฒนาและสร้างแพลตฟอร์มที่ช่วยให้ผู้ใช้สามารถสร้างเว็บไซต์เว็บไซต์โดยไม่ต้องมีความรู้ทางเทคนิคด้านการเขียนโปรแกรมเป็นพิเศษ
- 1.3.2 สามารถใช้งานฟีเจอร์พื้นฐานที่จำเป็นสำหรับการสร้างเว็บไซต์ เช่น การสร้างแบบฟอร์มในการป้อนข้อมูล การจัดการฐานข้อมูลพื้นฐาน การจัดการผู้ใช้งาน และการจัดการสิทธิ์การเข้าถึงข้อมูล
- 1.3.3 สามารถเชื่อมต่อกับ API 外部 API ภายนอกเพื่อให้ผู้ใช้สามารถนำเอาข้อมูลหรือบริการจากแหล่งข้อมูลภายนอกมาใช้ในเว็บไซต์ที่พัฒนาได้
- 1.3.4 สามารถปรับแต่งและการปรับเปลี่ยนรูปแบบของเว็บไซต์ได้อย่างยืดหยุ่น ผู้ใช้สามารถปรับเปลี่ยนเค้าโครงหรือรูปแบบทั้งหมดของเว็บไซต์ได้ตามต้องการ ได้รวมถึงการปรับแต่งสี รูปแบบเดียวกัน และส่วนประกอบอื่น ๆ
- 1.3.5 สามารถปรับแต่งและการปรับเปลี่ยนรูปแบบของเว็บไซต์ได้อย่างยืดหยุ่น ผู้ใช้สามารถปรับเปลี่ยนเค้าโครงหรือรูปแบบทั้งหมดของเว็บไซต์ได้ตามต้องการ ได้รวมถึงการปรับแต่งสี รูปแบบเดียวกัน และส่วนประกอบอื่น ๆ
- 1.3.6 สามารถตรวจสอบความถูกต้องและการทดสอบเบื้องต้น ผู้ใช้สามารถตรวจสอบการทำงานของเว็บไซต์และทดสอบความถูกต้องของฟังก์ชันพื้นฐานได้ก่อนการเผยแพร่
- 1.3.7 สามารถให้ผู้ใช้สร้างหน้าเว็บไซต์หลายหน้าได้ โดยให้ผู้ใช้สร้างหน้าเว็บไซต์ใหม่และกำหนดลำดับการเรียงลำดับหน้าเว็บไซต์
- 1.3.8 สามารถให้ผู้ใช้สร้างเมนูและการนำทางในเว็บไซต์ได้ โดยให้ผู้ใช้สร้างเมนูหลักและเมนูย่อย และกำหนดลำดับการเรียงลำดับของเมนู
- 1.3.9 ไม่สามารถใช้งานกับ API ที่ซับซ้อนได้
- 1.3.10 ไม่สามารถ Custom CSS ได้
- 1.3.11 ไม่สามารถ แชร์หน้า Project ไว้ใช้สำหรับหลายคน User ได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1.4.1 สามารถใช้งานฟีเจอร์พื้นฐานที่จำเป็นสำหรับการสร้างเว็บไซต์ เช่น การสร้างแบบฟอร์มในการป้อนข้อมูล การจัดการฐานข้อมูลพื้นฐาน การจัดการผู้ใช้งาน และการจัดการสิทธิ์การเข้าถึงข้อมูล
- 1.4.2 สามารถพัฒนาและสร้างแพลตฟอร์มที่ช่วยให้ผู้ใช้สามารถสร้างเว็บไซต์ได้โดยไม่ต้องมีความรู้ทางเทคนิคด้านการเขียนโปรแกรมระดับสูง
- 1.4.3 สามารถเชื่อมต่อกับ API ภายนอกเพื่อให้ผู้ใช้สามารถนำเอาข้อมูลหรือบริการจากแหล่งข้อมูลภายนอกมาใช้ในเว็บไซต์ที่พัฒนาได้
- 1.4.4 สามารถปรับแต่งและการปรับเปลี่ยนรูปแบบของเว็บไซต์ได้อย่างยืดหยุ่น ผู้ใช้สามารถปรับเปลี่ยนเค้าโครงหรือรูปแบบทั้งหมดของเว็บไซต์ได้ตามต้องการ ได้รวมถึงการปรับแต่งสี รูปแบบเดียวกัน และส่วนประกอบอื่น ๆ
- 1.4.5 สามารถตรวจสอบความถูกต้องและการทดสอบเบื้องต้น ผู้ใช้สามารถตรวจสอบการทำงานของเว็บไซต์และทดสอบความถูกต้องของฟังก์ชันพื้นฐานได้ก่อนการเผยแพร่
- 1.4.6 สามารถให้ผู้ใช้สร้างหน้าเว็บไซต์หลายหน้าได้ โดยให้ผู้ใช้สร้างหน้าเว็บไซต์ใหม่และกำหนดลำดับการเรียงลำดับหน้าเว็บไซต์
- 1.4.7 สามารถให้ผู้ใช้สร้างเมนูและการนำทางในเว็บไซต์ได้ โดยให้ผู้ใช้สร้างเมนูหลักและเมนูย่อย และกำหนดลำดับการเรียงลำดับของเมนู

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

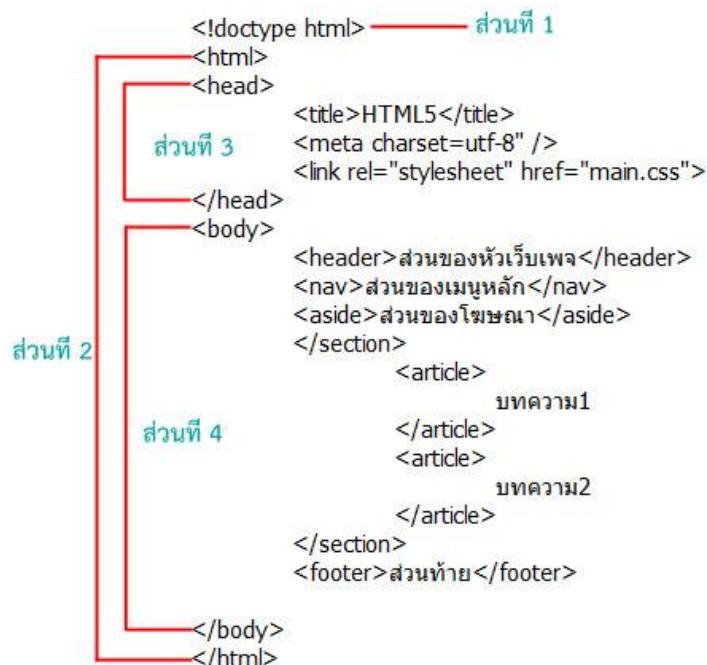
2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 Hyper Text Markup Language

Hyper Text Markup Language หรือ HTML เป็นภาษาคอมพิวเตอร์ที่ใช้สร้างหน้าเว็บ (Web Page) ในรูปแบบของไฟล์ HTML (คือไฟล์ที่มีนามสกุลเป็น .htm หรือ .html) ซึ่งมีเว็บเบราว์เซอร์ (Web Browser) เป็นโปรแกรมที่ใช้แปลงไฟล์ HTML เพื่อแสดงผลในรูปของหน้าเว็บ

ไฟล์ HTML เป็นไฟล์รหัสอักขระ (ASCII) ถูกบันทึกในรูปของไฟล์เอกสาร (Text File) ที่สามารถสร้างจากโปรแกรมสร้างไฟล์ข้อความ (Text Editor) เช่น Notepad หรือ Word Processing ทั่วๆ ไป ซึ่งลักษณะของไฟล์ HTML ประกอบไปด้วยแท็ก (Tag) ต่างๆ ที่เป็นคำสั่งของ HTML ซึ่งแท็กจะอยู่ภายใต้เครื่องหมาย <tag>

แท็กใน HTML แบ่งเป็น 2 ประเภท คือ คอนเทนเนอร์แท็ก (Container Tag) และแท็กเปล่า (Empty Tag) โดยที่คอนเทนเนอร์แท็ก ประกอบไปด้วยแท็กเปิด และแท็กปิด โดยที่แท็กปิดจะมีเครื่องหมาย / นำหน้าแท็ก เช่น <H1> . . . </H1> ส่วนแท็กเปล่า จะมีแท็กเปิดอย่างเดียว เช่น <HR> ซึ่งแท็กจะถูกเรียกด้วยตัวอักษร พิมพ์ใหญ่ หรือพิมพ์เล็กก็ได้ จะไม่มีผลต่อการแสดงผลของเว็บเบราว์เซอร์ เช่น
,
,
 หรือ
 เว็บเบราว์เซอร์จะแปลงความหมายเหมือนกัน



รูป 2.1 โครงสร้างไฟล์ HTML

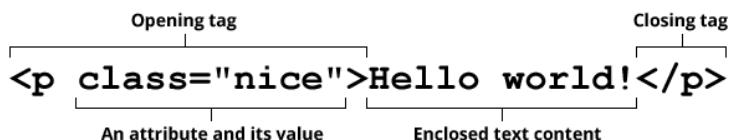
2.1.2 HTML Elements

HTML Elements คือส่วนหนึ่งของ HTML Document ที่ใช้ในการกำหนดโครงสร้างของหน้าเว็บไซต์ได้อย่างถูกต้อง และมีความสอดคล้องกับมาตรฐาน HTML และเนื้อหาของหน้าเว็บไซต์ บาง HTML Elements จะแสดงส่วนที่สามารถมองเห็นได้บนหน้าเว็บ เช่น ข้อความรูปภาพ หรือปุ่ม ในขณะที่ HTML Elements บางส่วน จะแสดงส่วนต่าง ๆ ของหน้า หรือให้ข้อมูลเกี่ยวกับเว็บไซต์ (Meta-Information) เช่น คำอธิบายหน้าเว็บ (Meta Description) หรือคีย์เวิร์ดสำคัญ (Main Keyword) ที่เกี่ยวข้องกับเนื้อหาของหน้าเว็บนั้น ๆ

HTML Elements ถูกสร้างขึ้นโดยการใช้ HTML Tags ซึ่งประกอบด้วยข้อความที่อยู่ระหว่างเครื่องหมาย วงเล็บ (<>) ประกอบด้วย 3 ส่วน ดังนี้

1. แท็กเปิด (Opening Tags) เป็นแท็กที่บ่งบอกว่าเนื้อหาของ Element จะเริ่มต้นที่ใด
2. แท็กปิด (Closing Tags) เป็นแท็กที่บ่งบอกถึงจุดสิ้นสุดของเนื้อหาของ Element โดยแท็กปิดจะเหมือนกับแท็กเปิด แต่จะมีการเพิ่มเครื่องหมาย (/) เข้ามาด้วย
3. ส่วนเนื้อหา จะถูกวางไว้ระหว่างแท็กเปิด และแท็กปิดของ Element

Anatomy of an HTML element



รูป 2.2 โครงสร้าง HTML Elements

2.1.3 HTML Attribute

HTML Attribute เป็นข้อมูลที่เพิ่มเข้าไปใน HTML Tag เพื่อกำหนดลักษณะขององค์ประกอบในเว็บไซต์ หรือจะมองเป็นข้อมูลที่ใช้ในการควบคุมองค์ประกอบนั้น ๆ โดยสามารถเพิ่ม Attribute ไปยังแท็กต่าง ๆ เพื่อปรับแต่ง หรือกำหนดลักษณะที่ต้องการ ได้ เช่น การกำหนดสีพื้นหลัง (Background Color), ขนาดตัวอักษร (Font Size), การลิงก์ไปยังหน้าอื่น (Hyperlink), และอื่น ๆ อีกมากmany

Html Attributes

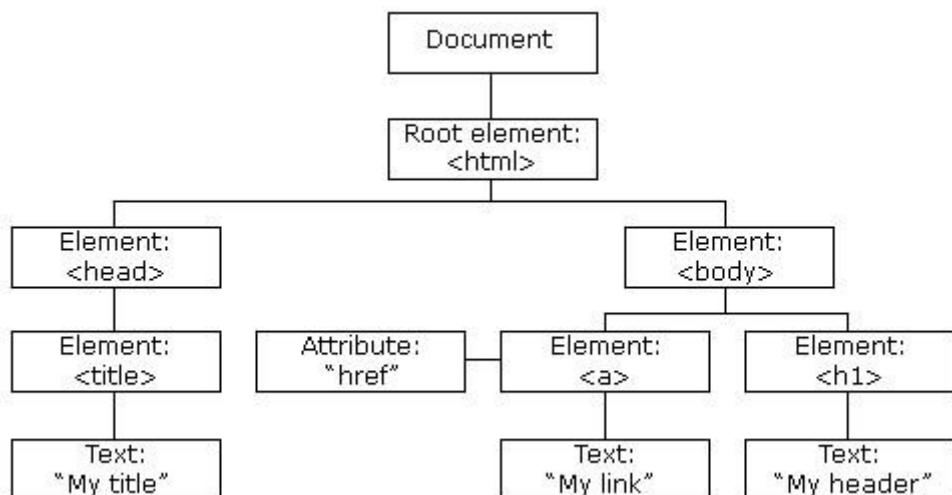


รูป 2.3 โครงสร้าง HTML Attribute

2.1.4 DOM (Document Object Model)

DOM ย่อมาจาก "Document Object Model" ซึ่งเป็นโมเดลที่ใช้ในการแสดงโครงสร้างแบบต้นไม้ของเอกสาร HTML, XML, หรือ XHTML บนเว็บเบราว์เซอร์ โดย DOM จะแสดงโครงสร้างของเอกสารในรูปแบบของโหนด (nodes) และเชื่อมโยนกันเหล่านี้เข้าด้วยกันเป็นโครงสร้างต้นไม้เพื่อจัดสิ่งและจัดการกับเนื้อหาภายในเอกสาร ได้โดยง่าย

DOM ประกอบด้วยโครงสร้างของโหนด (nodes) ที่แบ่งออกเป็นหลายประเภท เช่น Element nodes สำหรับแท็ก HTML, Text nodes สำหรับข้อความภายในเอกสาร, และ Attribute nodes สำหรับแอ็ตทริบิวต์ของ Element นอกจากนี้ DOM ยังให้เมทอดและคุณสมบัติต่างๆ ที่อนุญาตให้นักพัฒนาเว็บเข้าถึงและจัดการกับเนื้อหาของเอกสาร HTML ได้ โดยสามารถเปลี่ยนแปลงโครงสร้างหรือเนื้อหาของเอกสาร ได้ด้วยการใช้ DOM API ที่มีให้ใช้งานในภาษาโปรแกรมต่างๆ เช่น JavaScript, Python, หรือ Java ซึ่งทำให้นักพัฒนาสามารถสร้างและแก้ไขเนื้อหาของเว็บไซต์ได้อย่างมีประสิทธิภาพและยืดหยุ่น ได้ตามต้องการ



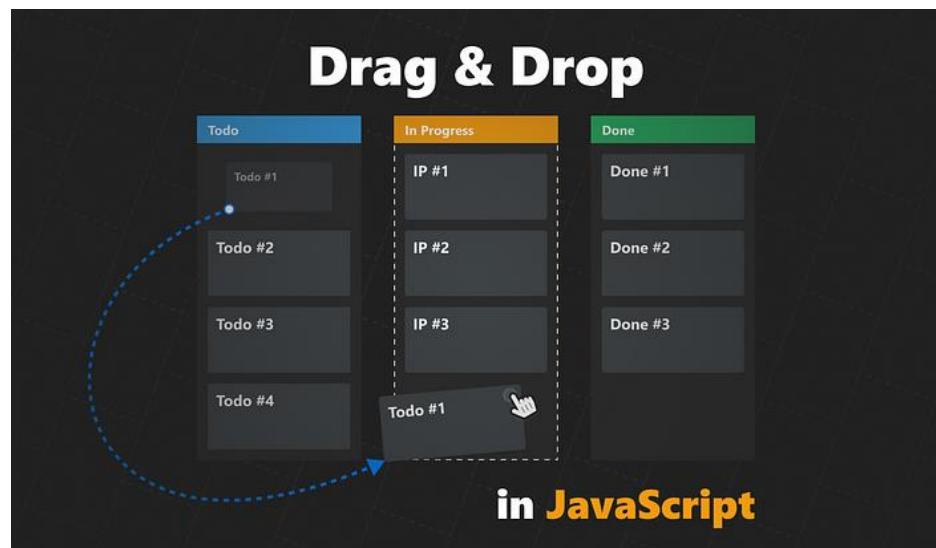
รูป 2.4 โครงสร้าง DOM

2.1.5 DOM Drag and Drop (DnD)

DOM Drag and Drop เป็นการใช้งาน DOM API เพื่อรองรับการลากและวาง (drag and drop) ขององค์ประกอบบนหน้าเว็บ เราสามารถใช้งาน DOM API และเหตุการณ์ต่าง ๆ เช่น ondragstart, ondragover, ondrop เพื่อจัดการกับการลากและวางขององค์ประกอบบนหน้าเว็บได้

การใช้งาน DOM Drag and Drop ประกอบด้วยขั้นตอนหลักๆ คือการเริ่มต้นลาก, การลาก, การวาง, และการสิ้นสุดการลาก เมื่อผู้ใช้ทำการลากและวางองค์ประกอบระบบจะส่งงานเหตุการณ์ต่าง ๆ เพื่อให้เราสามารถจัดการกับกระบวนการต่าง ๆ นี้ได้อย่างเหมาะสม

การใช้งาน DOM Drag and Drop เป็นที่นิยมในการสร้างประสบการณ์ผู้ใช้ที่มีประสิทธิภาพและสะดวกสบาย เช่น การจัดเรียงหรือเรียงลำดับข้อมูล, การลากและวางไอเท็มในรูปแบบออนไลน์, หรือการใช้เครื่องมือลากและวางในเว็บแอปพลิเคชันการทำงานที่มีการจัดการข้อมูลที่ซับซ้อน



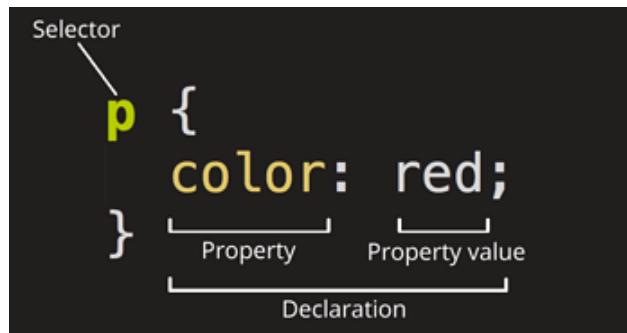
รูป 2.5 Drag and Drop

2.1.6 Cascading Style Sheets

CSS ย่อมาจากคำว่า Cascading Style Sheets ซึ่งเป็นภาษาคอมพิวเตอร์ (ภาษา Style Sheets) ที่ใช้สำหรับตกแต่งเอกสาร HTML หรือกำหนดรูปแบบ และลักษณะขององค์ประกอบบนหน้าเว็บไซต์ โดย CSS จะมีลักษณะเป็นภาษาที่มีรูปแบบในการเขียน Syntax แบบเฉพาะ และได้ถูกกำหนดมาตรฐานโดย W3C ที่ถือเป็นภาษาหนึ่ง สำหรับใช้ในการตกแต่งเว็บไซต์ ที่ได้รับความนิยมกันอย่างแพร่หลาย

การกำหนดลักษณะของข้อความ เช่น ขนาด สี ตัวอักษร หรือกำหนดรูปแบบขององค์ประกอบ เช่น กล่องข้อความ รูปภาพ และปุ่ม รวมถึงการจัดวางองค์ประกอบต่าง ๆ ภายใน

เว็บไซต์ ขณะที่เว็บเบราว์เซอร์ทำการแสดงหน้าเว็บ ก็จะแปลงโค้ดเหล่านี้ เพื่อแสดงเนื้อหาตามสีสไตล์ที่ได้กำหนดไว้



รูป 2.6 โครงสร้าง CSS Property

2.1.7 Tailwind CSS

Tailwind CSS คือ เครื่องมือ CSS framework ที่มุ่งเน้นการพัฒนาและออกแบบเว็บแอปพลิเคชัน ได้อย่างรวดเร็วและมีความสามารถในการปรับแต่งสไตล์ให้ง่ายขึ้น โดยใช้เทคนิคการเขียนคลาส CSS ที่เรียกว่า "utility classes" หรือ "utility-first" approach ซึ่งช่วยลดเวลาในการเขียน CSS และทำให้การบริหารจัดการสไตล์ของเว็บง่ายขึ้นเป็นอย่างมาก เนื่องจากมาพร้อมกับ class สำเร็จรูปสุดยอดประสิทธิภาพที่ใช้งานได้ทันทีในกรณีที่ต้องการเปลี่ยน UI หลักของเฟรมเวิร์ก นั้นทำให้นักพัฒนาแทนไม่ต้องเข้าไปแก้ไขที่ไฟล์ CSS หลักเลย

2.1.8 JavaScript

JavaScript คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต ที่กำลังได้รับความนิยมอย่างสูง Java JavaScript เป็นภาษาสคริปต์เชิงวัตถุ (ที่เรียกว่า "สคริปต์" (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเรามีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปทีละคำสั่ง" (interpret) หรือเรียกว่า อ็อบเจกต์โอเรียลเต็ต (Object Oriented Programming) ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางผ่านไคลเอนต์ และทางผ่านเซิร์ฟเวอร์

การทำงานของ JavaScript เกิดขึ้นบนบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้นไม่ว่าจะใช้เซิร์ฟเวอร์อะไรที่ไหน ก็ยังคงสามารถใช้ JavaScript ในเว็บเพจได้ ต่างกับภาษาสคริปต์อื่น เช่น Perl, PHP หรือ ASP ซึ่งต้องแปลความและทำงานที่ตัวเครื่องเซิร์ฟเวอร์ (เรียกว่า server-side

script) ดังนั้นจึงต้องใช้บันเชิร์ฟเวอร์ ที่สนับสนุนภาษาเหล่านี้เท่านั้น อย่างไรก็ดี จากลักษณะดังกล่าวก็ทำให้ JavaScript มีข้อจำกัด คือไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ใช้ เพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานลักษณะนี้ จึงยังคงต้องอาศัยภาษา server-side script อีก

2.1.9 React

React (บางครั้งเรียกเป็น React.js หรือ ReactJS) คือไลบรารี jawsdkripที่ถูกมองว่าเป็นตัวช่วยให้สามารถสร้าง UI (User Interface หรือองค์ประกอบของเว็บที่เชื่อมต่อกับผู้ใช้งานโดยตรง) ได้แม่นยำและรวดเร็วมากยิ่งขึ้น และส่งผลให้การแสดงผลมีความเป็นระบบคงเส้นคงวามากขึ้นไปพร้อมๆ กัน สิ่งนี้ถูกนำมาใช้กันอย่างแพร่หลายในการเขียนโค้ดสำหรับเว็บไซต์และแอปพลิเคชันมีประโยชน์กับคนที่ทำงานในฐานะ Front-end จึงนับว่าเป็นหนึ่งในสิ่งที่คนทำงานในสายนักพัฒนาจำเป็นต้องเรียนรู้ 3 องค์ประกอบของ React คือ

Component: โค้ดที่แบ่งออกเป็นบล็อกเล็กๆ หลายอัน คอมโพเนนต์อันหนึ่งจะแทนที่ชิ้นส่วนของปุ่มหรือองค์ประกอบ UI ของหน้าเว็บ ซึ่งคอมโพเนนต์เหล่านี้สามารถนำมาใช้ซ้ำได้ตลอดการสร้างระบบ คอมโพเนนต์จะทำให้ส่วนต่างๆ ของหน้าเว็บทำงานอย่างเป็นอิสระต่อกัน

JSX: JSX เป็นส่วนขยายสำหรับการเขียนโค้ดภาษา JavaScript ทำให้นักพัฒนาสามารถเขียนโค้ดที่เหมือน HTML ภายในคอมโพเนนต์ แต่เมื่องหลังจะถูกแปลงเป็นภาษา JavaScript ได้เอง

Props: ใช้อธิบายวิธีการส่งผ่านข้อมูลจากคอมโพเนนต์หลักไปยังคอมโพเนนต์ย่อย เพื่อการแสดงผลบางประการที่แตกต่างกันไปในคอมโพเนนต์หลักชิ้นเดียวกันโดยไม่ต้องไปเขียนโค้ดชิ้นใหม่ทั้งหมด ทำให้สามารถใช้งานคอมโพเนนต์ได้อย่างหลากหลายมากขึ้น เช่น ปุ่มกดที่มีดีไซน์ UI เหมือนกันเกิดจากคอมโพเนนต์อันเดียวกัน แต่ Text บนปุ่มในแต่ละหน้าเว็บอาจไม่เหมือนกัน

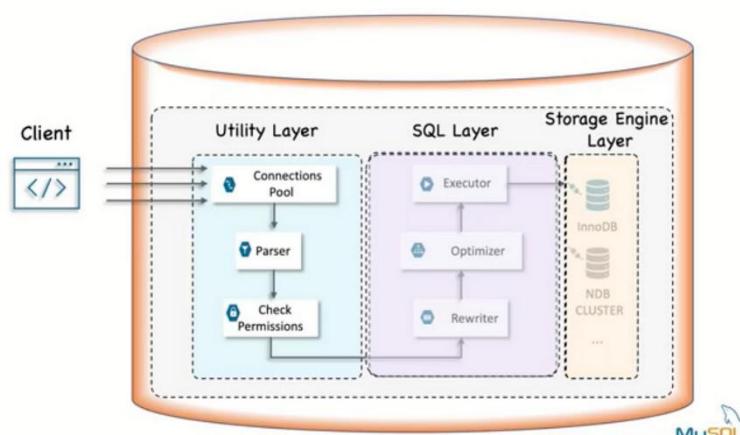
2.1.10 API

API คือการเชื่อมต่อระบบเพื่อให้ซอฟต์แวร์สามารถเข้าถึงและอัพเดทข้อมูลได้อย่างสะดวก โดย REST เป็นหนึ่งในสถาปัตยกรรมของ API ที่ใช้ HTTP Protocol ระหว่าง Client และ Server โดย Client ทำการส่ง HTTP requests เพื่อร้องขอข้อมูลหรือทำการกระทำการต่างๆ บน Server และ Server จะตอบกลับด้วยข้อมูลในรูปแบบต่างๆ เช่น JSON, XML, HTML ตามที่ร้องขอมา การใช้ REST API ช่วยให้บริษัทหรืออุตสาหกรรมต่างๆ อำนวยความสะดวก ลดกำลังคน และลดความผิดพลาดได้

2.1.11 MySQL

MySQL คือ ระบบจัดการฐานข้อมูล หรือ Database Management System (DBMS) แบบข้อมูลเชิงสัมพันธ์ หรือ Relational Database Management System (RDBMS) ซึ่งเป็นระบบฐานข้อมูลที่จัดเก็บรวบรวมข้อมูลในรูปแบบตาราง โดยมีการแบ่งข้อมูลออกเป็นแถว (Row) และในแต่ละแถวแบ่งออกเป็นคอลัมน์ (Column) เพื่อเชื่อมโยงระหว่างข้อมูลในตารางกับข้อมูลในคอลัมน์ที่กำหนด แทนการเก็บข้อมูลที่แยกออกจากกัน โดยไม่มีความเชื่อมโยงกัน ซึ่งประกอบด้วยข้อมูล (Attribute) ที่มีความสัมพันธ์เชื่อมโยงกัน (Relation) โดยใช้ RDBMS

MySQL มีหน้าที่จัดเก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่งภาษา Structured Query Language หรือ SQL เพื่อจัดการกับฐานข้อมูลโดยเฉพาะ เป็นภาษามาตรฐานของระบบฐานข้อมูล เชิงสัมพันธ์และเป็นระบบเปิด (Open System) ที่มีโครงสร้างของภาษาที่เข้าใจง่าย ไม่ซับซ้อน และนิยมใช้งานร่วมกับภาษาโปรแกรม PHP รวมถึงภาษาอื่น ๆ ที่สามารถทำงานร่วมกับฐานข้อมูล MySQL ได้หลากหลาย เช่น C, C++, Python, Java เป็นต้น อีกทั้ง MySQL ยังได้รับการออกแบบและปรับให้มีความเหมาะสมสำหรับการพัฒนา Website และ Web Application ทำให้สามารถรองรับการทำงานได้ทุกแพลตฟอร์ม รวมถึงการอนุญาตให้ผู้ใช้งานหลายคนสามารถใช้งานพร้อมกันได้ (Multi-user) นอกจากนี้ยังสามารถจัดการและสร้างฐานข้อมูลจำนวนมากรวมถึงประมวลผลหลาย ๆ งาน ได้พร้อมกัน (Multi-threaded) อย่างสมบูรณ์ จึงทำให้ MySQL เป็นตัวเลือกยอดนิยมสำหรับธุรกิจการพาณิชย์อิเล็กทรอนิกส์ หรือ Electronic Commerce (E-Commerce) และเหมาะสมสำหรับการนำไปใช้งานสร้างเว็บไซต์ทั่วไป เพราะมีความแม่นยำ ครอบคลุม ช่วยให้เข้าถึงข้อมูลได้อย่างรวดเร็ว

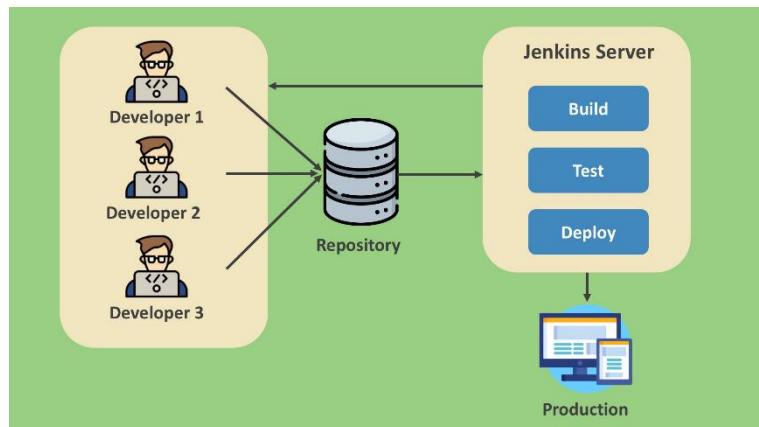


รูป 2.7 MySQL Architecture

2.1.12 Jenkins

Jenkins คือ Tools ประเภท Server-Based Application สำหรับการทำ Deployment แบบอัตโนมัติโดยทำงานตามหลักการของ CI/CD(Continuous Integration/Continuous Delivery) เพื่อแก้ปัญหากระบวนการรวม Code เข้าด้วยกันของ Developer สามารถลดระยะเวลาทำงานและถึงมอบซอฟต์แวร์ได้อย่างมีประสิทธิภาพ Jenkins นั้นถูกสร้างขึ้นจากภาษา Java เป็นซอฟต์แวร์ Open Source หรือสามารถใช้งานได้ฟรีและสามารถติดตั้งผ่าน Docker ได้ การทำงานแบบอัตโนมัติของ Jenkins นั้นสามารถทำงานได้ตั้งแต่ Build, Test และ Deploy ซอฟต์แวร์ นั้นมายความว่าทุกๆครั้งที่มีการ Commit Code เกิดขึ้น Jenkins จะทำการ Build และ Test ให้โดยอัตโนมัติจากรวมถึงแก้ผลไปยัง Developer ทำให้ได้รับการแจ้งเตือนทันทีว่า Code ที่ Commit ไปนั้นมีปัญหาหรือไม่

จากการสภารถปัจจุบันของ Jenkins จะพบว่า Developer แต่ละคนต้องการ Commit Code ไปยัง Repository เช่น GitHub จากนั้น Jenkins Server จะเริ่มทำการ Build Code หาก Build Fail ก็จะแจ้งเตือนไปยัง Developer นอกจากการ Build แล้ว Jenkins ยังสามารถเขียน Scripts ขึ้นมาทดสอบโดยเชื่อมผ่าน Jenkins ได้อีกด้วย



รูป 2.8 Jenkins Architecture

2.1.13 Golang

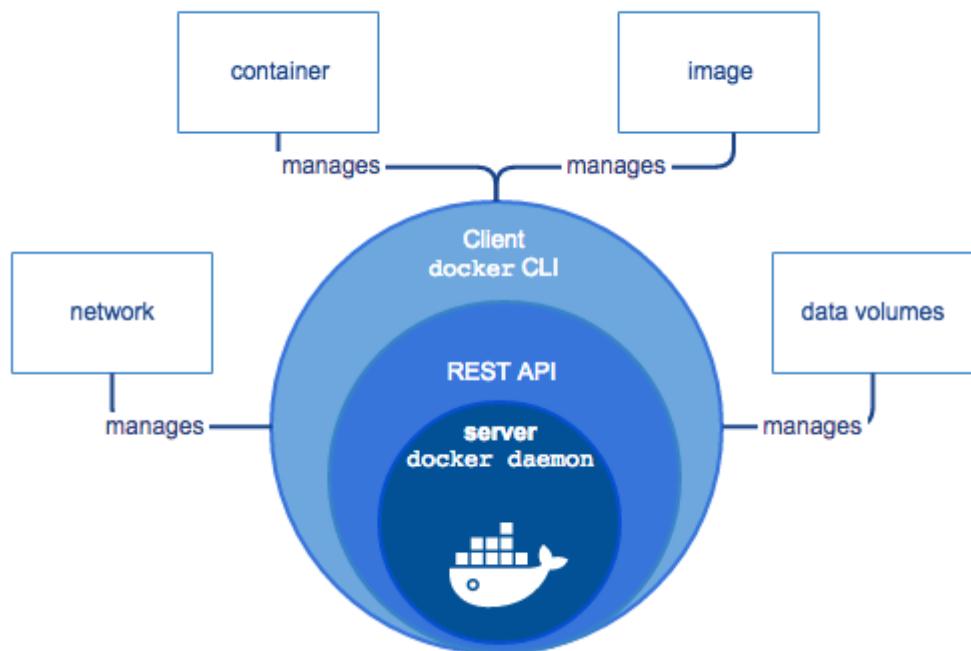
Go หรือ Golang เป็นภาษาโปรแกรมมิ่งที่มีการพัฒนาโดย Google และถูกออกแบบมาเพื่อให้มีประสิทธิภาพสูง มีความปลอดภัย และเหมาะสมกับการพัฒนาโปรแกรมแบบ Concurrent หรือการทำงานพร้อมกันหลาย-thread (Concurrency) ซึ่งเป็นส่วนหนึ่งของความเป็นภาษาซึ่งมีความเป็นเนื้อหาเด่นของ Golang

Go ถูกออกแบบขึ้นมาในปี 2009 โดย Rob Pike, Ken Thompson, และ Robert Griesemer ซึ่งเป็นผู้พัฒนาที่มีชื่อเสียงในโลกโปรแกรมมิ่ง ความเร็วในการคอมไพล์ Go สูง และช่วยให้โปรแกรมที่เขียนด้วย Go มีประสิทธิภาพสูงในการทำงาน โดยเฉพาะการทำงานกับเว็บและ

เซิร์ฟเวอร์ภาษา Go เป็นภาษาแบบสามารถคอมไพล์ได้และเป็นภาษาแบบ Statically typed ซึ่งหมายความว่าต้องประกาศตัวแปรก่อนใช้งาน และมีเครื่องมือช่วยการจัดการข้อผิดพลาด (Error Handling) ที่ดี โดยเซตของฟังก์ชันและการใช้งานของ Go คือจะเก็บในรูปแบบของ Module ที่ช่วยให้การจัดการ Go คือและการแบ่งแยก Go คือมีความสะดวกสบายและเป็นระเบียบมากขึ้น

2.1.14 Docker

Docker คือ เครื่องมือแบบ Open-source ที่ช่วยจำลองสภาพแวดล้อม (Environment) ในการรัน Service หรือ Server ตามหลักการสร้าง Container เพื่อจัดการกับ Library ต่างๆ อีกทั้งยังมีเครื่องมือในการจัดการในเรื่องของ Version Control เพื่อง่ายต่อการจัดการ และปรับปรุงการบริการซึ่งในปัจจุบันในโลกของการพัฒนา Software มีรูปแบบการทำงานแบบ Agile ที่เน้นความรวดเร็วในการส่งมอบงานในแต่ละขั้นตอน Docker จึงเป็นที่รู้จักในวงกว้างและเริ่มเข้ามามีบทบาทอย่างมากในโลกของการพัฒนา Software



รูป 2.9 Docker Architecture

2.1.15 GitLab

GitLab เป็น Git Repository Manager ใช้เพื่อช่วย Deploy Application แบบอัตโนมัติ และ Continuous Integration and Deployment (CI/CD) ทำการ Integrated กับเครื่องมือต่างๆ ดังนั้นกระบวนการ Build, Scan (Source Code Quality Scan), Test ต่างๆ (Unit Test, Functional Test, Web App Test, Load Test), การ Deploy และ Monitoring ซึ่งเป็นกระบวนการของ DevOps นอกจากนี้แล้ว GitLab ยังสามารถช่วยในเรื่องการเพิ่มความปลอดภัย ลดช่องโหว่ หรือ กีดกัน DevSecOps ด้วยการทำ Security Scan ในส่วนต่างๆ ได้แก่ Static Application Security Testing (SAST), Dynamic Application Security Testing (DAST), Dependency Scanning ซึ่งเป็นส่วนหนึ่งของการทำ Software Composition Analysis (SCA) และ Container Image Scan หรือ Docker Image Scan ซึ่งก็เป็นส่วนหนึ่งของการทำ SCA เช่นกัน

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 Performance optimization of web applications using Qwik

งานวิจัยนี้เป็นการศึกษาและวิเคราะห์เกี่ยวกับแนวคิดและเทคโนโลยีที่เกี่ยวข้องกับการพัฒนาแอปพลิเคชันเว็บ (Web Applications) และการจัดการการโต้ตอบของเว็บ เน้นไปที่การโต้ตอบของเว็บแบบหน้าเดียว (Single Page Applications, SPA) และการสร้างหน้าเว็บโดยใช้การренд์ร์บนฝั่งลูกค้า (Client Side Rendering, CSR) ซึ่งเป็นแนวโน้มที่แพร่หลายในการพัฒนาเว็บแอปพลิเคชันในปัจจุบัน

งานวิจัยนี้สำรวจวิธีการ CSR และ SSR (Server Side Rendering) ใน การพัฒนาแอปพลิเคชันเว็บ โดยวิเคราะห์เกี่ยวกับเวลาในการรีเฟรชและความล่าช้าที่เกิดขึ้น และนำผลการวิจัยมาเปรียบเทียบกับกรอบการพัฒนาที่ใช้งานอย่างแพร่หลายอย่าง React.js และ Next.js ซึ่งเป็นกรอบการพัฒนาที่มีการใช้งานอย่างแพร่หลายในการพัฒนาแอปพลิเคชันเว็บ

Server-Side Rendering (SSR) คือ การ Render หน้าเว็บไซต์บนเซิร์ฟเวอร์แทนบนブラัวเซอร์ หรือพูดให้เข้าใจง่าย ๆ ก็คือการแสดงผลบนฝั่งเซิร์ฟเวอร์นั่นเอง ซึ่งการแสดงผลนี้จะเกิดขึ้นทุกครั้งที่มีการส่งคำขอ (Request) โดยผู้ใช้งานจะได้รับข้อมูลที่อัปเดตใหม่ในรูปแบบ XML หรือ JSON ที่ส่งมาโดยตรงจากเซิร์ฟเวอร์ ทำให้ไม่ต้องรอการ渲染บนฝั่ง Client หรือผู้ใช้งาน แต่ต้องรอผลลัพธ์ที่ส่งมาโดยตรงจากเซิร์ฟเวอร์ จึงช่วยลดเวลาในการโหลดหน้าเว็บลงได้มาก แต่ SSR ไม่สามารถปรับเปลี่ยนเนื้อหาบนหน้าเว็บได้โดยอัตโนมัติ ต้องมีการตั้งค่าให้เซิร์ฟเวอร์ทราบว่าต้องแสดงผลแบบ SSR ให้กับผู้ใช้งาน

Client-Side Rendering (CSR) จะเป็นข้อต่างๆ กับ SSR คือ หน้าเพจทั้งหมดจะถูก Render ที่ฝั่ง Client หรือผู้ใช้งาน โดยเมื่อผู้ใช้งานส่งคำขอไปยังเซิร์ฟเวอร์ เซิร์ฟเวอร์จะส่งไฟล์ HTML ที่มีโครงสร้างของหน้าเว็บที่ต้องการ ให้ผู้ใช้งาน自行解析 ตามไฟล์ JavaScript ที่ติดต่ออยู่ในหน้าเพจ ทำให้สามารถเปลี่ยนแปลงเนื้อหาบนหน้าเพจได้โดยอัตโนมัติ ไม่ต้องรอการโหลดใหม่ แต่ CSR ไม่สามารถปรับเปลี่ยนเนื้อหาบนหน้าเพจได้โดยอัตโนมัติ ต้องมีการตั้งค่าให้เซิร์ฟเวอร์ทราบว่าต้องแสดงผลแบบ CSR ให้กับผู้ใช้งาน

บทที่ 3

การออกแบบและพัฒนา

ระบบ Front-end Builder หรือ Low-Code Platform เป็นแนวคิดใหม่ในการสร้างเว็บไซต์และแอปพลิเคชัน โดยเฉพาะสำหรับผู้ที่ไม่มีประสบการณ์หรือทักษะการเขียนโค้ด HTML และ CSS อย่างเชี่ยวชาญ แนวคิดของ Low-Code จะทำให้งานสร้างเว็บไซต์เป็นเรื่องง่ายและไม่ซับซ้อนเลย Low-Code Platform ช่วยให้นักพัฒนาและผู้สร้างแอปพลิเคชันมีความสามารถสร้างแอปพลิเคชันได้อย่างรวดเร็วโดยไม่จำเป็นต้องมีระดับทักษะการเขียนโค้ดสูง แนวคิดนี้ช่วยให้ผู้ที่ไม่มีความรู้ด้านการเขียนโปรแกรมหรือไม่มีความเชี่ยวชาญในการเขียนโค้ดสามารถออกแบบและพัฒนาเว็บไซต์หรือแอปพลิเคชันได้แบบง่ายๆ (APIเพิ่มเติม)

สรุปแนวคิด Low-Code คือวิธีใหม่ในการสร้างแอปพลิเคชันที่ช่วยลดความซับซ้อนในการเขียนโค้ดและเปิดโอกาสให้นักพัฒนาและผู้ที่ไม่มีความรู้ทางด้านโค้ดสามารถสร้างแอปพลิเคชันได้อย่างรวดเร็วและง่ายยิ่งขึ้น ได้

3.1 การออกแบบสถาปัตยกรรมของระบบ (System Architecture)

3.1.1 Three-Tier architecture



รูป 3.1 Three-Tier Architecture ของระบบเว็บไซต์ Front-end Builder

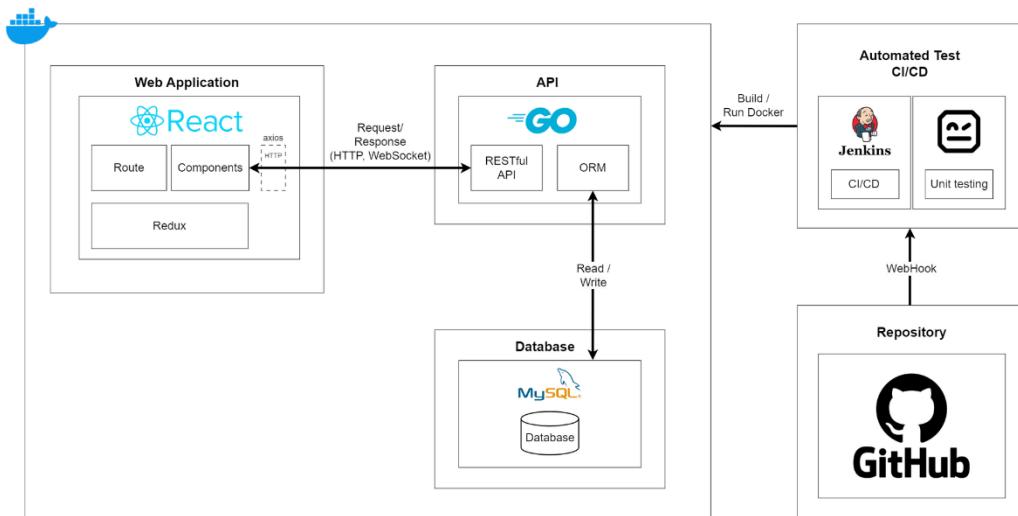
สถาปัตยกรรมของระบบ Three-Tier Architecture จะแบ่งออกเป็น 3 ชั้นดังนี้

ชั้นที่ 1 Presentation Layer สำหรับประมวลผลส่วนการติดต่อผู้ใช้งาน ซึ่งในส่วนนี้จะใช้ Vue.js สำหรับสร้างเว็บแอพพลิเคชันในฝั่งโคลอีนในรูปแบบของ HTML, Tailwindcss

ชั้นที่ 2 Service Layer สำหรับทำหน้าที่ติดต่อ Function ประมวลผลการทำงานและประสานงานระหว่าง Data Layer และ Presentation Layer ในส่วนนี้จะใช้ Node.js และ Express JS

ชั้นที่ 3 Data layer ส่วนที่ใช้ติดต่อกับฐานข้อมูล จัดเก็บข้อมูล และเข้าถึงข้อมูล โดยจะใช้ MySQL ในการจัดการกับระบบฐานข้อมูล

3.1.2 System architecture diagram



รูป 3.2 System architecture diagram ของระบบเว็บไซต์ Front-end Builder

3.2 ข้อกำหนดโครงการ (System Requirements Specification)

ความต้องการของระบบ Front-end Builder

ตาราง 3.1 รายการความสามารถของระบบ

ID	Details	Type	Priority
ส่วนการใช้งานสำหรับผู้ใช้งานสำหรับผู้ใช้งานทุกประเภท			
R01	ชื่อผู้ใช้และรหัสผ่าน: ระบบจะต้องให้ผู้ใช้ป้อน Username และ Password หรือ Google เพื่อเข้าสู่ระบบ	<ul style="list-style-type: none"> ● Login ● Functional 	Must have
R02	การจำกัดการเข้าสู่ระบบ: ระบบต้องกำหนดระยะเวลาหรือจำนวนครั้งที่ผู้ใช้สามารถล็อกอินเข้าสู่ระบบได้	<ul style="list-style-type: none"> ● Login ● Functional 	Must have
R03	ระบบจะต้องบังคับให้ User กรอกข้อมูลตามเงื่อนไขที่ต้องการ Username (Text), Password(Text), First Name(Text), Last Name(Text), Gmail(Text)	<ul style="list-style-type: none"> ● Register ● Functional 	Must have
R04	ข้อมูลส่วนบุคคล: ระบบควรระบุข้อมูลส่วนบุคคลที่จำเป็นเพื่อการลงทะเบียน เช่น ชื่อจริง, นามสกุล, ที่อยู่อีเมล	<ul style="list-style-type: none"> ● Register ● Functional 	Must have
R05	รายการโปรเจกต์: ระบบควรแสดงรายการโปรเจกต์ที่ผู้ใช้เคยสร้างในพื้นที่ทำงานโดยแสดงชื่อ โปรเจกต์ และรายละเอียดสั้น ๆ เพื่อให้ผู้ใช้สามารถเข้าถึงและเรียกคุ้มโปรเจกต์ได้ง่ายๆ	<ul style="list-style-type: none"> ● Workspace ● Functional 	Must have
R06	การแก้ไขและการลบโปรเจกต์: ระบบควรให้ผู้ใช้มีความสามารถในการแก้ไขข้อมูลของโปรเจกต์ที่เคยสร้างและลบโปรเจกต์หากต้องการ	<ul style="list-style-type: none"> ● Workspace ● Functional 	Must have

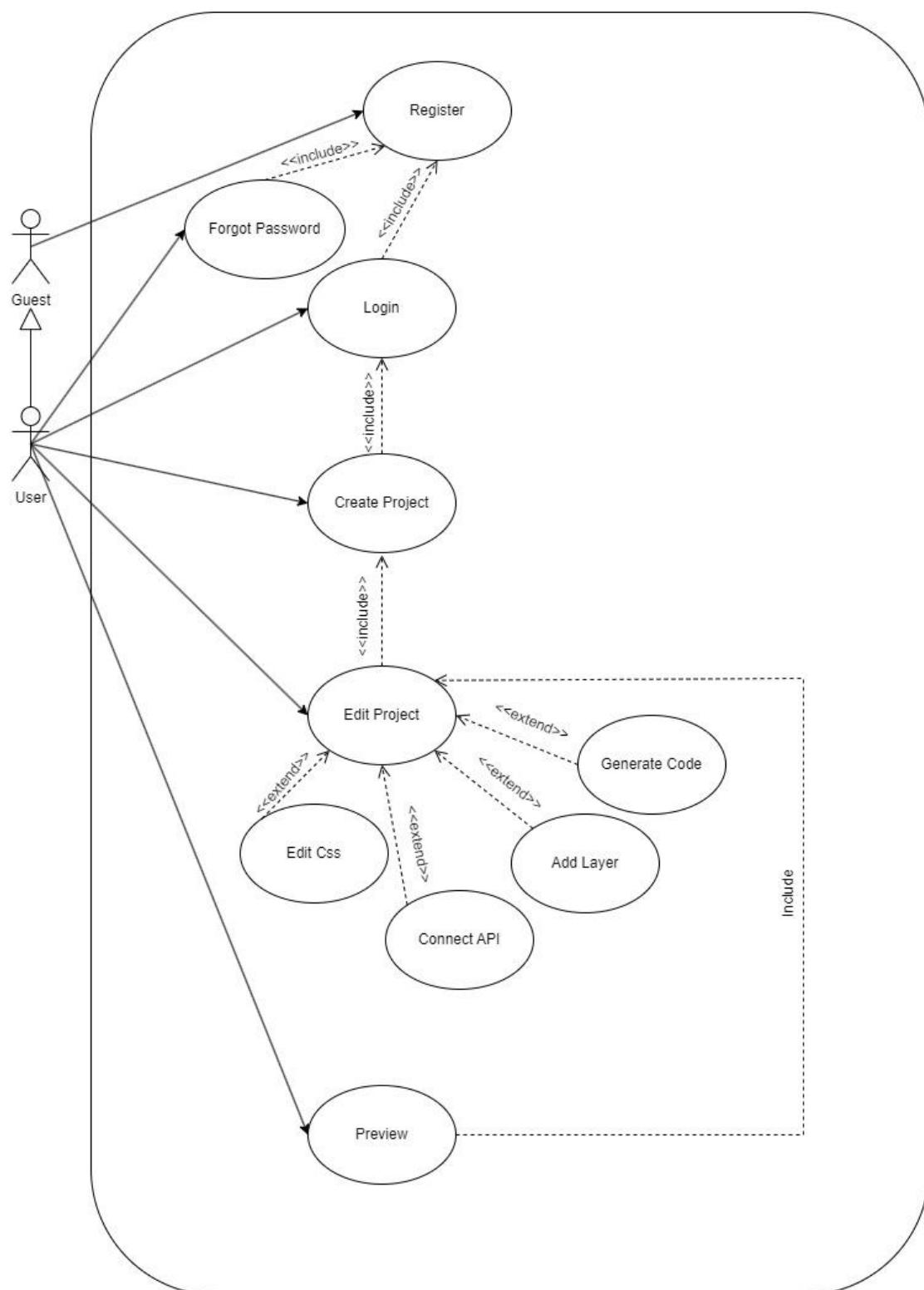
ตาราง 3.1 รายการความสามารถของระบบ (ต่อ)

ID	Detail	Type	Priority
ส่วนการใช้งานสำหรับผู้ใช้งานสำหรับผู้ใช้งานทุกประเภท			
R07	การเพิ่มโปรเจกต์ใหม่: ผู้ใช้สามารถสร้างโปรเจกต์ใหม่ในพื้นที่ทำงานโดยระบบควรให้ผู้ใช้กรอกข้อมูลเบื้องต้นเมื่อสร้างโปรเจกต์ใหม่ เช่นชื่อโปรเจกต์และรายละเอียด.	<ul style="list-style-type: none"> ● Workspace ● Functional 	Must have
R08	ระบบจะส่งรหัส Verify Code ไปยัง E-mail ผู้ใช้	<ul style="list-style-type: none"> ● Forget password ● Functional 	Must have
R09	ผู้ใช้ต้องนำรหัส Verify Code เพื่อมาเย็บขั้น	<ul style="list-style-type: none"> ● Forget password ● Functional 	Must have
R10	การเพิ่ม Page ใหม่ได้ระบบสามารถมีความสามารถในการสร้างหน้า (Page) ใหม่ในพื้นที่ทำงาน ซึ่งผู้ใช้สามารถกำหนดชื่อและรายละเอียดของหน้าได้ตอนสร้าง	<ul style="list-style-type: none"> ● Working space ● Functional 	Must have
R11	การแสดง Layers ของ Element ต่างๆ ระบบการแสดงรายการ Layers หรือรายการของ Element ที่อยู่บนหน้าทำงาน ซึ่งผู้ใช้สามารถเลือก Element และคลิกและการกับ Layers ได้และสามารถแก้ไข Style(CSS) Element.	<ul style="list-style-type: none"> ● Working space ● Functional 	Must have
R12	การเชื่อมต่อ API ระบบควรมีความสามารถในการเชื่อมต่อ กับ API ภายนอกเพื่อนำข้อมูลหรือผลลัพธ์จาก API มาแสดงในหน้าทำงาน ซึ่งผู้ใช้ควรสามารถกำหนดการเรียกใช้API และการแสดงผลได้	<ul style="list-style-type: none"> ● Working space ● Functional 	Must have

ตาราง 3.1 รายการความสามารถของระบบ (ต่อ)

ID	Detail	Type	Priority
ส่วนการใช้งานสำหรับผู้ใช้งานสำหรับผู้ใช้งานทุกประเภท			
R13	<p>การแสดง Code และการดาวน์โหลด ระบบ ควรมีความสามารถในการแสดง Code ที่ถูกสร้างขึ้นจากหน้าทำงาน ซึ่งผู้ใช้ สามารถดู Code ที่ถูกสร้างขึ้นจากการ ออกแบบหน้า</p> <p>ทำงานและระบบควรมีความสามารถใน การให้ผู้ใช้ดาวน์โหลด Code ที่ถูกสร้างมา ได้เพื่อใช้ในการนำไปใช้งานหรือบูรณาใน ภาษาหลัง</p>	<ul style="list-style-type: none"> ● Working space ● Functional 	Must have
R14	<p>การแสดงหน้าตัวอย่างและข้อมูลจาก API ระบบควรมีความสามารถในการแสดงหน้า ตัวอย่าง (Preview) ของหน้าที่ผู้ใช้ออกแบบ โดยแสดงผลอย่างคล้ายกับสถานะจริง</p> <p>ของหน้าทำงาน ระบบควรมีความสามารถ ในการทำการเชื่อมต่อกับ API จริงและดึง^{ดึง} ข้อมูลจาก API เพื่อแสดงในหน้าตัวอย่าง</p>	<ul style="list-style-type: none"> ● Preview ● Functional 	Must have

3.3 การออกแบบแบบบัญญาค์ (Use case diagram)



รูป 3.3 แผนภารท Use Case Diagram ของระบบเว็บไซต์ Front-end Builder

ตาราง 3.2 รายละเอียด Use Case Sign up

Use Case Name	Sign up
Use Case ID	UC-01
Actor	Guest
Description	ผู้ใช้สามารถสมัครสมาชิกก่อนเข้าใช้งานเว็บไซต์
Preconditions	เข้าสู่ Web Application
Flow of Events	<ol style="list-style-type: none"> 1. เข้าสู่หน้า Sign in Page 2. คลิกเข้าสู่หน้า Sign Up 2. กรอกข้อมูล First name , Last name , Email , Password, และ Confirm password 3. กดปุ่มลงทะเบียน
Postconditions	ลงทะเบียนสำเร็จ

ตาราง 3.3 รายละเอียด Use Case Forgot Password

Use Case Name	Forgot Password
Use Case ID	UC-02
Actor	User
Description	ผู้ใช้สามารถขอเปลี่ยนพาสเวิร์ด
Preconditions	ต้องเป็นสมาชิกของระบบ
Flow of Events	<ol style="list-style-type: none"> 1. เข้าสู่หน้า Forgot Password 2. กรอก Email 3. กดปุ่ม Send E-mail 4. รับรหัสยืนยันทาง E-mail 5. นำรหัสยืนยันมากรอก 6. กดปุ่ม Verify 7. กรอกรหัสผ่านใหม่ 8. กดปุ่ม Reset Password
Postconditions	เปลี่ยนรหัสผ่านสำเร็จ

ตาราง 3.4 รายละเอียด Use Case Sign in (Username, Password)

Use Case Name	Sign in
Use Case ID	UC-03
Actor	User
Description	ผู้ใช้สามารถเข้าสู่ระบบ
Preconditions	ต้องเป็นสมาชิกของระบบ
Flow of Events	<ol style="list-style-type: none"> 1. เข้าสู่หน้า Sign in page 2. กรอก Username และ Password 3. กดปุ่ม Login
Postconditions	เข้าสู่ระบบสำเร็จ

ตาราง 3.5 รายละเอียด Use Case Create Project

Use Case Name	Create Project
Use Case ID	UC-04
Actor	User
Description	ส่วนที่สามารถสร้าง Project ใหม่ได้ และสามารถเข้าไปทำงานใน Project ต่างๆ ได้
Preconditions	ต้องเข้าสู่ระบบ
Flow of Events	<ol style="list-style-type: none"> 1. เข้าสู่หน้า Work Space 2. กดปุ่ม New Project 3. ตั้งชื่อ Project 4. กดปุ่ม Create 5. ดับเบิล Project ที่จะเข้าไปทำงาน
Postconditions	สร้างโปรเจกต์สำเร็จ

ตาราง 3.6 รายละเอียด Use Case Create and Use Element

Use Case Name	Create and Use Element
Use Case ID	UC-05
Actor	User
Description	สร้างและใช้งานเครื่องมือ
Preconditions	ต้องเข้าสู่ระบบ สร้างโปรเจกต์
Flow of Events	1. สร้าง Project เพื่อเริ่มใช้งาน 2. เลือกเมนู Toolbox 3. เลือก Elements ต่างๆ เพื่อทดสอบการใช้งาน 4. ทดลองใช้งาน Elements โดยให้ผู้ใช้สร้าง Element และเลือก Elements ต่างๆ เพื่อแทรก Element ใหม่ ให้อยู่ภายใต้ Element ที่สร้าง
Postconditions	สร้าง Element สำเร็จ

ตาราง 3.7 รายละเอียด Use Case Format CSS

Use Case Name	Format CSS
Use Case ID	UC-06
Actor	User
Description	สร้างและใช้งานเครื่องมือ
Preconditions	ต้องเข้าสู่ระบบ สร้างโปรเจกต์ สร้าง Element
Flow of Events	1. เลือก Element ที่ต้องการจัดการ CSS 2. เลือกเมนู Properties 3. เลือกเมนู Style 4. เลือก Style ที่ต้องการ เพื่อทดสอบการใช้งาน 5. ทดลองใช้งาน Style โดยให้ผู้ใช้เลือก Style ที่ต้องการใช้งาน เช่น Fill เพื่อเปลี่ยนสี Background
Postconditions	ปรับเปลี่ยนスタイル Element สำเร็จ

ตาราง 3.8 รายละเอียด Use Case API

Use Case Name	API
Use Case ID	UC-07
Actor	User
Description	สร้างและใช้งานเครื่องมือร่วมกับ API
Preconditions	ต้องเข้าสู่ระบบ สร้างโปรเจกต์ สร้าง Element
Flow of Events	1. เลือก Element Div (row, column) 2. เลือกเมนู Properties 3. เลือกเมนู Data 4. เลือก Event เป็น OnLoad เพื่อใช้งาน API เมื่อเริ่มโหลดหน้าเว็บ 5. เลือก Edit Action เพื่อตั้งค่าการทำงานของ API 6. ใส่ลิงก์ Api และ Element, Json(Key : Value) เพื่อ map ข้อมูลให้กับ Element 7. ทดลองใช้งาน API โดยให้ผู้ใช้ต้องนำ API ที่จัดการข้อมูลมาเลี้ยวจากหลังบ้าน มาแสดงผล
Postconditions	เชื่อม API เพื่อใช้งานร่วมกับ Element สำเร็จ

ตาราง 3.9 รายละเอียด Use Case Preview

Use Case Name	Preview
Use Case ID	UC-08
Actor	User
Description	ส่วนที่สามารถแสดงหน้าเว็บไซต์หลังจากเชื่อมต่อกับ API
Preconditions	ต้องเข้าสู่ระบบ สร้างโปรเจกต์
Flow of Events	1. กดปุ่ม Preview 2. ทดลองระบบ
Postconditions	ดูตัวอย่างเว็บไซต์ใส่ข้อมูลจริงตามที่ API ส่งมา

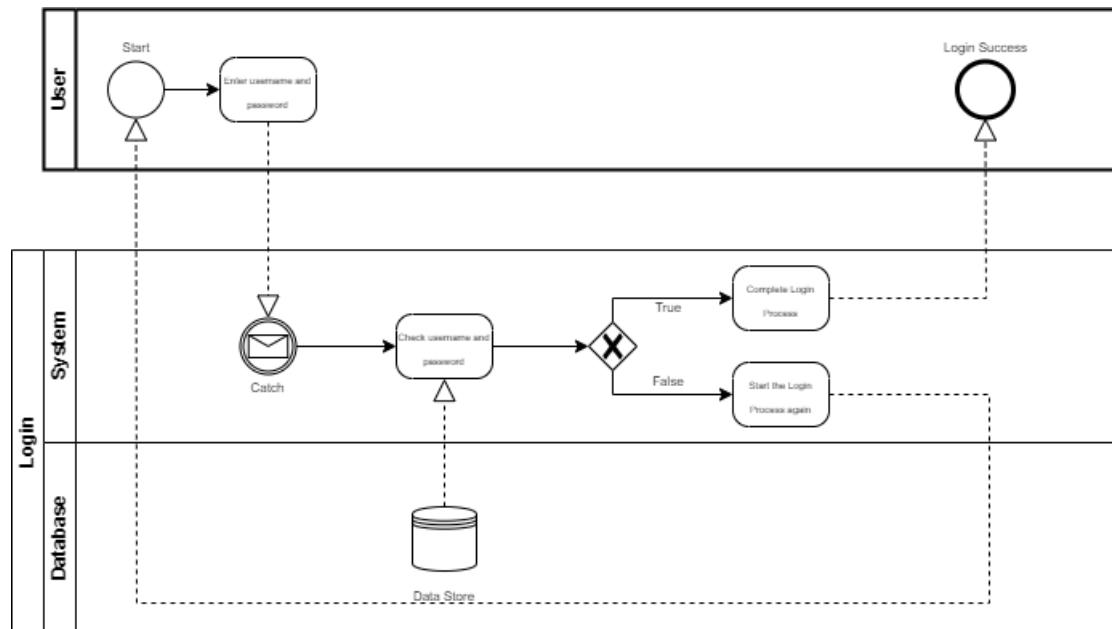
ตาราง 3.10 รายละเอียด Use Case Manual JavaScript

Use Case Name	Manual JavaScript
Use Case ID	UC-09
Actor	User
Description	ใช้งาน Manual JavaScript
Preconditions	ต้องเข้าสู่ระบบ สร้างโปรเจกต์
Flow of Events	1. เลือก Element ที่ต้องการใช้งาน JavaScript 2. เลือกเมนู Properties 3. เลือกเมนู Data 4. เลือก Event เป็น OnClick 5. เลือก Edit Action 6. ทดลองใช้งาน JavaScript เพื่อเปลี่ยนข้อมูลภายใน Text
Postconditions	สามารถเขียน Manual Java Script ได้

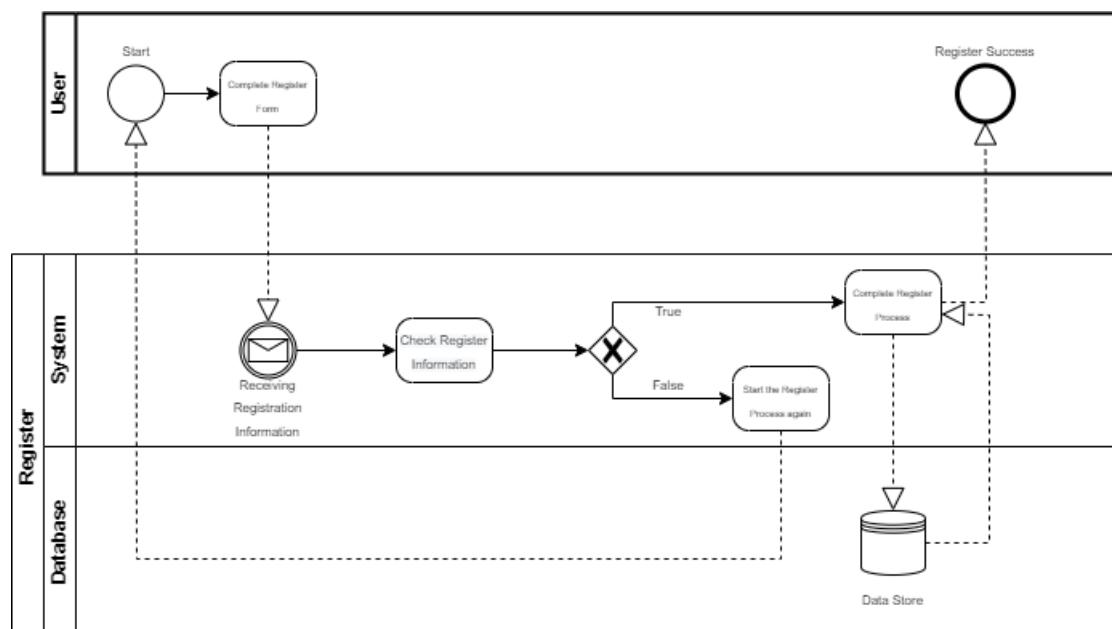
ตาราง 3.11 รายละเอียด Use Case Generate Code

Use Case Name	Generate Code
Use Case ID	UC-10
Actor	User
Description	ใช้งาน Generate Code
Preconditions	ต้องเข้าสู่ระบบ สร้างโปรเจกต์
Flow of Events	1. เลือกปุ่ม Generate Code เพื่อแสดงโค๊ด 2. ทดลองใช้งาน Generate Code โดยจะแสดงโค๊ดของหน้าเพจที่อยู่ใน Workspace ปัจจุบัน
Postconditions	Generate Code สำเร็จ

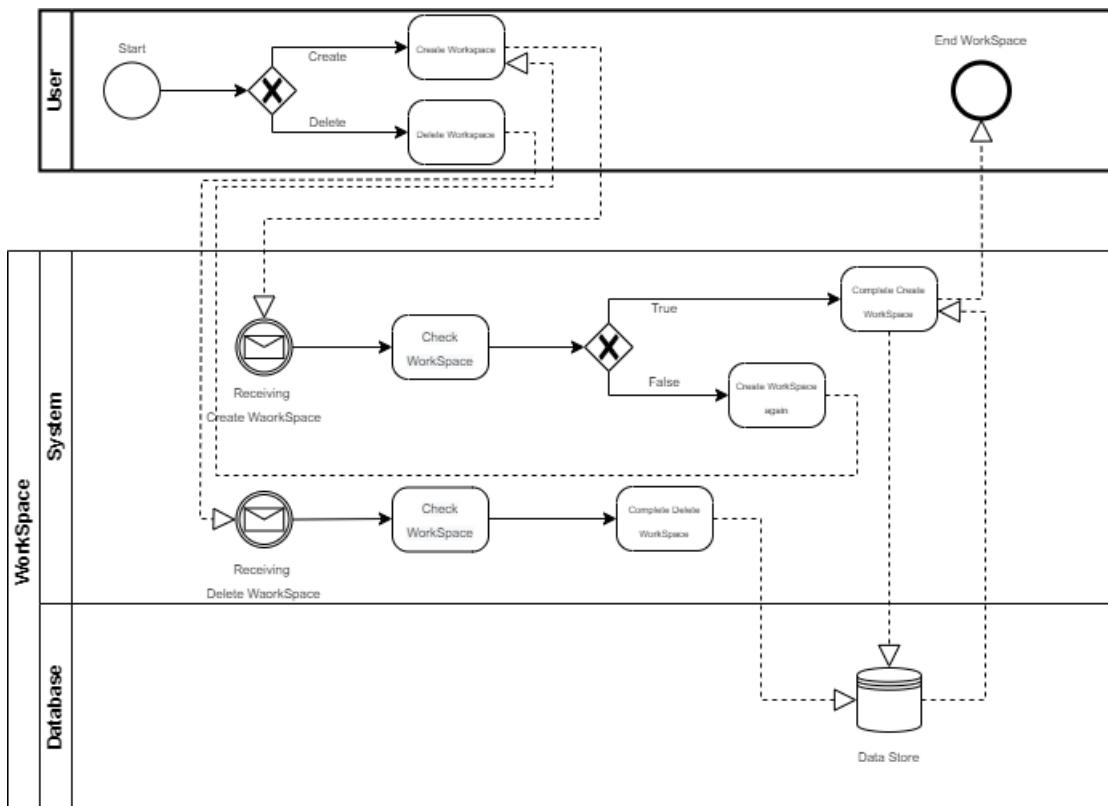
3.4 BPMN (Business Process Modeling Notation)



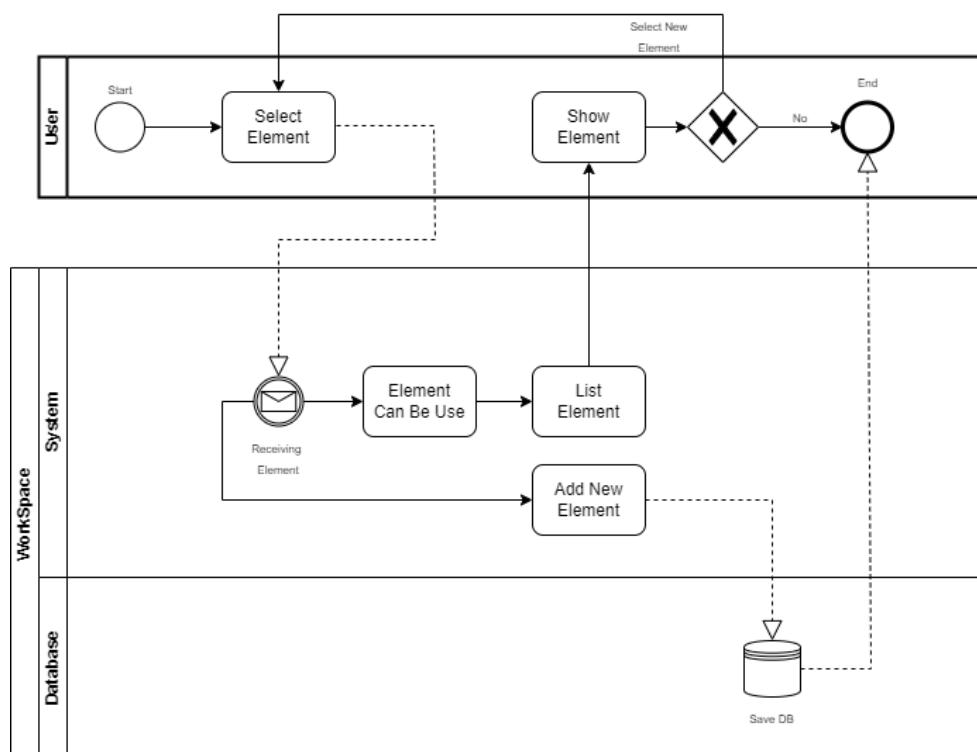
รูป 3.4 การออกแบบ (BPMN-Login) ของระบบเว็บไซต์ Front-end Builder



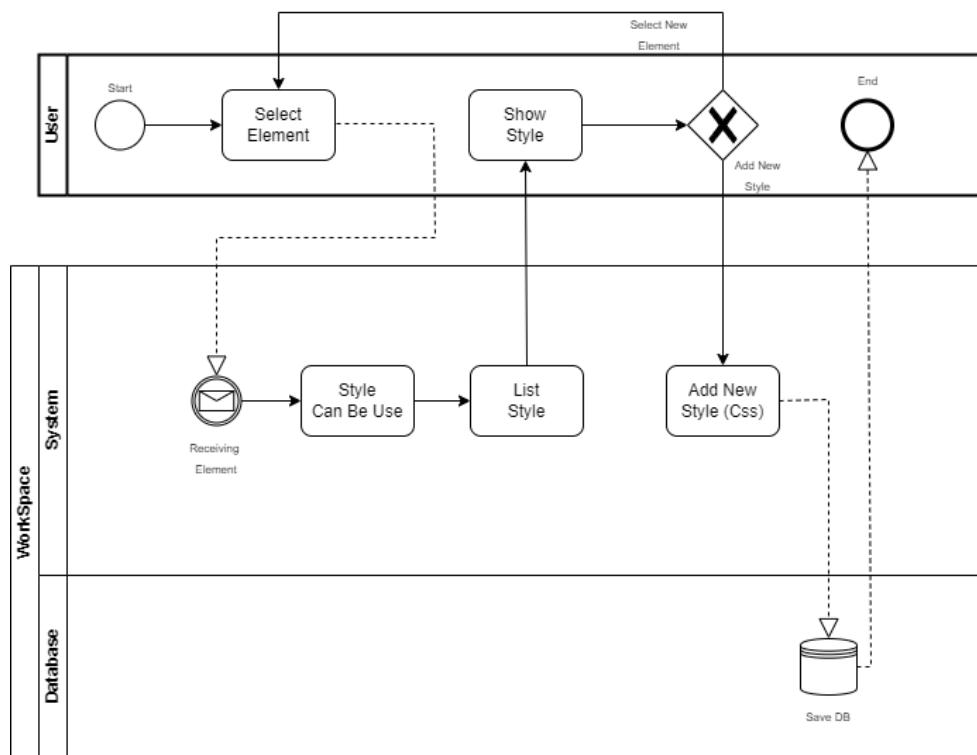
รูป 3.5 การออกแบบ (BPMN-Register) ของระบบเว็บไซต์ Front-end Builder



รูป 3.6 การออกแบบ (BPMN-Workspace) ของระบบเว็บไซต์ Front-end Builder

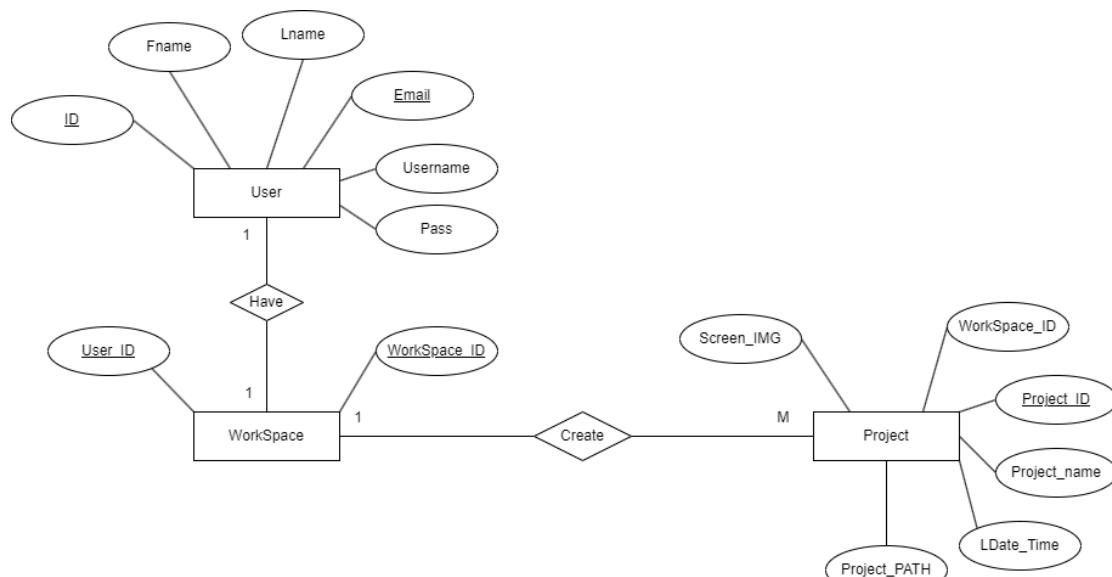


รูป 3.7 การออกแบบ (BPMN-Select Element) ของระบบเว็บไซต์ Front-end Builder



รูป 3.8 การออกแบบ (BPMN-Style Element) ของระบบเว็บไซต์ Front-end Builder

3.5 การออกแบบฐานข้อมูล (Er-diagram)



รูป 3.9 การออกแบบ (ER-Diagram) ของระบบเว็บไซต์ Front-end Builder

ตาราง 3.12 รายละเอียดฐานข้อมูลตาราง User

คีย์	ชื่อแอดминистратор	ชนิดตัวแปร	รายละเอียด
PK	Email	Varchar(50)	อีเมล
	Fname	varchar(100)	ชื่อ
	Lname	varchar(100)	นามสกุล
FK	ID	Int(3)	ลำดับการใช้งาน
	Username	varchar(30)	ผู้ใช้งาน
	Pass	varchar(30)	รหัสผ่าน

ตาราง 3.13 รายละเอียดฐานข้อมูลตาราง Workspace

คีย์	ชื่อแอดминистратор	ชนิดตัวแปร	รายละเอียด
PK	Workspace_ID	int(3)	รหัสพืนที่การทำงาน
FK	User_ID	int(3)	ผู้ใช้งาน

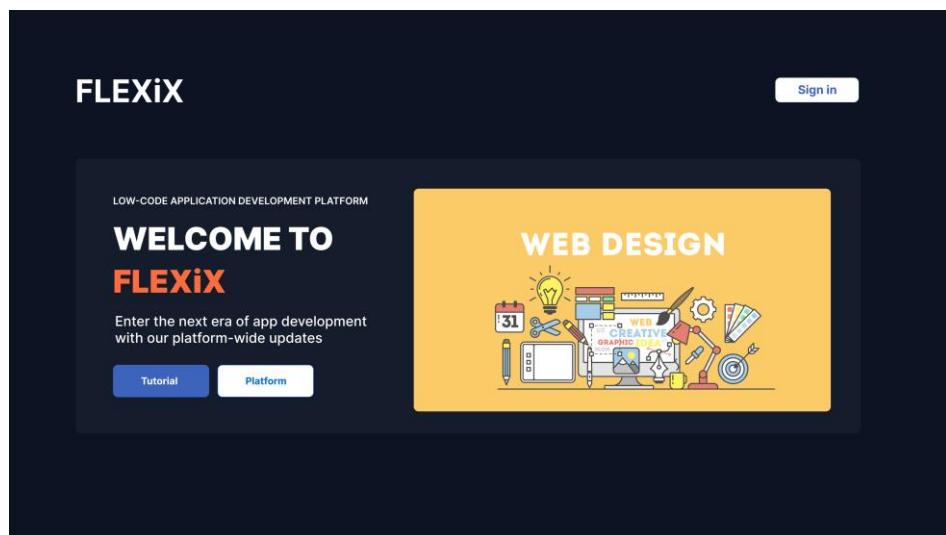
ตาราง 3.14 รายละเอียดฐานข้อมูลตาราง Project

คีย์	ชื่อแอดминистратор	ชนิดตัวแปร	รายละเอียด
PK	Project_ID	int(3)	รหัสโครงการ
	Project_name	varchar(30)	ชื่อโครงการ
	Workspace_ID	int(3)	รหัสโครงการ
	Project_PATH	Varchar(255)	ลิงก์โครงการ
	LDate_time	Date	เวลาการทำงานล่าสุด
	Screen_IMG	varchar(255)	รูปภาพโครงการ

3.6 การออกแบบส่วนติดต่อผู้ใช้ (User interface Design)

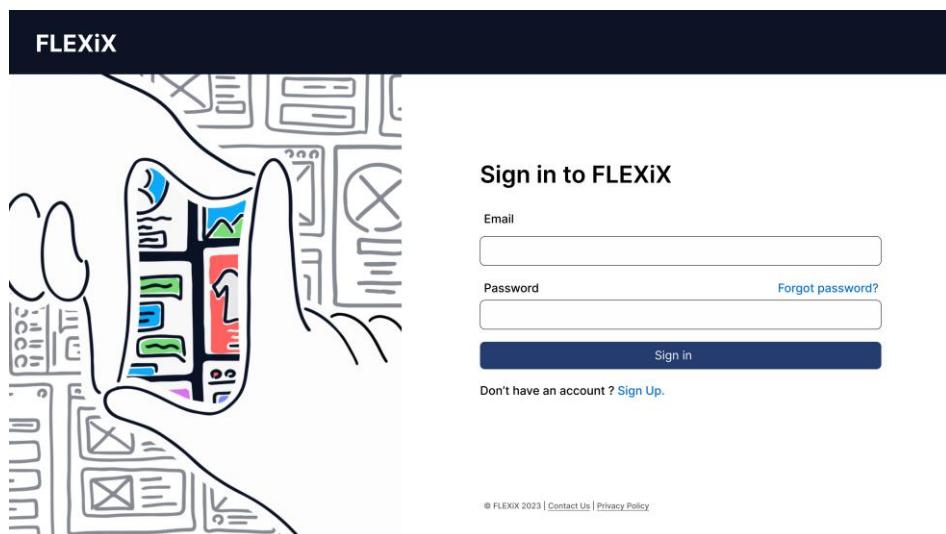
1) การออกแบบ UI Web Application หน้าหลัก

หน้าหลัก (Home) เป็นหน้าที่จะอธิบายเกี่ยวกับระบบของ Front-end Builder คืออะไร หน้าหลักสามารถไปยังหน้า Sign in เพื่อให้ผู้ใช้เข้าสู่ระบบและเข้าไปสร้างโปรเจ็ค



รูป 3.10 หน้าหลักเว็บไซต์ (Home)

2) ให้ผู้ใช้ใส่ E-mail, Password ที่ลงทะเบียนไว้เพื่อเข้าใช้งาน



รูป 3.11 หน้าเข้าสู่ระบบ (Login)

3) การออกแบบ UI Web Application ส่วนของ Sign up

ระบุ First Name, Last Name, Email, Password จากนั้นให้ผู้ใช้ยอมรับข้อตกลงกับระบบ
เมื่อกด Next ระบบจะนำไปยังหน้า Sign in เพื่อเข้าใช้งาน

Sign up

First Name Last Name

Email

Password

Confirm Password

I agree to [FLEXiX Privacy Policy](#)

I would like to receive communication from FLEXiX via email

Next

Already have an account? [Sign in](#)

รูป 3.12 หน้าเข้าสมัครสมาชิก (Sign Up)

4) การออกแบบ UI Web Application ส่วนของ Forgot Password(1)

ให้ผู้ใช้ระบุ Email ที่เคยสมัครสมาชิก

Forgot password

Enter Email Address

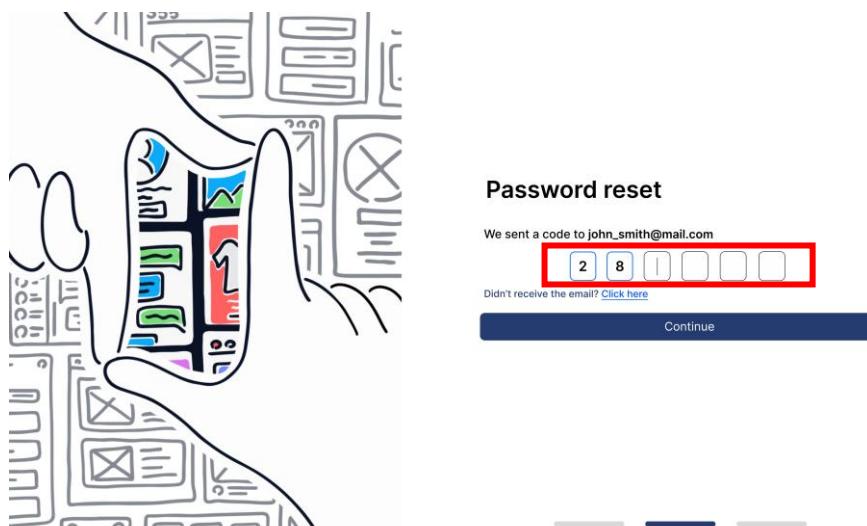
Send

[← Back to sign in](#)

รูป 3.13 หน้าลืมรหัสผ่าน (Forgot Password)

5) การออกแบบ UI Web Application ส่วนของ Forgot Password(2)

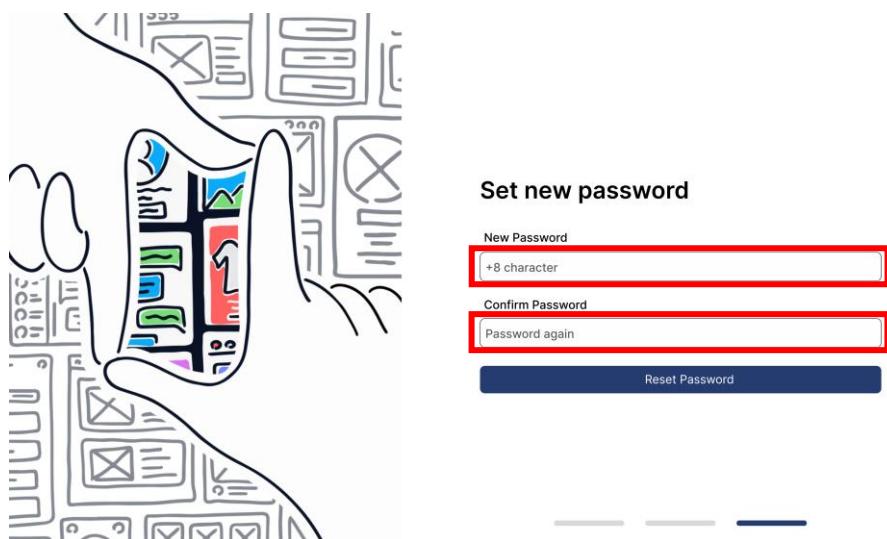
ผู้ใช้จะได้รับรหัสยืนยันทาง Email ให้ผู้ใช้นำมาใส่ในช่องที่กำหนดไว้



3.14 หน้าลืมรหัสผ่าน (Password Reset)

6) การออกแบบ UI Web Application ส่วนของ Forgot Password(3)

ให้ผู้ใช้ระบุรหัสผ่านใหม่เพื่อทำการตั้งรหัสผ่านใหม่



รูป 3.15 หน้าตั้งค่ารหัสผ่านใหม่ (Set new password)

7) การออกแบบ UI Web Application ส่วนของ Platform

หน้าอธิบายเกี่ยวกับ Platform ของเว็บไซต์

FLEXiX

PLATFORM

การพัฒนาแบบ Low-Code

- 1 Low-Code คืออะไร ?
- 2 คุณสมบัติของ Low-Code
- 3 ใครคือคนที่ต้องการ Low-Code ?
- 4 Low-Code vs High Code



**Low-Code
คืออะไร ?**

Low-code เป็นวิธีการพัฒนาแอปพลิเคชันที่ยกระดับการเขียนโค้ดจาก การใช้ข้อความไปสู่การใช้ส่วนกราฟิก

Low-code คือแพลตฟอร์มหรือเครื่องมือที่ช่วยให้นักพัฒนาสร้างแอปพลิเคชันโดยไม่ต้องเขียนโค้ดเดิมรูปแบบหรือเขียนโค้ดภาษา บันทึกโดยคำนับขั้นตอนในการพัฒนาและแก้ไขการสร้างแอปพลิเคชันเร็วขึ้นและง่ายขึ้น

**คุณสมบัติของ
Low-Code**

Low-code มีคุณสมบัติหลักอย่างที่ช่วยให้นักพัฒนาสามารถสร้างแอปพลิเคชันได้อย่างรวดเร็ว

การสร้างด้วยการลากและวาง (Drag-and-Drop) สามารถสร้าง UI และเพิ่มฟังก์ชันโดยไม่ต้องใช้การลากและวางของค่าประกอบแบบกราฟิก ไม่ต้องเขียนโค้ดเข้าไปเองในกระบวนการนี้

**ใครคือคนที่ต้องการ
Low-Code ?**

ในบทบาทของนักพัฒนาซอฟต์แวร์

Low-Code เป็นเครื่องมือสำคัญในการเพิ่มความเร็วในการพัฒนาโปรแกรม และแอปพลิเคชันโดยช่วยให้ผู้พัฒนาสร้างและแก้ไขโค้ดได้อย่าง ง่ายและมีค่าโดยบากบานมาก เช่นเดียวกับการทำงานของนักพัฒนาซอฟต์แวร์เพิ่มประสิทธิภาพในการจัดการโปรเจกต์ของพวงมาลัย

**Low-Code
vs High Code**

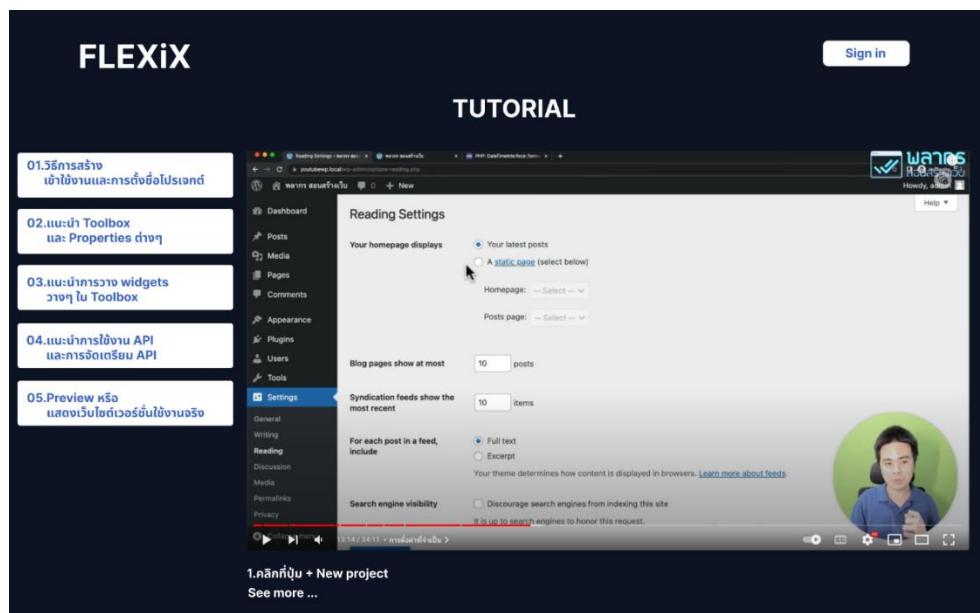
**การพัฒนาแอปแบบ low-code และแบบเดิม
แตกต่างอย่างไร**

การพัฒนาแบบ low-code ต่างจากการพัฒนาแอปแบบเดิมๆ ที่ต้องเขียนโค้ดมาโดย low-code ช่วยเพิ่มความสามารถในการสร้างแอปพลิเคชันให้กับผู้คนจำนวนมากโดยใช้ความรู้ขั้นต่ำ และเครื่องมือง่ายๆ การพัฒนาแบบเดิมต้องอาศัยนักพัฒนา มืออาชีพที่มีความเชี่ยวชาญในการเขียนโค้ดด้วยตนเอง

รูป 3.16 หน้าบอกจุดประสงค์เว็บไซต์ (Platform)

8) การออกแบบ UI Web Application ส่วนของ Tutorial

หน้าสอนการใช้งานเว็บไซต์ (Tutorial) สำหรับแสดงการสอนใช้งานเว็บทั้งหมด



รูป 3.17 หน้าสอนการใช้งานเว็บไซต์ (Tutorial)

9) การออกแบบ UI Web Application ส่วนของ Workspace

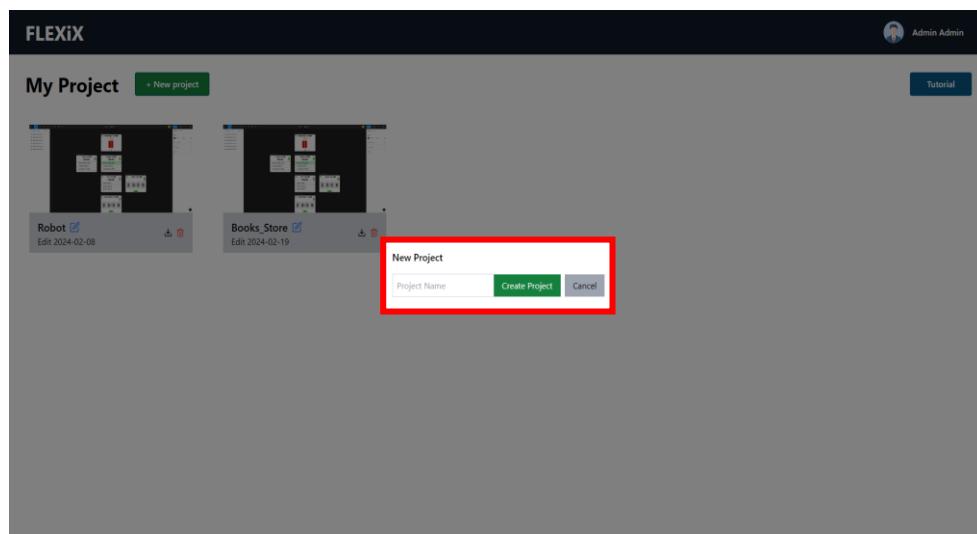
หน้าแสดงโปรเจกต์ทั้งหมดที่อยู่ภายใต้ Workspace ของผู้ใช้ ผู้ใช้สามารถ Create Project ได้จากหน้านี้



รูป 3.18 หน้าแสดงโปรเจกต์ที่มีทั้งหมด (Workspace)

10) การออกแบบ UI Web Application ส่วนของ Pop-up (Create Project)

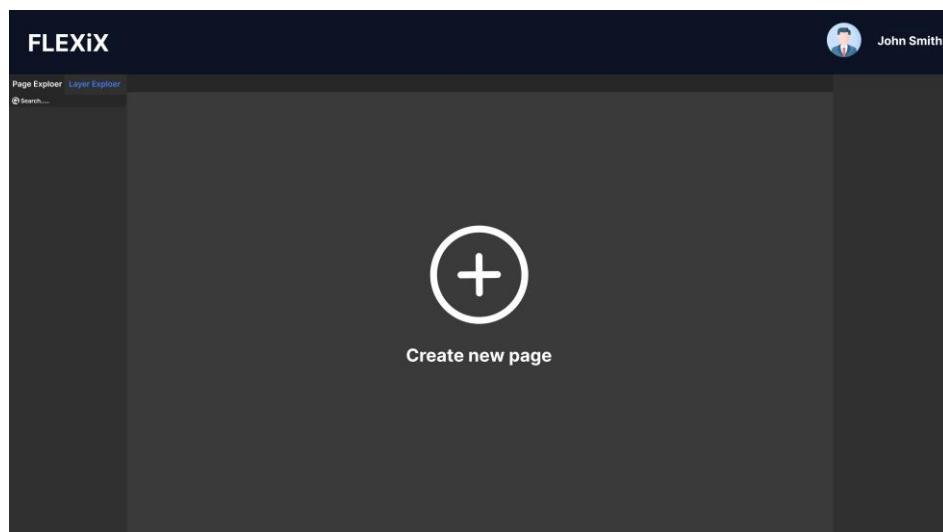
ผู้ใช้ระบบต้อง Project ที่ต้องการสร้าง



รูป 3.19 หน้าป้อนข้อมูลสร้างโปรเจกต์ (Create project)

11) การออกแบบ UI Web Application ส่วนของ Project Space

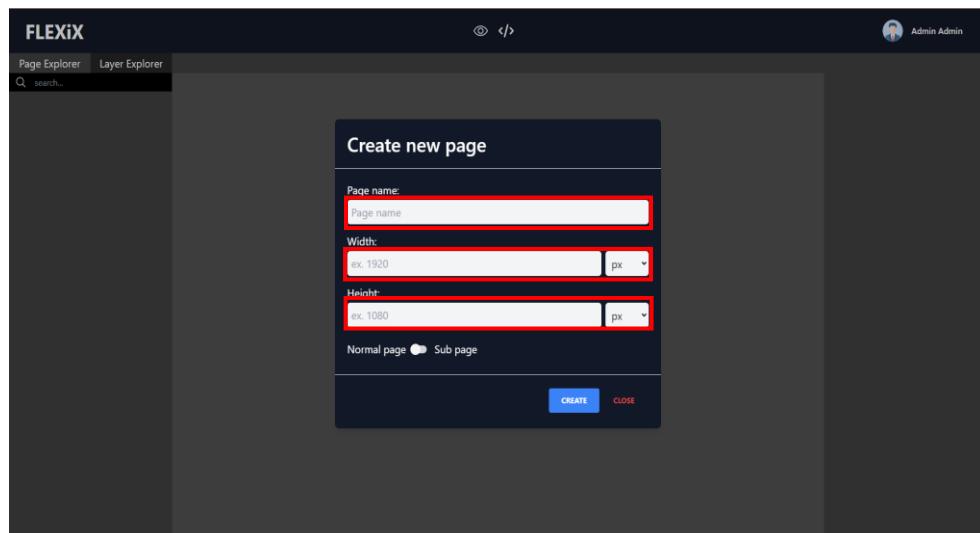
หน้าสำหรับสร้าง Page(ครั้งแรก) เพื่อใช้งานระบบ Front-end Builder



รูป 3.20 หน้าสร้างหน้าออกแบบ (Create new page)

12) การออกแบบ UI Web Application ส่วนของ Pop-up (Create Page)

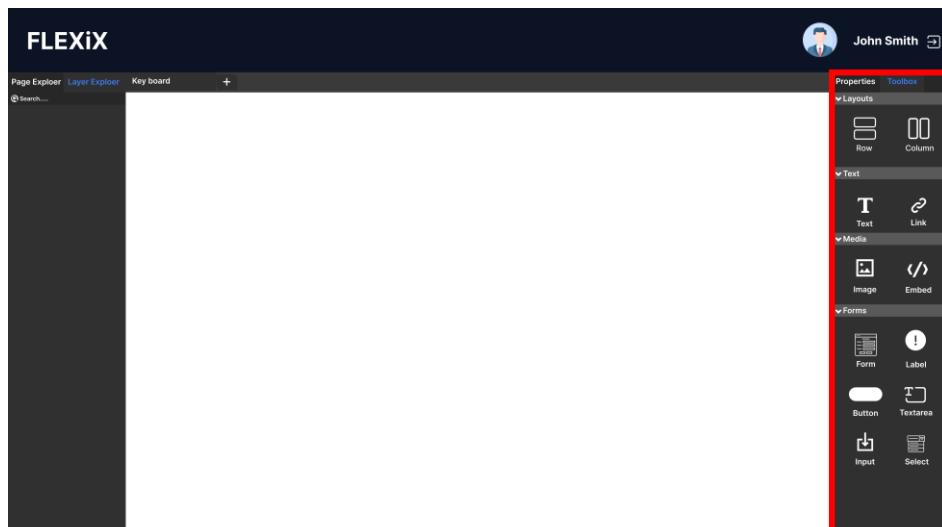
หน้าป๊อปอัพสำหรับสร้าง Page ผู้ใช้สามารถระบุขนาดหน้า Page และตั้งชื่อ Page



รูป 3.21 หน้าสร้างหน้าออกแบบ (Create new page)

13) การออกแบบ UI Web Application ส่วนของ Project Space

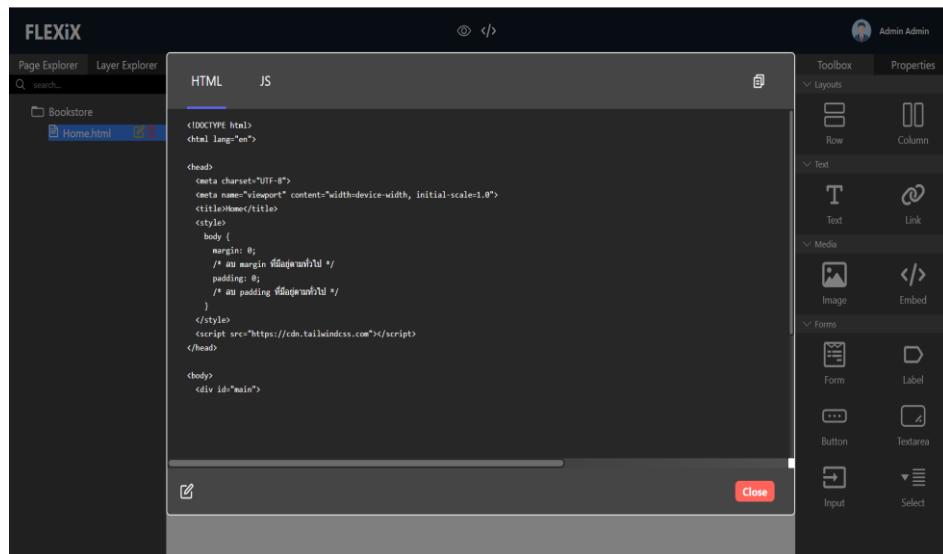
หน้าสำหรับให้ผู้ใช้สามารถดูและแก้ไขโครงสร้างเว็บไซต์ Front-end



รูป 3.22 หน้าการทำงาน (Working)

14) การออกแบบ UI Web Application ส่วนของ Pop-up (Generate Code)

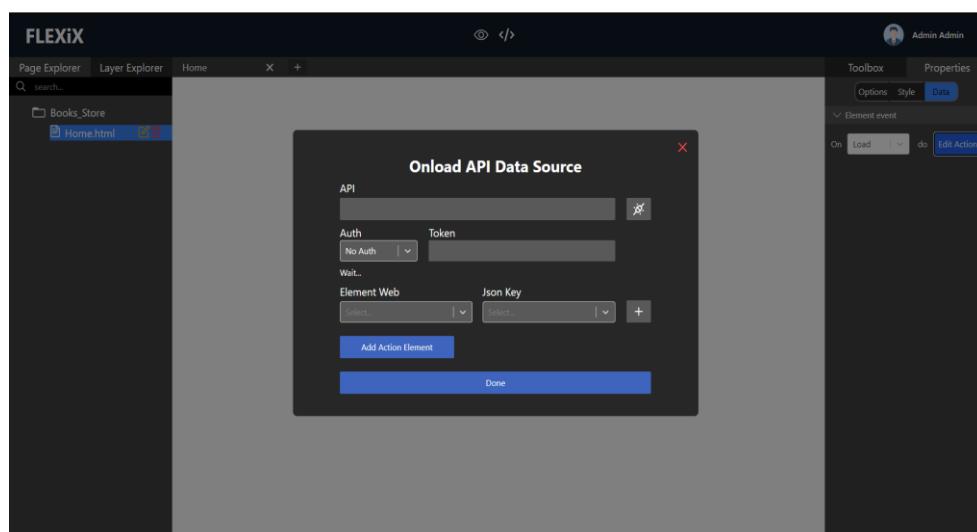
หน้าสำหรับแสดง Code ที่ Generate ออกมาราบบกการออกแบบของผู้ใช้



រូប 3.23 หน้าໂគែកដែលបង្ហាញកិច្ចការណ៍ (Code)

15) การออกแบบ UI Web Application ส่วนของ API(Create API)

หน้าการสร้าง API Connector สำหรับการนำข้อมูลใน API มาแสดงในទូទៅ Preview



រូប 3.24 หน้าការសរុប API (Create API)

บทที่ 4

การทดสอบ และผลการทดสอบ

จากการพัฒนา Front-end Builder ผู้ใช้ทำได้แบ่งการทดสอบการใช้งานออกแบบ 3 ส่วน ได้แก่

- 1) การทดสอบการใช้งานของเว็บแอปพลิเคชัน
- 2) การทดสอบใช้งานการสร้าง Elements
- 3) การทดสอบใช้งานสีタイล์ CSS จัดการรูปแบบ Elements
- 4) การทดสอบใช้งาน API ร่วมกับ Elements
- 5) การทดสอบใช้งาน Manual JavaScript
- 6) การทดสอบใช้งาน Preview Page
- 7) การทดสอบใช้งาน Generate Code

4.1 ทดสอบการใช้งานเว็บแอปพลิเคชัน

4.1.1 วัตถุประสงค์

เพื่อทดสอบการใช้งานเว็บแอปพลิเคชันให้สามารถใช้งานเว็บแอปพลิเคชันได้โดยใช้งานง่าย ปลอดภัย น่าเชื่อถือ เสถียร และมีประสิทธิภาพ อีกทั้งทดสอบการทำงานร่วมกันระหว่างส่วนติดต่อผู้ใช้ (Front-end) และส่วนระบบจัดการเว็บไซต์ (Back-end)

4.1.2 วิธีการทดสอบ

- 1) ทดสอบตาม Logic ของระบบ ความถูกต้องของปุ่ม ลิงก์ และ Flow ของระบบว่าถูกต้องตาม Wireframe ที่ได้อธิบายไว้ในหัวข้อ ? หรือไม่
- 2) ทดสอบการทำงานของแต่ละส่วน หรือฟังก์ชัน (Function) ว่าถูกต้องตาม Use Case ของระบบที่ได้อธิบายไว้ในหัวข้อ ? หรือไม่
- 3) ทดสอบการทำงานร่วมกันระหว่างส่วนติดต่อผู้ใช้ (Front-end) และส่วนระบบจัดการเว็บไซต์ (Back-end) ว่าสามารถทำงานร่วมกันได้อย่างถูกต้องหรือไม่
- 4) บันทึกผลการทดสอบ

ตาราง 4.1 ผลการทดสอบ Use Case ของการ Sign up

Test Case ID	TC-01
Use Case Name	Register
Use Case ID	UC-01
Actor	ผู้ใช้งาน
Preconditions	เข้าสู่ Web Application
Flow of Events	<ol style="list-style-type: none"> 1. เข้าสู่หน้า Sign up Page 2. กรอกข้อมูลตามที่ระบบต้องการ 3. กดปุ่มลงทะเบียน
ผลการทดสอบที่คาดหวัง	ลงทะเบียนสำเร็จ
ผลการทดสอบจริง	ลงทะเบียนสำเร็จ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน

Valid Register Log		Generated 2024/02/22 22:14:52 UTC-07:00 2 minutes 39 seconds ago
Test Statistics		
All Tests	Total: 1 Pass: 1 Fail: 0 Skip: 0 Elapsed: 00:00:34	Pass / Fail / Skip
Statistics by Tag	Total: Pass: Fail: Skip: Elapsed: Pass / Fail / Skip	
No Tags		
Statistics by Suite	Total: Pass: Fail: Skip: Elapsed: Pass / Fail / Skip	
Valid Register	1 1 0 0 00:00:35	Pass / Fail / Skip
Test Execution Log		
-	Valid Register	00:00:34.639
Full Name:	Valid Register	
Source:	E:\GitHub\Re-Factor-FastAPI\tests\functional\valid_register.robot	
Start / End / Elapsed:	2024/02/22 22:14:17.526 / 2024/02/22 22:14:52.165 / 00:00:34.639	
Status:	PASS	
-	Valid Register	00:00:34.401
Full Name:	Valid Register Valid Register	
Start / End / Elapsed:	2024/02/22 22:14:17.763 / 2024/02/22 22:14:52.164 / 00:00:34.401	
Status:	PASS	
+ [KEYWORD]	mouse Open Login Page	00:00:05.471
+ [KEYWORD]	mouse Submit To Register Page	00:00:01.367
+ [KEYWORD]	mouse Input First Name Ros	00:00:01.087
+ [KEYWORD]	mouse Input Last Name Ter	00:00:01.070
+ [KEYWORD]	mouse Input Email un_admin1@gmail.com	00:00:01.091
+ [KEYWORD]	mouse Input Password Unadmin12345	00:00:01.077
+ [KEYWORD]	mouse Input Confirm Password Unadmin12345	00:00:01.077
+ [KEYWORD]	mouse Submit Register	00:00:01.047
+ [KEYWORD]	mouse Store Alert	00:00:01.096
+ [KEYWORD]	mouse Login Page Should Be Open	00:00:01.094
+ [TEARDOWN]	Imantroy Close Browser	00:00:03.093

รูป 4.1 ผลลัพธ์ของ TC01 Use case Sign up

Sign Up

First Name Last Name

Email

Password

Confirm Password

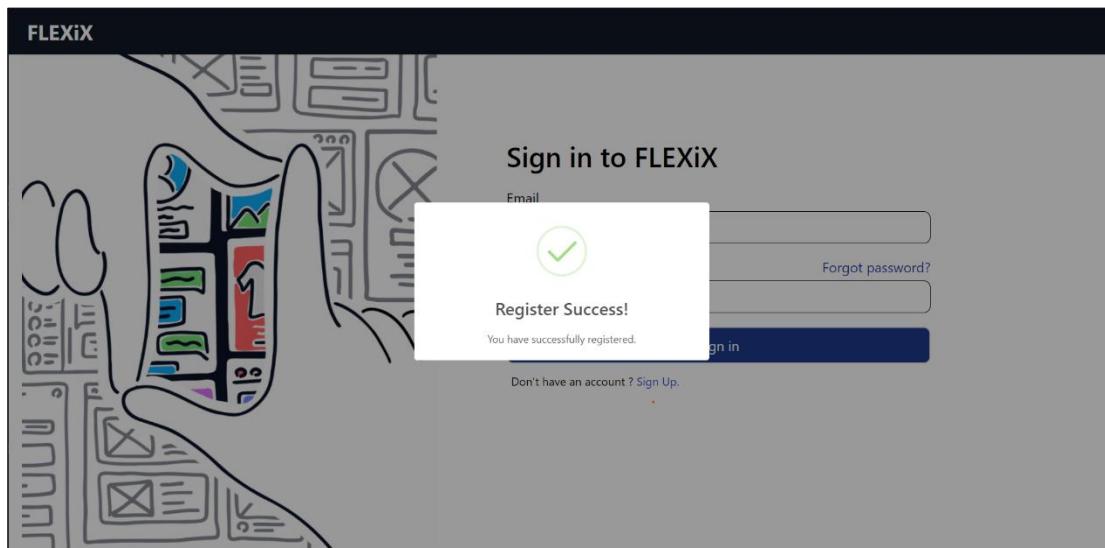
What you'll enjoy:

- Free access to the FLEXiX platform
- Access to templates, widgets, test API
- Self-paced courses and certifications

[Next](#)

Already have an account? [Sign In.](#)

รูป 4.2 กรอกข้อมูล First Name, Last Name, Email, Password, Confirm Password



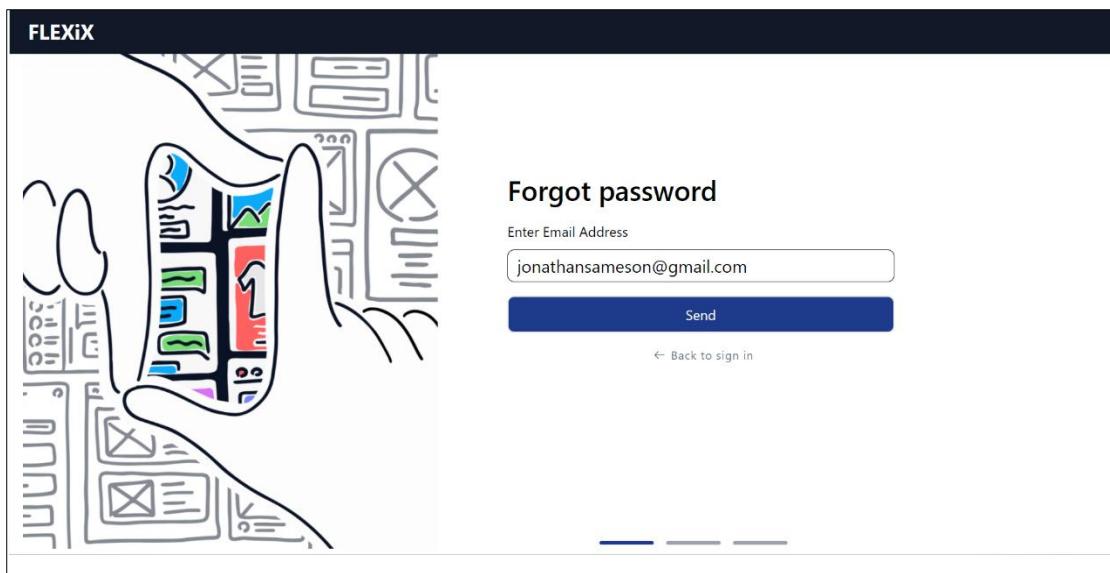
รูป 4.3 สมัครสมาชิกแล้วเสร็จสมบูรณ์

ตาราง 4.2 ผลการทดสอบ Use Case ของการ Forgot Password

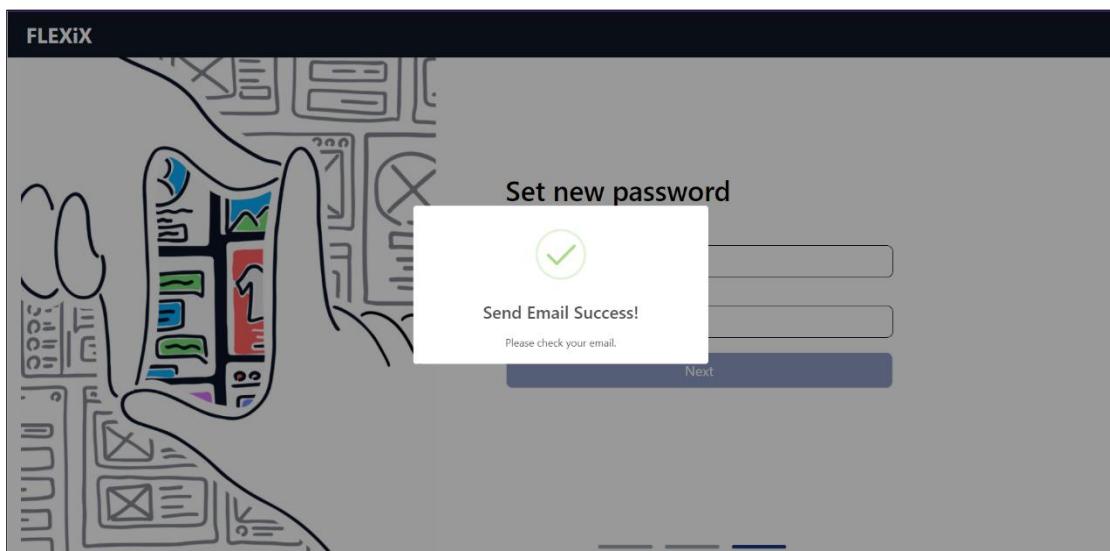
Test Case ID	TC-02
Use Case Name	Forgot Password
Use Case ID	UC-02
Actor	ผู้ใช้งาน
Preconditions	ต้องเป็นสมาชิกของระบบ
Flow of Events	<ol style="list-style-type: none"> 1. เข้าสู่หน้า Forgot Password 2. กรอก Email 3. กดปุ่ม Send E-mail 4. รับรหัสยืนยันทาง E-mail 5. นำรหัสยืนยันมากรอก 6. กดปุ่ม Verify 7. กรอกรหัสผ่านใหม่ 8. กดปุ่ม Reset Password
ผลการทดสอบที่คาดหวัง	เปลี่ยนรหัสผ่านสำเร็จ
ผลการทดสอบจริง	เปลี่ยนรหัสผ่านสำเร็จ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน

Valid Forgot Log		Generated 2024/02/22 22:38:15 UTC-07:00 9 seconds ago
Test Statistics		
All Tests	Total: 1 Pass: 1 Fail: 0 Skip: 0 Elapsed: 00:00:46	Pass / Fail / Skip
No Tags	Total: 1 Pass: 1 Fail: 0 Skip: 0 Elapsed: 00:00:46	Pass / Fail / Skip
Valid Forget	Total: 1 Pass: 1 Fail: 0 Skip: 0 Elapsed: 00:00:46	Pass / Fail / Skip
Test Execution Log		
- [KEYWORD] Valid Forget	Full Name: Valid Forget	00:00:46.240
Source: E:\Github\illeFlexx-Frontend\test_Flexx\valid_forget.robot		
Start / End / Elapsed: 2024/02/22 22:37:29.449 / 2024/02/22 22:38:15.689 / 00:00:46.240		
Status: 1 test total, 1 passed, 0 failed, 0 skipped		
- [TEST] Valid Forgot Password		00:00:46.009
Full Name: Valid Forget.Valid Forgot Password		
Source: E:\Github\illeFlexx-Frontend\test_Flexx\valid_forget.robot		
Start / End / Elapsed: 2024/02/22 22:37:29.680 / 2024/02/22 22:38:15.689 / 00:00:46.009		
Status: 1 test total, 1 passed, 0 failed, 0 skipped		
+ [KEYWORD] mouse Open LoginPage		00:00:05.370
+ [KEYWORD] mouse Click To forgot Page		00:00:03.658
+ [KEYWORD] mouse Input Email "u_admin1@gmail.com"		00:00:03.114
+ [KEYWORD] mouse Submit Send OTP		00:00:03.045
+ [KEYWORD] mouse Store Alert		00:00:06.016
+ [KEYWORD] mouse Input OTP "567890"		00:00:03.059
+ [KEYWORD] mouse Click Continue OTP		00:00:03.641
+ [KEYWORD] mouse Store Alert		00:00:03.006
+ [KEYWORD] mouse Input New Password "Admin12345"		00:00:03.074
+ [KEYWORD] mouse Input Confirm New Password "Admin12345"		00:00:03.071
+ [KEYWORD] mouse Click Next Confirm Password		00:00:03.640
+ [KEYWORD] mouse Store Alert		00:00:03.005
+ [KEYWORD] mouse Login Page Should Be Open		00:00:01.904
+ [TEARDOWN] tearDown Close Browser		00:00:03.104

รูป 4.4 ผลลัพธ์ของ TC02 Use case Forgot



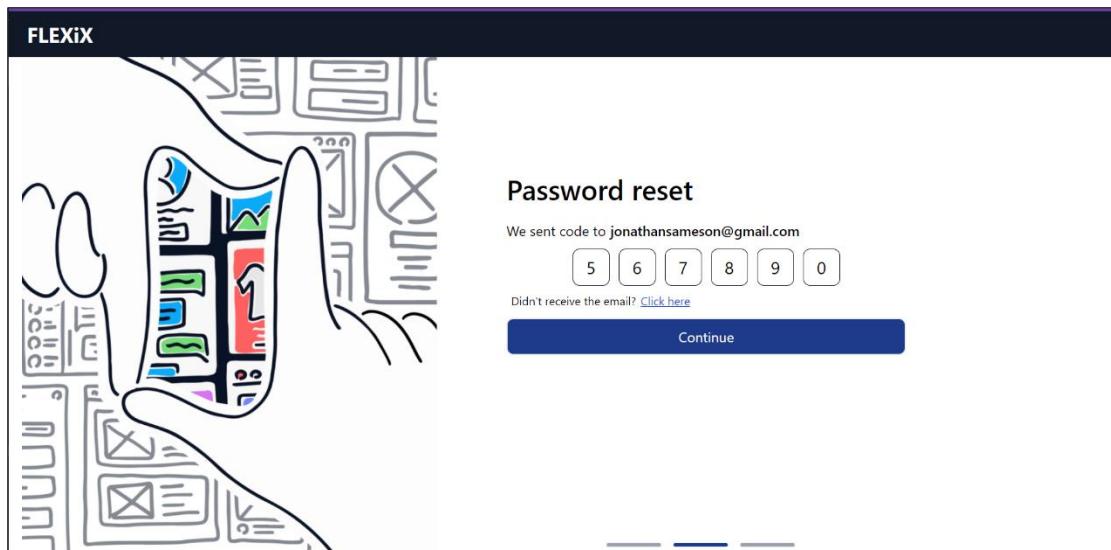
ຮູບ 4.5 ທຳການກອກຂໍ້ມູນ Email



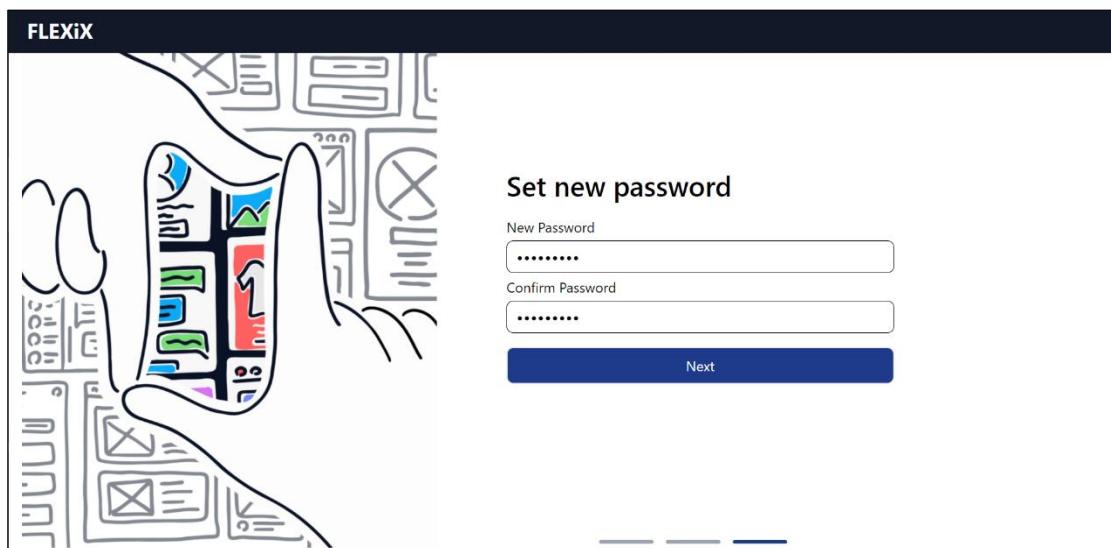
ຮູບ 4.6 ສ່າງຮັສ OTP ໄປທີ່ Email ຜູ້ໃຊ້



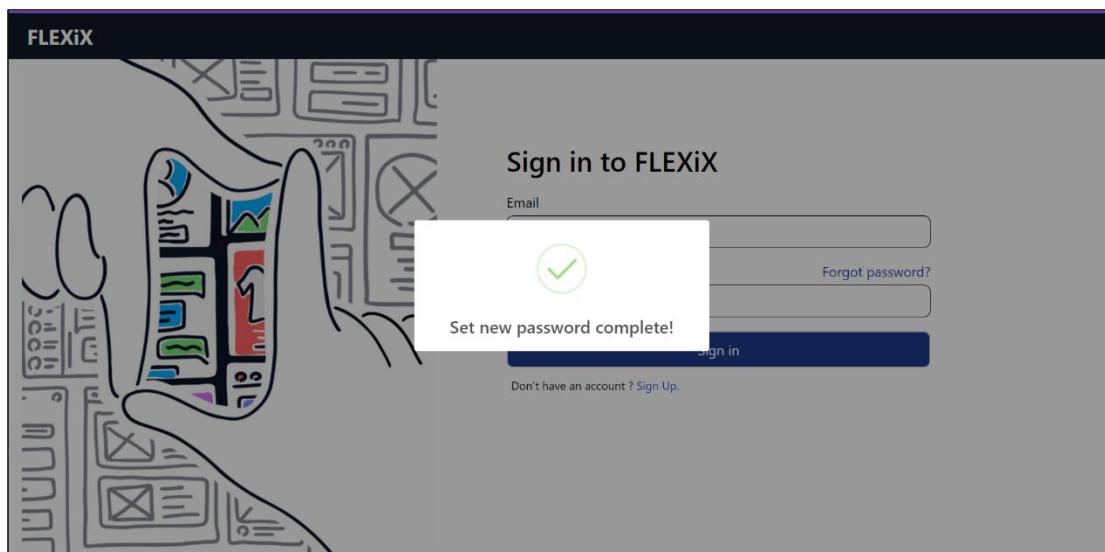
ຮູບ 4.7 ຮັສ OTP



รูป 4.8 ทำการกรอกรหัส OTP



รูป 4.9 ทำการกรอกข้อมูล New Password, Confirm Password



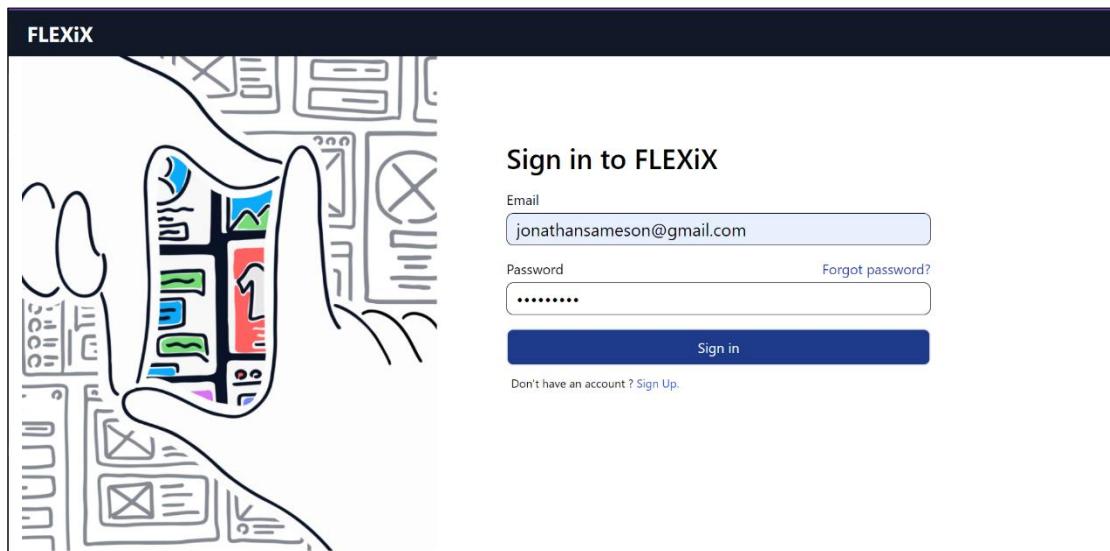
รูป 4.10 เป้าหมายรหัสผ่านเสร็จสิ้น

ตาราง 4.3 ผลการทดสอบ Use Case ของการ Sign in

Test Case ID	TC-03
Use Case Name	Sign in
Use Case ID	UC-03
Actor	ผู้ใช้งาน
Preconditions	ต้องเป็นสมาชิกของระบบ
Flow of Events	<ol style="list-style-type: none"> 1. เข้าสู่หน้า Sign in 2. กรอก Username และ Password 3. กดปุ่ม Login
ผลการทดสอบที่คาดหวัง	เข้าสู่ระบบสำเร็จ
ผลการทดสอบจริง	เข้าสู่ระบบสำเร็จ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน

Valid Login Log						
Test Statistics						
Total Statistics	0 Total	0 Pass	0 Fail	0 Skip	Elapsed 00:00:23	Pass / Fail / Skip
All Tests	1 Total	1 Pass	0 Fail	0 Skip	Elapsed 00:00:23	Pass / Fail / Skip
No Tags	Statistics by Tag	0 Total	0 Pass	0 Fail	0 Skip	Elapsed 00:00:23
Valid Login	Statistics by Suite	0 Total	0 Pass	0 Fail	0 Skip	Elapsed 00:00:23
Valid Login	Valid Login	1 Total	1 Pass	0 Fail	0 Skip	Elapsed 00:00:23
Test Execution Log						
- [INFO] Valid Login	Full Name:	Valid Login				00:00:23.004
	Source:	E:\Github\file\FlexiX-FrontendTest_FlexiX\valid_login.robot				
	Start / End / Elapsed:	20240222 21:47:07.336 / 20240222 21:47:30.349 / 00:00:23.004				
	Status:	1 test total, 1 passed, 0 failed, 0 skipped				
- [TEST] Valid Login	Full Name:	Valid Login				00:00:22.773
	Start / End / Elapsed:	20240222 21:47:07.566 / 20240222 21:47:30.339 / 00:00:22.773				
	Status:	[PASSED]				
• [KEYWORD] mouse Open Login Page						00:00:05.423
• [KEYWORD] www Input Email	admin@admin.com					00:00:03.119
• [KEYWORD] www Input The Password	Admin1234					00:00:03.057
• [KEYWORD] www Submit Credentials						00:00:03.549
• [KEYWORD] www Store Alert						00:00:03.005
• [KEYWORD] www Workspace Page Should Be Open						00:00:02.008
• [TEARDOWN] selenium Close Browser						00:00:03.103

ຮູບ 4.11 ຜົດລັບພື້ນຂອງ TC03 Use case Sign in



ຮູບ 4.12 ທຳການກຽມອຸບໜຸດ Email, Password



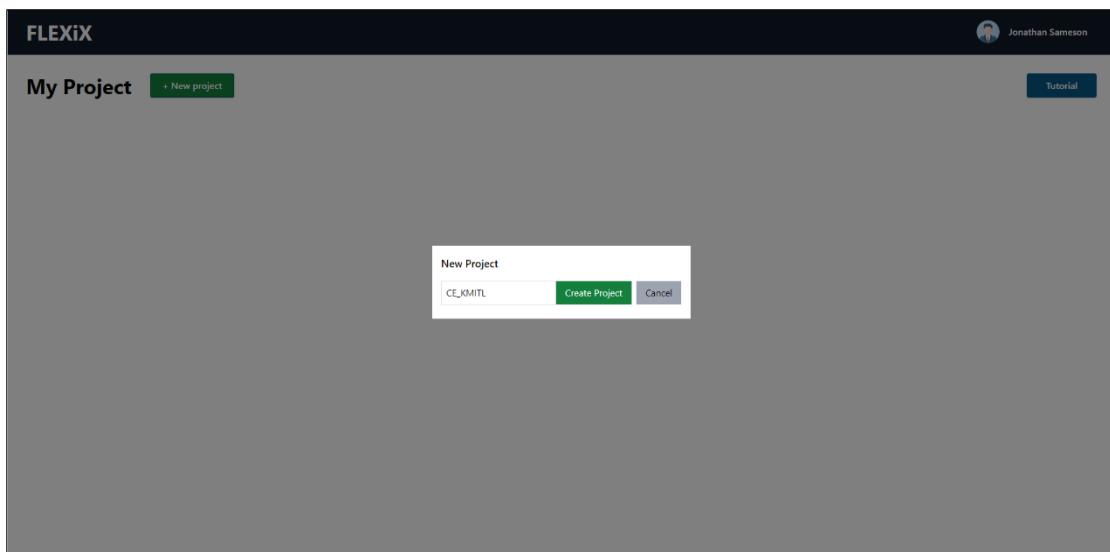
ຮູບ 4.13 Login ສໍາເລັບເຂົ້າສູ່ໜ້າ Project

ตาราง 4.4 ผลการทดสอบ Use Case ของ Project

Test Case ID	TC-04
Use Case Name	Project
Use Case ID	UC-04
Actor	ผู้ใช้งาน
Preconditions	ต้องเข้าสู่ระบบ
Flow of Events	<ol style="list-style-type: none"> 1. เข้าสู่หน้า Project 2. กดปุ่ม New Project 3. ตั้งชื่อ Project 4. กดปุ่ม Create 5. ดับเบิล Project ที่จะเข้าไปทำงาน
ผลการทดสอบที่คาดหวัง	สร้างโปรเจกต์สำเร็จ
ผลการทดสอบจริง	สร้างโปรเจกต์สำเร็จ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน

Valid Create Project Log	
Test Statistics	Generated: 2024/02/22 22:18:50 UTC-07:00 3 minutes ago (estimated)
All Tests	Total: 1 Pass: 1 Fail: 0 Skip: 0 Elapsed: 00:00:35 Pass / Fail / Skip
Statistics by Tag	Total: 1 Pass: 1 Fail: 0 Skip: 0 Elapsed: 00:00:35 Pass / Fail / Skip
No Tags	Total: 1 Pass: 1 Fail: 0 Skip: 0 Elapsed: 00:00:35 Pass / Fail / Skip
Statistics by Suite	Total: 1 Pass: 1 Fail: 0 Skip: 0 Elapsed: 00:00:35 Pass / Fail / Skip
Valid Create Project	1 test total, 1 passed, 0 failed, 0 skipped
Test Execution Log	
- [INFO] Valid Create Project	00:00:35.124
Full Name: Valid Create Project	
Status: PASSED	
Start / End / Elapsed: 2024/02/22 22:18:15.975 / 2024/02/22 22:18:50.854 / 00:00:35.124	
Keywords:	
+ [INFO] Valid Create Project	00:00:34.877
Full Name: Valid Create Project Valid Create Project	
Status: PASSED	
Start / End / Elapsed: 2024/02/22 22:18:15.975 / 2024/02/22 22:18:50.852 / 00:00:34.877	
Keywords:	
+ [KEYWORD] mouse Open Login Page	00:00:05.369
+ [KEYWORD] mouse Input Email admin@admin.com	00:00:03.106
+ [KEYWORD] mouse Input The Password Admin1234	00:00:03.045
+ [KEYWORD] mouse Submit Credentials	00:00:03.006
+ [KEYWORD] mouse Store Alert	00:00:02.038
+ [KEYWORD] mouse Click New Project	00:00:02.007
+ [KEYWORD] mouse Input Name Project Robotframework	00:00:02.006
+ [KEYWORD] mouse Click Create Project	00:00:02.006
+ [KEYWORD] mouse Store Alert	00:00:02.006
+ [KEYWORD] mouse Workspace Page Should Be Open	00:00:02.006
+ [TEARDOWN] selenium Close Browser	00:00:03.099

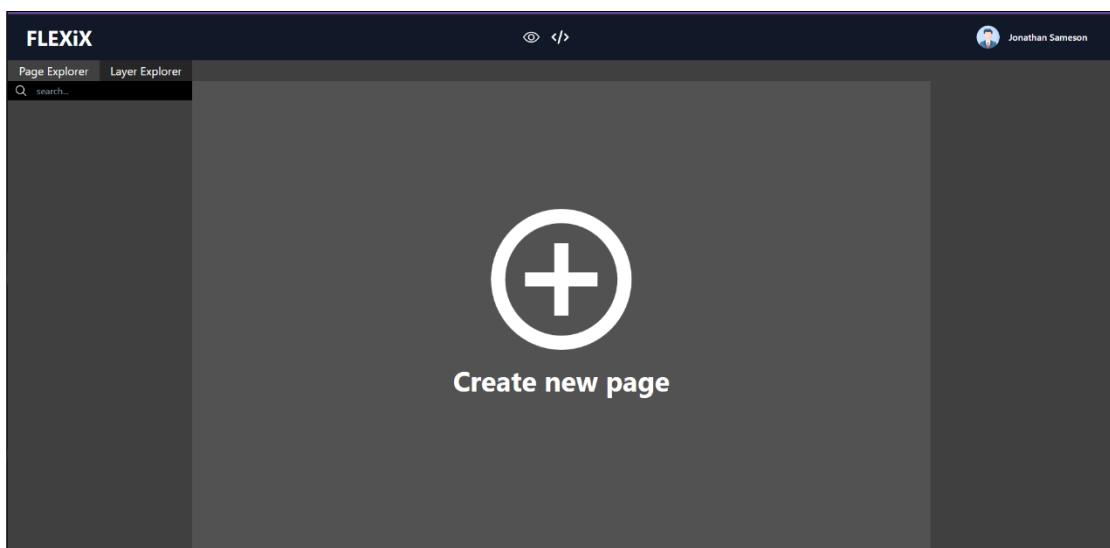
รูป 4.14 ผลลัพธ์ของ TC04 Use case Project



รูป 4.15 กรอกชื่อ Project



รูป 4.16 สร้าง Project สำเร็จ



รูป 4.17 หน้า Workspace

4.2 การทดลองใช้งานการสร้าง Elements

4.2.1 วัตถุประสงค์

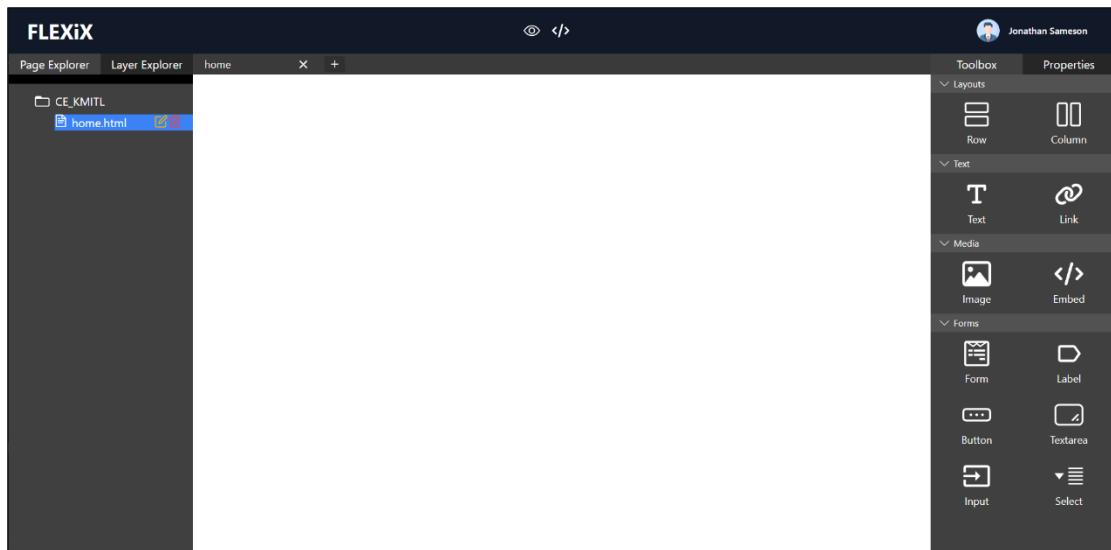
เพื่อทดสอบการใช้งานฟังก์ชันการสร้าง Elements ภายใน Workspace เพื่อให้ผู้ใช้งานสามารถใช้งานระบบได้

ตาราง 4.5 ผลการทดลอง Use Case ของการ Element

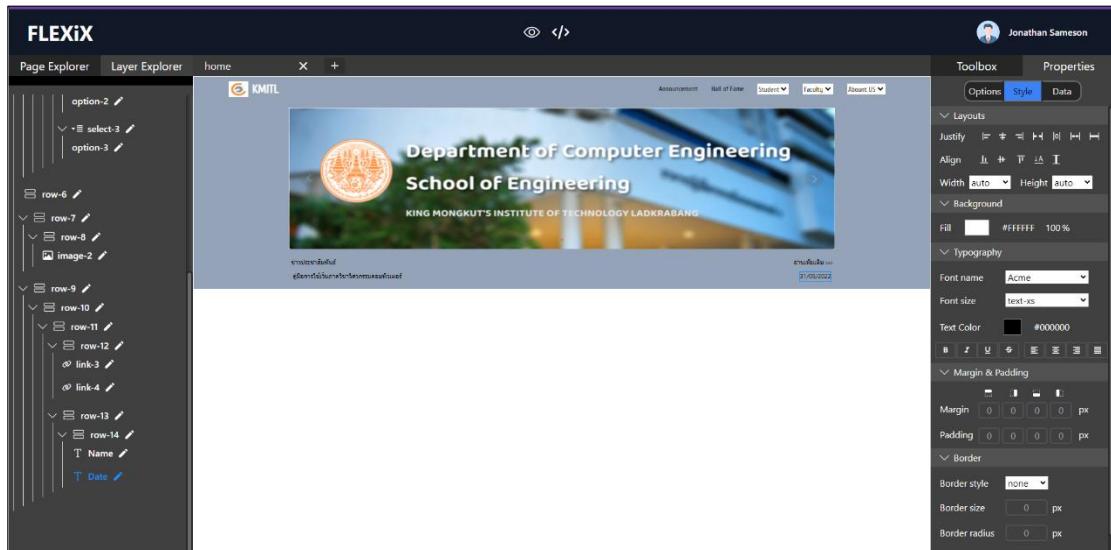
Test Case ID	TC-05
Use Case Name	Create and Use Element
Use Case ID	UC-05
Actor	ผู้ใช้งาน
Preconditions	ต้องเข้าสู่ระบบ ต้องสร้าง Project
Flow of Events	1. สร้าง Project เพื่อเริ่มใช้งาน 2. เลือกเมนู Toolbox 3. เลือก Elements ต่างๆ เพื่อทดสอบการใช้งาน 4. ทดลองใช้งาน Elements โดยให้ผู้ใช้สร้าง Element และเลือก Elements ต่างๆ เพื่อแทรกร Element ใหม่ ให้อยู่ภายใต้ Element ที่สร้าง
ผลการทดสอบที่คาดหวัง	สร้าง Element สำเร็จ
ผลการทดสอบจริง	สร้าง Element สำเร็จ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน

Valid Add Element Log		Generated: 2024/02/23 00:51:44 UTC+07:00 1 minute 8 seconds ago
Test Statistics		
Total Statistics	0 Total 0 Pass 0 Fail 0 Skip 0 Elapsed 0:00:27 Pass / Fail / Skip	0:00:27
All Tests	1 1 0 0 0 0:00:27	Pass
Statistics by Tag	0 Total 0 Pass 0 Fail 0 Skip 0 Elapsed 0:00:27 Pass / Fail / Skip	
No Tags	0 0 0 0 0 0:00:27	
Statistics by Suite	0 Total 0 Pass 0 Fail 0 Skip 0 Elapsed 0:00:28 Pass / Fail / Skip	
Valid Add Element	1 1 0 0 0 0:00:28	Pass
Test Execution Log		
- NOTE: Valid Add Element	Valid Add Element	00:00:27.670
Full Name:	Valid Add Element	
Source:	E:\Software\Python\Frontend-Test\FlaskProject\add_element_robott	
Start / End / Elapsed:	2024/02/23 00:51:16.777 / 2024/02/23 00:51:44.397 / 00:00:27.670	
Status:	1 test total: 1 passed, 0 failed, 0 skipped	
- PASS: Valid Add Element	Valid Add Element	00:00:27.433
Full Name:	Valid Add Element	
Source:	E:\Software\Python\Frontend-Test\FlaskProject\add_element_robott	
Start / End / Elapsed:	2024/02/23 00:51:16.952 / 2024/02/23 00:51:44.395 / 00:00:27.433	
Status:	PASS	
- KEYWORD: mouse Open Login Page		00:00:05.428
• KEYWORD: mouse Input Email admin@admin.com		00:00:03.118
• KEYWORD: mouse Input The Password Admin1234		00:00:03.072
• KEYWORD: mouse Submit Credentials		00:00:03.061
• KEYWORD: mouse Store Alert		00:00:03.005
• KEYWORD: mouse Click Project		00:00:02.305
• KEYWORD: mouse Open Workspace Page		00:00:02.297
• KEYWORD: mouse Click Add Element		00:00:02.045
• KEYWORD: mouse Close Browser		00:00:03.112

รูป 4.18 ผลลัพธ์ของ TC05 Use case Use Element



รูป 4.19 หน้าการทำงาน Workspace



รูป 4.20 การจัดวาง Element

4.3 การทดลองใช้งานสีトイล์ CSS จัดการรูปแบบ Elements

4.3.1 วัตถุประสงค์

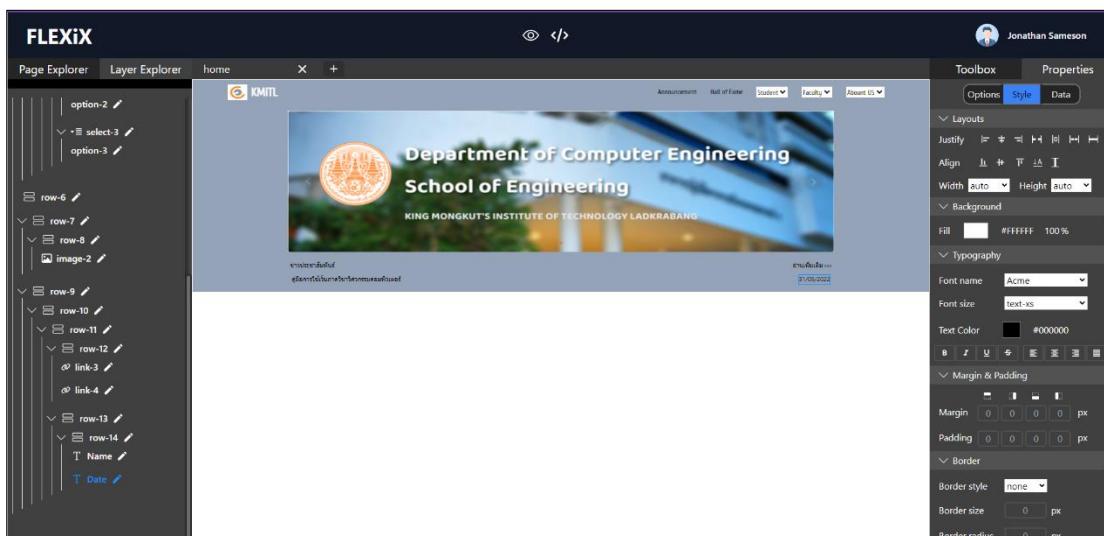
เพื่อทดสอบการใช้งานฟังก์ชัน CSS จัดการรูปแบบ Elements เพื่อให้ผู้ใช้งานสามารถใช้งานการปรับแต่งรูปแบบได้

ตาราง 4.6 ผลการทดลอง Use Case ของการ Format CSS

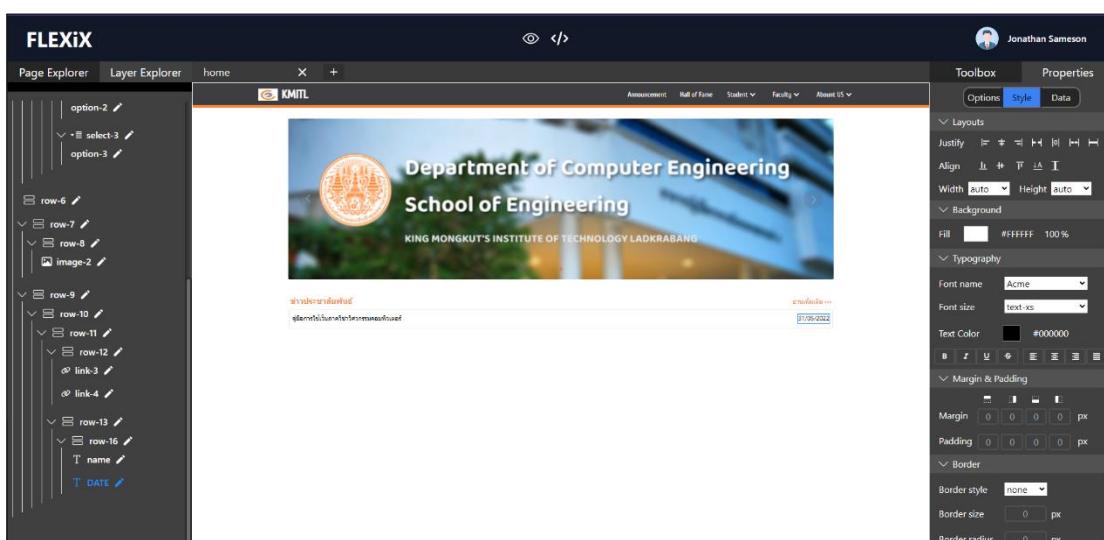
Test Case ID	TC-06
Use Case Name	Format CSS
Use Case ID	UC-06
Actor	ผู้ใช้งาน
Preconditions	ต้องเข้าสู่ระบบ ต้องสร้าง Project ต้องสร้าง Element
Flow of Events	1. เลือก Element ที่ต้องการจัดการ CSS 2. เลือกเมนู Properties 3. เลือกเมนู Style 4. เลือก Style ที่ต้องการ เพื่อทดสอบการใช้งาน 5. ทดลองใช้งาน Style โดยให้ผู้ใช้เลือก Style ที่ต้องการใช้งาน เช่น Fill เพื่อเปลี่ยนสี Background
ผลการทดสอบที่คาดหวัง	ปรับแต่ง Element สำเร็จ
ผลการทดสอบจริง	ปรับแต่ง Element สำเร็จ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน

Valid Style Css Log						
Test Statistics						
All Tests	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
1	1	0	0	0	00:00:39	Pass
No Tags	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
1	1	0	0	0	00:00:39	Pass
Statistics by Suite	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
Valid Style Css	1	1	0	0	00:00:40	Pass
Test Execution Log						
- Suite Valid Style Css	Full Name:	Valid Style Css	Generated:			00:00:39.563
	Source:	E:\github\file\FlexiFrontend\Test_Flexio\Valid_style_css.robot				
	Start / End / Elapsed:	2024/02/23 02:29:42 184 / 2024/02/23 02:30:21 747 / 00:00:39.563				
	Status:	1 test total, 1 passed, 0 failed, 0 skipped				
- Test Valid Add Element	Valid Style Css Valid Add Element					00:00:39.294
	Full Name:	Valid Style Css				
	Source:	E:\github\file\FlexiFrontend\Test_Flexio\Valid_style_css.robot				
	Start / End / Elapsed:	2024/02/23 02:29:42 452 / 2024/02/23 02:30:21 746 / 00:00:39.294				
	Status:	Pass				
- Step Open Login Page	Valid Style Css	Open Login Page				00:00:05.446
- Step Input Email	Valid Style Css	admin@admin.com				00:00:03.121
- Step Input The Password	Valid Style Css	Admin1234				00:00:03.071
- Step Submit Credentials	Valid Style Css					00:00:03.052
- Step Store Alert	Valid Style Css					00:00:03.037
- Step Click Project	Valid Style Css					00:00:02.309
- Step Open Workspace Page	Valid Style Css					00:00:02.213
- Step Click Open Layers	Valid Style Css					00:00:02.146
- Step Click Element Row	Valid Style Css					00:00:02.206
- Step Click Properties	Valid Style Css					00:00:02.082
- Step Click Click Color Element	Valid Style Css					00:00:02.290
- Step Add Color To Element	Valid Style Css	#000000				00:00:03.105
- Step Click Toolbox	Valid Style Css					00:00:02.082
- Step Save And Close	Valid Style Css	Save And Close Browser				00:00:03.103

รูป 4.21 ผลลัพธ์ของ TC06 Use case Format CSS



รูป 4.22 เลือก Element ที่ปรับแต่ง Style



รูป 4.23 ผลลัพธ์การปรับแต่ง CSS สำเร็จ

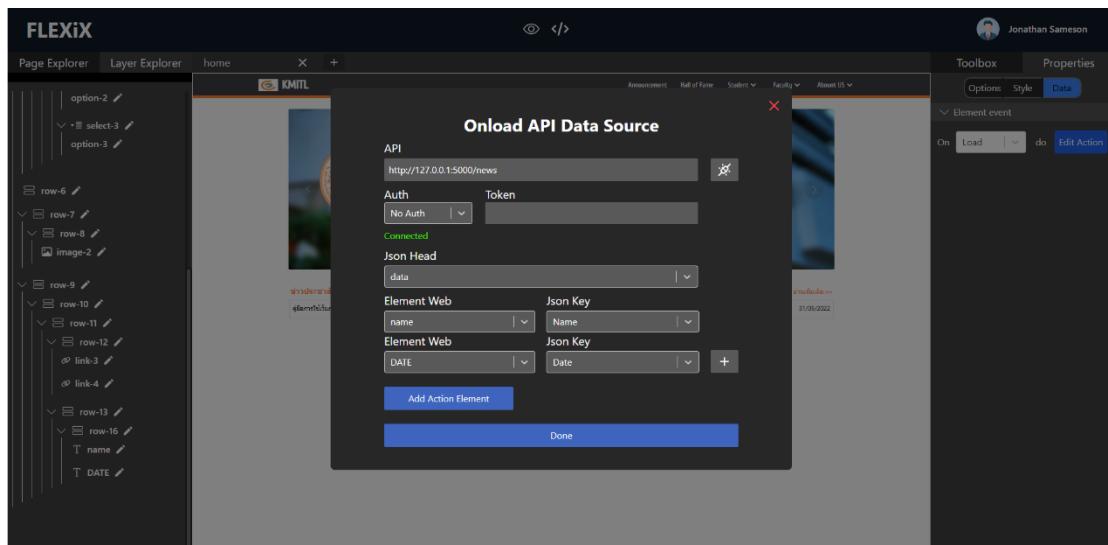
4.4 การทดสอบใช้งาน API ร่วมกับ Elements

4.4.1 วัตถุประสงค์

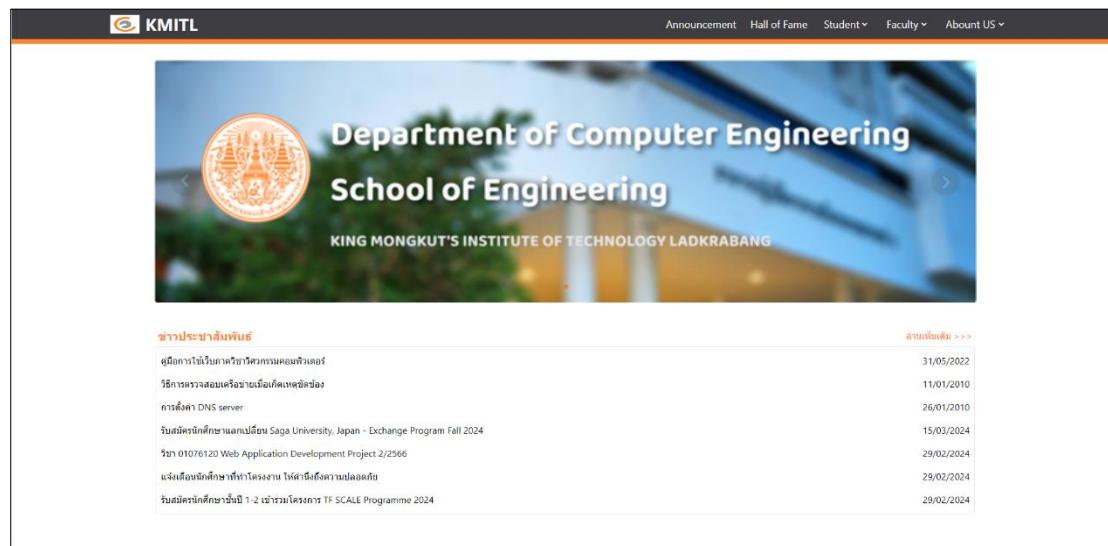
เพื่อทดสอบการใช้งาน API ร่วมกับ Elements เพื่อใช้แสดงผลข้อมูลบนหน้าเว็บจากข้อมูลที่ส่งมาจาก API ที่อยู่ในรูปแบบ Json เพื่อให้เว็บไซต์สามารถในการใช้งาน

ตาราง 4.7 ผลการทดสอบ Use Case ของการ API

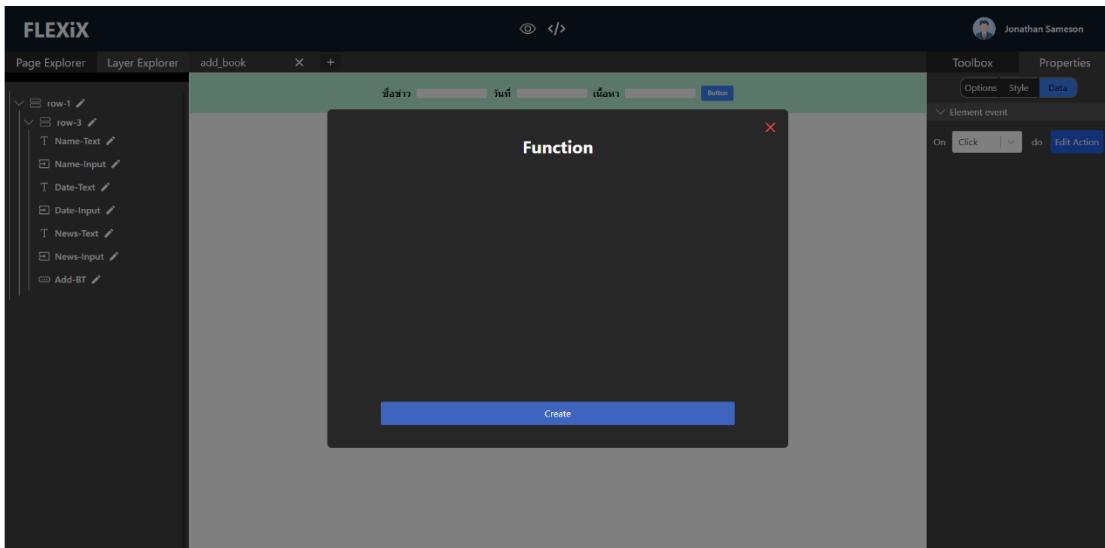
Test Case ID	TC-07
Use Case Name	API
Use Case ID	UC-07
Actor	ผู้ใช้งาน
Preconditions	ต้องเข้าสู่ระบบ ต้องสร้าง Project ต้องสร้าง Element
Flow of Events	1. เลือก Element Div (row, column) 2. เลือกเมนู Properties 3. เลือกเมนู Data 4. เลือก Event เป็น OnLoad เพื่อใช้งาน API เมื่อเริ่มโหลดหน้าเว็บ 5. เลือก Edit Action เพื่อตั้งค่าการทำงานของ API 6. ใส่ลิงก์ Api และ Element, Json(Key : Value) เพื่อ map ข้อมูลให้กับ Element 7. ทดลองใช้งาน API โดยให้ผู้ใช้ต้องนำ API ที่จัดการข้อมูลมาเลี้ยวจากหลังบ้าน มาแสดงผล
ผลการทดสอบที่คาดหวัง	ปรับแต่ง Element สำเร็จ
ผลการทดสอบจริง	ปรับแต่ง Element สำเร็จ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน



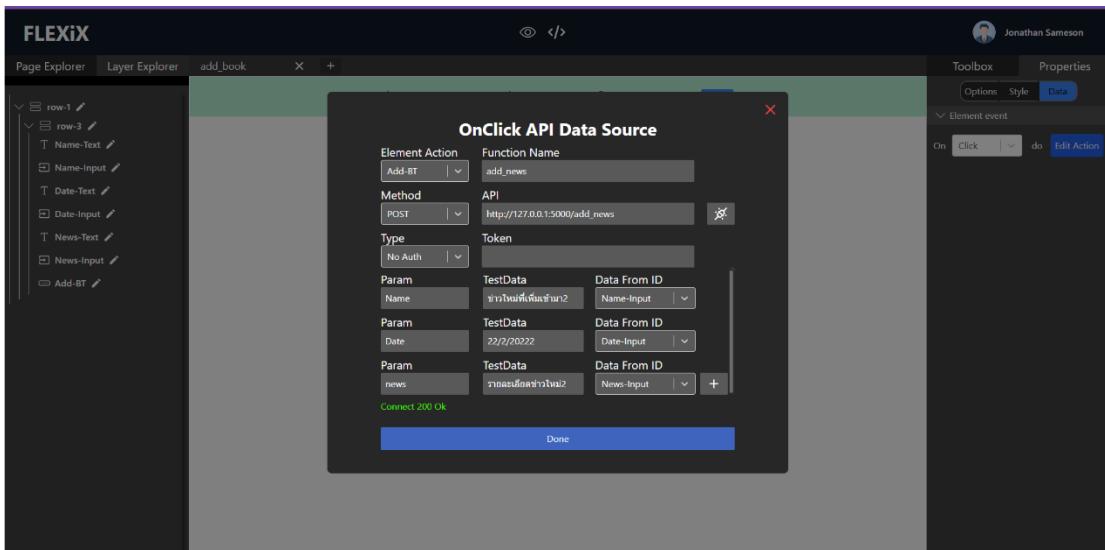
รูป 4.24 การเชื่อมต่อ API แบบ GET



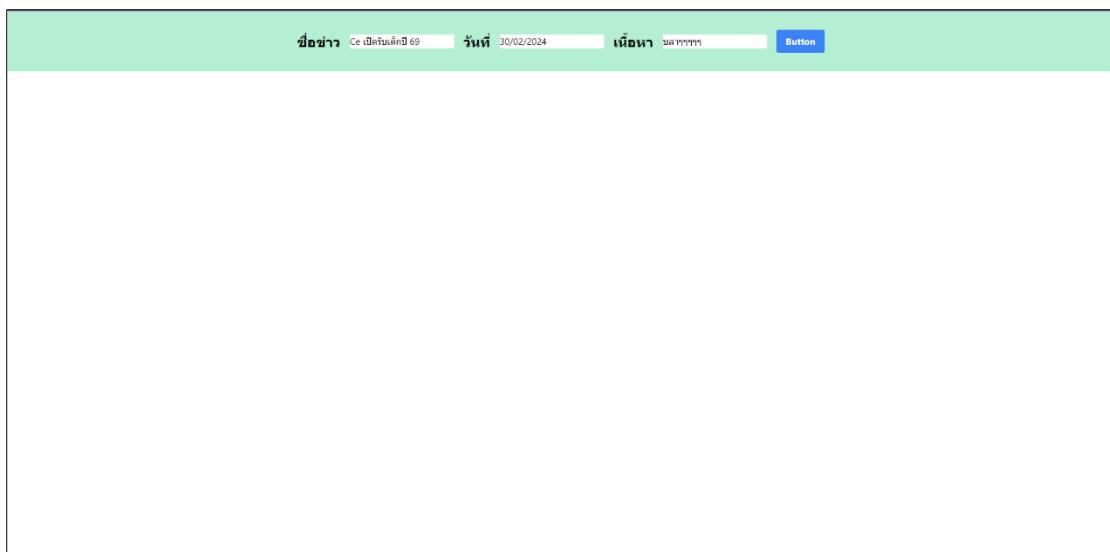
รูป 4.25 ผลลัพธ์การดึงข้อมูล API แบบ GET



รูป 4.26 การเชื่อมต่อ API แบบ POST



รูป 4.27 สร้าง Function สำหรับการยิง API



รูป 4.28 ตัวอย่างการใส่ข้อมูล

ข่าวประชาสัมพันธ์	อ่านเพิ่มเติม >>
เรียนเชิญนักศึกษาเข้าร่วมโครงการอบรมพัฒนาทักษะ IT ด้านอาชญากรรมทางไซเบอร์	31/05/2022
บริการตรวจสอบเครือข่ายเบื้องต้นของอุปกรณ์	11/01/2010
การติดตั้ง DNS server	26/01/2010
นักศึกษาเข้าร่วมแลกเปลี่ยน Saga University, Japan - Exchange Program Fall 2024	15/03/2024
งาน 01076120 Web Application Development Project 2/2566	29/02/2024
บังคับใช้กฎหมาย ให้กับบุคลากร ไม่ต้องมีเอกสารแนบท้าย	29/02/2024
นักศึกษาเข้าร่วมโครงการ TF SCALE Programme 2024	29/02/2024
Ce เมืองเดือนที่ 69	30/02/2024

รูป 4.29 ผลลัพธ์ของข้อมูลที่ผ่านการส่งข้อมูล API

4.5 การทดสอบใช้งาน Preview Page

4.5.1 วัตถุประสงค์

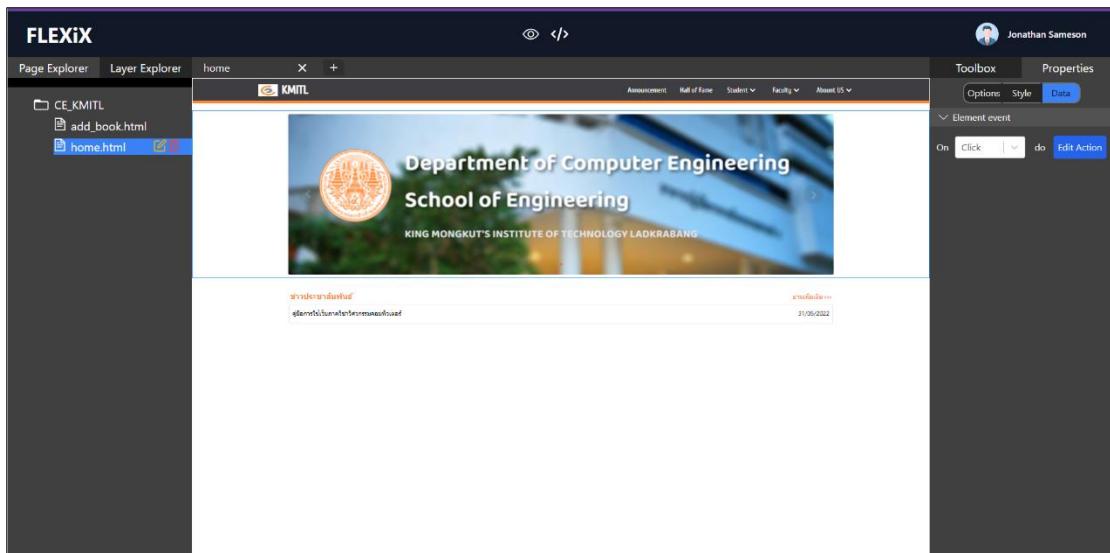
เพื่อทดสอบการใช้งานฟังก์ชันการ Preview Page เพื่อให้ผู้ใช้งานสามารถดูผลลัพธ์ของเว็บไซต์ได้

ตาราง 4.8 ผลการทดสอบ Use Case ของการ Preview

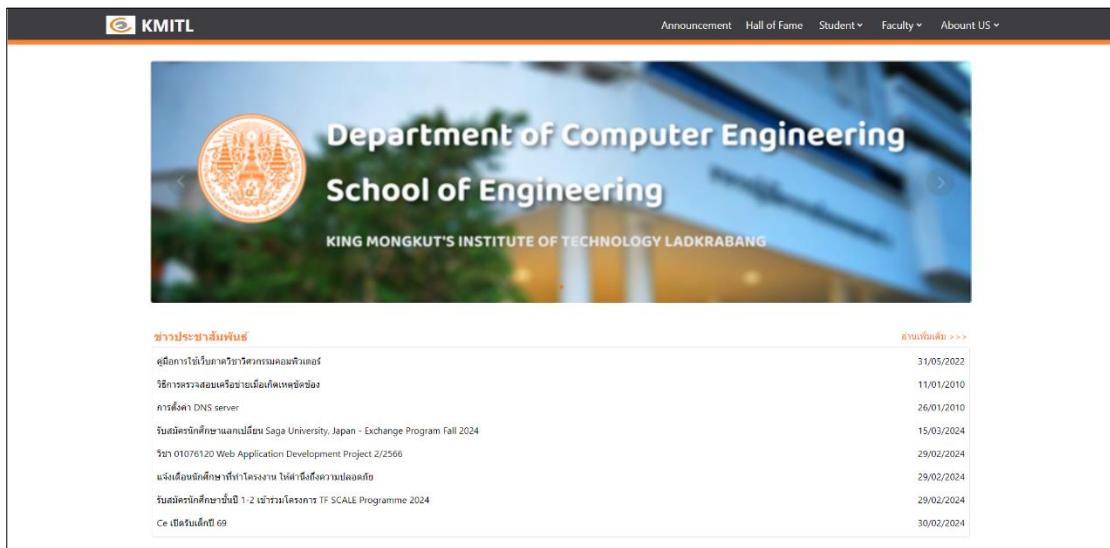
Test Case ID	TC-08
Use Case Name	Preview
Use Case ID	UC-08
Actor	ผู้ใช้งาน
Preconditions	ต้องเข้าสู่ระบบ ต้องสร้าง Project
Flow of Events	1. กดปุ่ม Preview 2. ทดสอบระบบ
ผลการทดสอบที่คาดหวัง	สามารถ Preview หน้าเว็บไซต์สำเร็จ
ผลการทดสอบจริง	สามารถ Preview หน้าเว็บไซต์สำเร็จ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน

Valid Preview Website Log		Generated: 2024/02/23 16:06:34 (UTC) 37 seconds ago
Test Statistics		
All Tests	Total: 1	Pass: 1 Fail: 0 Skip: 0 Elapsed: 00:00:27 Pass / Fail / Skip: Pass
No Tags	Total: 0	Pass: 0 Fail: 0 Skip: 0 Elapsed: 0 Pass / Fail / Skip: Pass / Fail / Skip
Statistics by Suite	Total: 1	Pass: 1 Fail: 0 Skip: 0 Elapsed: 00:00:28 Pass / Fail / Skip: Pass
Valid Preview Website	1	1 0 0 00:00:28 Pass / Fail / Skip: Pass
Test Execution Log		
- [INFO] Valid Preview Website	Full Name: Valid Preview Website	00:00:27.593
	Source: E:\GitHub\Ref-Flex-Frontend\Text_Flex\valid_preview_website\robot	
	Start / End / Elapsed: 2024/02/23 16:06:07.336 / 2024/02/23 16:06:34.929 / 00:00:27.593	
	Status: 1 test total, 1 passed, 0 failed, 0 skipped	
- [INFO] Valid Preview Website	Full Name: Valid Preview Website	00:00:27.347
	Start / End / Elapsed: 2024/02/23 16:06:07.581 / 2024/02/23 16:06:34.928 / 00:00:27.347	
	Status: PASS	
• [KEYWORD] mouse Open Login Page		00:00:05.365
• [KEYWORD] mouse Input Email admin@admin.com		00:00:03.115
• [KEYWORD] mouse Input Password Admin1234		00:00:03.067
• [KEYWORD] mouse Submit Credentials		00:00:03.047
• [KEYWORD] mouse Store Alert		00:00:03.096
• [KEYWORD] mouse Click Project		00:00:03.293
• [KEYWORD] mouse Open Workspace Page		00:00:02.312
• [KEYWORD] mouse Click Button Preview		00:00:02.034
+ [TEARDOWN] browserClose Close Browser		00:00:03.108

รูป 4.30 ผลลัพธ์ของ TC08 Use case Preview



รูป 4.31 กดปุ่มเพื่อแสดงผลลัพธ์เว็บไซต์



รูป 4.32 ผลลัพธ์การ Preview Page

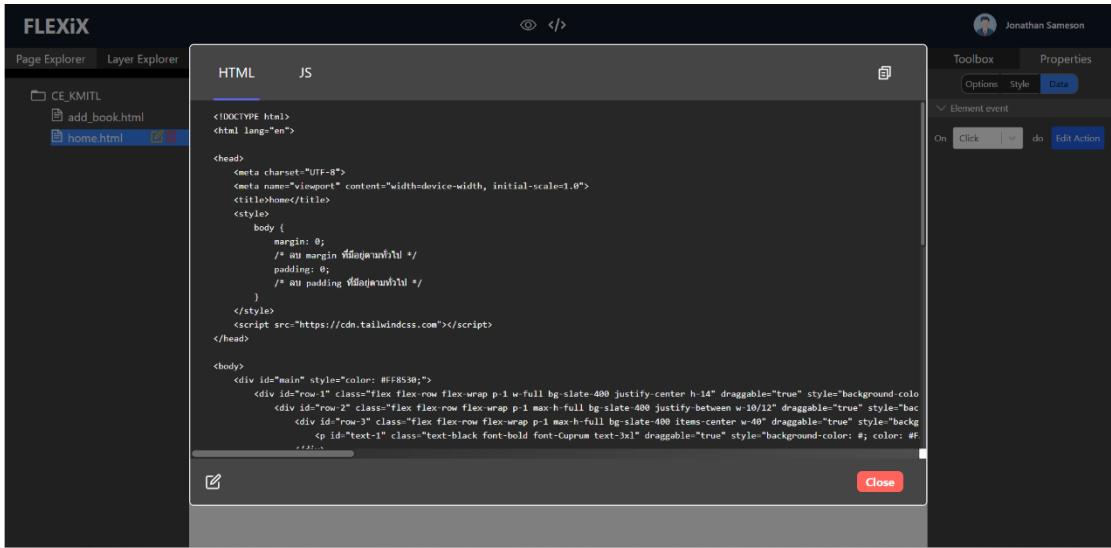
4.6 การทดลองใช้งาน Manual JavaScript

4.6.1 วัตถุประสงค์

เพื่อทดสอบการใช้งานฟังก์ชันการเขียน JavaScript เพื่อให้ผู้ใช้งานสามารถใช้ JavaScript ที่นักออกแบบจากที่ระบบเตรียมไว้ให้ได้

ตาราง 4.9 ผลการทดลอง Use Case ของการ Manual JavaScript

Test Case ID	TC-09
Use Case Name	API
Use Case ID	UC-09
Actor	ผู้ใช้งาน
Preconditions	ต้องเข้าสู่ระบบ ต้องสร้าง Project ต้องสร้าง Element
Flow of Events	1. เลือก Element ที่ต้องการใช้งาน JavaScript 2. เลือกเมนู Properties 3. เลือกเมนู Data 4. เลือก Event เป็น OnClick 5. เลือก Edit Action 6. ทดลองใช้งาน JavaScript เพื่อเปลี่ยนข้อมูลภายใน Text
ผลการทดสอบที่คาดหวัง	สามารถเขียน Manual Java Script สำเร็จ
ผลการทดสอบจริง	สามารถเขียน Manual Java Script สำเร็จ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน



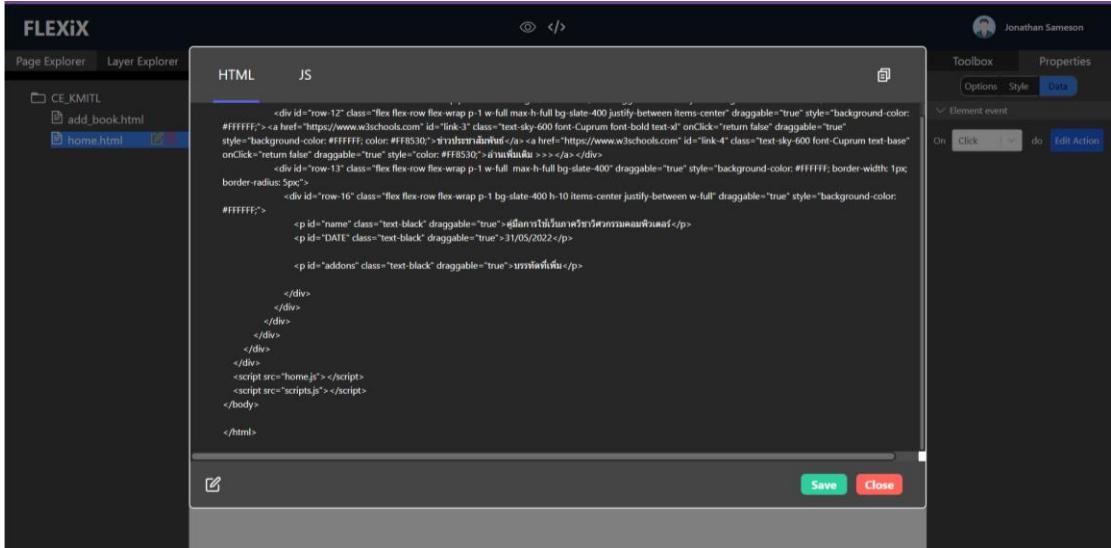
```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Home</title>
    <style>
        body {
            margin: 0;
            /* Առ margin մնայածին */
            padding: 0;
            /* Առ padding մնայածին */
        }
        </style>
        <script src="https://cdn.tailwindcss.com"></script>
    </head>

    <body>
        <div id="main" style="color: #FF8530;">
            <div id="row-1" class="flex flex-row flex-wrap p-1 w-full bg-slate-400 justify-center h-14" draggable="true" style="background-color: #FF8530; color: #FF8530; font-weight: bold; text-align: center; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: center; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">
                <div id="row-2" class="flex flex-row flex-wrap p-1 w-full bg-slate-400 justify-between w-10/12" draggable="true" style="background-color: #FF8530; color: #FF8530; font-weight: bold; text-align: center; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: space-around; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">
                    <div id="row-3" class="flex flex-row flex-wrap p-1 w-full bg-slate-400 items-center w-40" draggable="true" style="background-color: #FF8530; color: #FF8530; font-weight: bold; text-align: center; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: center; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">
                        <div id="text-1" class="text-black font-bold font-Cuprum text-3xl" style="background-color: #FF8530; color: #FF8530; font-weight: bold; font-size: 30px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">ՀԱՅԱՍՏԱՆ</div>
                    </div>
                </div>
            </div>
        </div>
    </body>

```

Հյու 4.33 հնա Code HTML



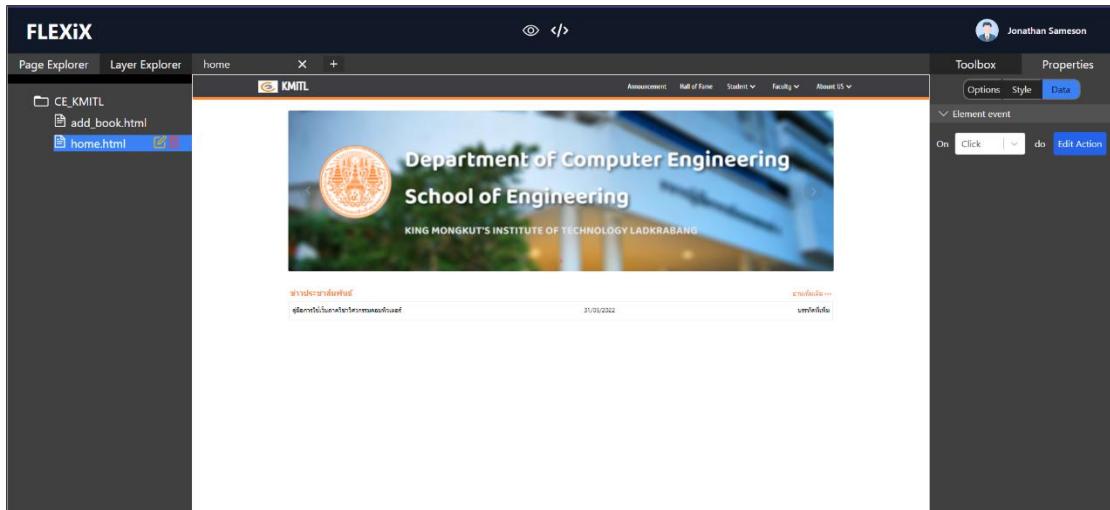
```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Home</title>
    <style>
        body {
            margin: 0;
            /* Առ margin մնայածին */
            padding: 0;
            /* Առ padding մնայածին */
        }
        </style>
        <script src="https://www.w3schools.com" id="link-3" class="text-sky-600 font-bold text-xf" onClick="return false" draggable="true" style="background-color: #FFFFFF; color: #FF8530; font-weight: bold; text-align: center; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: center; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">
            <a href="https://www.w3schools.com" id="link-4" class="text-sky-600 font-Cuprum text-base" onClick="return false" draggable="true" style="background-color: #FFFFFF; color: #FF8530; font-weight: bold; text-align: center; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: center; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">Ինչու այսպիսով կարող են լինել այսպիսով</a>
        </script>
        <script src="home.js" ></script>
        <script src="scripts.js" ></script>
    </head>

    <body>
        <div id="row-12" class="flex flex-row flex-wrap p-1 w-full max-h-full bg-slate-400 justify-between items-center" draggable="true" style="background-color: #FF8530; color: #FF8530; font-weight: bold; text-align: center; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: space-around; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">
            <div id="row-13" class="flex flex-row flex-wrap p-1 w-full max-h-full bg-slate-400" draggable="true" style="background-color: #FFFFFF; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: center; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">
                <div id="row-16" class="flex flex-row flex-wrap p-1 bg-slate-400 h-10 items-center justify-between w-full" draggable="true" style="background-color: #FFFFFF; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: space-around; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">
                    <p id="name" class="text-black" draggable="true" style="background-color: #FFFFFF; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: center; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">Անուն Ազգանուն</p>
                    <p id="DATE" class="text-black" draggable="true" style="background-color: #FFFFFF; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: center; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">31/05/2022</p>
                    <div id="addons" class="text-black" draggable="true" style="background-color: #FFFFFF; border: 1px solid black; border-radius: 5px; padding: 5px; margin-bottom: 10px; width: fit-content; height: 14px; align-items: center; justify-content: center; gap: 10px; font-size: 14px; font-family: sans-serif; letter-spacing: 1px; text-decoration: none; transition: all 0.3s ease; position: relative; z-index: 1; ">Առաջ առ առաջ</div>
                </div>
            </div>
        </div>
    </body>

```

Հյու 4.34 սկզբանական Code HTML



รูป 4.35 ผลลัพธ์หลังจากเพิ่ม Code HTML

4.7 การทดลองใช้งาน Generate Code

4.7.1 วัตถุประสงค์

เพื่อทดสอบการใช้งานฟังก์ชันการ Generate Code เพื่อให้ผู้ใช้งานสามารถคัดลอก หรือดาวน์โหลดโค้ดที่ทำกับระบบไปใช้งานได้ เพื่อนำไปพัฒนาต่อ

ตาราง 4.10 ผลการทดลอง Use Case ของการ Generate Code

Test Case ID	TC-10
Use Case Name	API
Use Case ID	UC-10
Actor	ผู้ใช้งาน
Preconditions	ต้องเข้าสู่ระบบ ต้องสร้าง Project ต้องสร้าง Element
Flow of Events	1. เลือกปุ่ม Generate Code เพื่อแสดงโค้ด 2. ทดลองใช้งาน Generate Code โดยจะแสดงโค้ดของหน้าเพจที่อยู่ใน Workspace ปัจจุบัน
ผลการทดสอบที่คาดหวัง	สามารถเขียน Generate Code สำเร็จ
ผลการทดสอบจริง	สามารถเขียน Generate Code สำเร็จ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน



ສັບ 4.36 ມໍານີ້ Project



ສັບ 4.37 ທຳການ Download Code Project

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Home</title>
    <style>
      body {
        margin: 0;
        /* no margin ດູງເກົ່າໄວ */
        padding: 0;
        /* no padding ດູງເກົ່າໄວ */
      }
      </style>
      <script src="https://cdn.tailwindcss.com"></script>
    </head>
    <body>
      <div id="main" style="color: #F08300;">
        <div id="row-1" class="flex flex-row flex-wrap p-1 w-full bg-slate-400 justify-center h-14" draggable="true" style="background-color: #E3042;">
          <div id="row-2" class="flex flex-row flex-wrap p-1 w-full bg-slate-400 justify-between w-10/12" draggable="true" style="background-color: #E3042;">
            <div id="row-3" class="flex flex-row flex-wrap p-1 max-h-full bg-slate-400 items-center w-40" draggable="true" style="background-color: #E3042;">
              <img id="text-1" class="text-black font-bold font-Cuprum text-3xl" draggable="true" style="background-color: #F08300; margin-left: 10px;"/>
            </div>
            <div id="row-4" class="flex flex-row flex-wrap p-1 max-h-full bg-slate-400 items-center justify-end w-5/12" draggable="true" style="margin-right: 20px; background-color: #E3042;">
              <div id="row-5" class="flex flex-row flex-wrap p-1 max-h-full bg-slate-400 justify-between w-1/12" draggable="true" style="background-color: #E3042;">
                <a href="#">
                  <select id="select-1" draggable="true" class="font-Cuprum text-lg" style="background-color: #E3042; color: #FFFFFF;">
                    <option id="option-2" value="null">Faculty</option>
                  </select><select id="select-2" draggable="true" style="background-color: #E3042; color: #FFFFFF;">
                    <option id="option-3" value="null">About US</option>
                  </select></div>
                </div>
              </div>
            </div>
          </div>
        </div>
        <div id="row-6" class="flex flex-row flex-wrap p-1 w-full bg-slate-400 h-auto" draggable="true" style="background-color: #FF8530;"></div>
        <div id="row-7" class="flex flex-row flex-wrap p-1 w-full bg-slate-400 justify-center h-auto" draggable="true" style="background-color: #FF8530; margin-top: 20px; color: #000000; background-color: #000000;">
          <div id="row-8" class="flex flex-row flex-wrap p-1 max-h-full bg-slate-400 w-9/12" draggable="true" style="background-color: #FF8530; margin-top: 20px; color: #000000; background-color: #000000;">
            <div id="row-9" class="flex flex-row flex-wrap p-1 w-full max-h-full bg-slate-400" draggable="true" style="margin-top: 20px; background-color: #FF8530;">
              <div id="row-10" class="flex flex-row flex-wrap p-1 w-full max-h-full bg-slate-400 justify-center" draggable="true" style="background-color: #FF8530;">
                <div id="row-11" class="flex flex-row flex-wrap p-1 w-full max-h-full bg-slate-400 w-9/12" draggable="true" style="background-color: #FF8530;">
                  <div id="row-12" class="flex flex-row flex-wrap p-1 w-full max-h-full bg-slate-400 justify-between items-center" draggable="true" style="background-color: #FF8530;">
                    <div id="row-13" class="flex flex-row flex-wrap p-1 bg-slate-400 h-10 items-center justify-between w-full" draggable="true" style="background-color: #FF8530;">
                      <a href="#">
                        <img alt="Logo" style="width: 100px; height: 100px; border-radius: 50%; border: 2px solid #FF8530; margin-right: 10px;"/>
                        <span>CE_KMITL</span>
                      </a>
                    </div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </body>
  </html>

```

ສັບ 4.38 Code ທີ Download

บทที่ 5

สรุปผลและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงภาพรวมของการพัฒนาระบบ ของ เว็บแอปพลิเคชัน Front-end Builder ที่ได้ พัฒนาเรียบร้อยแล้วโดยในส่วนที่หนึ่งจะกล่าวสรุปภาพรวมของการพัฒนาแอปพลิเคชันที่สำเร็จ ส่วนที่สอง จะกล่าวถึงปัญหาที่เกิดขึ้นระหว่างการทำงาน และการพัฒนาแอปพลิเคชัน และ ส่วน สุดท้าย จะกล่าวถึง แนวทางการแก้ปัญหา และ การพัฒนาต่อในอนาคตเพื่อให้เป็นแอปพลิเคชันที่ สมบูรณ์มากขึ้น

5.1 สรุปผลที่ได้จากการพัฒนาโปรแกรม

การพัฒนาเว็บแอปพลิเคชัน Front-end Builder โปรแกรมสามารถสร้างหน้า Front end เพื่อที่จะแสดงข้อมูลในรูปแบบเว็บไซต์และช่วยประหยัดเวลาในการเขียนโค้ดสำหรับสร้างหน้า เว็บไซต์และโปรแกรมสามารถนำข้อมูล API ของผู้ใช้มาทำการทดสอบได้ทันทีโดยไม่ต้องลง ตามที่ต้องการหรือไม่

5.2 ปัญหา และอุปสรรค

- 1) Requirement ไม่ชัดเจนทำให้ต้องแก้ไขในส่วนที่ออกแบบไปแล้วใหม่
- 2) เนื่องจากหน้าเว็บไซต์มีการออกแบบไม่เหมาะสมกับการทำ Responsive ทำให้เมื่อใช้งานกับบาง Device และจะไม่สามารถแสดงผลออกมาก็ได้มากนัก
- 3) เนื่องจากผู้จัดทำมีประสบการณ์น้อยในการเขียนโปรแกรมภาษาบางชนิดที่ใช้สร้าง เว็บไซต์ รวมไปถึงการใช้ Library บางตัวที่ใช้สร้างทำให้จึงต้องใช้เวลาในการศึกษา นาน การพัฒนาเว็บไซต์จึงเป็นไปได้ช้า

5.3 แนวทางการพัฒนาต่อ

แนวทางการพัฒนาเว็บแอปพลิเคชัน Front-end Builder ต่อจากโครงการนี้ คือ

- 1) พัฒนาในส่วนของการนำเข้าสู่ สาธารณะ (Publish)
- 2) พัฒนาในส่วนของการใช้งานเว็บแอปพลิเคชัน โดยผู้ใช้สามารถจะแชร์ไปยังเพื่อน ทำงานร่วมกันกับคนในทีมได้
- 3) ปรับปรุงเว็บแอปพลิเคชัน Front-end Builder ให้มีความสวยงาม ให้น่าสนใจ และมี การใช้งานง่ายตอบโจทย์ผู้ใช้มากขึ้น

บรรณานุกรม

Asria. (2565, 20 กรกฎาคม). HTML Language and HTML Structure. สืบค้น 27 ตุลาคม 2565.

จาก <https://asria.org/html-language-and-html-structure/>

Divmagic. (2565, 19 กรกฎาคม). Tailwind CSS คืออะไร? สืบค้น 27 ตุลาคม 2565.

จาก <https://divmagic.com/th/blog/tailwind-css>

Got. (2565, 4 กรกฎาคม). Got คืออะไร? ทำความรู้จักกับ Got แพลตฟอร์ม e-Commerce สืบค้น 27 ตุลาคม 2565.

จาก <https://www.got.co.th/service/36/show>

Open Landscape. (2565, 7 กรกฎาคม). MySQL คืออะไร? ทำความรู้จักกับ MySQL ฐานข้อมูลแบบ Relational Database Management System สืบค้น 27 ตุลาคม 2565.

จาก <https://blog.openlandscape.cloud/mysql>

Skouldio. (2565, 20 มกราคม). Docker คืออะไร? ทำความรู้จักกับ Docker เครื่องมือสำหรับบรรจุและรันแอปพลิเคชัน สืบค้น 27 ตุลาคม 2565.

จาก <https://blog.skouldio.com/what-is-docker/>

Techterrotor. (2565, 2 กุมภาพันธ์). Jenkins คืออะไร? ทำความรู้จักกับ Jenkins เครื่องมืออัตโนมัติสำหรับการพัฒนาซอฟต์แวร์ สืบค้น 27 ตุลาคม 2565.

จาก <https://www.techterrotor.com/2022/02/jenkins.html>

TechupTH. (2565, 15 กรกฎาคม). React คืออะไร? ทำความรู้จักกับ React กรอบงานพัฒนาเว็บไซต์ สืบค้น 27 ตุลาคม 2565.

จาก <https://www.techupth.com/articles/react>

Ultimate Python. (2565, 10 กรกฎาคม). What Can Python Do? สืบค้น 27 ตุลาคม 2565.

จาก <https://www.ultimatepython.co/post/what-can-python-do>