

เว็บแอปพลิเคชันจัดเก็บและค้นหากรณีทดสอบ

WEB APPLICATION FOR STORING AND SEARCHING
TEST CASES



ชโยดม พานิชเจริญ

สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตาม

หลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2566

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WEB APPLICATION FOR STORING AND SEARCHING TEST CASES



A COOPERATIVE EDUCATION SUBMITTED IN
PARTIAL FULFILLMENT OF THE REQUIREMENT FOR
THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE)
DEPARTMENT OF COMPUTER SCIENCE, SCHOOL OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2023

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อสหกิจศึกษา เว็บแอปพลิเคชันจัดเก็บและค้นหากรณีทดสอบ
Web Application for Storing and Searching Test Cases
ชื่อนักศึกษา นายชโยดม พานิชเจริญ รหัสนักศึกษา 63050114
ปริญญา วิทยาศาสตร์บัณฑิต ปริญญาตรี
ภาควิชา วิทยาการคอมพิวเตอร์
ปีการศึกษา 2566
อาจารย์ที่ปรึกษา ผศ.ดร. ปัทมา เจริญพร

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์) ประจำปีการศึกษา 2566

คณะกรรมการสอบ	ลายมือชื่อ
ผศ.ดร. อนันตพร ทรราชคุมาตย์ ประธานกรรมการ	
ผศ.ดร. ปัทมา เจริญพร กรรมการและอาจารย์ที่ปรึกษา	

ลิขสิทธิ์ของคณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อสหกิจศึกษา	เว็บแอปพลิเคชันจัดเก็บและค้นหากรณีทดสอบ
ชื่อนักศึกษา	นายชโยดม พานิชเจริญ รหัสนักศึกษา 63050114
ปริญญา	วิทยาศาสตร์บัณฑิต ปริญญาตรี
ภาควิชา	วิทยาการคอมพิวเตอร์
คณะ	วิทยาศาสตร์
มหาวิทยาลัย	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)
ปีการศึกษา	2566
อาจารย์ที่ปรึกษา	ผศ.ดร. ปัทมา เจริญพร

บทคัดย่อ

ในสหกิจศึกษาครั้งนี้มีวัตถุประสงค์เพื่อตอบสนองความต้องการของพนักงานฝ่าย QA ของบริษัท ซี.เจ.เอ็กซ์เพรส กรุ๊ปจำกัด โดยได้มีการพัฒนาแอปพลิเคชันในรูปแบบของเว็บแอปพลิเคชันโดยใช้หลักการ CRUD โดยที่มีการออกแบบโดยเฉพาะ โดยสหกิจครั้งนี้จะมีการศึกษาเครื่องมือและเทคโนโลยีต่างๆ รวมไปถึงการวิเคราะห์ความต้องการ จากนั้นนำมาออกแบบระบบและทำการพัฒนาตามกระบวนการที่วางเอาไว้โดยจะมีการออกแบบ กรณีการใช้งาน(Use Case) แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์ (Sequence Diagram) และแผนภาพจำลองข้อมูล (ER Diagram) จากผลลัพธ์ที่มีการทดสอบโปรแกรม และการสอบถามด้วยแบบสอบถามเพื่อวัดผลแสดงให้เห็นว่าสามารถพัฒนาเว็บแอปพลิเคชันได้ตรงตามความต้องการ จึงสามารถสรุปได้ว่าเว็บแอปพลิเคชันจัดเก็บและค้นหากรณีทดสอบทำให้กระบวนการค้นหาเอกสารกรณีทดสอบเพื่อนำไปใช้อ้างอิงของพนักงานฝ่าย QA มีประสิทธิภาพและมีความรวดเร็วมากยิ่งขึ้น

คำสำคัญ : เว็บแอปพลิเคชัน หลักการCRUD

Title	Web Application for Storing and Searching Test Cases
Students	Mr. Chayodom Panitcharoen Student ID 63050114
Degree	Bachelor of Science (Computer Science)
Department	Computer Science
Faculty	Science
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)
Academic Year	2023
Advisor	Asst. Prof. Dr. Patama Charoenporn.

Abstract

This Cooperative education aims to meet the needs of QA employees at C.J. Express Group Co.,Ltd. Applications have been developed in the form of web applications using CRUD concepts. With a special design This Cooperative education will study various technology tools, including requirement analysis. Then use it to design the system and develop it according to the laid out process by designing Use Cases, Sequence diagrams, and ER Diagrams. The results of program testing and questionnaire-based evaluations show that the web application has been developed according to the requirements. Therefore, it can be concluded that the Web application for storing and searching test cases has made the process of searching for test case documents for reference by QA department employees more efficient and significantly faster.

Keywords : web application, CRUD

กิตติกรรมประกาศ

การทำสหกิจศึกษาครั้งนี้สำเร็จลุล่วงได้ เนื่องจากความกรุณาของ ผศ.ดร.ปัทมา เจริญพร ผู้เป็นที่ปรึกษาและ ผศ.ดร. อนันตพร ทรรษकुณาคัย ผู้เป็นกรรมการ ที่ให้คำปรึกษาและแก้ไขปัญหาที่เกิดขึ้น รวมไปถึงการตรวจทานแก้ไขและชี้แนะข้อบกพร่อง จนกระทั่งสหกิจศึกษาครั้งนี้เสร็จสิ้น ผู้เขียนจึงขอขอบพระคุณอาจารย์อย่างสูง

ขอขอบพระคุณบริษัท ซี.เจ.เอ็กซ์เพรส กรุ๊ปจำกัดรวมถึงพี่ ธวัชศักร บูรณะถาวร ที่ได้มอบโอกาสในการทำสหกิจศึกษา มอบความรู้และประสบการณ์ในการทำงาน รวมไปถึงการให้คำปรึกษาปัญหาต่างๆและชี้แนะการแก้ไขข้อบกพร่องในการทำงาน ทำให้การทำงานสามารถดำเนินไปได้ด้วยดี

ขอขอบพระคุณอาจารย์ทุกท่านในภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ ที่ได้มอบความรู้ทั้งด้านวิชาการ คุณธรรม จริยธรรมและด้านประสบการณ์ในการทำงานในสายงานต่างๆตลอด 4 ปี การเรียนรู้ที่ผ่านมา ผู้เขียนได้นำความรู้มาพัฒนาและปรับใช้ในการทำงาน จนสหกิจศึกษาครั้งนี้สำเร็จลุล่วง

สุดท้ายนี้ขอขอบพระคุณครอบครัว เพื่อนและผู้ที่เกี่ยวข้องในการทำสหกิจครั้งนี้ ที่ให้การสนับสนุน ทำให้สหกิจศึกษาครั้งนี้สำเร็จลุล่วง

ชโยดม พานิชเจริญ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญรูป.....	ช
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 วัตถุประสงค์.....	1
1.3 ขอบเขต.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 เอกสารและเทคโนโลยีที่เกี่ยวข้อง.....	3
2.1 JavaScript.....	3
2.2 NodeJS.....	4
2.3 HTML.....	4
2.4 CSS.....	5
2.5 Express.js.....	5
2.6 SQL.....	6
2.7 MySQL.....	7
2.8 ORM Sequelize.....	7
2.9 หลักการ CRUD.....	8
2.10 React.....	9
2.11 Axios.....	10
2.12 Material UI.....	11
2.13 Jira Software.....	12
2.14 Postman.....	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
บทที่ 3 วิธีการดำเนินงาน	13
3.1 ขั้นตอนการดำเนินงาน.....	13
3.2 การออกแบบระบบ	14
บทที่ 4 ผลการดำเนินงาน.....	25
4.1 หน้าจอปฏิสัมพันธ์ของระบบ (User Interface).....	25
4.2 ผลการทดสอบโปรแกรม.....	28
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ	39
5.1 สรุปผลการดำเนินงาน	39
5.2 ข้อเสนอแนะในการพัฒนา	40
เอกสารอ้างอิง	41

สารบัญตาราง

ตารางที่	หน้า
3.1 Use case Description ของ จัดเก็บข้อมูล Test Case	16
3.2 Use case Description ของ ค้นหาข้อมูล Test Case.....	16
3.3 Use case Description ของ แก้ไขข้อมูล Test Case	17
3.4 Use case Description ของ ลบข้อมูล Test Case	17
3.5 Use case Description ของ ดาวน์โหลดไฟล์ Test Case	18
3.6 Entity Relationship Diagram ระบบจัดเก็บและค้นหา Test case	24



สารบัญรูป

รูปที่	หน้า
2.1 ตัวอย่างการพัฒนาด้วยภาษา JavaScript.....	3
2.2 ตัวอย่างการใช้งาน HTML ในรูปแบบ JSX	5
2.3 ตัวอย่างการใช้งาน Express.js เพื่อสร้าง RESTful service	6
2.4 ตัวอย่างการใช้งาน MySQL Workbench เพื่อค้นหาข้อมูล.....	7
2.5 ตัวอย่างการใช้งาน Sequelize เพื่อเชื่อมต่อฐานข้อมูล MySQL	8
2.6 ตัวอย่างการเขียน React ในรูปแบบ Component.....	10
2.7 การใช้งาน axios ในการร้องขอข้อมูลผ่าน API.....	11
2.8 การใช้งาน Postman เพื่อทดสอบ API ค้นหาข้อมูล.....	12
3.1 ตัวอย่างการค้นหาด้วย Jira Software	13
3.2 แผนภาพยูสเคสระบบจัดเก็บและค้นหา Test case	15
3.3 Sequence Diagram ระบบจัดเก็บข้อมูล Test Case	19
3.4 Sequence Diagram ระบบค้นหา Test Case.....	20
3.5 Sequence Diagram ระบบแก้ไขข้อมูล Test Case.....	21
3.6 Sequence Diagram ระบบลบข้อมูล Test Case	22
3.7 Sequence Diagram ระบบดาวน์โหลดไฟล์ Test Case.....	23
3.8 Entity Relationship Diagram ระบบจัดเก็บและค้นหา Test case	24
4.1 หน้าค้นหาข้อมูล	25
4.2 หน้าจัดเก็บข้อมูล.....	26
4.3 หน้าแก้ไขข้อมูล	27
4.4 หน้าลบข้อมูล.....	27
4.5 ผลการทดสอบ API (ID 1-8).....	29
4.6 ผลการทดสอบ API (ID 9-16).....	30
4.7 ผลการทดสอบ API (ID 17-22)	31
4.8 ผลการทดสอบการทำงาน (ID 1-8).....	32
4.9 ผลการทดสอบการทำงาน (ID 9-18).....	33
4.10 ผลแบบสอบถามหัวข้อ”แอปพลิเคชันสามารถเพิ่มประสิทธิภาพในการค้นหาได้”	34
4.11 ผลแบบสอบถามหัวข้อ”แอปพลิเคชันสามารถจัดเก็บข้อมูล Test Case ได้อย่างมีประสิทธิภาพ”	35

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.12 ผลแบบสอบถามหัวข้อ”แอปพลิเคชันสามารถดาวน์โหลดไฟล์ได้อย่างมีประสิทธิภาพ”	35
4.13 ผลแบบสอบถามหัวข้อ”แอปพลิเคชันสามารถแก้ไขข้อมูล Test Case ได้อย่างมีประสิทธิภาพ”	36
4.14 ผลแบบสอบถามหัวข้อ”แอปพลิเคชันสามารถลบ Test Case ได้อย่างมีประสิทธิภาพ” ...	37
4.15 ผลแบบสอบถามหัวข้อ”ส่วนต่อประสานกับผู้ใช้ (User Interface) สามารถเข้าใจได้ง่ายและสะดวกต่อการใช้งาน”	37
4.16 ข้อคิดเห็นและข้อเสนอแนะเพิ่มเติม.....	38



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบัน บริษัท ซี.เจ. เอ็กซ์เพรส กรุ๊ป จำกัด เป็นบริษัทที่ดำเนินธุรกิจค้าปลีกที่ได้มีการพัฒนาซอฟต์แวร์สำหรับใช้ในบริษัท โดยเลือกใช้เครื่องมือจัดการโครงการงาน JIRA Software เพื่อจัดการโครงการงานและติดตามการพัฒนาซอฟต์แวร์ในรูปแบบการพัฒนาแบบ Agile อย่างไรก็ตาม การใช้งาน JIRA Software ในการบันทึกผลเอกสารกรณีทดสอบ(Test Case)สามารถทำได้อย่างมีประสิทธิภาพ แต่กลับมีข้อจำกัดในการค้นหาเอกสารกรณีทดสอบที่เก็บบันทึกไว้ ทำให้สามารถค้นหาได้โดยยากโดยการพัฒนา software ของบริษัท ซี.เจ. เอ็กซ์เพรส กรุ๊ป นั้นจะมีการพัฒนา version เพื่อเพิ่มประสิทธิภาพอยู่ตลอดเวลา ซึ่งทุกการพัฒนาของทุก version เมื่อสำเร็จจะได้รับการตรวจสอบด้วย manual test ทุกครั้ง หากสามารถอ้างอิงวิธีทดสอบการจากเอกสาร test case เก่าจะสามารถลดระยะเวลาในการเขียนเอกสาร test case ใหม่ได้โดยไม่ต้องเขียนเริ่มใหม่ทั้งหมด ทำให้สามารถทำ regression testing ได้โดยใช้เวลาลดลง และอีกทั้งยังสามารถทำให้เกิดความเข้าใจวิธีการทดสอบเมื่อมีการเปลี่ยนผ่านพนักงานได้ แต่หากไม่มีเอกสาร test case เก่าจะทำให้เกิดความล่าช้าและสูญเสียเวลาในการพัฒนาซอฟต์แวร์

เพื่อรองรับความต้องการในการจัดเก็บและค้นหาเอกสาร กรณีทดสอบที่มีประสิทธิภาพ จึงตัดสินใจพัฒนาเว็บแอปพลิเคชันเพื่อจัดเก็บและค้นหาเอกสารกรณีทดสอบอย่างมีประสิทธิภาพ โดยเว็บแอปพลิเคชันนี้จะช่วยให้ทีมพัฒนาสามารถบันทึกเอกสารกรณีทดสอบที่สำคัญได้โดยง่าย และยังมีการค้นหาที่สามารถใช้งานได้ง่ายและตรงความต้องการ

1.2 วัตถุประสงค์ของงาน

- 1.) เพื่อเพิ่มประสิทธิภาพให้ทีมพัฒนาสามารถค้นหากรณีทดสอบในระบบและลดเวลาค้นหาข้อมูลที่สำคัญในการพัฒนาซอฟต์แวร์
- 2.) เพื่อพัฒนาเว็บแอปพลิเคชันสามารถช่วยทีมพัฒนาจัดเก็บกรณีทดสอบที่สำคัญได้ง่ายและสะดวกสบาย

1.3 ขอบเขตของงาน

- 1.) พัฒนาเว็บแอปพลิเคชันที่สามารถค้นหากรณีทดสอบโดยคำค้นหาที่กำหนดไว้
- 2.) พัฒนาเว็บแอปพลิเคชันสามารถจัดเก็บกรณีทดสอบได้
- 3.) เว็บแอปพลิเคชันสามารถแก้ไขข้อมูลได้
- 4.) เว็บแอปพลิเคชันสามารถลบข้อมูลได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1.) ได้เว็บแอปพลิเคชันที่สามารถช่วยในการค้นหากรณีทดสอบที่สำคัญได้อย่างมีประสิทธิภาพ
- 2.) ได้เว็บแอปพลิเคชันที่สามารถช่วยในการช่วยจัดเก็บกรณีทดสอบได้ง่ายและสะดวก



บทที่ 2

เอกสารและเทคโนโลยีที่เกี่ยวข้อง

ในการพัฒนาเว็บแอปพลิเคชันสำหรับการจัดเก็บกรณีทดสอบ ทางผู้จัดทำได้ศึกษาค้นคว้าเพิ่มเติมเกี่ยวกับเอกสารและการใช้งานแอปพลิเคชัน เพื่อนำมาสร้างเว็บแอปพลิเคชัน ดังต่อไปนี้

2.1 JavaScript

JavaScript เป็นภาษาโปรแกรมที่ถูกพัฒนาโดย Brendan Eich ในปี 1995 มีลักษณะการเขียนแบบโพรโทไทป์ มีโครงสร้างภาษาที่คล้ายกับ Java และ C เนื่องจาก JavaScript ได้รับการพัฒนามาจากภาษาซี

JavaScript ใช้ในการพัฒนาหน้าเว็บไซต์แบบโต้ตอบกับผู้ใช้(interactive) ร่วมกับภาษา HTML และ CSS เพื่อให้เว็บไซต์สามารถตอบสนองผู้ใช้งานได้มากขึ้น โดย JavaScript จะฝังอยู่ในเว็บเบราว์เซอร์(client-side) ในด้านการใช้งานได้หลายรูปแบบเช่น ตรวจสอบข้อมูลก่อนเพิ่มข้อมูลเข้าระบบ สร้างเนื้อหาที่มีการเปลี่ยนแปลงได้ในระบบ การพัฒนาหน้าเว็บไซต์ด้วยภาษา JavaScript โดยจะเป็นส่วนหนึ่งของ HTML โดยเพิ่มด้านในส่วน of <head> tag โดยจะใช้ <script> tag เพื่อเรียกใช้ JavaScript ในปัจจุบันมีเทคโนโลยีที่ชื่อว่า Nodejs เป็น Cross platform runtime environment ทำให้ลดข้อจำกัดของ JavaScript

โดยการนำ JavaScript มาใช้งานในส่วนการพัฒนาโปรแกรมทั้งหมดไม่ว่าจะเป็นในส่วนของอินเตอร์เฟซผู้ใช้และเซิร์ฟเวอร์โดยตัวอย่างรูปที่ 2.1 แสดงการใช้งาน JavaScript โปรแกรมเพื่อทำการลบไฟล์ภายในเซิร์ฟเวอร์

```
async function deleteFile(fileName) {
  const filePath = `./excel/${fileName}`;

  fs.unlink(filePath, (err) => {
    if (err) {
      console.error('Error delete file:', err);
    } else {
      console.log('Successfully deleted file:', fileName);
    }
  });
}
```

รูปที่ 2.1 ตัวอย่างการพัฒนาด้วยภาษา JavaScript

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 NodeJS

NodeJS เป็น Cross platform runtime environment เพื่อจะสามารถ สามารถรันโค้ด JavaScript ได้ในหลายระบบปฏิบัติการ เช่น Windows, Linux, macOS ฯลฯ โดยหากไม่มี NodeJS ภาษา JavaScript จะสามารถทำงานบนฝั่งเว็บเบราว์เซอร์เพียงอย่างเดียว ด้วยเหตุนี้ทำให้ง่ายต่อการพัฒนาส่วน front-end และ back-end เพราะสามารถพัฒนาด้วยภาษา JavaScript ที่เหมือนกัน เมื่อมีการนำ NodeJS มาใช้จะสามารถนำภาษา JavaScript มาพัฒนาในส่วนของเซิร์ฟเวอร์ได้ เนื่องจากปัจจุบัน JavaScript เป็นภาษาที่ได้รับความนิยมเป็นอันดับต้นๆ ทำให้มีแหล่งเรียนรู้ framework และ library ที่หลากหลายให้เลือกใช้ตัวอย่างเช่น Express.js ที่มีชุดคำสั่งเพื่อจัดการและสร้าง API ในรูปแบบ RESTful API ในการใช้งาน NodeJS ใช้เพื่อสร้างในส่วนของเว็บเซิร์ฟเวอร์และเว็บแอปพลิเคชันที่จะรับและแสดงข้อมูลให้ผู้ใช้

โดยการนำ NodeJS มาใช้นั้นใช้เพื่อให้สามารถพัฒนาอินเทอร์เน็ตเพสผู้ใช้และเซิร์ฟเวอร์บนระบบปฏิบัติการ Windows และใช้ Node Package Manager เพื่อใช้จัดการกับ package ต่างๆที่ได้ติดตั้งเพิ่มเติมเช่น React, Axios, Express

2.3 HTML

HTML (Hypertext Markup Language) เป็นภาษาโปรแกรมที่ใช้สำหรับสร้างหน้าเว็บไซต์ โดยใช้ Text Editor เช่น Notepad เป็นเครื่องมือในการเขียน โครงสร้างของ HTML จะประกอบไปด้วย Tag ต่างๆ เพื่อให้เว็บเบราว์เซอร์แปลงออกมาเป็นหน้าเว็บไซต์ โดย Tag ต่างๆก็จะมีหน้าที่แตกต่างกันออกไป เช่น แท็ก h1 จะเป็นแท็กที่กำหนดขนาดตัวอักษรให้มีขนาดลำดับที่ 1 และแท็ก a ที่จะเป็นแท็กสำหรับกำหนดลิ้งค์เพื่อไปหน้าอื่นๆ สามารถใช้ร่วมกับ Java Script และ CSS เพื่อเพิ่มการตอบสนองกับผู้ใช้และความสวยงาม โดยการใช้งาน HTML ใช้ในส่วนเว็บแอปพลิเคชันเพื่อสร้างโครงสร้างตัวหนังสือช่องกรอกข้อมูลและปุ่มต่างๆให้ผู้ใช้งานได้เห็นบนหน้าเว็บไซต์

โดยการใช้งาน HTML นั้นจะใช้อยู่ในส่วนอินเทอร์เน็ตเพสผู้ใช้ทั้งหมดองค์ประกอบต่างๆที่สร้างขึ้นล้วนต้องประกอบไปด้วยภาษา HTML ทั้งหมดรวมไปถึง Material UI ซึ่งเป็น React component library ก็มีการใช้งานภาษา HTML อยู่เบื้องหลัง โดยตัวอย่างรูปที่ 2.2 แสดงการใช้งาน HTML บางส่วนร่วมกับ MUI เพื่อสร้าง แถบเครื่องมือแอป(AppBar)ในรูปแบบ JSX

```

<AppBar position="static" sx={{ bgcolor: '#0F9A4F', boxShadow: '0 2px 2px -2px rgba(0,0,0,0.2)' }}>
  <Toolbar>
    <HomeIcon
      fontSize="large"
      sx={{ mx: 2, mt: -1, cursor: "pointer" , '&:hover': { color: '#FBE108'}}
      onClick={handleHomeIconClick}/>
    <Typography variant="h6" sx={{ textDecoration: 'none', color: 'inherit', flexGrow: 1}}>
      TESTCASE STORAGE
    </Typography>
  </Toolbar>
</AppBar>

```

รูปที่ 2.2 ตัวอย่างการใช้งาน HTML ในรูปแบบ JSX

2.4 CSS

CSS (Cascading Style Sheets) เป็นภาษาสำหรับการจัดรูปแบบการแสดงผลของ HTML โดยสามารถกำหนดตำแหน่ง สี ขนาด ฟรอนต์ และสีพื้นหลังทำให้เกิดความสวยงามของเว็บไซต์ การใช้งาน CSS นั้นจะทำการตกแต่งโครงสร้างที่สร้างไว้แล้วคือ HTML ตัวอย่างเช่น จัดองค์ประกอบของหน้า ตัวหนังสือ และปุ่มต่างๆ ใส่สีเพื่อให้ผู้ใช้สามารถเข้าใจได้ง่ายมากขึ้นและเพื่อเพิ่มความสวยงาม

โดยการนำ CSS มาใช้งานส่วนใหญ่จะเป็นการปรับแต่งด้านสี ขนาด ช่องว่างต่างโดยจะเขียนอยู่ภายใต้ MUI เพื่อปรับ Component ต่างๆให้เป็นไปตามต้องการและสวยงามมากยิ่งขึ้นโดยตัวอย่างรูปที่ 2.2 เป็นการสร้าง แถบเครื่องมือแอป(AppBar)ในรูปแบบ JSX และมีการใช้ภาษา CSS เพื่อปรับแต่งความสวยงามเพิ่มเติมโดย MUI จะมีฟังก์ชัน sx ให้เรียกใช้เพื่อปรับแต่งด้วย CSS และจาดตัวอย่างรูปจะมีการใช้งาน CSS ดังนี้

```

<AppBar position="static" sx={{ bgcolor: '#0F9A4F',
  boxShadow: '0 2px 2px -2px rgba(0,0,0,0.2)' }}>

```

จากโค้ดด้านบนเป็นส่วนหนึ่งของรูปที่ 2.2 แสดงการใช้งาน CSS เพื่อตกแต่งสีพื้นหลังในส่วน “bgcolor: '#0F9A4F'” และการเพิ่มเงาในส่วนของ “boxShadow: '0 2px 2px -2px rgba(0,0,0,0.2)'”

2.5 Express.js

Express เป็น framework ของ NodeJS ถูกออกแบบไว้ใช้สำหรับพัฒนา Web Service ใน server side อย่างรวดเร็วและง่ายโดย Express จะมีคำสั่งไว้จัดการการ routing middleware HTTP response และ request นอกจากนี้ Express ยังมีวิธีการจัดการข้อผิดพลาดอย่างมีประสิทธิภาพและมีความยืดหยุ่นในการใช้งานสามารถใช้งานกับ library อื่นได้หลากหลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการใช้ Express นั้นจะใช้เพื่อสร้างในฝั่งของเว็บเซิร์ฟเวอร์ที่จะให้บริการเว็บแอปพลิเคชันโดยการสร้างบริการจะเป็นในรูปแบบ RESTful service เพื่อให้เว็บแอปพลิเคชันสามารถทำการรับและส่งข้อมูลเพื่อประมวลผลในฝั่งเซิร์ฟเวอร์

```
router.get('/searchAll', async (req, res) => {
  try {
    const jsonTestCases = await dbhandler.searchAllTC();
    res.send(jsonTestCases);
  } catch (error) {
    console.error('Error /searchAll:', error);
    res.status(500).send('Internal Server Error');
  }
});
```

รูปที่ 2.3 ตัวอย่างการใช้งาน Express.js เพื่อสร้าง RESTful service

จากรูปที่ 2.3 แสดงตัวอย่างการใช้งาน Express.js ในการสร้าง RESTful service โดยใช้วิธีการ HTTP รูปแบบ GET ในเส้นทาง “/searchAll” เพื่อรองรับการร้องขอข้อมูลจากฐานข้อมูลทั้งหมดเพื่อตอบกลับข้อมูลในรูปแบบ JSON และได้มีการจัดการข้อผิดพลาดเมื่อไม่สามารถดึงข้อมูลจากฐานข้อมูลได้

2.6 SQL

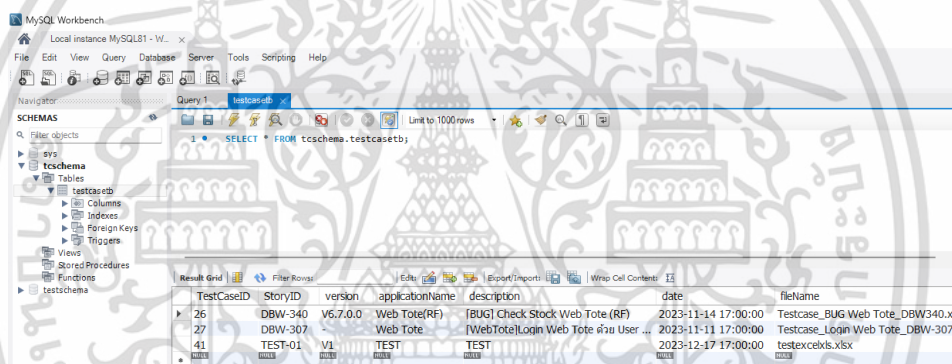
SQL (Structured Query Language) เป็นภาษาโปรแกรมที่ไว้ใช้ในการ ค้นหา จัดเก็บ คำฉนวน ในฐานข้อมูลซึ่งฐานข้อมูลเป็นรูปแบบฐานข้อมูลเชิงสัมพันธ์ที่จะเก็บข้อมูลในรูปแบบตารางโดยมีแถวและคอลัมน์ ในการใช้งาน SQL นั้นจะใช้เพื่อสร้างฐานข้อมูลและตาราง จัดเก็บและค้นหาหรือค้นหาข้อมูลตามเงื่อนไขต่างๆที่ซับซ้อน

โดยการใช้งานภาษา SQL นั้นจะใช้ร่วมกับ Sequelize ซึ่งเป็น Object Relational Mapping เพื่อค้นหาข้อมูลในรูปแบบมีเงื่อนไขเช่นการค้นหาข้อมูลแบบการจับคู่แบบรูปแบบ (pattern matching) โดยจะใช้ LIKE Operator ในการค้นหาข้อมูลเช่น “SELECT * FROM testcasesb WHERE StoryID LIKE ‘DBW%’“ เป็นเงื่อนไขในการเลือกข้อมูลโดยจะเลือกเฉพาะแถวที่ค่าในคอลัมน์ StoryID เริ่มต้นด้วย 'DBW' ตามด้วยอักขระใดๆ หรือไม่มีอักขระเลย

2.7 MySQL

MySQL เป็น ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ แบบโอเพ่นซอร์ส ใช้ภาษา SQL ในการสืบค้น และจัดการข้อมูล MySQL เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์แบบโอเพ่นซอร์สที่ทรงพลังและเป็นที่ยอมรับ ซึ่งช่วยให้ผู้ใช้สามารถจัดการและจัดเก็บข้อมูลในลักษณะที่มีโครงสร้าง ใช้งานง่าย และปลอดภัย นอกจากนี้ MySQL ยังสามารถปรับแต่งได้สูงด้วยฟังก์ชันและคุณสมบัติที่หลากหลายที่สามารถปรับแต่งให้ตรงกับความต้องการของผู้ใช้และแอปพลิเคชันต่างๆ

ใช้งาน MySQL เพราะหากไม่มีระบบจัดการฐานข้อมูลเชิงสัมพันธ์ก็จะไม่สามารถใช้งานฐานข้อมูลเชิงสัมพันธ์ได้โดย MySQL นั้นมีโปรแกรมชื่อว่า MySQL Workbench เป็นเครื่องมือที่ใช้สำหรับจัดการฐานข้อมูล MySQL Server ซึ่งเป็นรูปแบบอินเทอร์เฟซผู้ใช้ที่ใช้ง่าย รูปที่ 2.4 ทำให้สามารถใช้งานและจัดการฐานข้อมูลได้ง่าย



รูปที่ 2.4 ตัวอย่างการใช้งาน MySQL Workbench เพื่อค้นหาข้อมูล

2.8 ORM Sequelize

Sequelize เป็น Object Relational Mapping (ORM) และเป็นไลบรารีของ JavaScript ซึ่งทำหน้าที่ในการจับคู่ข้อมูลกับฐานข้อมูลเชิงสัมพันธ์และแปลงเป็นข้อมูลรูปแบบ Object Oriented Language และจะสามารถแปลงข้อมูล Object Oriented Language ไปเป็นข้อมูลเชิงสัมพันธ์ (Relational Database) โดย Sequelize จะสามารถใช้งานได้กับหลายระบบจัดการฐานข้อมูล เช่น Oracle, Postgres, MySQL, MariaDB, SQLite, SQL Server ฯลฯ ทำให้เกิดความง่ายต่อการใช้งานกับฐานข้อมูลที่มีการเปลี่ยนแปลงบ่อยและไม่เปลี่ยนแปลงเนื่องจาก Sequelize มีการจัดการคำสั่ง SQL ในการสร้างหรือปรับเปลี่ยน Schema ให้ใช้ในรูปแบบที่ง่ายและ Sequelize มีคำสั่งต่างๆในการสร้าง ค้นหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปลี่ยนแปลง และลบ ให้สามารถเรียกใช้ได้ง่ายหรือใช้คำสั่ง SQL แบบดั้งเดิมสามารถทำได้หากต้องการ

การนำ Sequelize มาใช้นั้นจะทำงานอยู่บน Nodejs และเขียนด้วยภาษา JavaScript ซึ่งจะนำมาใช้ในการเชื่อมต่อกับระบบจัดการฐานข้อมูลเชิงสัมพันธ์ เพื่อที่จะทำการค้นหา จัดเก็บ แก้ไข และลบข้อมูลในฐานข้อมูลเชิงสัมพันธ์โดยตัวอย่างการใช้งานดังรูปที่ 2.5

```
const sequelize = new Sequelize(DB_NAME, DB_USER, DB_PASSWORD, {
  host: DB_HOST,
  dialect: 'mysql',
  define: { timestamps: false }
});

const TestCase = sequelize.define('testcasesb', {
  TestCaseID: {
    type: DataTypes.INTEGER,
    allowNull: true,
    unique: true,
    primaryKey: true
  },
  StoryID: {
    type: DataTypes.STRING,
    allowNull: false
  },
  version: {
    type: DataTypes.STRING,
    allowNull: false
  },
  applicationName: {
    type: DataTypes.STRING,
    allowNull: false
  },
  description: {
    type: DataTypes.STRING,
  },
  date: {
    type: DataTypes.DATE,
    allowNull: false
  },
  fileName: {
    type: DataTypes.STRING,
    allowNull: false,
    unique: true
  }
}, {
  tableName: DB_TABLE_NAME
});
```

รูปที่ 2.5 ตัวอย่างการใช้งาน Sequelize เพื่อเชื่อมต่อฐานข้อมูล MySQL

จากรูปที่ 2.5 เป็นวิธีการเชื่อมต่อ MySQL ด้วย Sequelize โดยส่วนโค้ดด้านบนของรูปจะเป็นการใส่ข้อมูลเพื่อที่จะเชื่อมต่อฐานข้อมูลซึ่งประกอบไปด้วย ชื่อฐานข้อมูล ชื่อผู้ใช้งาน รหัสผ่านผู้ใช้งาน และชนิดของระบบจัดการฐานข้อมูล ซึ่งโค้ดด้านล่างจะเป็นการสร้าง Object เพื่ออ้างอิงถึงตัวแปรและชนิดของข้อมูลในแต่ละคอลัมน์โดยต้องกำหนดให้ตรงกับฐานข้อมูลที่สร้างขึ้น

2.9 หลักการ CRUD

CRUD ถูกย่อมาจาก Create, Read, Update และ Delete ซึ่งเป็นแนวคิดสำหรับโปรแกรมการจัดการข้อมูลขั้นพื้นฐานโดย

Create หมายถึง การสร้างข้อมูลใหม่เพื่อบันทึกลงในฐานข้อมูล

Read หมายถึง การอ่านข้อมูลจากฐานข้อมูล

Update หมายถึง การแก้ไขข้อมูลในฐานข้อมูล

Delete หมายถึง การลบข้อมูลออกจากฐานข้อมูล

โดยจากหลักการ CRUD สามารถนำไปใช้ในโปรแกรมได้ในหลากหลายรูปแบบรวมไปถึงเว็บแอปพลิเคชันที่จะใช้ SQL และ Restful API ในการสร้างเว็บแอปพลิเคชันที่สามารถแก้ไขข้อมูล โดยหากจำแนกคำสั่งต่างๆตามทฤษฎี CRUD จะจำแนกได้ดังนี้ SQL C(Create) = INSERT, R(Read) = SELECT, U(Update) = UPDATE, D(Delete) = DELETE และ Restful API C(Create) = PUT/POST, R(Read) = GET, U(Update) = PUT/POST/PATCH, D(Delete) = DELETE โดยหลักการนี้จะนำไปใช้ในการสร้างเว็บแอปพลิเคชันนี้

2.10 React

เป็น library ของ JavaScript ถูกพัฒนาขึ้นโดย Facebook มีความสามารถในการช่วยสร้างอินเตอร์เฟซผู้ใช้ให้มีความรวดเร็วและตรงตามที่ต้องการมากยิ่งขึ้น โดยจุดเด่นอยู่ที่ 3 องค์ประกอบหลักของ React ประกอบด้วย Component , state และ Props

Component ซึ่งเป็นโค้ดที่แบ่งออกเป็นส่วนย่อยๆของการทำงานหลายอันเมื่อสร้างแล้วจะสามารถนำมาใช้ซ้ำในแต่ละส่วนได้โดยไม่ต้องเขียนขึ้นมาใหม่ทั้งหมดตัวอย่างเช่นปุ่มต่างๆที่ได้มีการใช้ซ้ำช่องกรอกข้อมูล เป็นต้น

State เป็นข้อมูลที่อยู่ในแต่ละ Component ซึ่งมักจะเกี่ยวข้องกับการเปลี่ยนแปลงต่างๆบนอินเตอร์เฟซผู้ใช้เช่น การคำนวณ การโต้ตอบหรือการบ่งบอกสถานะบางอย่างที่ต้องมีการโต้ตอบโดยทันที ซึ่งการใช้ State จะช่วยให้เกิดการตอบสนองต่อประสบการณ์ผู้ใช้งานมากขึ้น

Props เป็นวิธีที่ช่วยให้เราส่งข้อมูลจาก Component หลักไปยัง Component ย่อยได้ ซึ่งเป็นวิธีที่ดีในการจัดการข้อมูลที่ต้องการนำมาแสดงหรือใช้งานใน Component ย่อยๆโดย Component

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักจะถูกเรียกว่า Parent Component และ Component ย่อยจะถูกเรียกว่า Child Component โดยตัวอย่างการใช้งาน Props เช่นการส่งข้อมูล ชื่อ อายุ สี เลขจำนวนหรือข้อมูลอื่นๆ ตามที่เราต้องการ ซึ่งจะสามารถตอบสนองการนำ Component ไปใช้ซ้ำและลดระยะเวลาในการพัฒนา

```
import React from "react";
import AppBar from '@mui/material/AppBar';
import Toolbar from '@mui/material/Toolbar';
import { Typography } from "@mui/material";
import HomeIcon from '@mui/icons-material/Home';
import { useNavigate } from 'react-router-dom';

function AppBarCJ(){
  const navigate = useNavigate();
  const handleHomeIconClick = () => {
    navigate('/');
  };
}

return(
  <AppBar position="static" sx={{ bgcolor: '#0F9A4F', boxShadow: '0 2px 2px -2px rgba(0,0,0,0.2)' }}>
    <Toolbar>
      <HomeIcon
        fontSize="large"
        sx={{ mx: 2, mt: -1, cursor: "pointer", '&:hover': { color: '#FBE108'}}
        onClick={handleHomeIconClick}/>
      <Typography variant="h6" sx={{ textDecoration: 'none', color: 'inherit', flexGrow: 1}}>
        TESTCASE STORAGE
      </Typography>
    </Toolbar>
  </AppBar>
);
}

export default AppBarCJ;
```

รูปที่ 2.6 ตัวอย่างการเขียน React ในรูปแบบ Component

จากรูปที่ 2.6 แสดงวิธีการเขียนโดยใช้ไลบรารี React โดยเขียนในรูปแบบ Component โดยแถบเครื่องมือแอป(AppBar) นี้สามารถนำมาใช้ซ้ำในหน้าค้นหาข้อมูลและการจัดเก็บข้อมูลซึ่งวิธีการใช้ Component นี้สามารถใช้งานในหน้าอื่นๆได้โดยใช้ “<AppBarCJ/>” ตามตำแหน่งที่ต้องการ

2.11 Axios

เป็น library ของ JavaScript สำหรับ HTTP Request เพื่อเชื่อมต่อ API Service ต่างๆโดย มักจะใช้ในการรับส่งข้อมูลในรูปแบบของ RestFul API ในส่วนอินเทอร์เน็ตเฟสผู้ โดย Axios จะมีความสามารถจัดการ HTTP method ต่างๆ เช่น GET, POST, PUT, DELETE ได้โดยง่ายต่อการใช้งาน นอกจากนี้ยังสามารถกำหนด data headers และการกำหนดความปลอดภัยอีกทั้งยังมีการจัดการ error อย่างเป็นระบบเมื่อเกิดปัญหาในการส่งหรือรับข้อมูลกับ API Services ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

const handleDeleteClick = async () => {
  try {
    await axios.delete(`${baseUrl}/deletedialog/delete`, {
      data: {
        TestCaseID: rowDSelect.TestCaseID,
        fileName: rowDSelect.fileName
      }
    });
    handleClose();
    handleSearch();
    alert(`Story with ID ${rowDSelect.StoryID} has been successfully deleted.`);
  } catch (error) {
    if (error.response && error.response.status === 500) {
      alert("Internal Server Error");
      handleClose();
    } else {
      handleClose();
      alert("Unknown error");
      console.error('Error:', error)
    }
  }
}

```

รูปที่ 2.7 การใช้งาน axios ในการร้องขอลบข้อมูลผ่าน API

จากรูปที่ 2.7 แสดงการใช้งาน axios ในส่วนอินเตอร์เฟซผู้ใช้เพื่อร้องขอข้อมูล HTTP ในรูปแบบ DELETE และมีการส่งข้อมูลเอกสารกรณีทดสอบและชื่อไฟล์เพื่อทำการลบเอกสารในฐานข้อมูลและไฟล์ภายในเซิร์ฟเวอร์นอกจากนี้ยังมีการดักจับข้อผิดพลาดที่อาจเกิดขึ้นและมีการแจ้งเตือนให้ผู้ใช้ทราบ

2.12 Material UI

เป็น library สำหรับ React ที่ทำให้ผู้พัฒนาสามารถสร้างอินเตอร์เฟซผู้ใช้ที่สวยงามและมีประสิทธิภาพได้ง่ายมากยิ่งขึ้น โดย Material UI ได้มีการนำแนวคิดในการใช้งานของ React โดยที่จะใช้แบ่งส่วนต่างๆเป็นในรูปแบบของ component ทำให้สามารถนำ Components พื้นฐานของ Material UI มาใช้งานได้ทันที เช่น Button, TextField, Checkbox อีกทั้งยังสามารถตอบสนอง Responsive Design ที่จะทำให้เว็บแอปพลิเคชันสามารถรองรับขนาดหน้าจอ อุปกรณ์ที่ผู้ใช้งานที่แตกต่างกันและการปรับขนาดการแสดงผลได้อย่างเหมาะสม

โดยการใช้งาน Material UI นั้นจะขึ้นอยู่กับหลายส่วนของอินเตอร์เฟซผู้ใช้โดยตัวอย่างดังรูปที่ 2.6 สามารถเห็นได้ว่ามีการใช้ Component ของ Material UI ที่หลากหลายในการสร้างแถบเครื่องมือแอป (AppBar) ซึ่งประกอบไปด้วย AppBar Toolbar HomeIcon Typography ซึ่งการใช้ Component เหล่านี้ทำให้ลดระยะเวลาในการพัฒนา

2.13 Jira Software

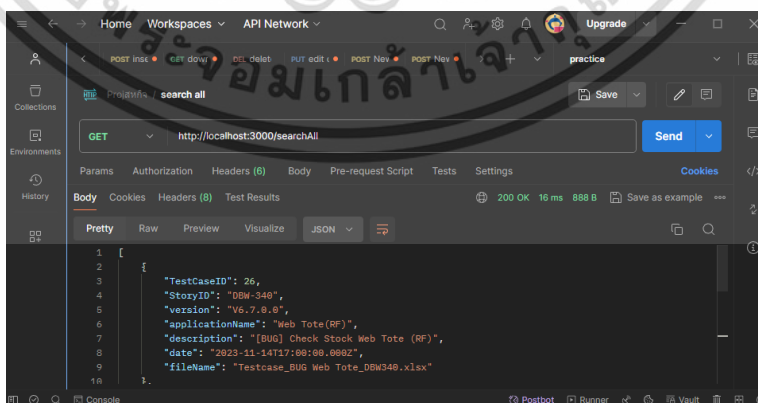
จिरะซอฟต์แวร์ เป็นเครื่องมือที่ใช้ในการจัดการโครงการและติดตามงาน โดยเฉพาะอย่างยิ่งในการติดตามปัญหาและการจัดการโครงการ โดย Jira Software มีความสามารถในการจัดการโครงการแบบ Agile และ Waterfall โดยสามารถติดตามงาน คำถาม และรายการงานอื่น ๆ ได้ นอกจากนี้ยังสามารถกำหนดค่า JIRA ให้ทำงานร่วมกับเครื่องมือการจัดการบริการจำนวนมาก ทำให้เป็นเครื่องมืออเนกประสงค์สำหรับการจัดการงาน

ซึ่งสหกิจศึกษาได้มีการศึกษาการทำงานและความสามารถของ Jira Software เพื่อนำข้อดีและข้อมูลที่สำคัญมาอ้างอิงเพื่อพัฒนาเว็บแอปพลิเคชันที่ตอบโจทย์ความต้องการของพนักงานฝ่าย QA ทำให้การพัฒนาโปรแกรมมีประสิทธิภาพมากขึ้น

2.14 Postman

เป็นเครื่องมือไว้ใช้สำหรับทดสอบ API Service ที่พัฒนาขึ้นโดยมีการจัดการทางด้าน ส่งข้อมูล Request และรับข้อมูล Response กลับมาได้ซึ่งจะมีวิธีการเรียก API Service ได้หลายวิธีประกอบด้วย GET, POST, PUT, DELETE, PATCH, HEAD ฯลฯ โดย Postman ยังมีความสามารถในการจัดการกับการรับส่งข้อมูลที่ซับซ้อนไม่ว่าจะเป็นการส่งข้อมูลผ่านทาง Headers, Body, Query Parameters หรือ Path Variables ทำให้สามารถจำลองสถานการณ์หลากหลายได้มากขึ้นและ Postman ยังมีความสามารถในการทำการเขียนโปรแกรมเพื่อทดสอบอัตโนมัติทำให้สามารถทดสอบความถูกต้องของ API ได้โดยอัตโนมัติที่ได้กำหนดเอาไว้

โดยการใช้งาน Postman นั้นใช้สำหรับการทดสอบ API ทั้งหมดในส่วนของการทำงานเซิร์ฟเวอร์ โดยตัวอย่างรูปที่ 2.8 เป็นการทดสอบ API ค้นหาข้อมูลและได้ผลลัพธ์เป็นข้อมูลภายในฐานข้อมูล



รูปที่ 2.8 การใช้งาน Postman เพื่อทดสอบ API ค้นหาข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

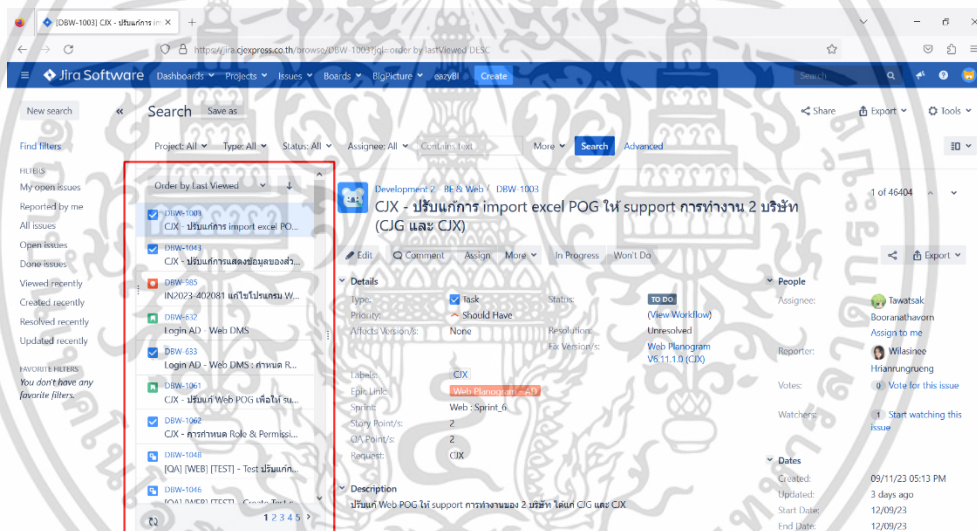
บทที่ 3

วิธีการดำเนินงาน

3.1 ขั้นตอนการดำเนินงาน

3.1.1 ศึกษาปัญหาและความต้องการของระบบ

ในการพัฒนาเว็บแอปพลิเคชันครั้งนี้ผู้พัฒนาได้ทำการศึกษาปัญหาที่เกิดขึ้นและความต้องการของระบบของฝ่าย QA (Quality Assurance) บริษัท ซี.เจ. เอ็กซ์เพรส เพื่อกำหนดขอบเขตของการพัฒนา ได้ทราบถึงปัญหาว่าระบบ Jira Software มีเครื่องมือสำหรับการค้นหาแต่ไม่ได้มีความเหมาะสมสำหรับการค้นหาไฟล์เอกสารสำหรับ QA เนื่องจากเครื่องมือที่ใช้ค้นหาไม่มีความซับซ้อนหลายขั้นตอน



รูปที่ 3.1 ตัวอย่างการค้นหาด้วย Jira Software

จากรูปที่ 3.1 แสดงให้เห็นว่าการค้นหานี้จำเป็นต้องกดไปยังหัวข้อรหัสต่างๆเพื่อดูรายละเอียดในแต่ละ StoryID หรือ TaskID เพื่อตรวจสอบอีกทีหนึ่งว่าเป็นข้อมูลที่ต้องการจริงหรือไม่ จากนั้นจึงทำการดาวน์โหลดไฟล์ที่ต้องการ จากสิ่งที่พบและจากความต้องการของพนักงาน จึงสามารถสรุปความต้องการของระบบว่าต้องการเว็บแอปพลิเคชันที่มีความสามารถในการ จัดเก็บ Test Case, ค้นหา Test Case, แก้ไข Test Case และลบข้อมูล Test Case ได้ผ่านบนเว็บเบราว์เซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 ศึกษาเครื่องมือที่ใช้ในการพัฒนาระบบ

ศึกษาเครื่องมือสำหรับการพัฒนาเซิร์ฟเวอร์แอปพลิเคชัน Node.js ด้วย Express Framework และศึกษาเครื่องมือที่ใช้ในส่วนของอินเทอร์เฟซผู้ใช้ HTML, CSS, JavaScript และ React (JavaScript Library) โดยมีไลบรารีอื่นๆเพื่อช่วยในการพัฒนา และ Postman สำหรับในการทดสอบระหว่างพัฒนาเซิร์ฟเวอร์แอปพลิเคชัน

3.1.3 ออกแบบระบบ

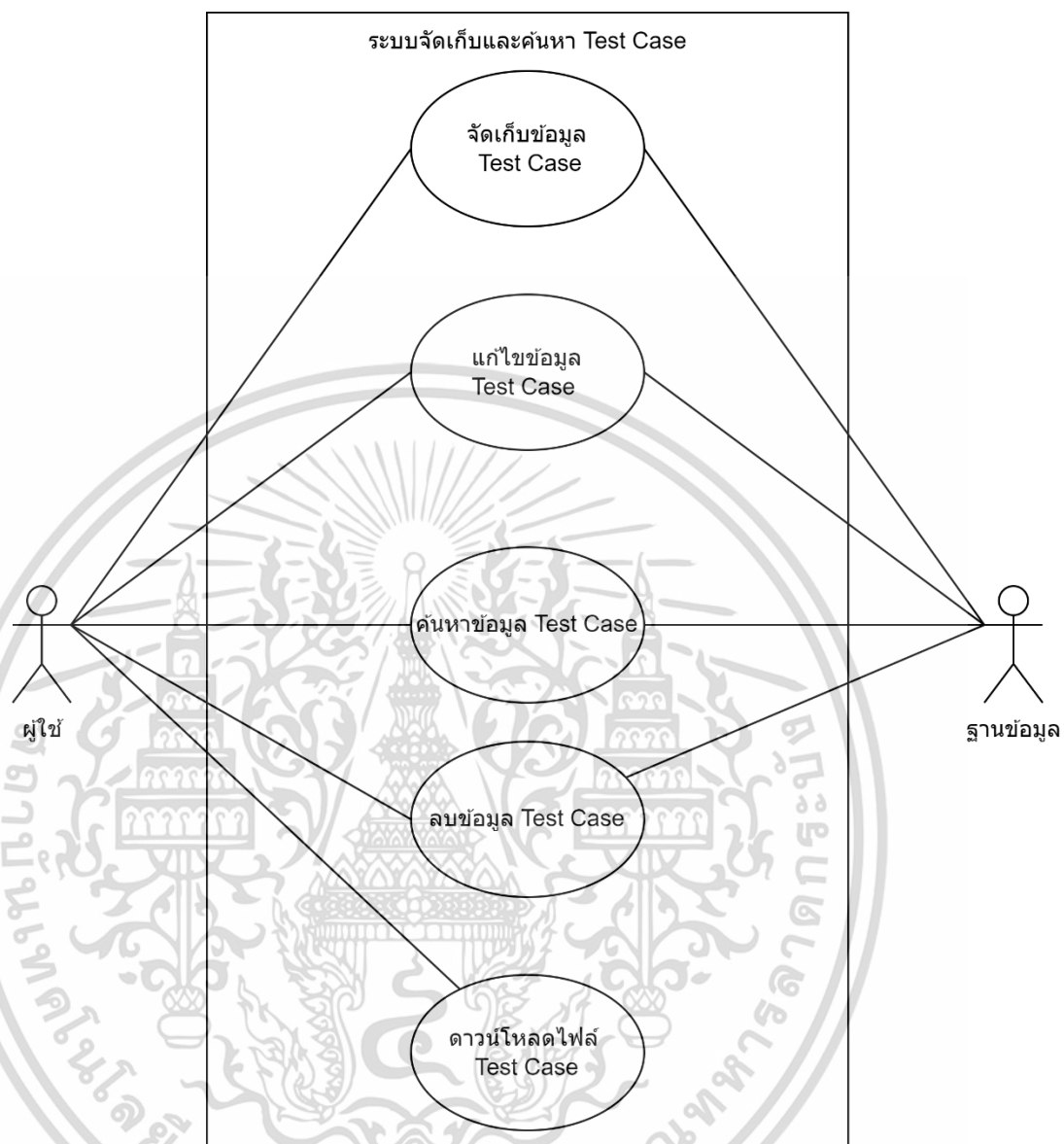
ผู้พัฒนาได้ทำการออกแบบ กรณีการใช้งาน(Use Case) แผนภาพการทำงานแบบลำดับ ปฏิสัมพันธ์ (Sequence Diagram) และแผนภาพจำลองข้อมูล (ER Diagram) เพื่อที่จะออกแบบการทำงานของโปรแกรมก่อนที่จะพัฒนาและทำให้การพัฒนาเป็นไปตามขอบเขตและแบบแผนตามลำดับ

3.2 การออกแบบระบบ

3.2.1 สร้างแผนภาพยูสเคส (Use Case Diagram)

แสดงแผนภาพของระบบจัดเก็บและค้นหา Test Case โดย User คือ ผู้ใช้งานและ Database คือ ฐานข้อมูลสำหรับจัดเก็บข้อมูล โดยผู้ใช้งานสามารถจัดเก็บ, ค้นหา, แก้ไขข้อมูล, ลบข้อมูล และ ดาวน์ โหลด Test Case ได้ โดยสามารถอธิบายการทำงานของระบบได้ดังนี้

- 1) ผู้ใช้งานสามารถจัดเก็บข้อมูลลงฐานข้อมูล
- 2) ผู้ใช้งานสามารถค้นหาข้อมูลจากฐานข้อมูล
- 3) ผู้ใช้งานสามารถแก้ไขข้อมูลที่มีอยู่ในฐานข้อมูลได้
- 4) ผู้ใช้งานสามารถลบข้อมูลที่มีอยู่ในฐานข้อมูล
- 5) ผู้ใช้งานสามารถดาวน์โหลดไฟล์ที่บันทึกไว้



รูปที่ 3.2 แผนภาพยูสเคสระบบจัดเก็บและค้นหา Test case

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.1 จัดเก็บข้อมูล Test Case

การจัดเก็บข้อมูล Test Case ใช้ในการรับข้อมูลเอกสารและไฟล์จากผู้ใช้เพื่อเก็บลงในฐานข้อมูล ไว้เพื่อใช้ในการค้นหาครั้งต่อไป

ตารางที่ 3.1 Use case Description ของ จัดเก็บข้อมูล Test Case

Use Case Name	จัดเก็บข้อมูล Test Case
Initiating Actor	ผู้ใช้งาน, Database
Actor's Goal	ผู้ใช้งานสามารถจัดเก็บข้อมูล Test Case
Participating Actors	-
Pre-condition	-
Post-condition	ข้อมูลและไฟล์ Test Case ถูกจัดเก็บ
Main flow	1.ผู้ใช้งานเข้าเว็บไซต์ 2.ผู้ใช้งานเข้าไปยังหน้าเพิ่ม Test Case 3.ผู้ใช้กรอกข้อมูลและไฟล์ 4.ผู้ใช้กดยืนยันจัดเก็บข้อมูล
Alternate Condition	กรณีที่มีชื่อไฟล์ซ้ำในฐานข้อมูลจะไม่สามารถจัดเก็บได้

3.2.1.2 ค้นหาข้อมูล Test Case

การค้นหาข้อมูล Test Case โดยผู้ใช้จะสามารถค้นหา ข้อมูล Test Case ที่ได้ทำการบันทึกไว้ในฐานข้อมูล

ตารางที่ 3.2 Use case Description ของ ค้นหาข้อมูล Test Case

Use Case Name	ค้นหาข้อมูล Test Case
Initiating Actor	ผู้ใช้งาน, Database
Actor's Goal	ผู้ใช้งานสามารถค้นหาข้อมูล Test Case
Participating Actors	-
Pre-condition	มีข้อมูล Test Case ในฐานข้อมูลและกดค้นหา
Post-condition	ผู้ใช้ได้รับข้อมูล Test Case

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Main flow	1.ผู้ใช้งานเข้าเว็บไซต์ 2.กดค้นหา Test Case
Alternate Condition	-

3.2.1.3 แก้ไขข้อมูล Test Case

การแก้ไขข้อมูล Test Case จะสามารถใช้งานได้หลังจากการค้นหาข้อมูลโดยผู้ใช้งานสามารถแก้ไขข้อมูลบางส่วนที่บันทึกไปแล้วในฐานข้อมูล

ตารางที่ 3.3 Use case Description ของ แก้ไขข้อมูล Test Case

Use Case Name	แก้ไขข้อมูล Test Case
Initiating Actor	ผู้ใช้งาน, Database
Actor's Goal	ผู้ใช้งานสามารถแก้ไขข้อมูล Test Case ที่มีอยู่
Participating Actors	-
Pre-condition	มีข้อมูล Test Case ในฐานข้อมูลและกดค้นหา
Post-condition	ข้อมูลถูกแก้ไขเป็นข้อมูลใหม่
Main flow	1.ผู้ใช้งานเข้าเว็บไซต์ 2.กดค้นหา Test Case 3.กดปุ่มแก้ไขในตารางข้อมูล
Alternate Condition	กรณีผู้ใช้แก้ไขข้อมูลเป็นค่าว่างจะไม่สามารถแก้ไขข้อมูลได้

3.2.1.4 ลบข้อมูล Test Case

การลบข้อมูล Test Case จะสามารถใช้งานได้หลังจากการค้นหาข้อมูลโดยผู้ใช้งานสามารถลบข้อมูลในฐานข้อมูลและไฟล์ที่บันทึกไว้ในเซิร์ฟเวอร์ได้

ตารางที่ 3.4 Use case Description ของ ลบข้อมูล Test Case

Use Case Name	ลบข้อมูล Test Case
Initiating Actor	ผู้ใช้งาน, Database
Actor's Goal	ผู้ใช้งานสามารถลบข้อมูล Test Case ที่มีอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Participating Actors	-
Pre-condition	มีข้อมูล Test Case ในฐานข้อมูลและกดค้นหา
Post-condition	ข้อมูลถูกลบ
Main flow	1.ผู้ใช้งานเข้าเว็บไซต์ 2.กดค้นหา Test Case 3.กดปุ่มลบข้อมูล
Alternate Condition	-

3.2.1.5 ดาวน์โหลดไฟล์ Test Case

การดาวน์โหลดไฟล์ Test Case จะสามารถใช้งานได้หลังจากการค้นหาข้อมูลโดยผู้ใช้งานทำการ download ไฟล์ที่ได้บันทึกไว้ในเซิร์ฟเวอร์

ตารางที่ 3.5 Use case Description ของ ดาวน์โหลดไฟล์ Test Case

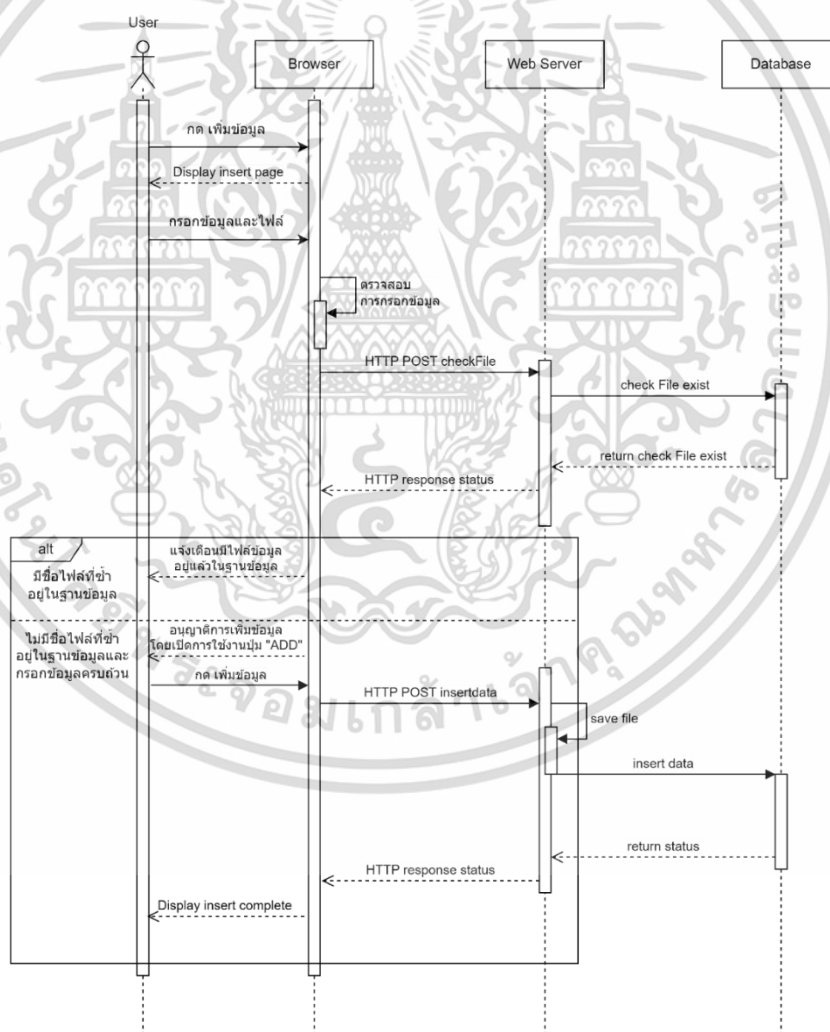
Use Case Name	ดาวน์โหลดไฟล์ Test Case
Initiating Actor	ผู้ใช้งาน
Actor's Goal	ผู้ใช้งานสามารถดาวน์โหลด Test Case ที่มีอยู่
Participating Actors	-
Pre-condition	มีข้อมูล Test Case ในฐานข้อมูลและกดค้นหา
Post-condition	ผู้ใช้งานได้รับไฟล์ Test Case
Main flow	1.ผู้ใช้งานเข้าเว็บไซต์ 2.กดค้นหา Test Case 3.กดปุ่มดาวน์โหลด
Alternate Condition	-

3.2.2 สร้างแผนภาพการทำงานแบบลำดับปฏิสัมพันธ์ (Sequence Diagram)

แผนงานการทำงานแบบลำดับปฏิสัมพันธ์ (Sequence Diagram) จะแสดงภาพของการทำงานของระบบโดยแผนภาพจะอธิบายลำดับเหตุการณ์ว่าเกิดขึ้นอย่างไร มีลำดับการทำงานอย่างไรในแต่ละส่วน

3.2.2.1 จัดเก็บข้อมูล Test Case

รูปที่ 3.3 แสดงรายละเอียดและความสัมพันธ์ของการทำงานของระบบการจัดเก็บข้อมูล Test Case โดยผู้ใช้งานจะต้องกดปุ่มเพิ่มข้อมูลเพื่อไปยังหน้าเพิ่มข้อมูล โดยหน้าเพิ่มข้อมูลจะมีการตรวจสอบว่าข้อมูลกรอกครบและไฟล์ไม่ซ้ำจึงจะอนุญาตให้เพิ่มข้อมูลได้ โดยในขั้นตอนที่ ผู้ใช้(User) ได้เลือกไฟล์จะมีการตรวจสอบว่ามีชื่อไฟล์ที่ซ้ำอยู่ในฐานข้อมูลหรือไม่หากมีข้อมูลอยู่แล้วจะแสดงแจ้งเตือนว่ามีไฟล์อยู่แล้วและไม่อนุญาตให้เพิ่มข้อมูล เมื่อผู้ใช้กรอกข้อมูลและไฟล์ที่ไม่ซ้ำในฐานข้อมูลเสร็จสิ้นระบบจะอนุญาตให้สามารถเพิ่มข้อมูลได้ ในขั้นตอนการเพิ่มข้อมูลจะเรียกใช้ API ในการส่งข้อมูลและไฟล์ไปบันทึกในฝั่งของเซิร์ฟเวอร์ และฐานข้อมูลเมื่อมีการบันทึกเสร็จสิ้นจะมีการตอบกลับไปยังหน้าผู้ใช้(UI)เพื่อให้รับรู้ว่าบันทึกเสร็จสิ้นพร้อมกลับข้อมูลที่กรอกไว้ในช่องกรอกข้อมูลเพื่อรองรับข้อมูลใหม่ต่อไป

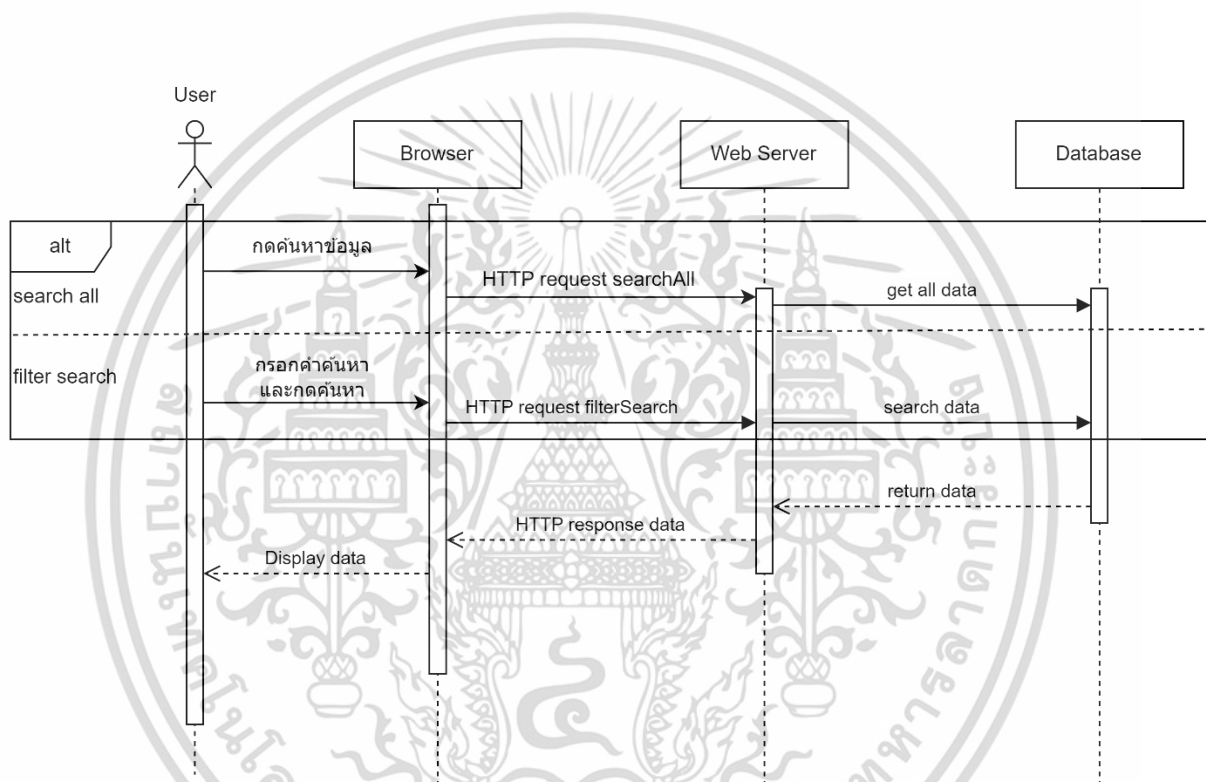


รูปที่ 3.3 Sequence Diagram ระบบจัดเก็บข้อมูล Test Case

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2.2 ค้นหา Test Case

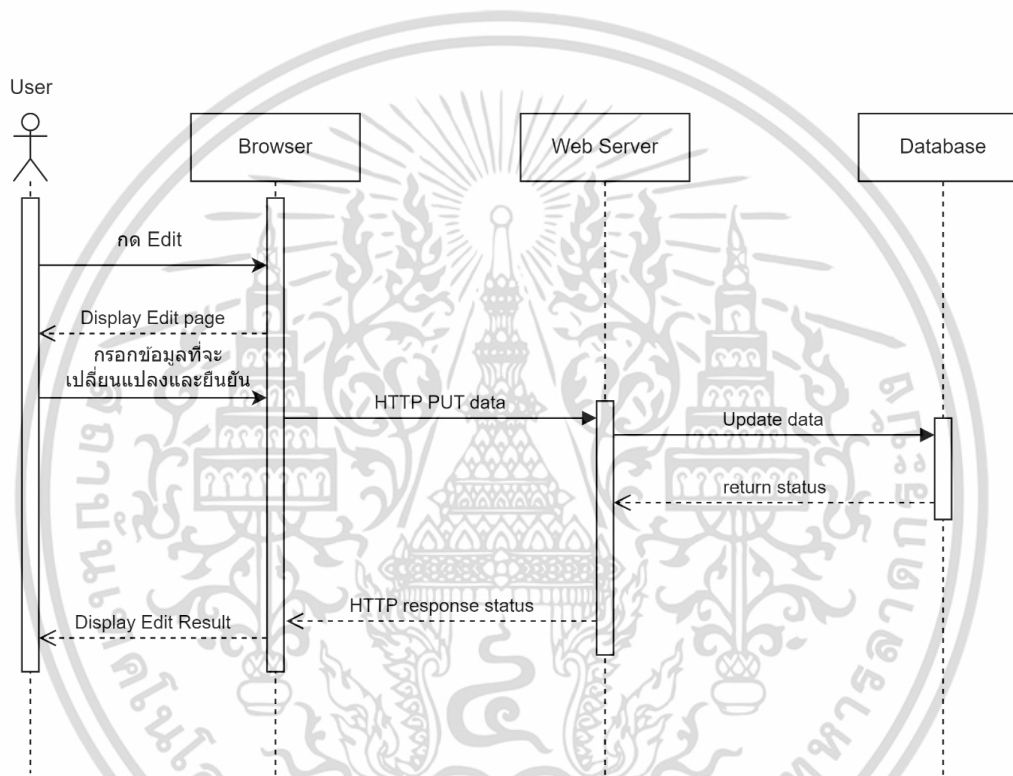
รูปที่ 3.4 แสดงรายละเอียดและความสัมพันธ์ของการทำงานของการทำงานของการค้นหา Test Case โดยผู้ใช้งาน (User) ทำการค้นหาข้อมูลโดยจะสามารถค้นหาข้อมูล Test Case ทั้งหมดหรือค้นหาด้วยการกรอกคำค้นหา โดยเริ่มต้นจากผู้โดยเริ่มต้นจากผู้ใช้ได้กดปุ่มค้นหาข้อมูลจากนั้นระบบจะทำการเรียกใช้ API ในการค้นหาข้อมูล จากนั้น Web Server จะค้นหาข้อมูลจาก Database(ฐานข้อมูล) แล้วส่งข้อมูลกลับมายัง Browser เพื่อแสดงผล



รูปที่ 3.4 Sequence Diagram ระบบค้นหา Test Case

3.2.2.3 แก้ไขข้อมูล Test Case

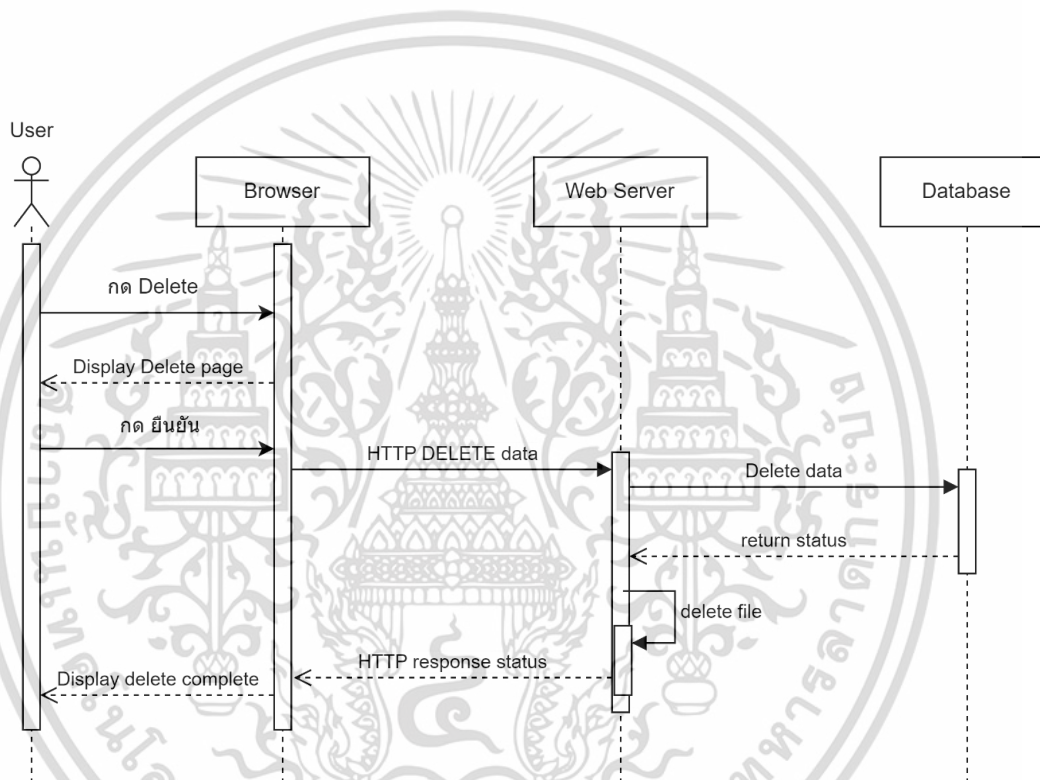
รูปที่ 3.5 แสดงรายละเอียดและความสัมพันธ์ของการทำงานของระบบการแก้ไขข้อมูล Test Case โดยผู้ใช้งาน (User) จะต่อผ่านการค้นหาและมีข้อมูลอยู่ในระบบก่อนจึงจะสามารถ แก้ไขข้อมูลที่มีอยู่ได้โดยเมื่อผู้ใช้กดที่ปุ่ม Edit(แก้ไข) จะทำการแสดงหน้าต่างไว้สำหรับแก้ไขข้อมูลที่มีอยู่เมื่อแก้ไขเสร็จสิ้น จะใช้งาน API ในการส่งข้อมูลไปส่วน Web Server เพื่อนำไปแก้ไขข้อมูลใน Database(ฐานข้อมูล) เมื่อแก้ไขสำเร็จจะส่งข้อมูลกลับไป browser เพื่อให้ผู้ใช้ ทราบถึงการแก้ไขเสร็จสิ้น



รูปที่ 3.5 Sequence Diagram ระบบแก้ไขข้อมูล Test Case

3.2.2.4 ลบข้อมูล Test Case

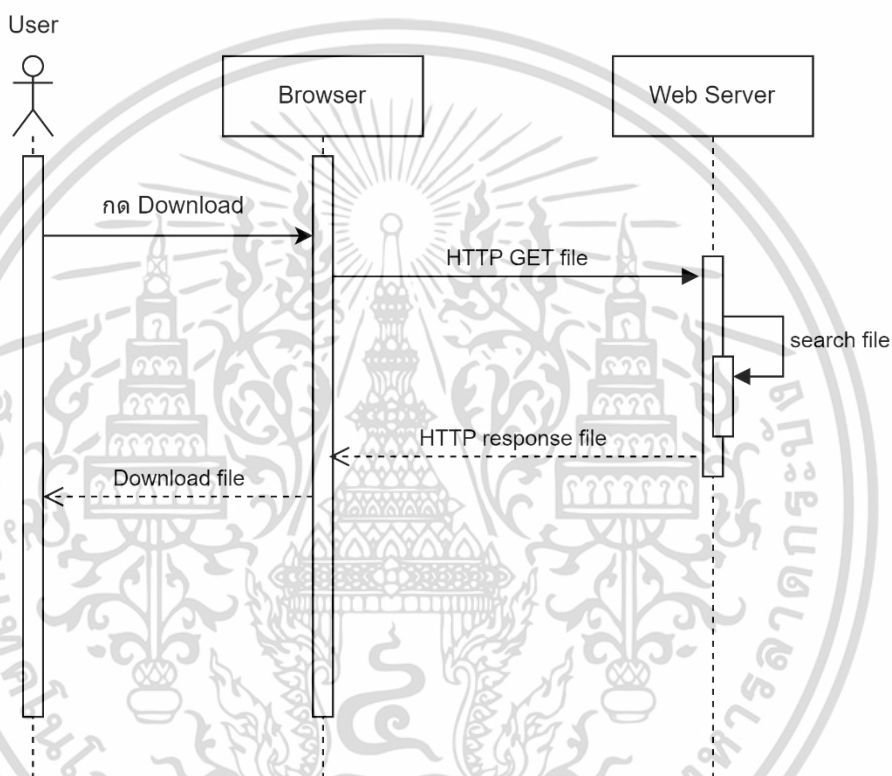
รูปที่ 3.6 แสดงรายละเอียดและความสัมพันธ์ของการทำงานของระบบการลบข้อมูล Test Case โดยผู้ใช้งาน (User) จะต้องผ่านการค้นหาและมีข้อมูลอยู่ในระบบก่อนจึงจะสามารถ ลบข้อมูลที่มีอยู่ได้โดยเมื่อผู้ใช้กดที่ปุ่ม Edit(ลบ) จะทำการแสดงหน้าต่างไว้สำหรับลบข้อมูลที่มีอยู่เมื่อลบเสร็จสิ้น จะใช้งาน API ในการลบข้อมูลจาก Web Server โดยจะมีการส่งข้อมูล TestCaseID เพื่อนำไปลบข้อมูลใน Database (ฐานข้อมูล) เมื่อลบข้อมูลเสร็จสิ้นจะส่งข้อมูลไปยัง Browser เพื่อให้ผู้ใช้ทราบถึงการลบข้อมูลเสร็จสิ้น



รูปที่ 3.6 Sequence Diagram ระบบลบข้อมูล Test Case

3.2.2.5 ดาวน์โหลดไฟล์ Test Case

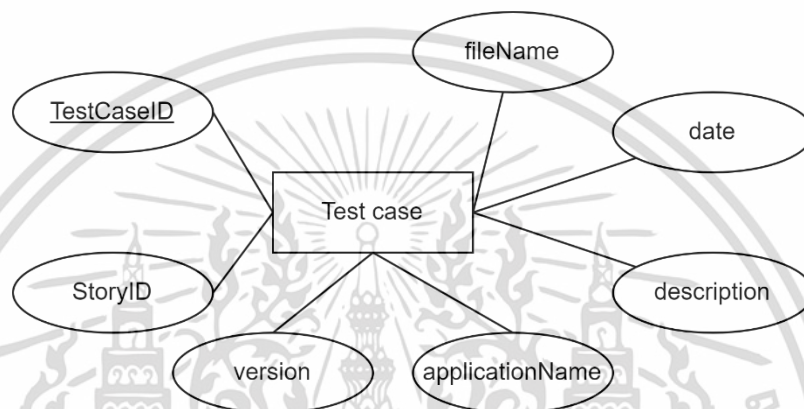
รูป 3.7 แสดงรายละเอียดและความสัมพันธ์ของการทำงานของกรณีทดสอบการดาวน์โหลดไฟล์ Test Case โดยผู้ใช้งาน(User) จะต้องผ่านการค้นหาและมีข้อมูลอยู่ในระบบก่อนจึงจะสามารถดาวน์โหลดไฟล์ Test Case ที่มีอยู่ได้โดยเมื่อผู้ใช้กดที่ปุ่ม Download(ดาวน์โหลด) จะทำการเรียกใช้ API ไปยัง Web Server เพื่อนำไฟล์ส่งกลับมายัง browser จากนั้นผู้ใช้งานจะนำไฟล์ไปใช้งานต่อไป



รูปที่ 3.7 Sequence Diagram ระบบดาวน์โหลดไฟล์ Test Case

3.2.3 สร้างแผนภาพจำลองข้อมูลด้วย Entity Relationship Diagram

โดยเว็บแอปพลิเคชันมีการออกแบบการจัดเก็บข้อมูลแบบระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database) โดยจะประกอบด้วยหัวข้อข้อมูลที่สำคัญในการค้นหาที่ได้มีการตกลงไว้ตามความต้องการ



รูปที่ 3.8 Entity Relationship Diagram ระบบจัดเก็บและค้นหา Test case

ตารางที่ 3.6 Entity Relationship Diagram Description ของระบบจัดเก็บและค้นหา Test case

Name	Data Type	Comment	Key
TestCaseID	int	รหัสกรณีทดสอบ	PK
StoryID	Varchar(45)	รหัสงานใน Jira	
version	Varchar(45)	เวอร์ชันของแอปพลิเคชัน	
applicationName	Varchar(255)	ชื่อของแอปพลิเคชัน	
description	Varchar(255)	ข้อความอธิบายเพิ่มเติม	
date	datetime	วันที่บันทึก	
fileName	Varchar(255)	ชื่อไฟล์สำหรับอ้างอิง	

บทที่ 4

ผลการดำเนินงาน

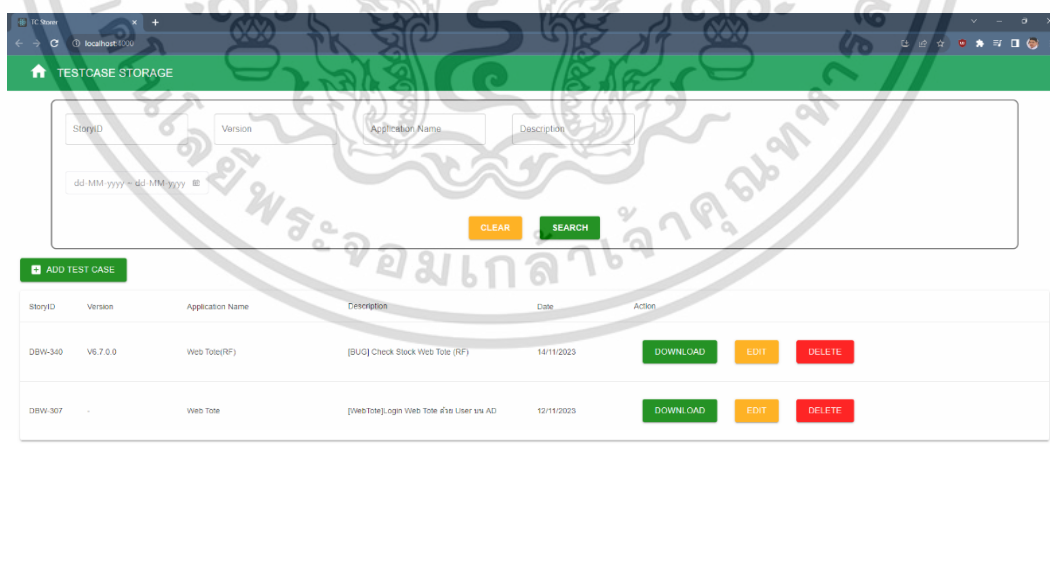
ในการพัฒนาครั้งนี้ผู้จัดทำได้ออกแบบหน้าจอปฏิสัมพันธ์ของระบบเพื่อรองรับการทำงานของระบบจัดเก็บและค้นหา Test Case และอีกทั้งยังช่วยเพิ่มประสิทธิภาพในการค้นหาและจัดเก็บ

4.1 หน้าจอปฏิสัมพันธ์ของระบบ (User Interface)

โดยหน้าจอแสดงผลส่วนติดต่อกับผู้ใช้สามารถแบ่งเป็น 4 ส่วนดังนี้

4.1.1 หน้าค้นหาข้อมูล Test Case

โดยหน้าแรกของเว็บไซต์เป็นหน้าสำหรับค้นหาข้อมูล Test Case โดยจะมีองค์ประกอบหลัก 3 ส่วนประกอบไปด้วย 1.ส่วนการค้นหาข้อมูล 2.ปุ่มสำหรับไปยังหน้าเพิ่มข้อมูล 3.ตารางแสดงผลพร้อมปุ่มกด ส่วนค้นหาข้อมูลประกอบไปด้วย Textfield สำหรับค้นหาด้วยตัวแปรต่างๆประกอบไปด้วย StoryID, Version, Application Name, Description และช่วงเวลาที่ยืนยันได้ ถัดมาด้านล่างของส่วนการค้นหาข้อมูลคือปุ่มสำหรับไปยังหน้าเพิ่มข้อมูลที่จะทำหน้าที่พาไปยังหน้าเพิ่มข้อมูล ด้านล่างสุดจะเป็นส่วนแสดงตารางแสดงผลการค้นหา



รูปที่ 4.1 หน้าค้นหาข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 หน้าจัดเก็บข้อมูล Test Case

หน้าสำหรับจัดเก็บข้อมูล Test Case จะมีช่องใช้สำหรับกรอกข้อมูลและปุ่มสำหรับเลือกไฟล์ โดยผู้ใช้จำเป็นต้องกรอกข้อมูลและไฟล์ที่ไม่ซ้ำในฐานข้อมูลและนามสกุลไฟล์เป็น .xlsx .xls เมื่อกรอกข้อมูลเสร็จสิ้นปุ่มเพิ่มข้อมูลจะเปิดให้สามารถทำงานได้

The screenshot shows a web browser window with the URL 'localhost:8000/mgmt'. The page title is 'TESTCASE STORAGE'. The main content is a form titled 'Add Test Case'. The form contains the following fields:

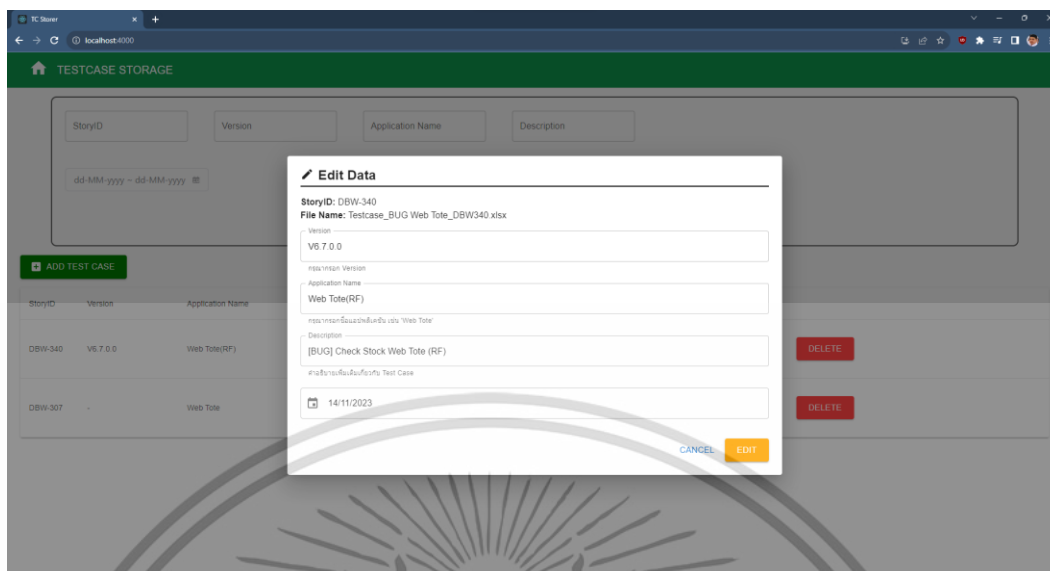
- StoryID (with a small text 'กรุณากรอก StoryID' below it)
- Version (with a small text 'กรุณากรอก Version' below it)
- Application Name (with a small text 'กรุณากรอกชื่อและนามสกุลไฟล์ของ Test Case' below it)
- Description (with a small text 'กรุณากรอกชื่อและนามสกุลไฟล์ของ Test Case' below it)

 Below the Description field is a file upload section with a placeholder '00/00/00000000' and a blue button labeled 'UPLOAD EXCEL FILE'. Underneath is a 'Selected file:' label. At the bottom right of the form, there are two buttons: a red 'BACK' button and a grey 'ADD' button.

รูปที่ 4.2 หน้าจัดเก็บข้อมูล

4.1.3 หน้าแก้ไขข้อมูล Test Case

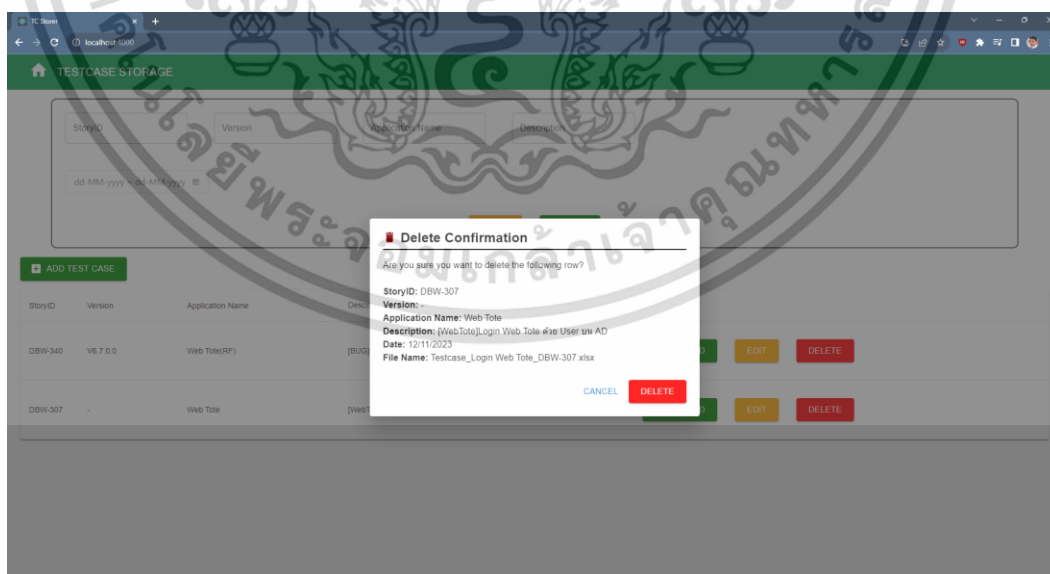
หน้าแก้ไขข้อมูลสามารถเปิดหลังจากค้นหาข้อมูลในหน้าค้นหาข้อมูล โดยกดที่ปุ่ม “EDIT” ในตารางแสดงข้อมูล หน้าแก้ไขข้อมูลจะถูกเปิดขึ้นมาพร้อมกับข้อมูลในแถวที่เลือก โดยสามารถแก้ไข Version, Application Name, Description และวันที่บันทึก เมื่อผู้ใช้กดปุ่ม แก้ไข(Edit) ก็จะมีการส่งข้อมูล API เพื่อแก้ไขข้อมูลจากนั้นจะปิดหน้าแก้ไขข้อมูลโดยอัตโนมัติ



รูปที่ 4.3 หน้าแก้ไขข้อมูล

4.1.4 หน้าลบข้อมูล Test Case

หน้าลบข้อมูลสามารถเปิดหลังจากค้นหาข้อมูลในหน้าค้นหาข้อมูล โดยกดที่ปุ่ม “DELETE” ในตารางแสดงข้อมูล หน้าลบข้อมูลจะถูกเปิดขึ้นมาพร้อมกับแสดงข้อมูลในแถวที่เลือก เพื่อให้ผู้ใช้ตรวจสอบการลบอีกครั้งหนึ่งก่อนที่ผู้ใช้จะกดปุ่ม Delete เพื่อส่ง API ในการลบข้อมูล



รูปที่ 4.4 หน้าลบข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลการทดสอบโปรแกรม

ผลการทดสอบโปรแกรมถูกแบ่งเป็น 2 ส่วนดังนี้

4.2.1 ด้านการทำงาน

ในส่วนการทดสอบโปรแกรมผู้จัดทำได้ทำการทดสอบในรูปแบบ Manual Test ในรูปแบบของบริษัท CJ Express ซึ่งเป็นการทดสอบโดยการใช้งานโปรแกรมโดยตรง เพื่อทดสอบความถูกต้องของการทำงานของโปรแกรม โดยการทดสอบนี้จะครอบคลุมการทดสอบทุกฟังก์ชันการทำงานของโปรแกรม ประกอบไปด้วยการทดสอบการทำงาน API การทดสอบฟังก์ชันการทำงานหลัก การทดสอบการทำงานของฟังก์ชันย่อย และการแสดงผลของส่วนต่อประสานกับผู้ใช้ (User Interface)

โดยความหมายในแต่ละคอลัมน์ของกรณีทดสอบมีดังนี้

ID หมายถึงรหัสการทดสอบ

Positive/Negative หมายถึงรูปแบบการงานเงื่อนไขซึ่ง Positive เป็นการวางเงื่อนไขโดยคาดหวังว่าโปรแกรมจะทำงานอย่างถูกต้องและ Negative เป็นเป็นการวางเงื่อนไขโดยคาดหวังว่าโปรแกรมจะทำงานไม่ถูกต้อง

Test Condition หมายถึงหัวข้อหรือสถานการณ์ที่ต้องการทดสอบ

Description(Scenario Description) หมายถึงการอธิบายจุดที่ต้องการตรวจสอบหรือทดสอบและการวางเงื่อนไขเพิ่มเติม

Step หมายถึงวิธีการทดสอบโดยจะอธิบายการทดสอบในแต่ละขั้นตอน

TestData หมายถึงข้อมูลหรือสิ่งที่ต้องจัดเตรียมไว้เพื่อใช้ในการทดสอบ

Expected Result หมายถึงผลที่คาดหวังไว้ล่วงหน้าโดยอ้างอิงจากความต้องการ(Requirement)

Actual Result หมายถึงผลที่เกิดขึ้นจากการทดสอบ

Status หมายถึงผลลัพธ์การทดสอบหากผลที่คาดหวังไม่ตรงกับผลที่เกิดขึ้นจะมีสถานะเป็น Fail และหากตรงกันจะได้สถานะเป็น True

จากรูปที่ 4.5 เป็นเอกสารการทดสอบเพื่อการทดสอบเว็บเซิร์ฟเวอร์ด้วย API ในส่วนของการค้นหาข้อมูล Test Case และ การจัดเก็บข้อมูล โดยผลลัพธ์ที่เกิดขึ้นเป็นไปตามผลลัพธ์ที่คาดหวังทั้งหมดโดยไม่มีข้อผิดพลาด

โครงการ :	TCStore(API)							
ผู้จัดเตรียม:	Chayodom Panitcharoen							
ผู้ทดสอบ	Chayodom Panitcharoen							
วันที่เริ่มทดสอบ :	13/11/2023							
วันที่สิ้นสุดการทดสอบ :	21/11/2023							
Environment :	DEV							
Version :	-							
ID	Positive/Negative	Test condition	Description (Scenario Description)	Step	TestData	Expected Result	Actual Result	Status
1.ค้นหาข้อมูล Test Case								
TC API 01	Positive	การค้นหาข้อมูลทั้งหมด (/searchAll)	ตรวจสอบการทำงาน API	1.input URL in postman 2.send request(GET)	Url : http://192.168.0.5:3000/searchAll	ระบบจะแสดงข้อมูลทั้งหมดในฐานข้อมูล	ระบบจะแสดงข้อมูลทั้งหมดในฐานข้อมูล	Pass
TC API 02	Negative	การค้นหาข้อมูลทั้งหมด (/searchAll)	ตรวจสอบการทำงาน API กรณีใส่ข้อมูลผิดรูปแบบ	1.input URL in postman 2.send request(GET)	Url : http://192.168.0.5:3000/searchAll	ระบบส่งข้อมูลกลับ Status 400	ระบบส่งข้อมูลกลับ Status 400	Pass
TC API 03	Positive	การค้นหาข้อมูลด้วยคำค้นหา (/filterSearch)	ตรวจสอบการทำงาน API	1.input URL in postman 2.input data 3.send request(GET)	Url : http://192.168.0.5:3000/filterSearch Data : {ApplicationName:web}	ระบบจะแสดงข้อมูลด้วยคำที่ค้นหา	ระบบจะแสดงข้อมูลด้วยคำที่ค้นหา	Pass
TC API 04	Positive	การค้นหาข้อมูลด้วยคำค้นหา (/filterSearch)	ตรวจสอบการทำงาน API กรณีใส่ข้อมูลมากกว่า 1 ตัวแปร	1.input URL in postman 2.input data 3.send request(GET)	Url : http://192.168.0.5:3000/filterSearch Data : {ApplicationName:web, StoryID:SDW}	ระบบจะแสดงข้อมูลด้วยคำที่ค้นหา	ระบบจะแสดงข้อมูลด้วยคำที่ค้นหา	Pass
TC API 05	Negative	การค้นหาข้อมูลด้วยคำค้นหา (/filterSearch)	ตรวจสอบการทำงาน API กรณีใส่ข้อมูลผิดรูปแบบ	1.input URL in postman 2.input data 3.send request(GET)	Url : http://192.168.0.5:3000/filterSearch	ระบบส่งข้อมูลกลับ Status 400	ระบบส่งข้อมูลกลับ Status 400	Pass
2.จัดเก็บข้อมูล Test Case								
TC API 06	Positive	จัดเก็บข้อมูล Test Case (/insert)	ตรวจสอบการจัดเก็บข้อมูล	1.input URL in postman 2.input data 3.send request(POST)	Url : http://192.168.0.5:3000/insertpage/insert Data : {"StoryID": "TEST-001", "version": "V2", "applicationName": "Test", "description": "Test insert", "date": "2023-07-26T00:00:00.000Z", "fileName": "Test.txt" }	ระบบบันทึกข้อมูลลงฐานข้อมูล	ระบบบันทึกข้อมูลลงฐานข้อมูล	Pass
TC API 07	Positive	จัดเก็บข้อมูล Test Case (/insert)	ตรวจสอบการจัดเก็บข้อมูล กรณีข้อมูลไม่ครบ	1.input URL in postman 2.input data 3.send request(POST)	Url : http://192.168.0.5:3000/insertpage/insert Data : {"StoryID": "TEST-001" }	ระบบส่งข้อมูลกลับ Status 400 และข้อความ'Bad Request: Incomplete data'	ระบบส่งข้อมูลกลับ Status 400 และข้อความ'Bad Request: Incomplete data'	Pass
TC API 08	Positive	จัดเก็บข้อมูล Test Case (/insert)	ตรวจสอบการจัดเก็บข้อมูล กรณีใส่ข้อมูลผิดรูปแบบ	1.input URL in postman 2.input data 3.send request(POST)	Url : http://192.168.0.5:3000/insertpage/insert	ระบบส่งข้อมูลกลับ Status 400	ระบบส่งข้อมูลกลับ Status 400	Pass

รูปที่ 4.5 ผลการทดสอบ API (ID 1-8)

จากรูปที่ 4.6 เป็นเอกสารการทดสอบเพื่อการทดสอบเว็บเซิร์ฟเวอร์ด้วย API ในส่วนของการจัดเก็บไฟล์ Test Case การแก้ไขข้อมูล Test Case และการลบข้อมูล Test Case โดยผลลัพธ์ที่เกิดขึ้นเป็นไปตามผลลัพธ์ที่คาดหวังทั้งหมดโดยไม่มีข้อผิดพลาด

3.จัดเก็บไฟล์ Test Case								
TC API 09	Positive	จัดเก็บข้อมูล Test Case (/insert)	ตรวจสอบการจัดเก็บไฟล์	1.input URL in postman 2.input file 3.send request(POST)	Url : http://192.168.0.5:3000/insertFile	ระบบบันทึกไฟล์	ระบบบันทึกไฟล์	Pass
TC API 10	Positive	จัดเก็บข้อมูล Test Case (/insert)	ตรวจสอบการจัดเก็บไฟล์กรณีนามสกุลไฟล์ไม่ใช่ .xlsx , .xls	1.input URL in postman 2.input file 3.send request(POST)	Url : http://192.168.0.5:3000/insertFile	ระบบไม่บันทึกไฟล์	ระบบไม่บันทึกไฟล์	Pass
TC API 11	Positive	จัดเก็บข้อมูล Test Case (/insert)	ตรวจสอบการจัดเก็บไฟล์กรณีส่งข้อมูลผิดรูปแบบ	1.input URL in postman 2.input file 3.send request(POST)	Url : http://192.168.0.5:3000/insertFile	ระบบส่งข้อมูลกลับ Status 400	ระบบส่งข้อมูลกลับ Status 400	Pass
4.แก้ไขข้อมูล Test Case								
TC API 12	Positive	แก้ไขข้อมูล Test Case (/update)	ตรวจสอบการแก้ไขข้อมูล	1.input URL in postman 2.input data 3.send request(PUT)	Url : http://192.168.0.5:3000/editpage/update	ระบบได้แก้ไขข้อมูลในฐานข้อมูลอย่างถูกต้อง	ระบบได้แก้ไขข้อมูลในฐานข้อมูลอย่างถูกต้อง	Pass
TC API 13	Negative	แก้ไขข้อมูล Test Case (/update)	ตรวจสอบการแก้ไขข้อมูลกรณีส่งข้อมูลผิดรูปแบบ	1.input URL in postman 2.input data 3.send request(PUT)	Url : http://192.168.0.5:3000/editpage/update	ระบบส่งข้อมูลกลับ Status 400	ระบบส่งข้อมูลกลับ Status 400	Pass
5.ลบข้อมูล Test Case								
TC API 14	Positive	ลบข้อมูล Test Case (/delete)	ตรวจสอบการลบข้อมูล	1.input URL in postman 2.input data 3.send request(DELETE)	Url : http://192.168.0.5:3000/deletepage/deleteData:{"TestCaseID": 36, fileName: 'test Case.xlsx'}	ข้อมูลในฐานข้อมูลและไฟล์ถูกลบ	ข้อมูลในฐานข้อมูลและไฟล์ถูกลบ	Pass
TC API 15	Negative	ลบข้อมูล Test Case (/delete)	ตรวจสอบการแก้ไขข้อมูลกรณีข้อมูลไม่มี TestCaseID	1.input URL in postman 2.input data 3.send request(DELETE)	Url : http://192.168.0.5:3000/deletepage/deleteData:{"}	ระบบส่งข้อมูลกลับ Status 400 และข้อความ"TestCaseID and fileName Required"	ระบบส่งข้อมูลกลับ Status 400 และข้อความ"TestCaseID and fileName Required"	Pass
TC API 16	Negative	ลบข้อมูล Test Case (/delete)	ตรวจสอบการแก้ไขข้อมูลกรณีส่งข้อมูลผิดรูปแบบ	1.input URL in postman 2.input data 3.send request(DELETE)	Url : http://192.168.0.5:3000/deletepage/delete	ระบบส่งข้อมูลกลับ Status 400	ระบบส่งข้อมูลกลับ Status 400	Pass

รูปที่ 4.6 ผลการทดสอบ API (ID 9-16)

จากรูปที่ 4.7 เป็นเอกสารการทดสอบเพื่อการทดสอบเว็บเซิร์ฟเวอร์ด้วย API ในส่วนของการดาวน์โหลดไฟล์ Test Case และ การตรวจสอบไฟล์

6.ดาวน์โหลดไฟล์ Test Case								
TC API 17	Positive	ดาวน์โหลดไฟล์ Test Case (/download)	ตรวจสอบไฟล์จากการดาวน์โหลด	1.input URL in postman 2.input data 3.send request(GET)	Url : http://192.168.0.5:3000/download Data:{fileName: 'test Case.xlsx'}	ได้รับไฟล์ในการดอกลับ	ได้รับไฟล์ในการดอกลับ	Pass
TC API 18	Negative	ดาวน์โหลดไฟล์ Test Case (/download)	ตรวจสอบไฟล์จากการดาวน์โหลด กรณี ไม่มีไฟล์ในฐานข้อมูล	1.input URL in postman 2.input data 3.send request(GET)	Url : http://192.168.0.5:3000/download Data:{fileName:'test.tx t'}	มีการแจ้งเตือน "File not found." (Status:404)	มีการแจ้งเตือน "File not found." (Status:404)	Pass
TC API 19	Negative	ดาวน์โหลดไฟล์ Test Case (/download)	ตรวจสอบไฟล์จากการดาวน์โหลด กรณี ส่งข้อมูลผิดรูปแบบ	1.input URL in postman 2.input data 3.send request(GET)	Url : http://192.168.0.5:3000/download	ระบบส่งข้อมูลกลับ Status 400	ระบบส่งข้อมูลกลับ Status 400	Pass
7.ตรวจสอบไฟล์								
TC API 20	Positive	ตรวจสอบไฟล์ Test Case (/checkfile)	ตรวจสอบการตรวจสอบไฟล์ กรณี ไม่มีชื่อไฟล์ซ้ำ	1.input URL in postman 2.input data 3.send request(POST)	Url : http://192.168.0.5:3000/download Data:{fileName: 'test Case2.xlsx'}	ได้รับข้อความ able to insert และ Status:200	ได้รับข้อความ able to insert และ Status:200	Pass
TC API 21	Positive	ตรวจสอบไฟล์ Test Case (/checkfile)	ตรวจสอบการตรวจสอบไฟล์ กรณี มีชื่อไฟล์ซ้ำ	1.input URL in postman 2.input data 3.send request(POST)	Url : http://192.168.0.5:3000/download Data:{fileName: 'test Case.xlsx'}	ได้รับข้อความ this file already exists และ Status:403	ได้รับข้อความ this file already exists และ Status:403	Pass
TC API 22	Positive	ตรวจสอบไฟล์ Test Case (/checkfile)	ตรวจสอบการตรวจสอบไฟล์ กรณี ส่งข้อมูลผิดรูปแบบ	1.input URL in postman 2.input data 3.send request(POST)	Url : http://192.168.0.5:3000/checkfile	ระบบส่งข้อมูลกลับ Status 400	ระบบส่งข้อมูลกลับ Status 400	Pass

รูปที่ 4.7 ผลการทดสอบ API (ID 17-22)

จากรูปที่ 4.8 เป็นเอกสารการทดสอบผ่านส่วนติดต่อผู้ใช้งานใช้โดยจะเป็นการทดสอบทั้งระบบตั้งแต่ในส่วนของการติดต่อผู้ใช้งานจนถึงเว็บเซิร์ฟเวอร์ โดยได้ทดสอบในส่วน การค้นหาข้อมูล Test Case และการจัดเก็บข้อมูล Test Case

ID	Positive/Negative	Test condition	Description (Scenario Description)	Step	TestData	Expected Result	Actual Result	Status
1.ค้นหาข้อมูล Test Case								
TC 01	Positive	การค้นหาข้อมูล Test Case	ตรวจสอบผลลัพธ์การค้นหากรณี ค้นหาโดยไม่กรอกคำค้นหา	1.open URL 2.Click Search	Url : http://192.168.0.5:4000	ระบบจะแสดงข้อมูลทั้งหมดในฐานข้อมูล	ระบบจะแสดงข้อมูลทั้งหมดในฐานข้อมูล	Pass
TC 02	Positive	การค้นหาข้อมูล Test Case	ตรวจสอบผลลัพธ์การค้นหากรณี ค้นหาโดยกรอกคำค้นหา 1 ตัวแปร	1.open URL 2.Input search data 3.Click Search	Url : http://192.168.0.5:4000	ระบบจะแสดงข้อมูลทั้งหมดในฐานข้อมูลตามคำค้นหาที่กำหนดไว้	ระบบจะแสดงข้อมูลทั้งหมดในฐานข้อมูลตามคำค้นหาที่กำหนดไว้	Pass
TC 03	Positive	การค้นหาข้อมูล Test Case	ตรวจสอบผลลัพธ์การค้นหากรณี ค้นหาโดยกรอกคำค้นหา มากกว่า 1 ตัวแปร	1.open URL 2.Input search data 3.Click Search	Url : http://192.168.0.5:4000	ระบบจะแสดงข้อมูลทั้งหมดในฐานข้อมูลตามคำค้นหาที่กำหนดไว้	ระบบจะแสดงข้อมูลทั้งหมดในฐานข้อมูลตามคำค้นหาที่กำหนดไว้	Pass
TC 04	Negative	การค้นหาข้อมูล Test Case	ตรวจสอบผลลัพธ์การค้นหากรณี ค้นหาโดยกรอกคำค้นหาที่ไม่มีข้อมูลในฐานข้อมูล	1.open URL 2.Input search data 3.Click Search	Url : http://192.168.0.5:4000	ระบบจะไม่แสดงข้อมูลเนื่องจากไม่พบข้อมูล	ระบบจะไม่แสดงข้อมูลเนื่องจากไม่พบข้อมูล	Pass
2.จัดเก็บข้อมูล Test Case								
TC 05	Positive	จัดเก็บข้อมูล Test Case	ตรวจสอบการจัดเก็บข้อมูล	1.open URL 2.Click Insert Test Case 3.Input Test Case data 4.Input Test Case File 5.Click Add	Url : http://192.168.0.5:4000	ระบบบันทึกข้อมูลลงฐานข้อมูลและมีการเก็บไฟล์ใน server	ระบบบันทึกข้อมูลลงฐานข้อมูลและมีการเก็บไฟล์ใน server	Pass
TC 06	Negative	จัดเก็บข้อมูล Test Case	ตรวจสอบการจัดเก็บข้อมูลกรณี ไฟล์เข้ากับข้อมูลที่มีอยู่ในระบบ	1.open URL 2.Click Insert Test Case 3.Input Test Case data 4.Input Test Case File	Url : http://192.168.0.5:4000	ระบบจะไม่อนุญาตให้เพิ่มข้อมูลได้และมีความแจ้งเตือน	ระบบจะไม่อนุญาตให้เพิ่มข้อมูลได้และมีความแจ้งเตือน	Pass
TC 07	Negative	จัดเก็บข้อมูล Test Case	ตรวจสอบการจัดเก็บข้อมูลกรณี นามสกุลไฟล์ไม่ใช่ .xlsx .xls	1.open URL 2.Click Insert Test Case 3.Input Test Case data 4.Input Test Case File	Url : http://192.168.0.5:4000	ระบบจะไม่อนุญาตให้เลือกไฟล์	ระบบจะไม่อนุญาตให้เลือกไฟล์	Pass
TC 08	Negative	จัดเก็บข้อมูล Test Case	ตรวจสอบการจัดเก็บข้อมูลกรณี กรอกข้อมูลไม่ครบ	1.open URL 2.Click Insert Test Case	Url : http://192.168.0.5:4000	ระบบจะไม่อนุญาตให้เพิ่มข้อมูลได้	ระบบจะไม่อนุญาตให้เพิ่มข้อมูลได้	Pass

รูปที่ 4.8 ผลการทดสอบการทำงาน (ID 1-8)

จากรูปที่ 4.8 เป็นเอกสารการทดสอบผ่านส่วนติดต่อผู้ใช้งานใช้โดยจะเป็นการทดสอบทั้งระบบตั้งแต่ในส่วนของการติดต่อผู้ใช้งานจนถึงเว็บเซิร์ฟเวอร์ โดยได้ทดสอบในส่วน การแก้ไขข้อมูล Test Case การลบข้อมูล Test Case การดาวน์โหลดไฟล์ และการตรวจสอบการทำงานเพิ่มเติมของส่วนติดต่อผู้ใช้งาน

3.แก้ไขข้อมูล Test Case								
TC 09	Positive	แก้ไขข้อมูล Test Case	ตรวจสอบการแก้ไขข้อมูล	1.open URL 2.Click Search 3.Click Edit 4.Edit data 5.Click EDIT	Url : http://192.168.0.5:4000	ระบบได้แก้ไขข้อมูลในฐานข้อมูลอย่างถูกต้อง	ระบบได้แก้ไขข้อมูลในฐานข้อมูลอย่างถูกต้อง	Pass
TC 10	Negative	แก้ไขข้อมูล Test Case	ตรวจสอบการแก้ไขข้อมูลกรณี ลบข้อมูลที่มีอยู่เป็นค่าว่าง	1.open URL 2.Click Search 3.Click Edit 4.Delete data 5.Click EDIT	Url : http://192.168.0.5:4000	ระบบไม่อนุญาตให้ยืนยันแก้ไขข้อมูล	ระบบไม่อนุญาตให้ยืนยันแก้ไขข้อมูล	Pass
TC 11	Negative	แก้ไขข้อมูล Test Case	ตรวจสอบการแก้ไขข้อมูลกรณี แก้ไขข้อมูลแล้วกดยกเลิก	1.open URL 2.Click Search 3.Click Edit 4.Edit data 5.Click Cancel	Url : http://192.168.0.5:4000	จะไม่เกิดการแก้ไขข้อมูลและ ปิด popup edit	จะไม่เกิดการแก้ไขข้อมูลและ ปิด popup edit	Pass
4.ลบข้อมูล Test Case								
TC 12	Positive	ลบข้อมูล Test Case	ตรวจสอบการลบข้อมูล	1.open URL 2.Click Search 3.Click DELETE 4.Click DELETE	Url : http://192.168.0.5:4000	ข้อมูลในฐานข้อมูลและไฟล์ถูกลบ	ข้อมูลในฐานข้อมูลและไฟล์ถูกลบ	Pass
TC 13	Negative	ลบข้อมูล Test Case	ตรวจสอบการลบข้อมูลกรณี ลบข้อมูลแล้วกดยกเลิก	1.open URL 2.Click Search 3.Click DELETE 4.Click Cancel	Url : http://192.168.0.5:4000	จะไม่เกิดการลบข้อมูลและปิด popup delete	จะไม่เกิดการลบข้อมูลและปิด popup delete	Pass
4.ดาวน์โหลดไฟล์ Test Case								
TC 14	Positive	ดาวน์โหลดไฟล์ Test Case	ตรวจสอบไฟล์จากการดาวน์โหลด	1.open URL 2.Click Search 3.Click Download	Url : http://192.168.0.5:4000	ผู้ใช้งานได้รับไฟล์ถูกต้องตามที่เก็บไว้	ผู้ใช้งานได้รับไฟล์ถูกต้องตามที่เก็บไว้	Pass
TC 15	Negative	ดาวน์โหลดไฟล์ Test Case	ตรวจสอบไฟล์จากการดาวน์โหลดกรณี ไม่มีไฟล์ในฐานข้อมูล	1.open URL 2.Click Search 3.Click Download	Url : http://192.168.0.5:4000	มีการแจ้งเตือน "File not found."	มีการแจ้งเตือน "File not found."	Pass
4.ตรวจสอบ User Interface								
TC 16	Negative	หน้าแก้ไขข้อมูล	ตรวจสอบการแสดงผลข้อมูลในหน้าแก้ไขข้อมูล	1.open URL 2.Click Search 3.Click Edit	Url : http://192.168.0.5:4000	มีการแสดงหน้าแก้ไขข้อมูลที่มีข้อมูลที่เลือกครบถ้วน	มีการแสดงหน้าแก้ไขข้อมูลที่มีข้อมูลที่เลือกครบถ้วน	Pass
TC 17	Negative	หน้าลบข้อมูล	ตรวจสอบการแสดงผลข้อมูลในหน้าลบข้อมูล	1.open URL 2.Click Search 3.Click DELETE	Url : http://192.168.0.5:4000	มีการแสดงหน้าลบข้อมูลที่มีข้อมูลที่เลือกครบถ้วน	มีการแสดงหน้าลบข้อมูลที่มีข้อมูลที่เลือกครบถ้วน	Pass
TC 18	Negative	ตารางแสดงข้อมูล	ตรวจสอบการจัดเรียง Column	1.open URL 2.Click Search 3.Click table sort	Url : http://192.168.0.5:4000	มีการจัดเรียง Column ที่เลือกได้ถูกต้อง	มีการจัดเรียง Column ที่เลือกได้ถูกต้อง	Pass

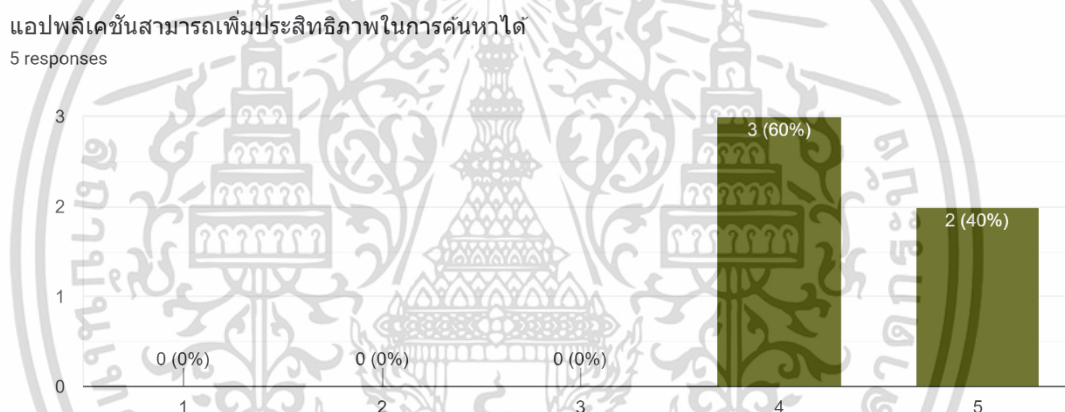
รูปที่ 4.9 ผลการทดสอบการทำงาน (ID 9-18)

4.2.2 ด้านความพึงพอใจของการใช้งาน

ผู้จัดทำได้ทำแบบสอบถามความพึงพอใจของการใช้งานเว็บแอปพลิเคชันต่อผู้ใช้งานที่เป็นผู้ทดสอบซอฟต์แวร์ของบริษัท ซี.เจ. เอ็กซ์เพรส เป็นจำนวน 5 คน ผ่านแบบสอบถามออนไลน์ของ Google form ใช้การให้คะแนนจาก 5 ถึง 1 โดยมีรายละเอียดผลการทดสอบดังนี้

1)แอปพลิเคชันสามารถเพิ่มประสิทธิภาพในการค้นหาได้

จากรูป 4.10 แสดงผลแบบสอบถามในหัวข้อ "แอปพลิเคชันสามารถเพิ่มประสิทธิภาพในการค้นหาได้" จากผลการตอบแบบสอบถามนี้สามารถสรุปได้ว่า ผู้ทดสอบซอฟต์แวร์ส่วนใหญ่มีความพึงพอใจในระดับดี(4 คะแนน) จำนวน 3 คน และดีมาก(5 คะแนน) จำนวน 2 คน ต่อความสามารถของแอปพลิเคชันในการเพิ่มประสิทธิภาพในการค้นหา



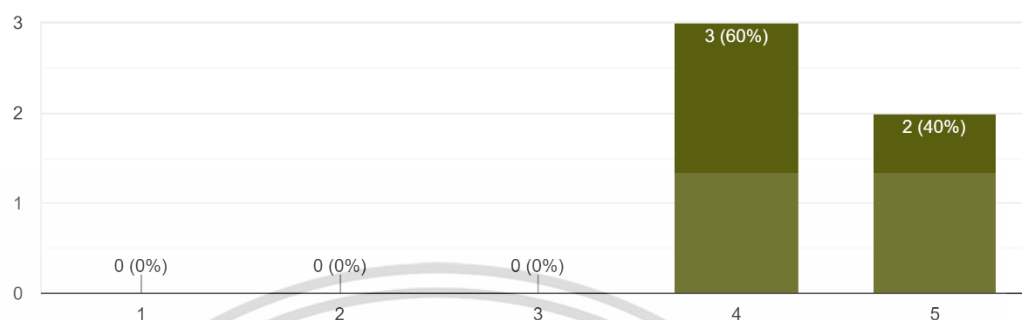
รูปที่ 4.10 ผลแบบสอบถามหัวข้อ”แอปพลิเคชันสามารถเพิ่มประสิทธิภาพในการค้นหาได้”

2)แอปพลิเคชันสามารถจัดเก็บข้อมูล Test Case ได้อย่างมีประสิทธิภาพ

จากรูป 4.11 แสดงผลแบบสอบถามในหัวข้อ ”แอปพลิเคชันสามารถจัดเก็บข้อมูล Test Case ได้อย่างมีประสิทธิภาพ” จากผลการตอบแบบสอบถามนี้สามารถสรุปได้ว่า ผู้ทดสอบซอฟต์แวร์ส่วนใหญ่มีความพึงพอใจในระดับดี(4 คะแนน) จำนวน 3 คน และดีมาก(5 คะแนน) จำนวน 2 คนต่อความสามารถของแอปพลิเคชันในการเพิ่มประสิทธิภาพในการจัดเก็บ

แอปพลิเคชันสามารถจัดเก็บข้อมูล Test Case ได้อย่างมีประสิทธิภาพ

5 responses



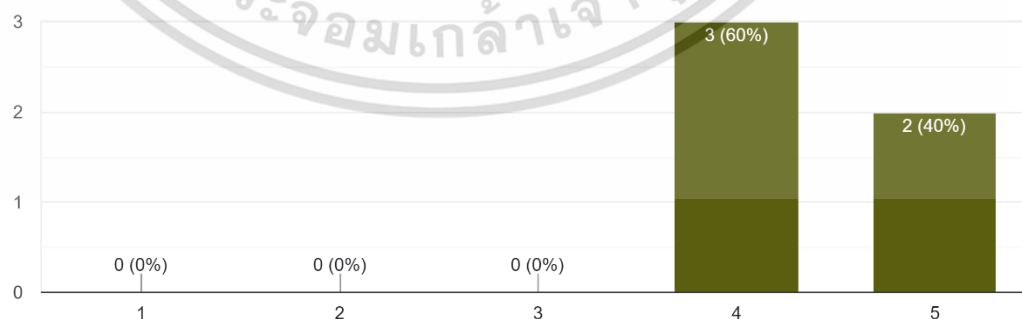
รูปที่ 4.11 ผลแบบสอบถามหัวข้อ”แอปพลิเคชันสามารถจัดเก็บข้อมูล Test Case ได้อย่างมีประสิทธิภาพ”

3)แอปพลิเคชันสามารถดาวน์โหลดไฟล์ได้อย่างมีประสิทธิภาพ

จากรูป 4.12 แสดงผลแบบสอบถามในหัวข้อ ”แอปพลิเคชันสามารถดาวน์โหลดไฟล์ได้อย่างมีประสิทธิภาพ” จากผลการตอบแบบสอบถามนี้สามารถสรุปได้ว่า ผู้ทดสอบซอฟต์แวร์ส่วนใหญ่มีความพึงพอใจในระดับดี(4 คะแนน) จำนวน 3 คน และดีมาก(5 คะแนน) จำนวน 2 คนต่อความสามารถของแอปพลิเคชันในการเพิ่มประสิทธิภาพในการดาวน์โหลดไฟล์

แอปพลิเคชันสามารถดาวน์โหลดไฟล์ได้อย่างมีประสิทธิภาพ

5 responses



รูปที่ 4.12 ผลแบบสอบถามหัวข้อ”แอปพลิเคชันสามารถดาวน์โหลดไฟล์ได้อย่างมีประสิทธิภาพ”

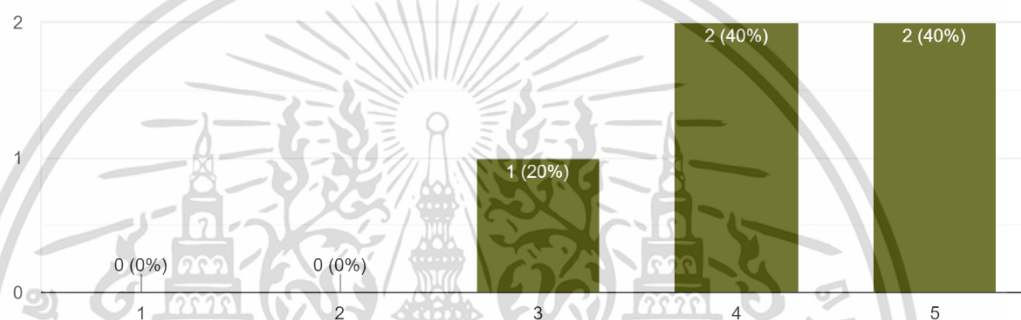
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4)แอปพลิเคชันสามารถแก้ไขข้อมูล Test Case ได้อย่างมีประสิทธิภาพ

จากรูป 4.13 แสดงผลแบบสอบถามในหัวข้อ ”แอปพลิเคชันสามารถแก้ไขข้อมูล Test Case ได้อย่างมีประสิทธิภาพ” จากผลการตอบแบบสอบถามนี้สามารถสรุปได้ว่า ผู้ทดสอบซอฟต์แวร์ส่วนใหญ่มีความพึงพอใจในระดับดี(4 คะแนน) จำนวน 2 คน และดีมาก(5 คะแนน) จำนวน 2 คนปานกลาง(2 คะแนน) จำนวน 1 คน

แอปพลิเคชันสามารถแก้ไขข้อมูล Test Case ได้อย่างมีประสิทธิภาพ

5 responses



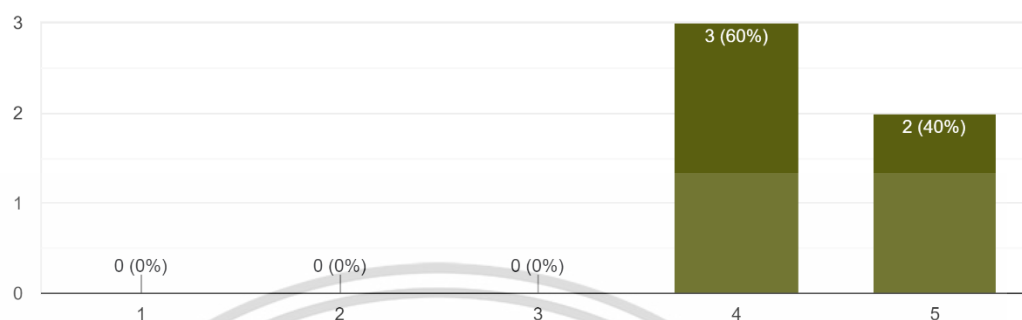
รูปที่ 4.13 ผลแบบสอบถามหัวข้อ”แอปพลิเคชันสามารถแก้ไขข้อมูล Test Case ได้อย่างมีประสิทธิภาพ”

5)แอปพลิเคชันสามารถลบ Test Case ได้อย่างมีประสิทธิภาพ

จากรูป 4.14 แสดงผลแบบสอบถามในหัวข้อ ”แอปพลิเคชันสามารถลบ Test Case ได้อย่างมีประสิทธิภาพ” จากผลการตอบแบบสอบถามนี้สามารถสรุปได้ว่าผู้ทดสอบซอฟต์แวร์ส่วนใหญ่มีความพึงพอใจในระดับดี(4 คะแนน) จำนวน 3 คน และดีมาก(5 คะแนน) จำนวน 2 คน

แอปพลิเคชันสามารถลบ Test Case ได้อย่างมีประสิทธิภาพ

5 responses



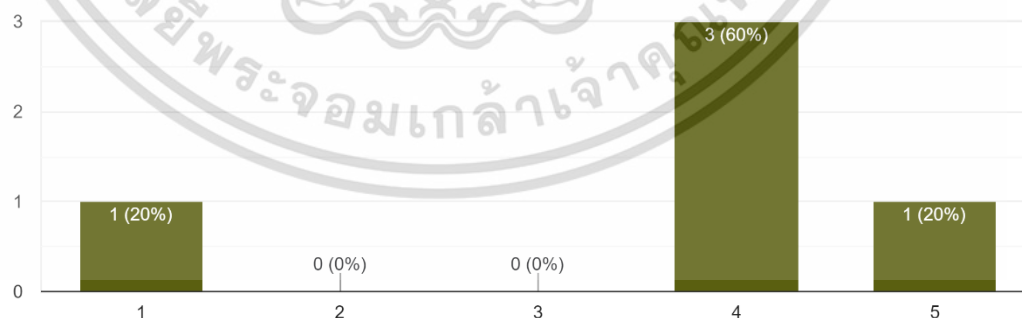
รูปที่ 4.14 ผลแบบสอบถามหัวข้อ”แอปพลิเคชันสามารถลบ Test Case ได้อย่างมีประสิทธิภาพ”

6)ส่วนต่อประสานกับผู้ใช้ (User Interface) สามารถเข้าใจได้ง่ายและสะดวกต่อการใช้งาน

จากรูป 4.15 แสดงผลแบบสอบถามในหัวข้อ ”ส่วนต่อประสานกับผู้ใช้ (User Interface) สามารถเข้าใจได้ง่ายและสะดวกต่อการใช้งาน” จากผลการตอบแบบสอบถามนี้สามารถสรุปได้ว่าผู้ทดสอบซอฟต์แวร์ส่วนใหญ่มีความพึงพอใจในระดับดี(4 คะแนน) จำนวน 3 คน ระดับดีมาก(5 คะแนน) จำนวน 1 คนและแย่(1 คะแนน) จำนวน 1 คน

ส่วนต่อประสานกับผู้ใช้ (User Interface) สามารถเข้าใจได้ง่ายและสะดวกต่อการใช้งาน

5 responses



รูปที่ 4.15 ผลแบบสอบถามหัวข้อ”ส่วนต่อประสานกับผู้ใช้ (User Interface) สามารถเข้าใจได้ง่ายและสะดวกต่อการใช้งาน”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) ข้อคิดเห็นและข้อเสนอแนะเพิ่มเติม

จากรูป 4.16 แสดงผลแบบสอบถามในหัวข้อ “ข้อคิดเห็นและข้อเสนอแนะเพิ่มเติม” จากผลการตอบแบบสอบถามนี้สามารถสรุปได้ว่า ผู้ทดสอบซอฟต์แวร์ต้องการเพิ่มความชัดเจนของตัวอักษรในส่วนต่อประสานกับผู้ใช้ การกรอกข้อมูลโดยอัตโนมัติในหน้าจัดเก็บข้อมูลกรณีทดสอบ การแสดงส่วนที่เปลี่ยนแปลงและการแก้ไขไฟล์เอกสารในหน้าแก้ไขข้อมูล

ข้อคิดเห็นและข้อเสนอแนะเพิ่มเติม

2 responses

อยากให้แยกชื่อ text box ให้ชัดเจน หรือทำให้เป็นตัวหนา ตอนนี้ก็เหมือนกับข้อความด้านล่างเดือนให้กรอก อยากให้ข้อความใต้ text box แสดงขึ้นมาตอน validate filed เช่น ถ้าไม่กรอก story ID แล้วกด ที่ช่องว่างให้เดือนเป็นข้อความสีแดง กรอก story ID

- ควรเพิ่มการแก้ไขไฟล์ testcase
- ควรเพิ่มการแสดง Modified ของข้อมูล
- วันที่ควร Auto Gen ให้เลย ถ้าเป็นวันที่ create

รูปที่ 4.16 ข้อคิดเห็นและข้อเสนอแนะเพิ่มเติม

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

5.1.1 ผลการดำเนินงาน

จากการพัฒนาเว็บแอปพลิเคชันจัดเก็บและค้นหากรณีทดสอบเพื่อนำมาเพิ่มประสิทธิภาพในขั้นตอนบันทึกเอกสารกรณีทดสอบที่สำคัญและการค้นหา สามารถสรุปผลการดำเนินงานได้ดังนี้

- 1) สามารถพัฒนาเว็บแอปพลิเคชันเพื่อใช้ในการช่วยจัดเก็บ ค้นหา แก้ไขและลบ Test Case ตามที่ได้ออกแบบไว้และช่วยลดเวลาในการค้นหา Test Case ในขั้นตอนการทดสอบโปรแกรมของพนักงานบริษัท ซี.เจ. เอ็กซ์เพรส
- 2) ได้ศึกษาสภาพแวดล้อมการทำงานจริงภายในบริษัท และได้ฝึกฝนการแก้ปัญหาที่เกิดขึ้นระหว่างการดำเนินงานในเวลาจำกัด

5.1.2 ผลการทดสอบเว็บไซต์

ในการทดสอบการทำงานของเว็บแอปพลิเคชันจัดเก็บและค้นหากรณีทดสอบผู้จัดทำได้ทำการทดสอบแต่ละส่วนของระบบด้วยวิธีการทดสอบด้วยมือ (Manual test) เช่นการทดสอบผลลัพธ์จากการค้นหาข้อมูล หรือการทดสอบการเพิ่มข้อมูล จากการทดสอบการทำงานได้สามารถสรุปผลได้ดังนี้

- 1) จากการทดสอบเว็บแอปพลิเคชันจัดเก็บและค้นหากรณีทดสอบพบว่าสามารถทำงานได้อย่างถูกต้องโดยไม่มีข้อผิดพลาด
- 2) การแสดงผลหรือการแสดงผลข้อมูลต่างๆของส่วนของการติดต่อผู้ใช้งาน ได้ถูกต้องตามที่ผู้ใช้งานและผู้พัฒนาได้ตกลงไว้

ในการทดสอบด้านความพึงพอใจของการใช้งานที่มีต่อระบบนั้นได้ผลลัพธ์ว่า ผู้ใช้งานส่วนใหญ่มีความพึงพอใจกับระบบจัดเก็บและค้นหาและหน้าส่วนประสานงานกับผู้ใช้ที่พัฒนาขึ้น โดยในมุมมองของ QA บริษัท ซี.เจ. เอ็กซ์เพรส นั้นพบว่าการใช้เว็บแอปพลิเคชันจัดเก็บและค้นหากรณีทดสอบ ช่วยให้สามารถจัดเก็บ ค้นหา แก้ไข และ ลบข้อมูลกรณีทดสอบได้อย่างมีประสิทธิภาพ

ทำให้ช่วยลดเวลาในการค้นหากรณีทดสอบในกระบวนการทดสอบซอฟต์แวร์ ในบางส่วนของหน้าส่วนประสานงานกับผู้ใช้มีการตอบสนองต่อผู้ใช้งานไม่มากพอ ทางผู้จัดทำได้ทำการวิเคราะห์ถึงปัญหาที่เกิดขึ้นและได้ทำการแก้ไขโปรแกรมได้เสร็จสิ้นตามความต้องการของผู้ใช้งาน

5.2 ข้อเสนอแนะในการพัฒนา

1) การเพิ่มความสามารถในการค้นหา ในปัจจุบันเว็บแอปพลิเคชันใช้การค้นหาแบบ LIKE ซึ่งอาจจะไม่สามารถรองรับการค้นหาที่มีความใกล้เคียงหรือพิมพ์ผิดได้ดีพอ ดังนั้นอาจจะมีการพัฒนาเว็บแอปพลิเคชันให้มีการค้นหาแบบคลุมเครือ(fuzzy search) ซึ่งจะช่วยรองรับการค้นหาที่มีความใกล้เคียงหรือพิมพ์ผิดได้ดีขึ้นซึ่งจะเป็นการช่วยเพิ่มประสิทธิภาพในการค้นหา

2) การเพิ่มความสามารถในการทำงานร่วมกับระบบอื่นๆในอนาคตเช่นระบบของบริษัท ซี.เจ. เอ็กซ์เพรสและ ระบบของ Jira Software โดยจะทำการดึงข้อมูลและไฟล์เพื่อให้ทำการจัดเก็บเป็นแบบอัตโนมัติหรือ ทำด้วยมือน้อยลงซึ่งจะช่วยลดเวลาให้ลดเวลาในการทำการจัดเก็บได้ดีขึ้น

เอกสารอ้างอิง

- amazon web services, 2023. JavaScript (JS) คืออะไร. [Online]. เข้าถึงได้จาก <https://aws.amazon.com/th/what-is/javascript/>
- amazon web services, 2023. SQL และ MySQL แตกต่างกันอย่างไรร. [Online]. เข้าถึงได้จาก <https://aws.amazon.com/th/compare/the-difference-between-sql-and-mysql/>
- aofleejay, 2017. สร้าง RESTful API ด้วย Express — Express 101. [Online]. เข้าถึงได้จาก <https://medium.com/@aofleejay/สร้าง-restful-api-ด้วย-express-express-101-ee37cc4952b4>
- Axios, 2020. What is Axios?. [Online]. เข้าถึงได้จาก <https://axios-http.com/docs/intro>
- BorntoDev, 2023. มาลองใช้ MUI ขึ้นหน้าเว็บบน React ให้ง่ายกว่าเดิมกัน. [Online]. เข้าถึงได้จาก <https://www.youtube.com/watch?v=DddmMf8D000>
- designil, 2020. React คืออะไร? ไขข้อสงสัยสำหรับมือใหม่ + แนวทางการหัด React ตั้งแต่เริ่มต้น. [Online]. เข้าถึงได้จาก <https://www.designil.com/react-คืออะไร/>
- KongRuksiam Official, 2021. พัฒนาเว็บด้วย Node.js & Express.js | สำหรับผู้เริ่มต้น. [Online]. เข้าถึงได้จาก <https://youtu.be/mDezAkh5gcE?si=LXfKlj4kGYVqgg62>
- Mawin, 2023. การใช้งาน Axios ในการเรียก API. [Online]. เข้าถึงได้จาก <https://www.borntodev.com/2023/11/03/การใช้งาน-axios-ในการเรียก-api-🍌/>
- mindphp.com, 2022. HTML คืออะไร เอชทีเอ็มแอล ภาษาคอมพิวเตอร์ที่ใช้ในการสร้างเว็บเพจ ใช้เขียนโปรแกรม ย่อมาจากอะไร. [Online]. เข้าถึงได้จาก <https://www.mindphp.com/คู่มือ/73-คืออะไร/2026-html-คืออะไร.html>
- Narongkon Kongprasert, 2019. CRUD คืออะไร. [Online]. เข้าถึงได้จาก <https://medium.com/@narongkonkongprasert/crud-คืออะไร-a897aa76ac7d>
- Nattapong Nakchun, 2020. React คืออะไร มาทำความเข้าใจกันเถอะ!. [Online]. เข้าถึงได้จาก <https://medium.com/@ome504/react-คืออะไร-มาทำความเข้าใจกันเถอะ-84f9417141a3>
- Sakul Montha, 2019. Jira คืออะไร แล้ว Epic, Story, Task และ Sub-Task ต่างกันอย่างไร. [Online]. เข้าถึงได้จาก <https://iamgique.medium.com/jira-คืออะไร-แล้ว-epic-story-task-และ-sub-task-ต่างกันอย่างไร-a928d10bece0>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-techupth, 2023. รู้จักกับ React เครื่องมือจำเป็นสำหรับ Front-end Developer. [Online]. เข้าถึงได้จาก <https://www.techupth.com/articles/react>

-Thapanon S, 2021. Node JS กับ Sequelize 101. [Online]. เข้าถึงได้จาก <https://www.borntodev.com/2021/07/02/node-js-กับ-sequelize-101/>

-Waerohanee Waehayee, 2019. Postman คืออะไร. [Online]. เข้าถึงได้จาก <https://medium.com/@waerohaneewaehayee/postman-คืออะไร-c6461461cc55>

-William, 2024. Web Application Architecture: The Latest Guide 2024. [Online]. เข้าถึงได้จาก <https://www.clickittech.com/devops/web-application-architecture/>

