



ระบบแนะนำเทคนิคการพัฒนาแอนิเมชันบนเว็บไซต์

โดย

นายพลภัฏฐ์ ตัณฑโอภาส

โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิทยาศาสตร์บัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์

ปีการศึกษา 2567

ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

ระบบแนะนำเทคนิคการพัฒนาแอนิเมชันบนเว็บไซต์

โดย

นายพลัฏฐ์ ตัณฑโอส

โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิทยาศาสตร์บัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์

ปีการศึกษา 2567

ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

Introduction to the development of frontend the web for efficiency

BY

MR Palat Tuntaopas

A FINAL-YEAR PROJECT REPORT SUBMITTED IN PARTIAL
FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
BACHELOR OF SCIENCE
COMPUTER SCIENCE
FACULTY OF SCIENCE AND TECHNOLOGY
THAMMASAT UNIVERSITY
ACADEMIC YEAR 2024
COPYRIGHT OF THAMMASAT UNIVERSITY

มหาวิทยาลัยธรรมศาสตร์
คณะวิทยาศาสตร์และเทคโนโลยี

รายงานโครงการพิเศษ

ของ

นายพลภัฏฐ์ ตัณฑโอภาส

เรื่อง

ระบบแนะนำเทคนิคการพัฒนาแอนิเมชันบนเว็บไซต์
ได้รับการตรวจสอบและอนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
เมื่อ วันที่ 30 พฤษภาคม พ.ศ. 2568

อาจารย์ที่ปรึกษา และ กรรมการสอบ

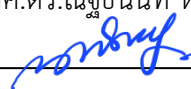
โครงการพิเศษ

กรรมการสอบโครงการพิเศษ

กรรมการสอบโครงการพิเศษ



(รศ.ดร.ณัฐชนนท์ หงส์วรธีธร)



(ผศ.ดร.นุชจรินทร์ อินทะหล้า)



(ผศ.ดร.ศาดนาฏ กิจศิริานุวัตร)

มหาวิทยาลัยธรรมศาสตร์
คณะวิทยาศาสตร์และเทคโนโลยี

รายงานโครงการพิเศษ

ของ

นายพลภัฏฐ์ ตัณฑโอภาส
เรื่อง

ระบบแนะนำเทคนิคการพัฒนาแอนิเมชันบนเว็บไซต์
ได้รับการตรวจสอบและอนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
เมื่อ วันที่ 30 พฤษภาคม พ.ศ. 2568

อาจารย์ที่ปรึกษา และ กรรมการสอบ

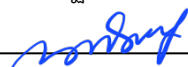
โครงการพิเศษ

กรรมการสอบโครงการพิเศษ

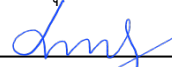
กรรมการสอบโครงการพิเศษ



(รศ.ดร.ณัฐชนนท์ หงส์วรธีธร)



(ผศ.ดร.นุชจรินทร์ อินทะหล้า)



(ผศ.ดร.ศัตนาฏ กิจศิริานุวัตร)

หัวข้อโครงการพิเศษ	ระบบแนะนำเทคนิคการพัฒนาแอนิเมชันบนเว็บไซต์
ชื่อผู้เขียน	นายพลัฏฐ์ ตัญญาโอภาส
ชื่อปริญญา	วิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
สาขาวิชา/คณะ/มหาวิทยาลัย	สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์
อาจารย์ที่ปรึกษาโครงการพิเศษ	รศ.ดร.ณัฐชนนท์ หงส์วิทธิธร
ปีการศึกษา	2567
บทคัดย่อ	

เนื่องจากผู้เขียนมีความสนใจในการพัฒนาแอปพลิเคชันต่างๆ บนเว็บไซต์ด้วย JavaScript โดยเฉพาะการใช้งานไลบรารีอย่าง GSAP (GreenSock Animation Platform) ซึ่งช่วยให้การสร้างแอนิเมชันมีความยืดหยุ่นและเขียนได้ง่ายมากขึ้น อย่างไรก็ตาม ในกระบวนการพัฒนาเว็บไซต์ <https://palat2023.com> ผู้เขียนพบปัญหาที่ทำให้เว็บไซต์มีอาการกระตุกหรือไม่ราบรื่นในการใช้งาน ซึ่งปัญหาดังกล่าวมักจะเกิดจากการที่การเขียนโค้ด JavaScript ที่ซับซ้อน หรือการจัดการทรัพยากรที่ไม่เหมาะสม ส่งผลให้การแสดงผลหน้าเว็บเกิดความล่าช้า ผู้เขียนจึงเริ่มต้นศึกษาและปรับปรุงการทำงานของเว็บไซต์ โดยเน้นไปที่การปรับปรุงประสิทธิภาพ (optimization) ของเว็บไซต์ในส่วนของ front-end เพื่อให้การแสดงผลและประสบการณ์ผู้ใช้งานมีความลื่นไหลมากขึ้น

อย่างไรก็ตาม ในระหว่างการเขียนเว็บไซต์ <https://palat2023.com> ผู้เขียนพบว่าการปรับปรุงประสิทธิภาพของเว็บไซต์ทางด้าน front-end นั้นยังขาดแคลนคำแนะนำหรือแนวทางที่ครอบคลุมในระดับขั้นกลางถึงขั้นสูง (intermediate to advanced) โดยเฉพาะอย่างยิ่งสำหรับผู้ที่มีความรู้พื้นฐานและต้องการยกระดับทักษะของตนเอง ผู้เขียนจึงเล็งเห็นถึงความสำคัญในการรวบรวมข้อมูล เทคนิค และวิธีการต่างๆ ที่มีประสิทธิภาพในการปรับปรุงการทำงานของเว็บไซต์ โดยเน้นไปที่การเพิ่มความเร็วและลดอาการกระตุกที่เกิดจากการทำงานของ JavaScript

รายงานฉบับนี้จึงถูกจัดทำขึ้นเพื่อเป็นแนวทางและเครื่องมือสำหรับนักพัฒนาที่ต้องการยกระดับทักษะการพัฒนาเว็บไซต์ของตนในด้านการปรับปรุงประสิทธิภาพ โดยผู้เขียนได้รวบรวมเทคนิคที่สำคัญ และแบ่งปันประสบการณ์จากการปรับปรุงเว็บไซต์จริง เพื่อให้สามารถนำไปประยุกต์ใช้ในโครงการของตนเองได้อย่างมีประสิทธิภาพมากยิ่งขึ้น

คำสำคัญ: Frontend, Javascript Effect and Animation, CSS Effect and Animation

Thesis Title	Introduction to the development of frontend the web for efficiency
Author	Palat Tuntaopas
Degree	Bachelor of Science
Major Field/Faculty/University	Computer Science Faculty of Science and Technology Thammasat University
Project Advisor	Assoc.Prof.Dr. Nuttanont Hongwarittorn
Project Co-Advisor	Assoc.Prof.Dr. Nuttanont Hongwarittorn
Academic Years	2024

ABSTRACT

As a web developer, I've always been fascinated by creating engaging effects on websites using JavaScript. During the development of https://palat2023.com, I faced a frustrating challenge: the site would occasionally lag or feel unresponsive. This problem often stemmed from overly complex JavaScript code or inefficient handling of resources, which led to a poor user experience.

Determined to solve this, I dove into optimizing the website's front-end performance. Along the way, I realized there's a lack of practical, intermediate-level resources for developers like me who want to go beyond the basics and build smoother, faster websites.

This report shares my journey of tackling these issues, the lessons I learned, and the techniques I used to improve performance. It's a hands-on guide for developers who want to enhance their skills and deliver seamless, enjoyable web experiences.

Keywords: Frontend, Javascript Effect and Animation, CSS Effect and Animation

กิตติกรรมประกาศ

รายงานฉบับนี้สำเร็จลุล่วงได้ด้วยความช่วยเหลือและการสนับสนุนจากบุคคลหลายท่าน ซึ่งผู้เขียนขอแสดงความขอบคุณอย่างจริงใจ

ก่อนอื่น ผู้เขียนขอขอบคุณ รศ.ดร.ณัฐชนน หงส์วริทธิ์ธร ที่ให้คำแนะนำและชี้แนะแนวทางที่เป็นประโยชน์ตลอดกระบวนการจัดทำรายงานฉบับนี้ คำแนะนำของท่านมีส่วนสำคัญอย่างยิ่งต่อความสำเร็จของงานชิ้นนี้

สุดท้าย ผู้เขียนขอขอบคุณ คุณพ่อ คุณแม่ ที่คอยสนับสนุนและให้กำลังใจมาโดยตลอด ทำให้ผู้เขียนสามารถดำเนินงานนี้จนสำเร็จลุล่วง

ขอขอบคุณ พี่พร้อมๆ และเพื่อนๆ ที่ให้คำปรึกษา ช่วยแลกเปลี่ยนความคิดเห็น และแบ่งปันประสบการณ์ ซึ่งช่วยเสริมความเข้าใจและเพิ่มมุมมองใหม่ๆ ให้กับผู้เขียน

ผู้เขียนหวังเป็นอย่างยิ่งว่ารายงานฉบับนี้จะเป็นประโยชน์ต่อผู้ที่สนใจและสามารถนำไปใช้พัฒนาต่อไปได้

นายพลัฏฐ์ ดันทโสภาส

สารบัญ

	หน้า
บทคัดย่อ	1
ABSTRACT	2
กิตติกรรมประกาศ	3
สารบัญ	4
สารบัญตาราง	8
สารบัญภาพ	9
รายการสัญลักษณ์และคำย่อ	11
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของโครงการ	3
1.4 ข้อจำกัดของโครงการ	4
1.5 ประโยชน์ของโครงการ	5
บทที่ 2 วรรณกรรมและงานวิจัยที่เกี่ยวข้อง	6
2.1 วรรณกรรมที่เกี่ยวข้อง	6
2.1.1 CSS	6
2.1.2 JavaScript	8
2.1.3 SVG	13

2.1.4	3D และภาพเสมือนจริง	14
2.1.5	Font Animation	15
2.2	สิ่งที่โครงานนี้จะสอน	17
2.2.1	CSS Animation	17
2.2.2	JavaScript Animation	18
2.2.3	SVG	18
2.2.4	3D และภาพเสมือนจริง	19
2.2.5	Font Animation	19
2.3	เว็บไซต์ที่สอนเนื้อหาที่จำเป็นต้องใช้	21
2.3.1	CSS	21
2.3.2	JavaScript	23
2.3.3	SVG	26
2.3.4	3D และภาพเสมือนจริง	28
2.3.5	Font Animation	28
2.4	การเปรียบเทียบเนื้อหาของบทเรียนระหว่างโครงานนี้กับ W3Schools	29
2.4.1	การเปรียบเทียบการเรียนรู้ CSS ระหว่างโครงานนี้กับ W3Schools	29
2.4.2	การเปรียบเทียบการเรียนรู้ Javascript ระหว่างโครงานนี้กับ W3Schools	31
2.4.3	การเปรียบเทียบการเรียนรู้ SVG ระหว่างโครงานนี้กับ W3Schools	32
2.5	ผู้เรียนควรมีพื้นฐานต่อไปนี้ก่อน	35
2.5.1	ผู้เรียนควรมีพื้นฐาน CSS ต่อไปนี้ก่อน	35
2.5.2	ผู้เรียนควรมีพื้นฐาน Javascript ต่อไปนี้ก่อน	36
2.5.3	ผู้เรียนไม่จำเป็นต้องมีพื้นฐานการทำ SVG มาก่อน	37
2.5.4	ผู้เรียนไม่จำเป็นต้องมีพื้นฐานการทำ 3D และภาพเสมือนจริง มาก่อน	37

2.5.5	ผู้เรียนไม่จำเป็นต้องมีพื้นฐานการทำ font animation มาก่อน	37
2.6	เว็บไซต์ที่สอนหรือรวบรวมเนื้อหาขั้น Intermediate ถึง Advance	38
บทที่ 3	วิธีการวิจัย	39
3.1	สถาปัตยกรรมของระบบ	39
3.2	การวิเคราะห์ขอบเขตและความต้องการของระบบ	40
3.3	ประเด็นที่น่าสนใจและสิ่งที่ท้าทาย	61
3.4	ผลลัพธ์ที่คาดหวัง	61
3.5	ระบบต้นแบบและผลลัพธ์เบื้องต้น	62
บทที่ 4	ทรัพยากรและแผนการดำเนินงาน	66
4.1	การจัดเตรียมฮาร์ดแวร์และซอฟต์แวร์	66
4.1.1	ภาษาที่ใช้ในการพัฒนา	66
4.1.2	โปรแกรมที่ใช้ในการพัฒนา	66
4.1.3	คอมพิวเตอร์ที่ใช้ในการพัฒนา	66
4.1.4	Computer specification	66
4.2	แผนการดำเนินงาน	67
4.3	ภาพการทำงานของระบบ	69
4.3.1	ภาพการทำงานของระบบและการแนะนำการพัฒนาการแสดงผลระบบส่วนหน้า (Frontend) เพื่อการแสดงผลอย่างมีประสิทธิภาพ	69
4.3.2	การวัดประสิทธิภาพของแอนิเมชันด้วย DevTools Performance	75
4.3.3	วิธีการเข้าถึง quiz	77
4.4	การทดสอบระบบส่วนเนื้อหา	80

	(7)
4.5 การทดสอบระบบโดยรวม	84
บทที่ 5 สรุป	93
5.1 สรุปผลการดำเนินงาน	93
5.2 ปัญหาและอุปสรรคที่พบ	94
5.3 ข้อเสนอแนะเพิ่มเติม	94
รายการอ้างอิง	97
ภาคผนวก	99
ภาคผนวก ก.	99
ภาคผนวก ข.	99
ภาคผนวก ค.	99

สารบัญตาราง

	หน้า
ตารางที่ 3.1 รายละเอียดกรณีการใช้งานของระบบ use case description	41
ตารางที่ 3.2 ตารางแสดง use case description เลือกบทเรียน	42
ตารางที่ 3.3 ตารางแสดง use case description คีลขาบทเรียน	43
ตารางที่ 3.4 ตารางแสดง use case description ทำแบบทดสอบ	44
ตารางที่ 3.5 ตารางแสดง use case description ส่งแบบทดสอบ	45
ตารางที่ 3.6 ตารางแสดง use case description คู่มือการทดสอบ	46
ตารางที่ 3.7 ตารางแสดง use case description เลือกแบบทดสอบ	47
ตารางที่ 3.8 ตารางแสดง use case description เพิ่มแบบทดสอบ	48
ตารางที่ 3.9 ตารางแสดง use case description ตรวจแบบทดสอบ	49
ตารางที่ 3.10 question_sets ประกอบด้วยลำดับข้อมูลในตาราง	59
ตารางที่ 3.11 questions ประกอบด้วยลำดับข้อมูลในตาราง	59
ตารางที่ 3.12 answers ประกอบด้วยลำดับข้อมูลในตาราง	59
ตารางที่ 3.13 choices ประกอบด้วยลำดับข้อมูลในตาราง	60
ตารางที่ 3.14 user_answer_sets ประกอบด้วยลำดับข้อมูลในตาราง	60
ตารางที่ 3.15 user_answers ประกอบด้วยลำดับข้อมูลในตาราง	60
ตารางที่ 4.1 การดำเนินงาน CS303	67
ตารางที่ 4.2 การดำเนินงาน CS403	68

สารบัญภาพ

	หน้า
ภาพที่ 2.1 CSS	6
ภาพที่ 2.2 JavaScript	9
ภาพที่ 2.3 React	12
ภาพที่ 2.4 SVG	13
ภาพที่ 2.5 ภาพ 3D	15
ภาพที่ 2.6 font animation	16
ภาพที่ 2.7 แบบเรียน CSS จาก W3Schools	21
ภาพที่ 2.8 ฟีเจอร์ Try it Yourself จาก W3Schools(CSS)	22
ภาพที่ 2.9 แบบเรียน JavaScript จาก W3Schools	23
ภาพที่ 2.10 ฟีเจอร์ Try it Yourself จาก W3Schools(Javascript)	24
ภาพที่ 2.11 แบบเรียน SVG จาก W3Schools	26
ภาพที่ 2.12 ฟีเจอร์ Try it Yourself จาก W3Schools(SVG)	26
ภาพที่ 3.1 สถาปัตยกรรมของระบบ	39
ภาพที่ 3.2 use case diagram	40
ภาพที่ 3.3 activity diagram เลือกบทเรียน	50
ภาพที่ 3.4 activity diagram ศึกษาบทเรียน	51
ภาพที่ 3.5 activity diagram ทำแบบทดสอบ	52
ภาพที่ 3.6 activity diagram ส่งแบบทดสอบ	53
ภาพที่ 3.7 activity diagram ดูผลการทดสอบ	54
ภาพที่ 3.8 activity diagram เลือกแบบทดสอบ	55
ภาพที่ 3.9 activity diagram เพิ่มแบบทดสอบ	56
ภาพที่ 3.10 activity diagram ตรวจแบบทดสอบ	57
ภาพที่ 3.11 Entity-Relationship Diagrams	58
ภาพที่ 3.12 หน้า course ของระบบ	62
ภาพที่ 3.13 หน้า บทเรียนตัวอย่าง (gsap)	63
ภาพที่ 3.14 หน้า quiz ส่วนปรนัย	64
ภาพที่ 3.15 หน้า quiz ส่วนอัตนัย	65

ภาพที่ 4.1 หน้า course	69
ภาพที่ 4.2 หน้า lesson	70
ภาพที่ 4.3 หน้า sublesson	70
ภาพที่ 4.4 หน้า sublesson	71
ภาพที่ 4.5 หน้า sublesson	71
ภาพที่ 4.6 หน้า quiz	77
ภาพที่ 4.7 หน้า กรอก ชื่อ และรหัสนักศึกษา	77
ภาพที่ 4.8 หน้า quiz	78
ภาพที่ 4.9 หน้า summit	78
ภาพที่ 4.10 หน้า ยืนยัน	79
ภาพที่ 4.11 หน้า แสดงคะแนน	79
ภาพที่ 4.12 หน้า กราฟผู้ทำ	84
ภาพที่ 4.13 หน้า กราฟประเมินระบบ	85
ภาพที่ 4.14 หน้า กราฟประเมิน quiz	86
ภาพที่ 4.15 หน้า กราฟการนำความรู้ไปใช้งานจริง	87
ภาพที่ 4.16 หน้า กราฟประเมินความเข้าใจพื้นฐาน	88
ภาพที่ 4.17 หน้า กราฟความคิดเห็นต่อการออกแบบ UI	89
ภาพที่ 4.18 หน้า กราฟประเมินการใช้งาน	89
ภาพที่ 4.19 หน้า กราฟประเมินเป้าหมาย	90
ภาพที่ 4.20 หน้า กราฟประเมินคุณภาพ	91
ภาพที่ 4.21 หน้า กราฟประเมินความพึงพอใจ	92
ภาพที่ 4.22 หน้า transition shorthand	72
ภาพที่ 4.23 หน้าคำถาม transition shorthand	73
ภาพที่ 4.24 หน้า transitionend	73

ภาพที่ 4.25 หน้าคำถาม transitionend	74
ภาพที่ 4.26 หน้า Animation บน GPU	74
ภาพที่ 4.27 หน้าคำถาม Animation บน GPU	75
ภาพที่ 4.28 หน้า Performance	76
ภาพที่ 4.29 หน้า เกณฑ์การประเมินเบื้องต้น	83
ภาพที่ 5.1 หน้า แบบฝึกหัดตัวอย่าง	95
ภาพที่ 5.2 หน้า แบบฝึกหัดตัวอย่าง	96

รายการสัญลักษณ์และคำย่อ

สัญลักษณ์/คำย่อ

คำเต็ม/คำจำกัดความ

CSS

Cascading Style Sheets

SVG

Scalable Vector Graphics

บทที่ 1 บทนำ

1.1 ความเป็นมาและความสำคัญของโครงการ

โครงการนี้เกิดขึ้นจากความตั้งใจของผู้เขียนที่ต้องการสร้างเว็บไซต์ที่น่าสนใจและสามารถดึงดูดความสนใจจากผู้ใช้งานได้มากขึ้นผู้เขียนเชื่อว่าแอนิเมชันมีบทบาทสำคัญใน

การทำให้เว็บไซต์ไม่น่าเบื่อและเพิ่มความมีชีวิตชีวาให้กับหน้าเว็บทั้งในแง่ของการเคลื่อนไหวของข้อความภาพหรือองค์ประกอบต่างๆบนหน้าเว็บเมื่อผู้ใช้งานเห็นแอนิเมชันที่สวยงามและน่าสนใจ จะช่วยให้ประสบการณ์การใช้งานเว็บดีขึ้นและทำให้ผู้ใช้งานรู้สึกถึงความใส่ใจของผู้ทำ

อย่างไรก็ตามเมื่อผู้เขียนทดลองเพิ่มแอนิเมชันที่ซับซ้อนบนเว็บไซต์ส่วนตัวก็พบว่ามีปัญหาเกิดขึ้นเช่นเว็บไซต์ช้าลงกระตุกหรือบางครั้งไม่สามารถแสดงแอนิเมชันได้อย่างราบรื่น โดยเฉพาะเมื่อใช้ JavaScript หรือไลบรารีอย่าง GSAP ซึ่งทำให้ผู้เขียนเกิดความกังวลเกี่ยวกับประสิทธิภาพของเว็บไซต์ที่อาจจะได้รับผลกระทบจากการเพิ่มแอนิเมชันเหล่านี้

เนื่องจากเว็บไซต์ที่มีแอนิเมชันที่ดูดีและทำงานได้ราบรื่นถือเป็นสิ่งสำคัญที่ทำได้ยาก ผู้เขียนจึงเริ่มศึกษาแนวทางต่าง ๆ เพื่อแก้ไขปัญหา และค้นหาวิธีที่จะสร้างแอนิเมชันที่ทั้งสวยงามและมีประสิทธิภาพ แต่พบว่าข้อมูลที่มีอยู่ส่วนใหญ่ยังไม่ครอบคลุม หรือนั่นแค่เทคนิคพื้นฐานที่ทำไม่สามารถแก้ปัญหาหรือปรับปรุงแอนิเมชันในระดับที่ซับซ้อนได้

โครงการนี้จึงเกิดขึ้นเพื่อรวบรวมแนวทางการพัฒนาแอนิเมชันที่มีประสิทธิภาพและสามารถใช้งานได้จริงในเว็บไซต์ที่มีการใช้งานสูง โดยการเน้นเทคนิคต่าง ๆ เช่น CSS Animation, JavaScript Animation, การใช้ SVG, การสร้างแอนิเมชันสามมิติ และการใช้เทคนิคการเคลื่อนไหวของฟอนต์เพื่อให้เว็บไซต์สามารถใช้งานได้อย่างราบรื่นและดึงดูดผู้ใช้งานโครงการนี้จะเป็นประโยชน์ทั้งในด้านการออกแบบ การเลือกใช้เครื่องมือและการปรับแต่งแอนิเมชันให้เหมาะสมกับเว็บไซต์ที่ต้องการความสวยงามและประสิทธิภาพในการทำงาน

การรวบรวมข้อมูลและการทดลองต่าง ๆ นี้จะช่วยให้นักพัฒนาหรือผู้ที่สนใจในการพัฒนาเว็บสามารถนำเทคนิคเหล่านี้ไปใช้ในงานจริงได้ โดยไม่เพียงแค่เพิ่มความสวยงามให้กับเว็บไซต์แต่ยังช่วยให้การใช้งานของเว็บไซต์ราบรื่นและตอบสนองได้ดีในทุกการใช้งาน

1.2 วัตถุประสงค์

ในการพัฒนาเว็บไซต์ที่มีแอนิเมชันผู้พัฒนามักต้องเผชิญกับความท้าทายในการทำให้แอนิเมชันมีความสวยงามและลื่นไหล โดยไม่ส่งผลกระทบต่อประสิทธิภาพของเว็บไซต์ แม้ว่าแอนิเมชันจะช่วยเพิ่มความน่าสนใจและทำให้เว็บไซต์ดูทันสมัยแต่หากใช้งานไม่เหมาะสมหรือขาดการปรับแต่งอย่างถูกต้อง อาจทำให้เว็บไซต์ช้าลงหรือกระตุกได้

ดังนั้นโครงการนี้จึงมีวัตถุประสงค์เพื่อศึกษาและรวบรวมแนวทางในการสร้างแอนิเมชันที่เหมาะสมและครอบคลุมเทคนิคการออกแบบและการใช้งานเครื่องมือในระดับต่าง ๆ เพื่อช่วยให้นักพัฒนาสามารถสร้างเว็บไซต์ที่ดึงดูดใจผู้ใช้งานและตอบสนองได้ดี

วัตถุประสงค์ของโครงการนี้มีดังต่อไปนี้:

1. เพื่อศึกษาและรวบรวมเทคนิคการสร้างแอนิเมชันสำหรับเว็บไซต์ให้มีความสวยงามและมีประสิทธิภาพสูง
2. เพื่อแก้ปัญหาความล่าช้าและการกระตุกของเว็บไซต์ที่เกิดจากการใช้แอนิเมชันที่ซับซ้อน
3. เพื่อสร้างแนวทางหรือคู่มือสำหรับนักพัฒนาที่ต้องการเพิ่มแอนิเมชันให้กับเว็บไซต์โดยไม่กระทบต่อประสิทธิภาพ
4. เพื่อเผยแพร่ความรู้เกี่ยวกับการสร้างแอนิเมชันเชิงลึกและการปรับปรุงประสิทธิภาพของแอนิเมชันในเว็บไซต์
5. เพื่อพัฒนาทักษะการสร้างและปรับแต่งแอนิเมชันของผู้เขียนและนำเสนอแนวคิดที่สามารถใช้ได้จริงในงานพัฒนาเว็บไซต์

วัตถุประสงค์เหล่านี้มุ่งเน้นการเพิ่มความน่าสนใจและการใช้งานที่ราบรื่นของเว็บไซต์รวมถึงสร้างแนวทางที่สามารถช่วยเหลือนักพัฒนาเว็บในทุกระดับให้สร้างแอนิเมชันได้อย่างมีประสิทธิภาพ

1.3 ขอบเขตของโครงการ

โครงการนี้เหมาะสำหรับผู้ที่มีพื้นฐานการเขียนโค้ดและการสร้างเว็บไซต์อยู่แล้ว โดยจะมุ่งเน้นการสอนเทคนิคระดับกลางถึงขั้นสูงเกี่ยวกับการสร้างแอนิเมชันให้เว็บไซต์ ทั้งในแง่ของการออกแบบและการเพิ่มประสิทธิภาพ

ขอบเขตของเนื้อหา:

(1) CSS Animation

1. ใช้ Transition และ Keyframes เพื่อสร้างแอนิเมชัน
2. เทคนิคทำให้แอนิเมชันลื่นไหล

(2) JavaScript Animation

1. ใช้ JavaScript และไลบรารีอย่าง GSAP เพื่อสร้างแอนิเมชันที่ซับซ้อน
2. แก้ปัญหาความหน่วงของแอนิเมชัน

(3) SVG Animation

1. สร้างแอนิเมชันด้วยภาพ Vector
2. ใช้ CSS และ JavaScript ร่วมกับ SVG

(4) 3D และภาพเสมือนจริง

1. สร้างแอนิเมชัน 3D ด้วย WebGL หรือ Three.js
2. ใช้แอนิเมชัน 3D ให้เหมาะกับเว็บไซต์

(5) Font Animation

1. ทำให้ตัวอักษรเคลื่อนไหว เช่น พิมพ์ข้อความหรือเปลี่ยนขนาด
2. ใช้ฟอนต์แบบปรับแต่งได้เพื่อสร้างแอนิเมชัน

(6) การปรับปรุงประสิทธิภาพ

1. ทำให้แอนิเมชันเร็วขึ้นและไม่กระตุก
2. ใช้เครื่องมือวิเคราะห์และปรับปรุงการทำงานของเว็บไซต์

1.4 ข้อจำกัดและกรอบการพัฒนาของระบบ

1. เนื้อหาถูกออกแบบมาสำหรับผู้ที่มีพื้นฐาน HTML, CSS และ JavaScript ระดับเบื้องต้นเท่านั้น
 2. ระบบครอบคลุมเฉพาะเทคนิคระดับกลางถึงขั้นสูงไม่เหมาะสำหรับแอนิเมชันที่มีความซับซ้อนเกินกว่าความสามารถของเครื่องมือพื้นฐานที่ใช้
 3. ระบบรองรับเฉพาะไลบรารีและเทคโนโลยีที่ได้รับความนิยม เช่น GSAP และ CSS Animation อาจไม่ครอบคลุมเครื่องมืออื่น ๆ
 4. การสร้างแอนิเมชันต้องคำนึงถึงความเสี่ยงของเว็บไซต์หากใช้ทรัพยากรมากเกินไป อาจทำให้เว็บไซต์ช้าลง
 5. เนื้อหามุ่งเน้นการสอนแบบเชิงปฏิบัติโดยเน้นตัวอย่างจริงและแนวทางที่สามารถนำไปใช้งานได้จริง
 6. ระบบจะทดสอบเฉพาะแอนิเมชันในเบราว์เซอร์ยอดนิยม เช่น Chrome, Firefox และ Safari โดยอาจมีข้อจำกัดในเบราว์เซอร์ที่ไม่ได้รับการสนับสนุน
 7. โครงการนี้มีการกำหนดระยะเวลาการพัฒนาในช่วงที่กำหนดไว้เท่านั้นทำให้การเพิ่มฟีเจอร์ใหม่ อาจไม่ได้ถูกรวมในเวอร์ชันแรก
 8. ระบบพัฒนาโดยใช้เครื่องมือฟรีและโอเพนซอร์สจึงอาจมีข้อจำกัดในการปรับแต่งบางส่วน
- โครงการนี้จะพัฒนาภายใต้ข้อกำหนดดังกล่าวเพื่อให้บรรลุเป้าหมายและประโยชน์สูงสุดสำหรับผู้ใช้งาน

1.5 ประโยชน์ของโครงการ

1. พัฒนาทักษะการสร้างแอนิเมชันในเว็บไซต์
2. เพิ่มประสิทธิภาพในการพัฒนาเว็บไซต์
3. เพิ่มความรู้ด้านการใช้งานไลบรารีและเครื่องมือที่ช่วยในการพัฒนา
4. เตรียมความพร้อมสำหรับการพัฒนาเว็บไซต์ระดับมืออาชีพ
5. เสริมสร้างความเข้าใจในเทคนิคและเครื่องมือใหม่ ๆ
6. สร้างความพึงพอใจให้กับผู้ใช้งาน

บทที่ 2

วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

2.1 วรรณกรรมที่เกี่ยวข้อง

2.1.1 CSS

CSS คือ ภาษาที่ใช้สำหรับตกแต่งเอกสาร HTML ให้มีหน้าตา สี สัน ระยะห่าง พื้นหลัง เส้นขอบและอื่นๆ ตามที่ต้องการ CSS ย่อมาจาก **Cascading Style Sheets** มีลักษณะเป็นภาษาที่มีรูปแบบในการเขียน **Syntax** แบบเฉพาะและได้ถูกกำหนดมาตรฐานโดย **W3C** เป็นภาษาหนึ่งในการตกแต่งเว็บไซต์ ได้รับความนิยมอย่างแพร่หลาย



ภาพที่ 2.1 CSS

โครงสร้างคำสั่งของ CSS

1. ตัวเลือก (Selector): ใช้ระบุองค์ประกอบ HTML ที่ต้องการกำหนดรูปแบบ เช่น `

` หมายถึงเลือกองค์ประกอบ `

` ทั้งหมดในหน้า
2. คำสั่ง (Declaration): ประกอบด้วยคุณสมบัติและค่าของคุณสมบัติที่ต้องการกำหนด เช่น `color: red;`
3. คุณสมบัติ (Property): คือสิ่งที่ต้องการปรับเปลี่ยนขององค์ประกอบ เช่น `color` หมายถึงการเปลี่ยนสี
4. ค่าของคุณสมบัติ (Property value): คือค่าที่กำหนดให้กับคุณสมบัตินั้น ๆ เช่น `red` หมายถึงให้สีเป็นสีแดง

```
p {
  color: red;
  width: 500px;
  border: 1px solid black;
}
```

สามารถใช้ CSS สร้างภาพเคลื่อนไหว เช่น การเปลี่ยนน้ำหนักของตัวอักษรได้โดยใช้ keyframes:

```
@keyframes letter-breathe {
  from,
  to {
    font-variation-settings: 'wght' 100;
  }
  50% {
```

```
font-variation-settings: 'wght' 900;

}

}
```

2.1.2 JavaScript

JavaScript คืออะไร

JavaScript คือ ภาษาคอมพิวเตอร์ภาษาหนึ่งที่มีไว้เพื่อจัดการเอฟเฟกต์หรือพฤติกรรมของหน้าเว็บไซต์ นอกเหนือจาก HTML ที่ใช้จัดการเรื่องของเนื้อหาของเว็บและ CSS ที่ใช้จัดการเรื่องของโครงสร้างและดีไซน์ของเว็บ

JavaScript ยังสามารถใช้งานในส่วนของโปรแกรมบนคอมพิวเตอร์และเซิร์ฟเวอร์ ยกตัวอย่างเช่น Node.js หรือในฐานข้อมูลก็สามารถใช้งานได้เช่นกัน เช่น MongoDB JavaScript สามารถช่วยเราจัดการงานต่างๆ โดยไม่จำเป็นที่จะต้องทำการโหลดหน้าเว็บใหม่ซ้ำๆ หากเราเคยเห็นการกดปุ่ม Button แล้ว ข้อความบางข้อความแสดงผลหรือซ่อนข้อความนั้นโดยไม่จำเป็นต้องโหลดหน้านั้นใหม่อีกครั้ง หรือมีการเปลี่ยนสีของตัวอักษร เปลี่ยนแปลงเนื้อหา คือ หน้าเว็บไซต์นั้นๆ ถูกเรียกใช้งานด้วย JavaScript แล้ว หากเราสรุปหน้าที่การทำงานของ JavaScript อย่างคร่าวๆ จะได้ดังนี้

1. JavaScript สามารถเปลี่ยนเนื้อหา HTML ได้
2. JavaScript สามารถเปลี่ยนค่าภายใน HTML Attributes ได้
3. JavaScript สามารถเปลี่ยนสไตล์ CSS ของ HTML ได้
4. JavaScript สามารถซ่อนส่วน HTML ที่ไม่ต้องการได้
5. JavaScript สามารถแสดงส่วน HTML ที่ต้องการได้



ภาพที่ 2.2 JavaScript

การเขียนประโยคใน JavaScript

โค้ดใน JavaScript นั้น จะประกอบไปด้วย Values, Operators, Expressions, Keywords และ Comments ขึ้นอยู่กับการเขียนของเราครับ เช่น หากเราต้องการเขียนประโยคที่บอกให้เบราว์เซอร์เขียนคำว่า Hello Dolly ไปใส่ภายในส่วนของ HTML ที่มีไอดี id="demo" เราจะได้ดังนี้

```
document.getElementById("demo").innerHTML = "Hello Dolly.";
```

โดยหากเรามาดูข้อกำหนดหลักๆ ของการเขียนประโยคใน JavaScript จะสรุปคร่าวๆ ได้แก่ ประโยคของ JavaScript นั้นจะลงท้ายด้วย Semicolons (;) เพื่อทำการแบ่งแยกประโยคแต่ละประโยคออกจากกัน

JavaScript จะไม่สนใจพื้นที่ว่างที่เป็น white space

เราสามารถแบ่งบรรทัดในการเขียนโค้ดได้ ในกรณีที่มีประโยคเกินกว่า 80 คำขึ้นไป

เราสามารถทำการจัดกลุ่มรวมในฟังก์ชันของ JavaScript ได้ โดยอยู่ภายในเครื่องหมายปิดปีกกา { }

ในกรณีที่เรากำลังการสร้างตัวแปร เราก็สามารถประกาศตัวแปรได้ ยกตัวอย่างเช่น

```
var x, y, z;    // ประกาศตัวแปร
x = 5; y = 6;  // กำหนดค่าให้ตัวแปร
z = x + y;     // คำนวณค่าตัวแปร
```

ในกรณีที่เป็นข้อความ (String) เราสามารถที่จะเขียนอยู่ภายในเครื่องหมายฟันทู แบบคู่และเดี่ยวก็ได้

ในกรณีที่เราไม่ต้องการให้ประโยคใดใน JavaScript ถูกใช้งาน เราจะใช้เครื่องหมายดับเบิ้ลแอส // หรือ /* และ */ เพื่อบอกให้รู้ว่าเป็นส่วนอธิบายหรือขยายความ

ในการกำหนดตัวแปรของ JavaScript นั้น เราจะต้องระบุให้ชัดเจน เพราะหากเรากำหนดแตกต่างกัน เพียงแค่ ตัวอักษรตัวใหญ่และตัวเล็กก็หมายถึงตัวแปรคนละตัวแล้ว เช่น

```
var lastname, lastName;
```

ในการกำหนดตัวแปรของ JavaScript สามารถกำหนดได้หลายแบบ เช่น มีเครื่องหมาย

Underscore เชื่อมประโยค first_name หรือตัวพิมพ์ใหญ่ FirstName หรือ ตัวพิมพ์เล็กอยู่ด้านด้าน firstName ทั้ง 3 แบบนี้เป็นที่นิยมในการกำหนดตัวแปร

ReactJS

ReactJS คืออะไร?

ReactJS เป็นไลบรารีของ JavaScript ที่ช่วยในการพัฒนาอินเทอร์เฟซผู้ใช้แบบไดนามิกและโต้ตอบได้ โดยถูกพัฒนาและเปิดให้ใช้งานโดย Facebook เพื่อให้การพัฒนาเว็บแอปพลิเคชันเป็นไปได้ง่ายและเร็วขึ้น

ใครควรเรียนรู้ ReactJS?

ผู้เริ่มต้นหรือผู้ที่ต้องการพัฒนาแอปพลิเคชันเว็บด้านหน้า (Front-end) และต้องการทำงานในสายงานนี้ ควรเรียนรู้ ReactJS เพื่อเพิ่มทักษะและความสามารถในการพัฒนา

ทำไมต้องเรียน ReactJS?

1. ใช้งานง่าย: ReactJS ไม่จำเป็นต้องเขียนโค้ดยาว สามารถใช้แนวคิดของคอมโพเนนต์ที่เขียนเพียงครั้งเดียวแต่ใช้งานได้หลายที่
2. รองรับโมดูลหลายตัว: ReactJS มีโมดูลต่าง ๆ ที่ช่วยให้การพัฒนาเป็นไปได้รวดเร็วและจัดการง่าย
3. สามารถพัฒนาแอปหลากหลาย: ด้วยความรู้ React สามารถพัฒนาเว็บเพจ แอปพลิเคชันบนเว็บ แอปมือถือ และแอป VR ได้
4. การย้ายระบบง่าย: มีการเรียนรู้ที่ไม่ซับซ้อน ทำให้ง่ายต่อการย้ายจากเทคโนโลยีอื่น ๆ มาสู่ ReactJS

ตัวอย่างโค้ด ReactJS

```

` `javascript
import React from 'react';
import ReactDOM from 'react-dom/client';

function Greeting(props) {
  return <h1>Welcome to Tutorialspoint</h1>;
}

const container = document.getElementById("root");

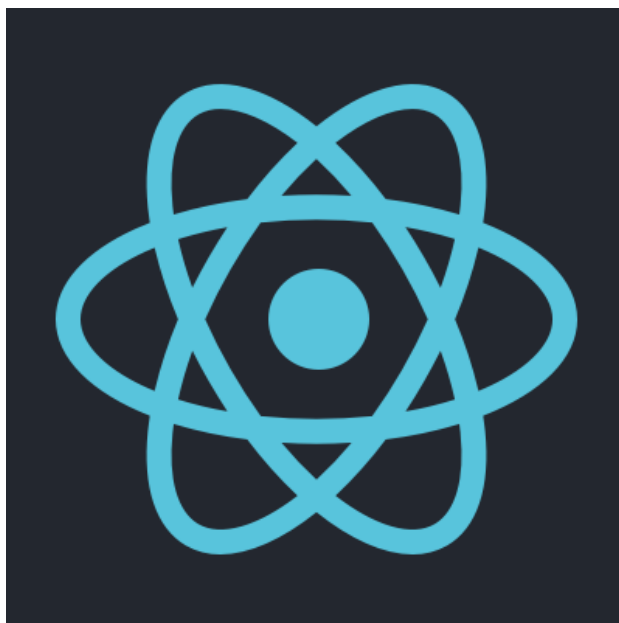
```

```
const root = ReactDOM.createRoot(container);
root.render(<Greeting />);
...
```

คุณสมบัติของ ReactJS

1. Virtual DOM: React ใช้ Virtual DOM ในการจัดการการเปลี่ยนแปลงของหน้าเว็บ ทำให้การอัปเดตมีประสิทธิภาพสูง
2. คอมโพเนนต์ที่ใช้ซ้ำได้: สามารถสร้างคอมโพเนนต์ครั้งเดียวและเรียกใช้ซ้ำในหลายส่วนของแอปพลิเคชัน
3. การผูกข้อมูลทางเดียว: ข้อมูลจะไหลจากคอมโพเนนต์หลักไปยังคอมโพเนนต์ย่อยเท่านั้น ทำให้การจัดการข้อมูลง่ายขึ้น

การเรียนรู้ ReactJS จำเป็นต้องมีพื้นฐานใน HTML, CSS และ JavaScript มาก่อน เพื่อเข้าใจแนวคิดเบื้องต้นและสามารถพัฒนาแอปพลิเคชันได้อย่างมีประสิทธิภาพ.



ภาพที่ 2.3 React

2.1.3 SVG

Scalable Vector Graphics (SVG) เป็นภาษามาร์กอัปที่ใช้ XML สำหรับการอธิบายกราฟิกแบบเวกเตอร์สองมิติ SVG เป็นมาตรฐานเปิดแบบข้อความสำหรับการอธิบายภาพที่สามารถแสดงผลได้อย่างชัดเจนในทุกขนาด และออกแบบมาเพื่อทำงานได้ดีร่วมกับมาตรฐานเว็บอื่น ๆ รวมถึง CSS, DOM, JavaScript, และ SMIL โดยที่ SVG จะคล้ายกับ HTML สำหรับข้อความ ภาพ SVG และพฤติกรรมที่เกี่ยวข้องจะถูกกำหนดในไฟล์ข้อความแบบ XML ซึ่งหมายความว่ามันสามารถค้นหา, ดัชนี, เขียนสคริปต์ และบีบอัดได้ นอกจากนี้ยังสามารถสร้างและแก้ไขได้ด้วยโปรแกรมแก้ไขข้อความหรือซอฟต์แวร์วาดภาพ เมื่อเทียบกับรูปแบบภาพแบบบิตแมปคลาสสิก เช่น JPEG หรือ PNG ภาพในรูปแบบ SVG สามารถแสดงผลได้ในทุกขนาดโดยไม่สูญเสียคุณภาพ และสามารถทำให้เป็นภาษาท้องถิ่นได้ง่ายโดยการอัปเดตข้อความภายในไฟล์ โดยไม่จำเป็นต้องใช้โปรแกรมกราฟิกในการทำเช่นนั้น หากใช้ไลบรารีที่เหมาะสม ไฟล์ SVG ยังสามารถทำให้เป็นภาษาท้องถิ่นได้ในขณะทำงาน

SVG ได้รับการพัฒนาโดย World Wide Web Consortium (W3C) ตั้งแต่ปี 1999



ภาพที่ 2.4 SVG

2.1.4 3D และภาพเสมือนจริง

3D Modeling คือกระบวนการสร้างแบบจำลองหรือรูปปั้น 3 มิติด้วยซอฟต์แวร์คอมพิวเตอร์ ทำให้สามารถเห็นแบบจำลองได้ 360 องศา ทั้งแนวตั้ง แนวนอน และแนวลึก แตกต่างจากภาพสองมิติที่ไม่มีแนวลึกให้ได้เห็น ซึ่งวัตถุประสงค์ของ 3D Modeling ก็เพื่อสร้างภาพหรือวัตถุให้ดูเสมือนจริงมากยิ่งขึ้น เช่น โมเดลโครงการบ้าน โมเดลการตกแต่งภายในบ้าน การออกแบบผลิตภัณฑ์ 3 มิติ การสร้างอนิเมชัน เป็นต้น ในกระบวนการสร้าง 3D Modeling มักเริ่มต้นด้วยการสร้างโมเดลพื้นฐานก่อน ซึ่งจะเป็นรูปร่างหรือโครงสร้างของวัตถุที่ต้องการทำเป็น 3D จากนั้นจึงจะเพิ่มรายละเอียดและลวดลาย เพื่อเข้ารูปด้วยซอฟต์แวร์ อย่าง Autodesk Maya, Blender, หรือ Cinema 4D เพื่อช่วยในการสร้างและปรับแต่งโมเดลออกมาเป็นรูปแบบไฟล์ 3D ที่สามารถนำไปใช้งานในโครงการต่าง ๆ ตามวัตถุประสงค์ที่กำหนดไว้ได้

3D Modeling ใช้ทำอะไรได้บ้าง มีประโยชน์กับธุรกิจอย่างไร

1. การสร้างแบบจำลองสามมิติ (3D Modeling) มีประโยชน์มากมายในอุตสาหกรรมและธุรกิจ ดังนี้
2. การออกแบบผลิตภัณฑ์: 3D Modeling เป็นเครื่องมือที่ช่วยในการออกแบบและพัฒนาสินค้าก่อนที่จะเข้าสู่กระบวนการผลิตจริง เช่น รถยนต์ อุปกรณ์อิเล็กทรอนิกส์ เป็นต้น
3. การออกแบบสถาปัตยกรรม: เช่น การสร้างแบบจำลองโครงการบ้าน ห้างสรรพสินค้า ระบบคมนาคมต่าง ๆ เพื่อให้สามารถวิเคราะห์โครงสร้าง และปรับปรุงการออกแบบก่อนการสร้างจริง
4. การสร้างภาพยนตร์ เกมส์ และภาพเคลื่อนไหว: ด้วยการสร้างโลกและตัวละคร 3 มิติ ทำให้ผู้ชมได้สัมผัสประสบการณ์ที่น่าตื่นเต้นและสมจริงมากยิ่งขึ้น
5. การออกแบบจำลองเพื่อการศึกษา : โดยเฉพาะทางการแพทย์ ที่ได้มีการสร้างโมเดลโครงสร้างภายในของร่างกาย หรือการฝึกผ่าตัดโดยใช้โมเดลจำลอง
6. การออกแบบเพื่อ 3D Printing : ไม่ว่าจะเป็นงานพุทธศิลป์ Art toys หรือว่าชิ้นส่วนอะไหล่ต่างๆที่ต้องการปริ้นสามมิติ ต้องผ่านกระบวนการออกแบบก่อนที่จะปริ้นสามมิติเพื่อให้ได้ไฟล์ .stl มาก่อนเพื่อนำไปพิมพ์สามมิติ



ภาพที่ 2.5 ภาพ 3D

2.1.5 font animation

CSS Text Animation คือ การใช้ CSS (Cascading Style Sheets) เพื่อเพิ่มเอฟเฟกต์การเคลื่อนไหวหรือการเปลี่ยนแปลงลักษณะต่าง ๆ ให้กับข้อความบนเว็บไซต์ โดยไม่ต้องใช้ JavaScript หรือโปรแกรมเสริมใดๆ ซึ่งช่วยเพิ่มความน่าสนใจและความโต้ตอบให้กับผู้ใช้ การใช้ CSS Text Animation ทำให้สามารถสร้างการเคลื่อนไหวของข้อความ เช่น การทำให้ข้อความเลื่อนเข้ามา (slide), การเปลี่ยนสี (color change), การกระโดด (bounce), หรือการพิมพ์ข้อความ (typing effect) เป็นต้น

วิธีการทำงาน:

Keyframes: ใช้สำหรับกำหนดลำดับการเปลี่ยนแปลงของคุณสมบัติ (เช่น ขนาด, ตำแหน่ง, ความโปร่งใส) ของข้อความในช่วงเวลาใดเวลาหนึ่ง

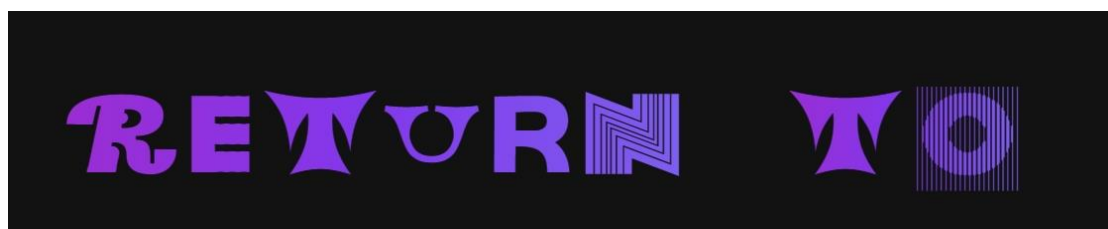
Animation Property: ใช้เพื่อควบคุมความเร็ว, การหน่วงเวลา, การวนซ้ำ และลักษณะการเคลื่อนไหวต่าง ๆ ที่จะใช้กับข้อความ

ตัวอย่างการใช้งาน:

Fade In: ทำให้ข้อความค่อยๆ ปรากฏขึ้นจากการที่มันโปร่งใสเป็นปกติ

Slide In: ทำให้ข้อความเลื่อนเข้าจากขอบของหน้าจอ

Typing Effect: ทำให้ข้อความดูเหมือนถูกพิมพ์ขึ้นมา



ภาพที่ 2.6 font animation

2.2 สิ่งที่โครงการนี้จะสอน

โครงการนี้ได้รวบรวมและออกแบบเนื้อหาสำหรับการสอน CSS Animation, JavaScript Animation, SVG, 3D และภาพเสมือนจริง (Virtual Reality), และ Font Animation โดยเน้นให้ผู้เรียนที่มีพื้นฐานเบื้องต้นสามารถพัฒนาทักษะไปสู่ระดับกลางและระดับสูง เนื้อหาเหล่านี้ถูกรวบรวมอย่างเป็นระบบ เพื่อแก้ไขปัญหาที่ไม่มีตำราเรียนหรือเว็บไซต์ใดที่ครอบคลุมหัวข้อเหล่านี้ทั้งหมดในรูปแบบที่เชื่อมโยงกัน

หัวข้อที่ครอบคลุมในโครงการนี้

2.2.1 CSS Animation

ตัวอย่างที่รวบรวมการสร้างการเคลื่อนไหวง่ายๆ เช่น การเปลี่ยนสีเมื่อผู้ใช้โฮเวอร์ หรือการเคลื่อนไหวขององค์ประกอบแบบวนลูป

สิ่งที่จะสอนในโครงการนี้

1. การใช้ @keyframes เพื่อสร้างแอนิเมชันขั้นสูง
2. การควบคุม Timing Functions เช่น ease, linear, และ cubic-bezier
3. การใช้ CSS Variables ร่วมกับ Animation

2.2.2 JavaScript Animation

ตัวอย่างที่รวบรวมการใช้ JavaScript แบบพื้นฐานเพื่อควบคุม DOM และทำให้เกิดการเปลี่ยนแปลง เช่น การย้ายตำแหน่งของวัตถุ

สิ่งที่จะสอนในคอร์สนี้

1. การใช้ JavaScript Animation API (requestAnimationFrame)
2. การใช้ไลบรารี GSAP เพื่อสร้างแอนิเมชันที่ซับซ้อนและลื่นไหล
3. การสร้าง Interaction โดยใช้ Event Listeners

2.2.3 SVG Animation

ตัวอย่างที่รวบรวมการใช้ SVG เพื่อวาดรูปทรงและสร้างการเคลื่อนไหวแบบง่ายๆ

สิ่งที่จะสอนในคอร์สนี้

1. การวาดกราฟิกแบบซับซ้อนด้วย SVG
2. การใช้ CSS และ JavaScript เพื่อสร้าง Path Animation
3. การปรับเปลี่ยนสี ขนาด และเอฟเฟกต์ของ SVG

2.2.4 3D และภาพเสมือนจริง

ตัวอย่างที่รวบรวมการใช้ CSS3 Transform เช่น rotate และ translate เพื่อสร้างวัตถุ 3D แบบพื้นฐาน

สิ่งที่จะสอนในคอร์สนี้

1. การสร้างวัตถุ 3D ที่ซับซ้อน เช่น กล้องหมุน
2. การใช้ WebGL และ Three.js เพื่อพัฒนาประสบการณ์เสมือนจริง
3. การผสมผสาน 3D และ Interactive Animation

2.2.5 Font Animation

ตัวอย่างที่รวบรวมการเปลี่ยนสีฟอนต์หรือขนาดเมื่อมีการโฮเวอร์

สิ่งที่จะสอนในคอร์สนี้

1. การใช้ Keyframes เพื่อทำให้ตัวอักษรเคลื่อนไหว เช่น ขยับตัวอักษรทีละตัว
2. การสร้าง Typing Effect โดยใช้ JavaScript และ CSS
3. การปรับแต่งฟอนต์เพื่อสร้างเอฟเฟกต์

เนื้อหาในคอร์สนี้มีความแตกต่างจากตำราหรือเว็บไซต์ทั่วไป เนื่องจากเราไม่ได้สอนเฉพาะหัวข้อแยกส่วน แต่เราเน้นการผสมผสานความรู้จากหลายหัวข้อ เช่น การใช้ SVG Animation ร่วมกับ JavaScript เพื่อสร้างแอนิเมชันที่ซับซ้อน หรือการใช้ CSS และ GSAP ในการควบคุมการเคลื่อนไหวในแบบ 3D

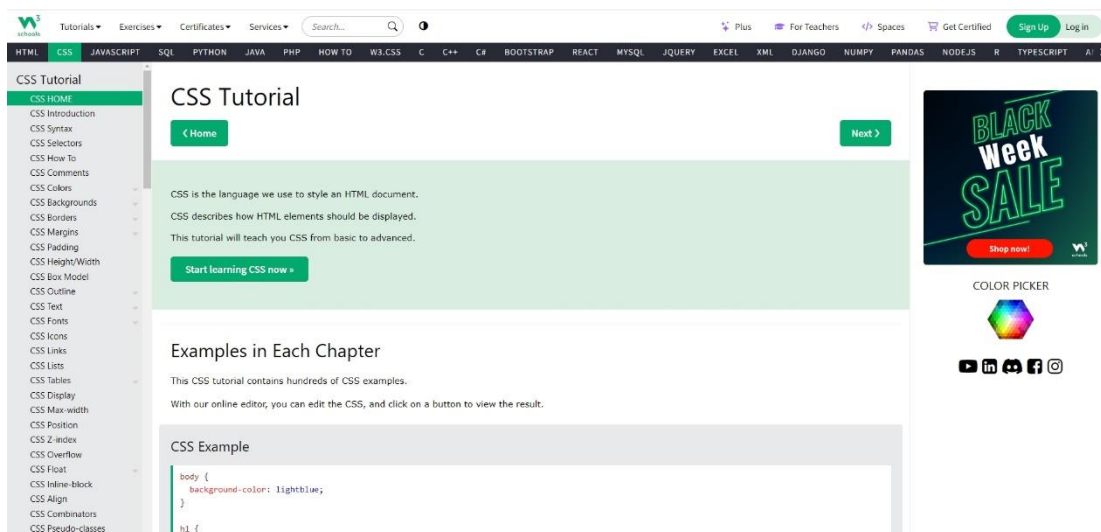
ตัวอย่างตำราหรือเว็บไซต์ที่มักมีข้อจำกัด เช่น

1. W3Schools ให้ข้อมูลเบื้องต้นที่เข้าใจง่าย แต่เนื้อหาไม่ครอบคลุมถึงการสร้างแอนิเมชันขั้นสูง
2. ตำราเฉพาะด้าน มักจะลึกเฉพาะเรื่อง เช่น การใช้ CSS Animation โดยไม่ได้รวมเทคนิคอื่นที่เกี่ยวข้อง

คอร์สนี้จึงออกแบบมาเพื่อให้ผู้เรียนสามารถพัฒนาแอนิเมชันได้ในทุกมิติ พร้อมทั้งตัวอย่างที่สามารถทดลองและปรับแต่งได้ทันที เพื่อให้เกิดการเรียนรู้ที่ครบวงจรและประยุกต์ใช้งานได้จริง

2.3 เว็บไซต์ที่สอนเนื้อหาที่จำเป็นต้องใช้

2.3.1 CSS

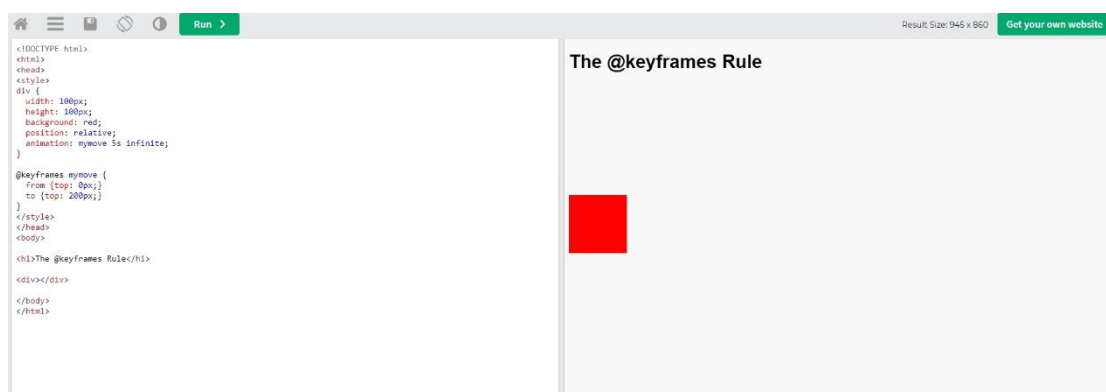


ภาพที่ 2.7 แบบเรียน CSS จาก W3Schools

แบบเรียน CSS จาก W3Schools

W3Schools เป็นแหล่งข้อมูลออนไลน์ที่ให้บริการเนื้อหาเกี่ยวกับ CSS และเทคโนโลยีเว็บอื่น ๆ โดยเน้นการสอนพื้นฐานสำหรับผู้เริ่มต้นที่ต้องการเรียนรู้การพัฒนาเว็บไซต์เว็บ W3Schools มีโครงสร้างเนื้อหาที่เป็นระบบครอบคลุมหัวข้อพื้นฐานของ CSS พร้อมตัวอย่างการใช้งานในแต่ละหัวข้ออย่างชัดเจนและอธิบายเนื้อหาอย่างละเอียด พร้อมทั้งแสดงตัวอย่างโค้ดที่เข้าใจง่ายเพื่อช่วยให้ผู้เรียนสามารถนำไปประยุกต์ใช้งานได้อย่างถูกต้อง

ฟีเจอร์ Try it Yourself



ภาพที่ 2.8 ฟีเจอร์ Try it Yourself จาก W3Schools(CSS)

ฟีเจอร์ Try it Yourself ช่วยให้ผู้เรียนสามารถทดลองโค้ดและดูผลลัพธ์ได้ทันที ผู้เรียนสามารถแก้ไขและทดลองโค้ดในพื้นที่ที่จำลองการทำงานของเว็บเบราว์เซอร์ การทดลองใช้งานสามารถทำได้โดยการคลิกปุ่ม “Try it Yourself” ในตัวอย่างโค้ด จากนั้นแก้ไขโค้ดและกด “Run” เพื่อดูผลลัพธ์

ตัวอย่างที่นำมาใน W3Schools มักเป็นโค้ดที่เข้าใจง่าย เพื่อให้ผู้เรียนสามารถต่อยอดความรู้ได้โดยไม่ซับซ้อน

ตัวอย่างการใช้ CSS เช่น การเปลี่ยนสีหรือขนาด

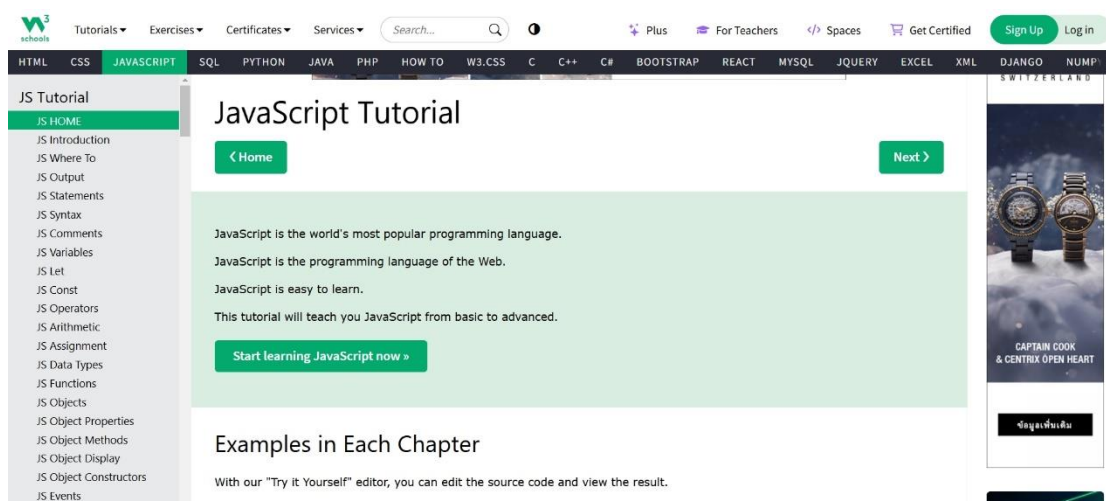
```
div {
width: 100px;
height: 100px;
background-color: blue;
transition: background-color 0.5s ease;
}
```

```
div:hover {
    background-color: red;
}
```

ผลลัพธ์

เมื่อผู้ใช้เลื่อนเมาส์ผ่านกล่องสีฟ้า กล่องจะเปลี่ยนเป็นสีแดงภายในเวลา 0.5 วินาที

2.3.2 Javascript



ภาพที่ 2.9 แบบเรียน JavaScript จาก W3Schools

แบบเรียน JavaScript จาก W3Schools

W3Schools เป็นแหล่งข้อมูลออนไลน์ที่เหมาะสมสำหรับผู้เริ่มต้นเรียนรู้ JavaScript โดยเนื้อหาครอบคลุมหัวข้อพื้นฐาน เช่น การใช้งานตัวแปร การเขียนฟังก์ชัน การควบคุมการทำงานของโปรแกรม และการจัดการเหตุการณ์ (Event Handling) บทเรียนใน W3Schools มีโครงสร้างที่ชัดเจน พร้อมตัวอย่างโค้ดที่เข้าใจง่าย เพื่อให้ผู้เรียนสามารถเรียนรู้ได้ด้วยตัวเองและนำไปประยุกต์ใช้กับการพัฒนาเว็บไซต์

พีเจอร် Try it Yourself



ภาพที่ 2.10 พีเจอร် Try it Yourself จาก W3Schools(JavaScript)

พีเจอร် Try it Yourself ช่วยให้ผูเรียนสามารถทดลองเขียนโค้ด JavaScript และดูผลลัพธ์ได้ทันที ผูเรียนสามารถปรับแต่งตัวอย่างโค้ดในพื้นที่ทดลองของ W3Schools หลังจากแก้ไขโค้ด สามารถกดปุ่ม "Run" เพื่อดูผลลัพธ์ที่แสดงผลลัพธ์แบบเรียลไทม์ในหน้าจ่อจำลอง

ตัวอย่างการใช้ JavaScript

ตัวอย่าง: การเปลี่ยนข้อความเมื่อคลิกปุ่ม

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1 id="demo">Hello World!</h1>
```

```
<button onclick="changeText()">Click me</button>
```

```
<script>
```

```
function changeText() {
```

```
    document.getElementById("demo").innerHTML = "You clicked the button!";
```

```
}
```

```
</script>
```



```
</body>
```

```
</html>
```

คำอธิบาย

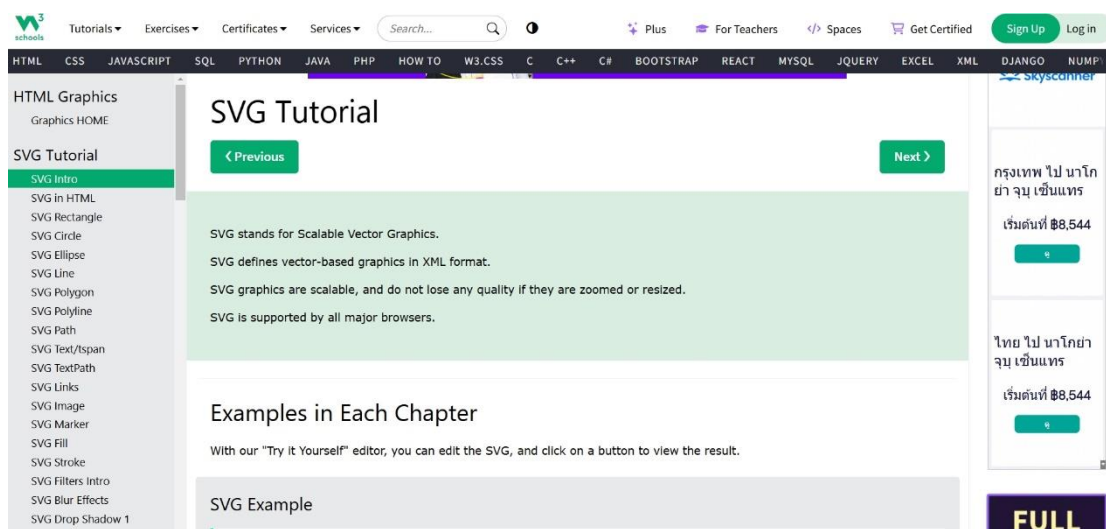
โค้ดนี้แสดงข้อความ "Hello World!" ในแท็ก <h1>

เมื่อคลิกปุ่ม ระบบจะเรียกใช้ฟังก์ชัน `changeText()` ซึ่งเปลี่ยนข้อความเป็น "You clicked the button!"

ประโยชน์ที่ได้จาก W3Schools

ผู้เรียนสามารถเข้าใจหลักการทำงานของ JavaScript ได้ง่ายด้วยตัวอย่างที่นำไปทดลองได้
ฟีเจอร์ Try it Yourself ช่วยให้การเรียนรู้เป็นแบบโต้ตอบ ทำให้เข้าใจการทำงานของโค้ดได้ดียิ่งขึ้น
เหมาะสำหรับการเรียนรู้ JavaScript ตั้งแต่พื้นฐานไปจนถึงการใช้งานในระดับที่ซับซ้อนมากขึ้น

2.3.3 SVG

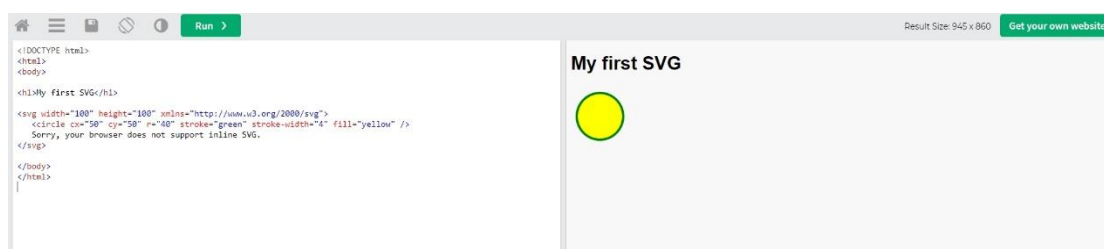


ภาพที่ 2.11 แบบเรียน SVG จาก W3Schools

แบบเรียน SVG จาก W3Schools เป็นแหล่งข้อมูลออนไลน์ที่ให้คำแนะนำในการเรียนรู้ SVG (Scalable Vector Graphics)

ซึ่งเป็นมาตรฐานสำหรับการสร้างกราฟิกแบบเวกเตอร์ที่สามารถขยายขนาดได้โดยไม่สูญเสียคุณภาพ บทเรียนใน W3Schools มีการอธิบายตั้งแต่พื้นฐานของการใช้งาน SVG เช่น การสร้างรูปทรงง่าย ๆ (เส้น, วงกลม, สี่เหลี่ยม) ไปจนถึงหัวข้อขั้นสูง เช่น การใช้เอฟเฟกต์ การแปลงรูป (Transformations) และการเพิ่มแอนิเมชันใน SVG

ฟีเจอร์ Try it Yourself



ภาพที่ 2.12 ฟีเจอร์ Try it Yourself จาก W3Schools(SVG)

ฟิเจอร์ Try it Yourself ช่วยให้ผู้ใช้สามารถทดลองเขียนโค้ด SVG และดูผลลัพธ์ได้ทันที

ผู้เรียนสามารถปรับแต่งตัวอย่างโค้ดในพื้นที่ทดลองของ W3Schools

หลังจากแก้ไขโค้ด สามารถกดปุ่ม "Run" เพื่อดูการแสดงผลกราฟิกแบบเรียลไทม์

ตัวอย่างการใช้งาน SVG

ตัวอย่าง: การสร้างวงกลมด้วย SVG

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1>My first SVG</h1>
```

```
<svg width="100" height="100" xmlns="http://www.w3.org/2000/svg">
```

```
<circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow" />
```

Sorry, your browser does not support inline SVG.

```
</svg>
```

```
</body>
```

```
</html>
```

คำอธิบาย

1. โค้ดนี้สร้างกราฟิกแบบวงกลมขนาดรัศมี 50 หน่วย
2. วงกลมถูกวางไว้ที่ตำแหน่งกึ่งกลางของพื้นที่ ($cx=50$, $cy=50$) และมีสีเติม (fill) เป็นสีเหลือง

ประโยชน์ที่ได้จาก W3Schools

ช่วยให้ผู้เรียนเข้าใจวิธีการสร้างและปรับแต่งกราฟิกด้วย SVG ได้อย่างง่ายดาย

ตัวอย่างโค้ดที่เข้าใจง่ายทำให้ผู้เรียนสามารถทดลองและพัฒนาความคิดสร้างสรรค์ได้ทันที

ฟีเจอร์ Try it Yourself ช่วยให้ผู้เรียนเห็นผลลัพธ์แบบเรียลไทม์

ทำให้การเรียนรู้เป็นไปอย่างมีประสิทธิภาพ

2.3.4 3D และภาพเสมือนจริง

เนื้อหาในเรื่องนี้มีน้อยและค่อนข้างกระจัดกระจายผู้เขียนเลยจะสอนเรื่องนี้ และเป็น case to case basis ควรพิจารณาเป็นรายกรณี

2.3.5 font animation

เนื้อหาในเรื่องนี้มีน้อยและค่อนข้างกระจัดกระจายผู้เขียนเลยจะสอนเรื่องนี้ และเป็น case to case basis ควรพิจารณาเป็นรายกรณี

2.4 การเปรียบเทียบเนื้อหาของบทเรียนระหว่างโครงงานนี้กับ W3Schools

2.4.1 การเปรียบเทียบการเรียนรู้ CSS ระหว่างโครงงานนี้กับ W3Schools

โครงงานนี้ รวบรวมและสอนเกี่ยวกับการใช้งาน CSS ในการสร้างแอนิเมชัน และเทคนิคต่างๆ เพื่อให้เว็บไซต์มีความน่าสนใจและมีประสิทธิภาพ ในขณะที่ W3Schools เป็นแหล่งเรียนรู้ออนไลน์ที่ให้ข้อมูลเกี่ยวกับ CSS ตั้งแต่พื้นฐานจนถึงระดับกลางโดยเน้น การสอนการใช้คุณสมบัติพื้นฐานของ CSS เช่น การจัดการตำแหน่ง, การจัดรูปแบบข้อความ, การเปลี่ยนแปลงสี, และการสร้างแอนิเมชันที่ไม่ซับซ้อน

ความแตกต่างในเนื้อหา:

W3Schools เน้นการสอน CSS ในส่วนของการจัดการองค์ประกอบพื้นฐาน เช่น การเปลี่ยนสีพื้นหลัง การจัดการขนาดฟอนต์ การใช้คลาสและไอดีในการเลือกองค์ประกอบ และการสร้างแอนิเมชันง่ายๆ เช่นการเปลี่ยนสีเมื่อเกิดการโฮเวอร์ (hover) หรือการใช้ @keyframes แบบเบื้องต้น

```
ตัวอย่างจาก W3Schools: div {
  width: 100px;
  height: 100px;
  background-color: blue;
  transition: background-color 0.5s ease;
}
div:hover {
  background-color: red;
}
```

ส่วนโครงงานนี้ มุ่งเน้นการใช้ CSS

ในการสร้างแอนิเมชันที่ซับซ้อนและมีประสิทธิภาพมากขึ้น เช่น การใช้ @keyframes เพื่อสร้างแอนิเมชันที่ยาวนานกว่า หรือการใช้ CSS Animation ร่วมกับ JavaScript เพื่อเพิ่มการควบคุมและสร้างผลลัพธ์ที่น่าสนใจมากขึ้น

นอกจากนี้ โครงงานยังได้เน้นการใช้ GSAP และ JavaScript ในการควบคุมแอนิเมชันที่มีความซับซ้อนและมีประสิทธิภาพสูงมากขึ้น ซึ่งไม่ครอบคลุมใน W3Schools

ตัวอย่างจากโครงงานนี้: @keyframes slideIn {

```
0% { transform: translateX(-100%); }
100% { transform: translateX(0); }
}
```

```
.element {
  animation: slideIn 1s ease-in-out;
}
```

โค้ดนี้ใช้ @keyframes เพื่อสร้างการเคลื่อนไหวประกอบจากด้านซ้ายไปยังตำแหน่งปกติ

สรุป:

W3Schools สอนเน้นที่ พื้นฐาน ของการใช้ CSS โดยเน้นการใช้งานฟิเจอร์พื้นฐานและแอนิเมชันเบื้องต้น ในขณะที่ โครงงานนี้ มุ่งเน้นการใช้ CSS และ JavaScript สำหรับการสร้างแอนิเมชันที่ ซับซ้อนมากขึ้น ใช้เครื่องมืออย่าง GSAP และ JavaScript เพื่อควบคุมแอนิเมชันที่มีประสิทธิภาพสูงและมีความซับซ้อนมากกว่า

ในส่วนของการทดลองรันโค้ดนั้น W3Schools มีฟิเจอร์ Try it Yourself แต่ผู้เรียนต้องกดปุ่ม run ทุกครั้งที่แก้ไขส่วนในระบบนี้ระบบจะ hot reload เองทำให้ผู้เรียนเห็นผลลัพธ์โดยไม่ต้องกดปุ่ม run ทุกครั้ง

2.4.2 การเปรียบเทียบการเรียนรู้ Javascript ระหว่างคอร์สนี้กับ W3Schools

คอร์สนี้มุ่งเน้นการสอนการใช้งาน JavaScript

ในการสร้างแอนิเมชันและการควบคุมการทำงานของเว็บไซต์ให้มีประสิทธิภาพสูงขึ้น โดยมีการใช้ไลบรารีต่างๆ เช่น GSAP และ JavaScript เพื่อสร้างการเคลื่อนไหวที่ซับซ้อนและลื่นไหล ในขณะที่ W3Schools เป็นแหล่งข้อมูลออนไลน์ที่สอน JavaScript ตั้งแต่พื้นฐานจนถึงระดับกลาง เน้นการอธิบายเกี่ยวกับฟังก์ชันการทำงานต่างๆ ของ JavaScript ที่สำคัญ เช่น การประกาศตัวแปร การควบคุมการไหลของโปรแกรม การทำงานกับ DOM (Document Object Model) และการจัดการเหตุการณ์ต่างๆ

ความแตกต่างในเนื้อหา

W3Schools จะเน้นที่การสอน JavaScript ในพื้นฐานที่สำคัญและใช้บ่อย เช่น การประกาศตัวแปร การใช้ฟังก์ชันพื้นฐาน การควบคุมการทำงานด้วยคำสั่ง if-else, switch, และการทำงานกับ DOM เพื่อให้ผู้เรียนสามารถเขียนโค้ดเพื่อเพิ่มความโต้ตอบให้กับเว็บไซต์ได้ ตัวอย่างเช่น การเพิ่มฟังก์ชันที่ตอบสนองต่อการคลิกปุ่มหรือการเปลี่ยนแปลงเนื้อหาในหน้าเว็บ

```
document.getElementById("myButton").addEventListener("click", function() {
    alert("Button clicked!");
});
```

ส่วนคอร์สนี้ มุ่งเน้นการใช้ JavaScript ในการสร้างแอนิเมชันที่มีความซับซ้อน เช่น การควบคุมการเคลื่อนไหวขององค์ประกอบบนหน้าเว็บ โดยใช้ GSAP (GreenSock Animation Platform) เพื่อให้แอนิเมชันมีความลื่นไหลและสามารถควบคุมเวลาและทิศทางการเคลื่อนไหวได้อย่างละเอียด นอกจากนี้ยังมีการใช้ JavaScript เพื่อสร้างแอนิเมชันร่วมกับ CSS เพื่อเพิ่มความหลากหลายในการควบคุมการเคลื่อนไหวของเว็บไซต์ ตัวอย่างโค้ดการใช้ GSAP ในคอร์สนี้

```
gsap.from(".box", { duration: 2, x: -200, opacity: 0, ease: "power2.out" });
```

โค้ดนี้จะทำให้ .box เคลื่อนที่จากตำแหน่งซ้าย 200px

ไปยังตำแหน่งปกติพร้อมกับการเพิ่มความโปร่งใสจาก 0 ไปยัง 1 ภายใน 2 วินาที

โดยใช้การควบคุมจาก GSAP ที่ช่วยให้แอนิเมชันมีความลื่นไหลและควบคุมได้ง่าย

สรุป

W3Schools เน้นการสอนพื้นฐานของ JavaScript เช่น การใช้ตัวแปร ฟังก์ชัน และการทำงานกับ DOM เพื่อให้ผู้เรียนสามารถเข้าใจวิธีการพัฒนาเว็บไซต์ที่โต้ตอบกับผู้ใช้ได้ ส่วน โครงงานนี้ เน้นการใช้ JavaScript และ GSAP เพื่อสร้างแอนิเมชันที่ซับซ้อนและลื่นไหลมากขึ้น เช่น การควบคุมแอนิเมชันขององค์ประกอบหลายตัวพร้อมกันและการสร้างประสบการณ์ผู้ใช้ที่ดียิ่งขึ้นด้วยการควบคุมการเคลื่อนไหว W3Schools มีฟีเจอร์ Try it Yourself ที่ช่วยให้ผู้เรียนทดลองโค้ดและดูผลลัพธ์ได้ทันที แต่ต้องกดปุ่ม Run ทุกครั้งที่มีการแก้ไขโค้ด ในขณะที่ โครงงานนี้ จะมีฟีเจอร์ Hot Reload ที่ทำให้ผลลัพธ์ปรากฏทันทีหลังจากการแก้ไขโค้ด โดยไม่ต้องกดปุ่ม Run

2.4.3 การเปรียบเทียบการเรียนรู้ SVG ระหว่างโครงงานนี้กับ W3Schools

โครงงานนี้มุ่งเน้นการใช้ SVG (Scalable Vector Graphics) ในการสร้างกราฟิกที่สามารถปรับขนาดได้โดยไม่สูญเสียความละเอียด และใช้ร่วมกับการสร้างแอนิเมชัน เช่น การสร้างภาพเคลื่อนไหว หรือการจัดการกับองค์ประกอบกราฟิกบนเว็บไซต์ให้มีการเคลื่อนไหวที่น่าสนใจ ขณะที่ W3Schools เป็นแหล่งเรียนรู้ออนไลน์ที่สอนเกี่ยวกับ SVG ตั้งแต่พื้นฐานจนถึงระดับกลาง โดยเน้นการสอนวิธีการสร้างและจัดการกับองค์ประกอบใน SVG รวมถึงการใช้งานร่วมกับ CSS และ JavaScript เพื่อเพิ่มความสามารถในการโต้ตอบกับผู้ใช้

W3Schools สอนพื้นฐานการใช้งาน SVG เบื้องต้น เช่น การสร้างเส้น รูปทรงพื้นฐาน เช่น สี่เหลี่ยม วงกลม และเส้นโค้ง รวมถึงการใช้งานคุณสมบัติต่างๆ เช่น การกำหนดสี และการปรับขนาดที่สามารถปรับเปลี่ยนได้ตามขนาดของหน้าจอ โดยมีตัวอย่างการสร้าง SVG เช่น


```
<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow" />
</svg>
```

โค้ดนี้จะสร้างวงกลมขนาด 40px ที่มีขอบสีเขียวและสีพื้นหลังเป็นสีเหลือง ภายในกรอบขนาด 100x100px

ส่วนโครงงานนี้ ใช้ SVG ในการสร้างกราฟิกที่สามารถปรับขนาดได้พร้อมกับการใช้ CSS Animation และ JavaScript ในการควบคุมการเคลื่อนไหวขององค์ประกอบต่างๆ เช่น การเคลื่อนไหวของเส้นทาง (path) หรือการสร้างแอนิเมชันที่สามารถควบคุมการแสดงผลในระยะเวลาและตำแหน่งได้ ตัวอย่างโค้ดจากโครงงานนี้ที่ใช้ SVG ร่วมกับ CSS เพื่อสร้างแอนิเมชัน

```
<svg width="200" height="200">
  <path d="M10 80 C 40 10, 65 10, 95 80 S 150 150, 180 80" stroke="black"
fill="transparent" stroke-width="5">
    <animate attributeName="d" values="M10 80 C 40 10, 65 10, 95 80 S 150 150, 180
80; M10 80 C 50 20, 75 30, 105 80 S 160 160, 190 80" dur="2s"
repeatCount="indefinite" />
  </path>
</svg>
```

โค้ดนี้ใช้ SVG path เพื่อสร้างเส้นโค้งที่เคลื่อนไหวเปลี่ยนแปลงรูปทรงไปตามเวลาที่กำหนด โดยการใช้ animate สำหรับการปรับเปลี่ยนค่าของ d attribute ซึ่งควบคุมเส้นทางของการเคลื่อนไหว

สรุป

W3Schools สอนการใช้งาน SVG ในระดับพื้นฐานที่เหมาะสมสำหรับการสร้างกราฟิกพื้นฐาน และการใช้งานร่วมกับ CSS และ JavaScript โดยเน้นที่การจัดการกับรูปทรงพื้นฐาน และการปรับขนาด ให้เหมาะสมกับขนาดหน้าจอ

ส่วนโครงการนี้ เน้นการใช้งาน SVG ในการสร้างแอนิเมชันที่มีความซับซ้อนมากขึ้น เช่น การควบคุม การเคลื่อนไหวขององค์ประกอบ SVG ด้วยการใช้ CSS Animation และ JavaScript เพื่อสร้างแอนิเมชันที่สามารถโต้ตอบกับผู้ใช้ได้ และสร้างความน่าสนใจให้กับเว็บไซต์

W3Schools ใช้ฟีเจอร์ Try it Yourself ที่ช่วยให้ผู้เรียนสามารถทดลองโค้ดและดูผลลัพธ์ได้ทันที แต่ต้องกดปุ่ม Run ทุกครั้งที่แก้ไขโค้ด ในขณะที่ โครงการนี้ มีฟีเจอร์ Hot Reload ที่ทำให้ผลลัพธ์แสดงทันทีหลังจากการแก้ไขโค้ดโดยไม่ต้องกดปุ่ม Run

2.5 ผู้เรียนควรมีพื้นฐานต่อไปนี้ก่อน

2.5.1 ผู้เรียนควรมีพื้นฐาน CSS ต่อไปนี้ก่อน

(1) Selectors and Properties

- 1.เข้าใจการใช้ selectors เช่น class, id, และ element เพื่อนำไปใช้กับองค์ประกอบต่างๆ ใน HTML
- 2.การกำหนดคุณสมบัติต่างๆ เช่น สี (color), ขนาดตัวอักษร (font-size), ความสูงและความกว้าง (height, width), ขอบ (border), ระยะห่าง (margin, padding)

(2) การจัดตำแหน่งและเค้าโครง (Layout)

- 1.การใช้ display, position, float, และ flexbox ในการจัดการเค้าโครงของเว็บเพจ เช่น การจัดเรียงองค์ประกอบทั้งในแนวนอนและแนวตั้ง

(3) การใช้ @keyframes และการสร้างแอนิเมชัน (CSS Animations)

- 1.การใช้ @keyframes เพื่อสร้างลำดับการเคลื่อนไหวหรือเปลี่ยนแปลงสไตล์ขององค์ประกอบ
- 2.การใช้ animation property เพื่อเชื่อมโยงการเคลื่อนไหวกับองค์ประกอบ

(4) การใช้ Transition

- 1.การใช้ transition เพื่อทำให้การเปลี่ยนแปลงคุณสมบัติขององค์ประกอบเกิดขึ้นอย่างลื่นไหล เช่น การเปลี่ยนสี, ขนาด, หรือสถานะอื่นๆ เมื่อมีการโต้ตอบกับผู้ใช้

การเรียนรู้พื้นฐานเหล่านี้จาก W3Schools จะช่วยให้ผู้เรียนเข้าใจหลักการใช้งาน CSS ได้ดีขึ้น และสามารถสร้างแอนิเมชันพื้นฐานหรือปรับปรุงการจัดระเบียบในเว็บไซต์ได้ ก่อนที่จะไปพัฒนาแอนิเมชันที่ซับซ้อนขึ้นในโครงการ

2.5.2 ผู้เรียนควรมีพื้นฐาน Javascript ต่อไปนี้ก่อน

(1) การประกาศตัวแปร (Variables)

เข้าใจวิธีการประกาศตัวแปรด้วย var, let, และ const
และการใช้งานตัวแปรในการเก็บข้อมูลต่างๆ เช่น ข้อความ, ตัวเลข,
หรือค่าต่างๆ

(2) ประเภทข้อมูล (Data Types)

รู้จักประเภทข้อมูลต่างๆ เช่น string, number, boolean, object, array,
และ null รวมถึงวิธีการใช้งานและแปลงประเภทข้อมูล

(3) การใช้ฟังก์ชัน (Functions)

เข้าใจการสร้างและใช้งานฟังก์ชัน
การรับพารามิเตอร์และการคืนค่าจากฟังก์ชัน เช่น การใช้ return
เพื่อส่งค่ากลับ

(4) การควบคุมการไหล (Control Flow)

เข้าใจการใช้คำสั่งควบคุม เช่น if, else, switch, for, while และ do-while
ในการควบคุมการทำงานของโค้ดตามเงื่อนไขหรือการวนลูป

(5) การทำงานกับออบเจกต์ (Objects)

รู้จักการสร้างและใช้งานออบเจกต์
การเข้าถึงและปรับเปลี่ยนคุณสมบัติของออบเจกต์
และการใช้เมธอดภายในออบเจกต์

(6) การทำงานกับอาร์เรย์ (Arrays)

เข้าใจการประกาศและใช้งานอาร์เรย์ การเข้าถึงค่าภายในอาร์เรย์
การเพิ่มหรือแก้ไขข้อมูลในอาร์เรย์ และการใช้เมธอดต่างๆ เช่น push(),
pop(), shift(), unshift(), map(), filter(), และ reduce()

(7) การจัดการกับเหตุการณ์ (Event Handling)

เข้าใจการใช้ addEventListener และการจัดการเหตุการณ์ต่างๆ เช่น
การคลิกปุ่ม, การเลื่อนเมาส์, หรือการกดปุ่มแป้นพิมพ์
เพื่อให้เว็บไซต์มีความโต้ตอบกับผู้ใช้

(8) การทำงานกับ DOM (Document Object Model)

เข้าใจการใช้ JavaScript ในการเข้าถึงและแก้ไข DOM เช่น การเปลี่ยนแปลงข้อความใน HTML, การเพิ่มหรือลบองค์ประกอบ HTML, การปรับแต่ง CSS ขององค์ประกอบ

(9) การใช้ Promise และ Asynchronous JavaScript

รู้จักการทำงานแบบอะซิงโครนัส เช่น การใช้ setTimeout(), setInterval(), fetch(), และการจัดการ Promise เพื่อให้สามารถทำงานพร้อมกันหลายๆ อย่างได้โดยไม่ทำให้เว็บไซต์หยุดชะงัก

การมีพื้นฐานเหล่านี้จะช่วยให้ผู้เรียนสามารถเข้าใจและนำ JavaScript ไปใช้ในโครงการได้อย่างมีประสิทธิภาพ ทั้งในเรื่องการพัฒนาแอนิเมชัน การจัดการ DOM และการสร้างฟังก์ชันที่ตอบสนองต่อการกระทำของผู้ใช้

2.5.3 ผู้เรียนไม่จำเป็นต้องมีพื้นฐานการทำ SVG มาก่อน

2.5.4 ผู้เรียนไม่จำเป็นต้องมีพื้นฐานการทำ 3D และภาพเสมือนจริง มาก่อน

2.5.5 ผู้เรียนไม่จำเป็นต้องมีพื้นฐานการทำ font animation มาก่อน

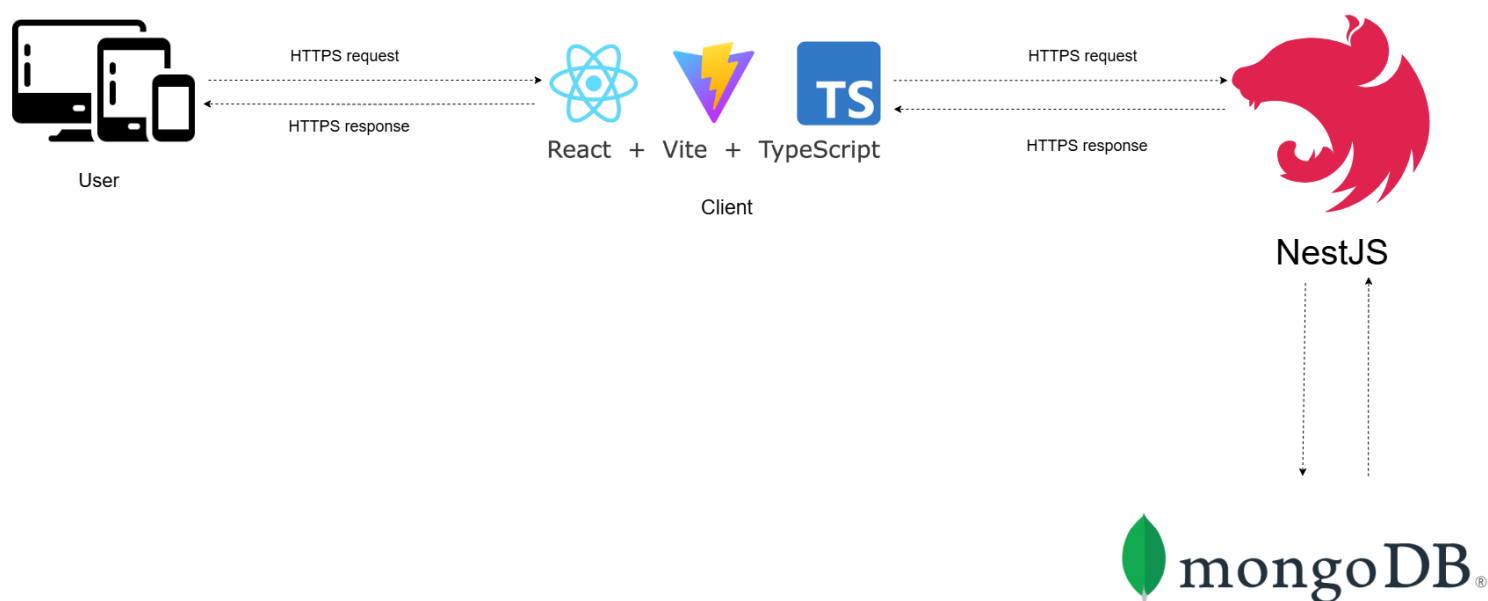
2.6 เว็บไซต์ที่สอนหรือรวบรวมเนื้อหาขั้น Intermediate ถึง Advance

- (1) <https://tympanus.net/codrops> รวบรวมทริคด้านfrontend
- (2) <https://www.awwwards.com> รวบรวมเว็บไซต์ที่ได้รับรางวัล
- (3) <https://www.youtube.com/@Hyperplexed> สอนทริคด้าน frontend
- (4) <https://www.youtube.com/@DesignCourse> สอนทริคด้าน frontend
- (5) <https://immersive-g.com> รวบรวมเว็บไซต์ที่ได้รับรางวัล
- (6) <https://css-tricks.com> สอนทริคด้าน CSS
- (7) <https://www.htmhell.dev> สอนทริคด้าน html
- (8) <https://codepen.io> รวบรวมทริคด้านต่างๆ

บทที่ 3

วิธีการวิจัย

3.1 สถาปัตยกรรมของระบบ

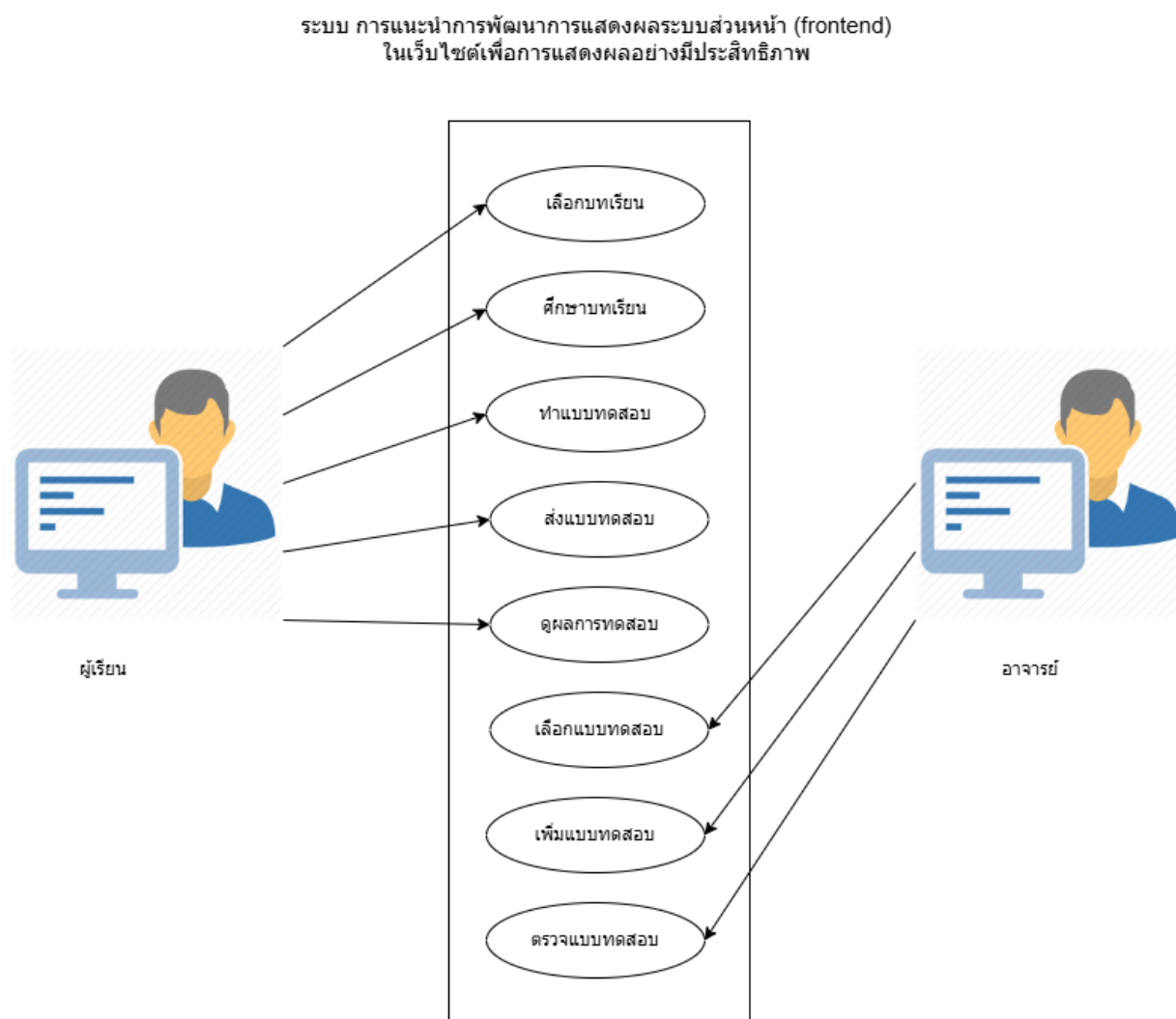


ภาพที่ 3.1 สถาปัตยกรรมของระบบ

จากรูปโครงสร้างสถาปัตยกรรมของระบบ ระบบแนะนำเทคนิคการพัฒนาแอนิเมชันบนเว็บไซต์แบ่งสถาปัตยกรรมของระบบได้ดังนี้

1. User คือผู้ใช้งาน ระบบแนะนำการพัฒนาการแสดงผลระบบส่วนหน้า (frontend) ในเว็บไซต์เพื่อการแสดงผลอย่างมีประสิทธิภาพ
2. Client คือ หน้าของเว็บแอปพลิเคชันระบบแนะนำเทคนิคการพัฒนาแอนิเมชันบนเว็บไซต์
3. NestJS คือ เป็นระบบ web server backend
4. MongoDB คือ database ที่ใช้เก็บข้อมูล

3.2 การวิเคราะห์ขอบเขตและความต้องการของระบบ



ภาพที่ 3.2 use case diagram

การออกแบบขั้นตอนการทำงานของระบบ

ตาราง 3.1 รายละเอียดกรณีการใช้งานของระบบ use case description

Use Case ID	Use Case Name	Description
UC1	เลือกบทเรียน	ผู้ใช้งานสามารถเลือกบทเรียนที่ต้องการศึกษาได้จากรายการบทเรียนที่มีอยู่ในระบบ
UC2	ศึกษาบทเรียน	ผู้ใช้งานสามารถศึกษาบทเรียนที่ต้องการศึกษาได้จากรายการบทเรียนที่มีอยู่ในระบบ
UC3	ทำแบบทดสอบ	ผู้ใช้งานสามารถเลือกทำแบบทดสอบที่เกี่ยวข้องกับบทเรียนเพื่อประเมินความเข้าใจ
UC4	ส่งแบบทดสอบ	ผู้ใช้งานส่งคำตอบที่ทำเสร็จแล้วให้ระบบประเมินผล
UC5	ดูผลการทดสอบ	ผู้ใช้งานสามารถดูผลการทดสอบและดูเฉลย
UC6	เลือกแบบทดสอบ	ผู้สอนสามารถเลือกแบบทดสอบจากรายการแบบทดสอบที่มีในระบบเพื่อกำหนดให้ผู้เรียน
UC7	เพิ่มแบบทดสอบ	ผู้สอนสามารถสร้างและเพิ่มแบบทดสอบใหม่ในระบบ เช่น การตั้งคำถามหรือกำหนดตัวเลือกคำตอบ
UC8	ตรวจแบบทดสอบ	ผู้สอนสามารถตรวจสอบและแก้ไขผลการทดสอบหรือคำตอบที่ระบบไม่สามารถประเมินได้(ส่วนอัตโนมัติ)

ตาราง 3.2 ตารางแสดง use case description เลือกบทเรียน

รหัสยูสเคส (Use Case ID)	UC1
ชื่อยูสเคส (User Case Name)	เลือกบทเรียน
ผู้ใช้งาน (Actor)	ผู้เรียน
คำอธิบาย (Description)	ผู้เรียนเลือกบทเรียนที่ต้องการศึกษาได้จาก รายการบทเรียนที่แสดงอยู่ในระบบ
เงื่อนไขก่อนหน้า (Pre-condition)	-
เงื่อนไขภายหลัง (Post-condition)	ระบบแสดงหน้าบทเรียนที่ผู้เรียนเลือก
กระแสหลัก (Basic Flow)	<ol style="list-style-type: none"> 1. ผู้เรียนเข้าสู่หน้ารายการบทเรียน 2. ผู้เรียนเลือกบทเรียนจากรายการ 3. ระบบแสดงเนื้อหาบทเรียนที่เลือก
กระแสรอง (Alternative Flow)	<p>2.a หากไม่มีบทเรียนที่เลือกไว้</p> <p>ระบบแสดงข้อความแจ้งเตือน</p>

ตาราง 3.3 ตารางแสดง use case description ศึกษาบทเรียน

รหัสยูสเคส (Use Case ID)	UC2
ชื่อยูสเคส (User Case Name)	ศึกษาบทเรียน
ผู้ใช้งาน (Actor)	ผู้เรียน
คำอธิบาย (Description)	ผู้เรียนสามารถอ่านดูหรือทดลองตามเนื้อหาบทเรียนที่ระบบแสดง
เงื่อนไขก่อนหน้า (Pre-condition)	ผู้เรียนเลือกบทเรียนสำเร็จ
เงื่อนไขภายหลัง (Post-condition)	-
กระแสหลัก (Basic Flow)	1. ระบบแสดงเนื้อหาบทเรียน 2. ผู้เรียนศึกษาเนื้อหา
กระแสรอง (Alternative Flow)	1.a หากเกิดข้อผิดพลาดในการโหลดเนื้อหา ระบบแจ้งเตือนและเสนอให้ลองใหม่

ตาราง 3.4 ตารางแสดง use case description ทำแบบทดสอบ

รหัสยูสเคส (Use Case ID)	UC3
ชื่อยูสเคส (User Case Name)	ทำแบบทดสอบ
ผู้ใช้งาน (Actor)	ผู้เรียน
คำอธิบาย (Description)	ผู้เรียนทำแบบทดสอบที่เกี่ยวข้องกับบทเรียน
เงื่อนไขก่อนหน้า (Pre-condition)	ผู้เรียนเลือกแบบทดสอบสำเร็จ
เงื่อนไขภายหลัง (Post-condition)	ระบบบันทึกคำตอบของผู้เรียน
กระแสหลัก (Basic Flow)	<ol style="list-style-type: none"> 1. ผู้เรียนเริ่มทำแบบทดสอบ 2. ผู้เรียนตอบคำถามแต่ละข้อ 3. ระบบบันทึกคำตอบ
กระแสรอง (Alternative Flow)	<p>3.a หากเกิดข้อผิดพลาดในการบันทึกคำตอบระบบแจ้งเตือนและเสนอให้ส่งใหม่</p>

ตาราง 3.5 ตารางแสดง use case description ส่งแบบทดสอบ

รหัสยูสเคส (Use Case ID)	UC4
ชื่อยูสเคส (User Case Name)	ส่งแบบทดสอบ
ผู้ใช้งาน (Actor)	ผู้เรียน
คำอธิบาย (Description)	ผู้เรียนส่งแบบทดสอบเพื่อประเมินผล
เงื่อนไขก่อนหน้า (Pre-condition)	ผู้เรียนทำแบบทดสอบ
เงื่อนไขภายหลัง (Post-condition)	ระบบประเมินผลและแสดงคะแนน
กระแสหลัก (Basic Flow)	<ol style="list-style-type: none"> 1. ผู้เรียนกดปุ่มส่งแบบทดสอบ 2. ระบบบันทึกแบบทดสอบ 3. ระบบประเมินผลแบบทดสอบ
กระแสรอง (Alternative Flow)	<p>2.a หากระบบไม่สามารถประเมินผลได้</p> <p>ระบบแจ้งผู้สอน</p>

ตาราง 3.6 ตารางแสดง use case description ดูผลการทดสอบ

รหัสยูสเคส (Use Case ID)	UC5
ชื่อยูสเคส (User Case Name)	ดูผลการทดสอบ
ผู้ใช้งาน (Actor)	ผู้เรียน
คำอธิบาย (Description)	ผู้เรียนดูผลคะแนนและคำตอบที่ถูกหรือผิด
เงื่อนไขก่อนหน้า (Pre-condition)	ผู้เรียนส่งแบบทดสอบสำเร็จ
เงื่อนไขภายหลัง (Post-condition)	-
กระแสหลัก (Basic Flow)	<ol style="list-style-type: none"> 1. ผู้เรียนเลือกดูผลการทดสอบ 2. ระบบแสดงคะแนนและคำตอบที่ถูกหรือผิด
กระแสรอง (Alternative Flow)	2.a หากเกิดข้อผิดพลาดระบบแจ้งผู้เรียนและเสนอให้ลองใหม่

ตาราง 3.7 ตารางแสดง use case description เลือกแบบทดสอบ

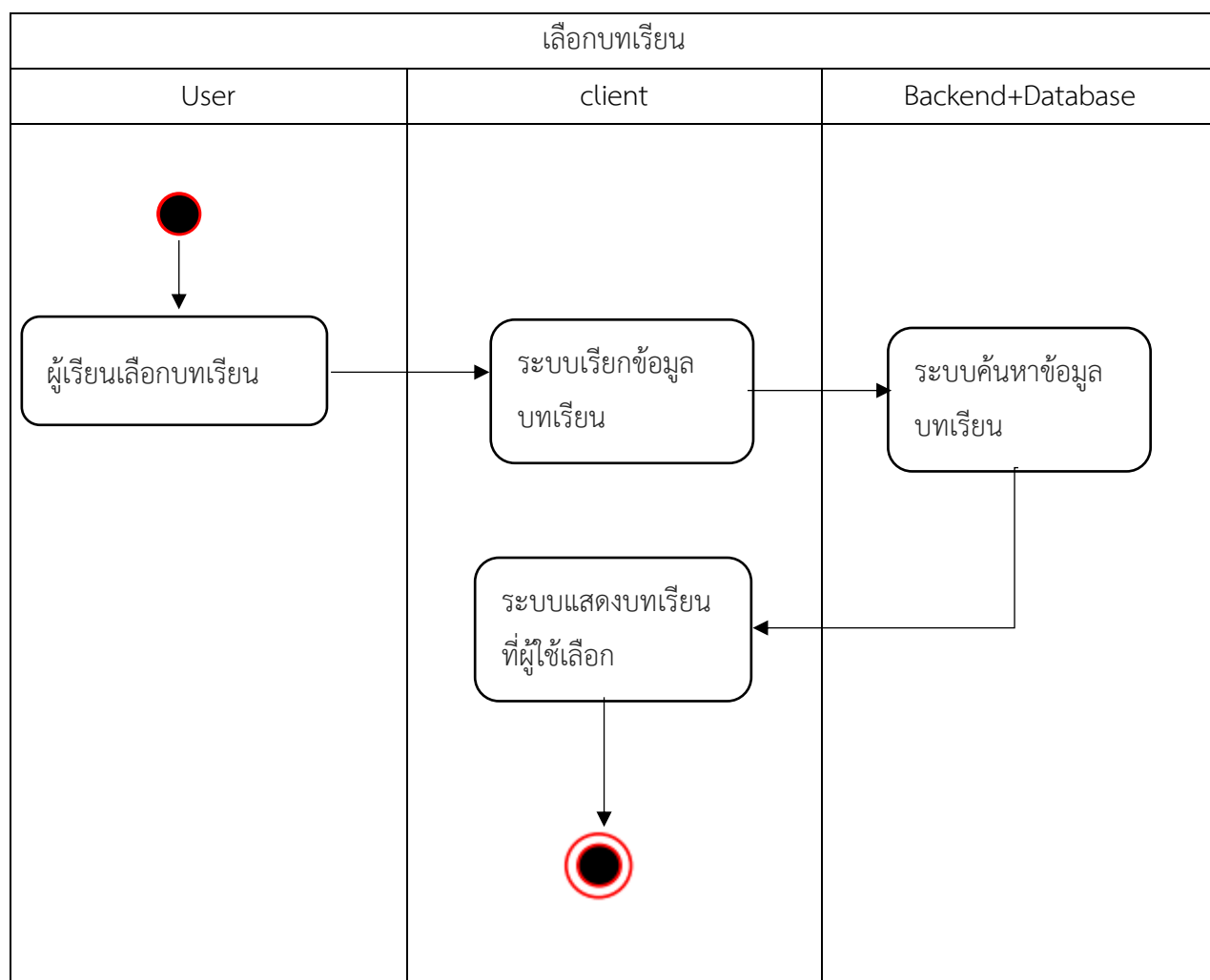
รหัสยูสเคส (Use Case ID)	UC6
ชื่อยูสเคส (User Case Name)	เลือกแบบทดสอบ
ผู้ใช้งาน (Actor)	ผู้สอน
คำอธิบาย (Description)	ผู้สอนเลือกแบบทดสอบจากรายการเพื่อบันทึก ผู้เรียน
เงื่อนไขก่อนหน้า (Pre-condition)	ผู้สอนเข้าสู่ระบบสำเร็จ
เงื่อนไขภายหลัง (Post-condition)	แบบทดสอบถูกกำหนดให้ผู้เรียน
กระแสหลัก (Basic Flow)	1. ผู้สอนเข้าหน้าจัดการแบบทดสอบ 2. ผู้สอนเลือกแบบทดสอบ 3. ระบบบันทึกแบบทดสอบที่เลือก
กระแสรอง (Alternative Flow)	2.a หากไม่มีแบบทดสอบในระบบ ระบบแจ้งเตือน

ตาราง 3.8 ตารางแสดง use case description เพิ่มแบบทดสอบ

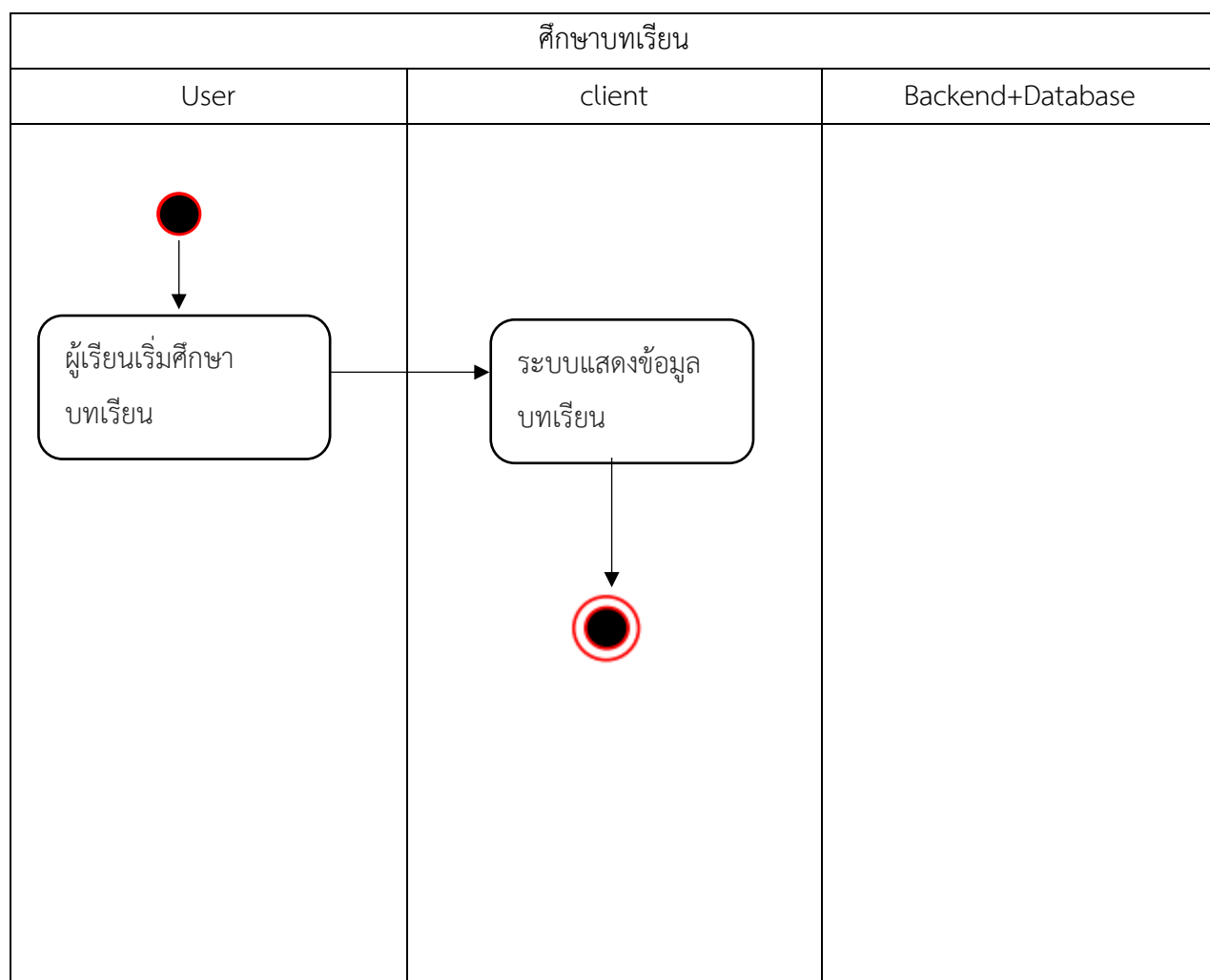
รหัสยูสเคส (Use Case ID)	UC7
ชื่อยูสเคส (User Case Name)	เพิ่มแบบทดสอบ
ผู้ใช้งาน (Actor)	ผู้สอน
คำอธิบาย (Description)	ผู้สอนเพิ่มแบบทดสอบใหม่ในระบบ
เงื่อนไขก่อนหน้า (Pre-condition)	ผู้สอนเข้าสู่หน้าจัดการแบบทดสอบสำเร็จ
เงื่อนไขภายหลัง (Post-condition)	แบบทดสอบใหม่ถูกเพิ่มในระบบ
กระแสหลัก (Basic Flow)	<ol style="list-style-type: none"> 1. ผู้สอนเลือกเพิ่มแบบทดสอบใหม่ 2. ผู้สอนกรอกข้อมูลแบบทดสอบ 3. ระบบบันทึกแบบทดสอบ
กระแสรอง (Alternative Flow)	-

ตาราง 3.9 ตารางแสดง use case description ตรวจแบบทดสอบ

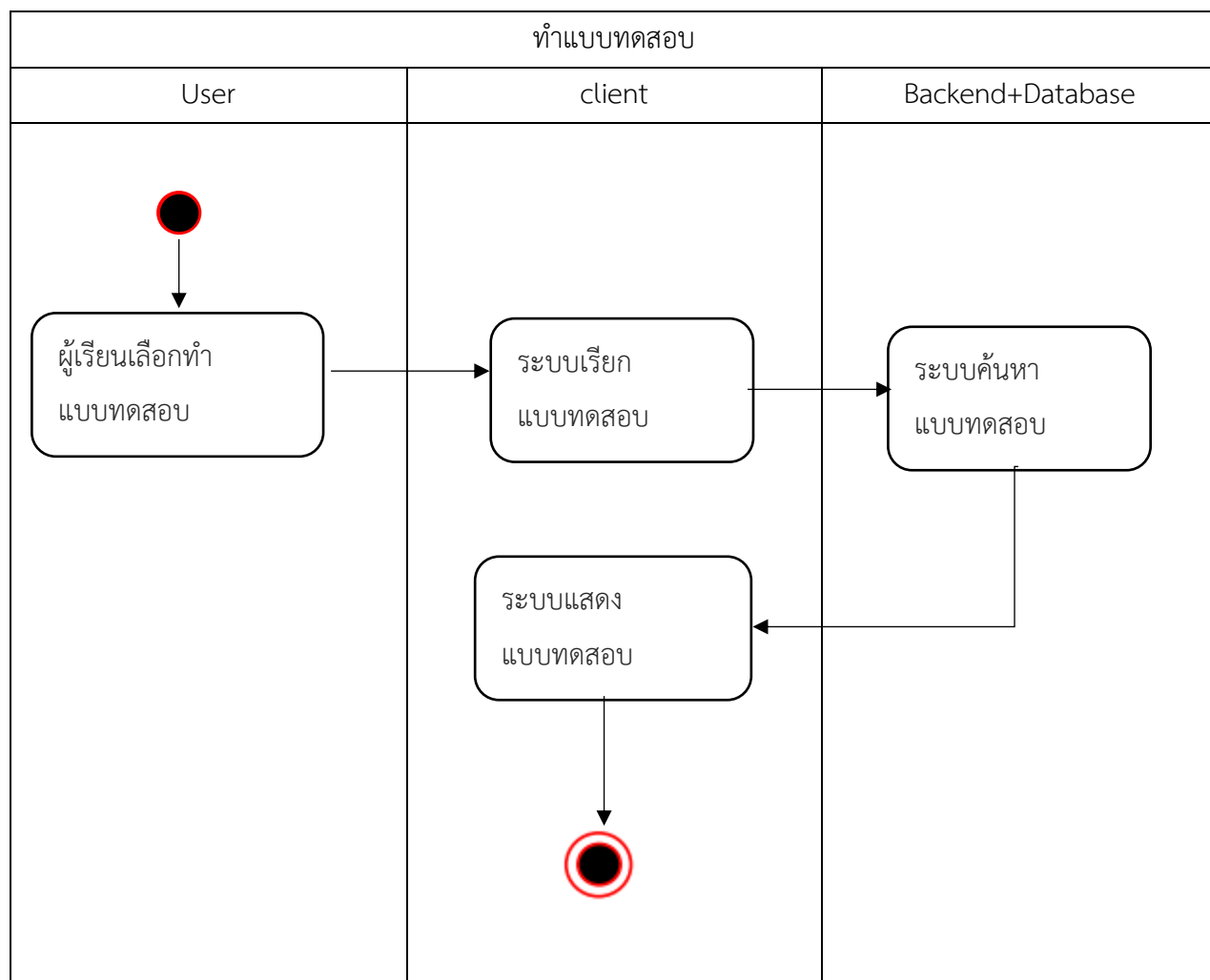
รหัสยูสเคส (Use Case ID)	UC8
ชื่อยูสเคส (User Case Name)	ตรวจแบบทดสอบ
ผู้ใช้งาน (Actor)	ผู้สอน
คำอธิบาย (Description)	ผู้สอนตรวจสอบและประเมินแบบทดสอบ
เงื่อนไขก่อนหน้า (Pre-condition)	มีแบบทดสอบที่ต้องการตรวจ
เงื่อนไขภายหลัง (Post-condition)	แบบทดสอบถูกตรวจสอบและประเมิน
กระแสหลัก (Basic Flow)	<ol style="list-style-type: none"> 1. ผู้สอนเลือกแบบทดสอบที่ต้องการตรวจ 2. ผู้สอนตรวจสอบคำตอบ 3. ผู้สอนบันทึกผลการตรวจ
กระแสรอง (Alternative Flow)	2.a หากไม่มีข้อผิดพลาดระบบแจ้งให้ผู้สอนทราบ



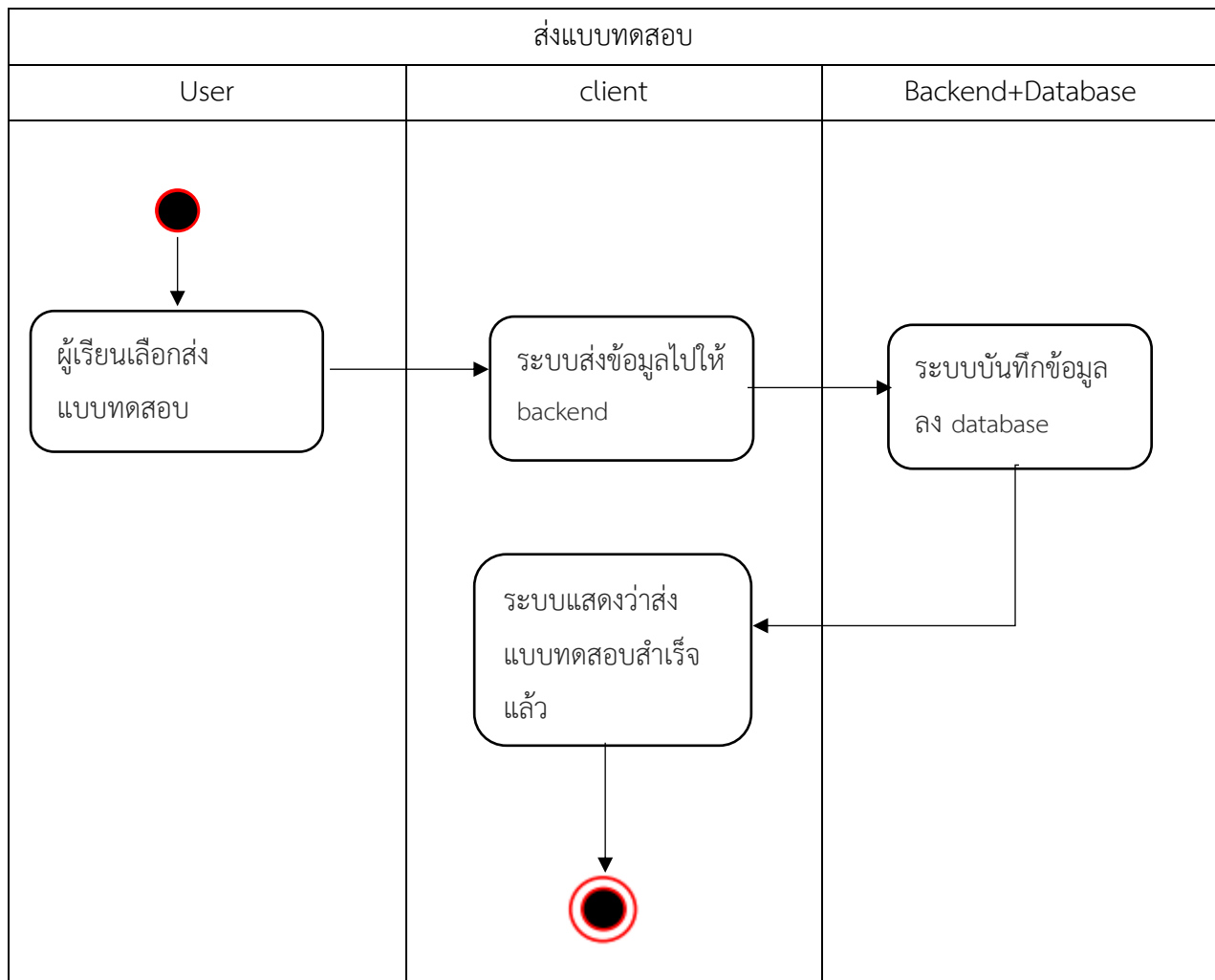
ภาพที่ 3.3 activity diagram เลือกบทเรียน



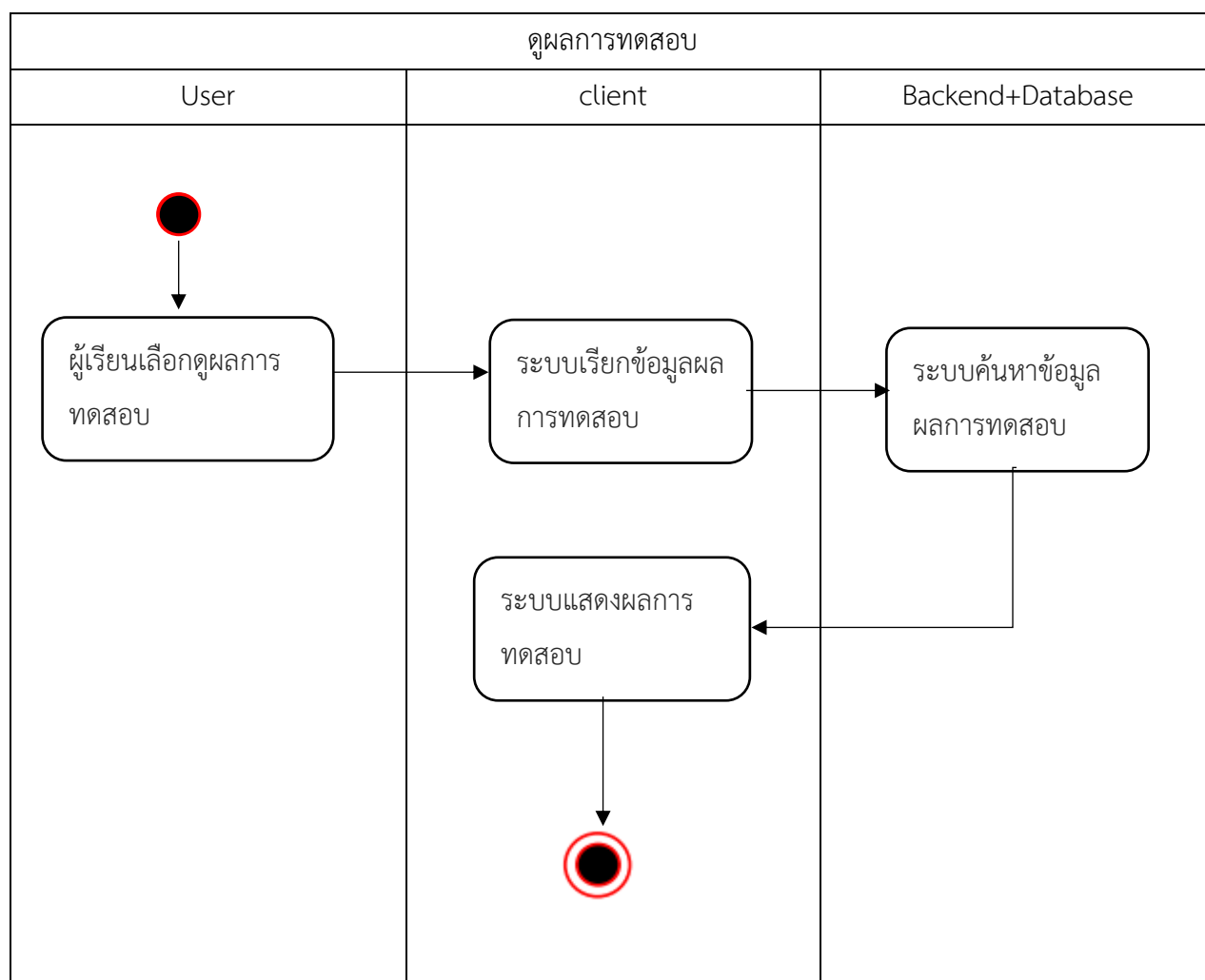
ภาพที่ 3.4 activity diagram ศึกษาบทเรียน



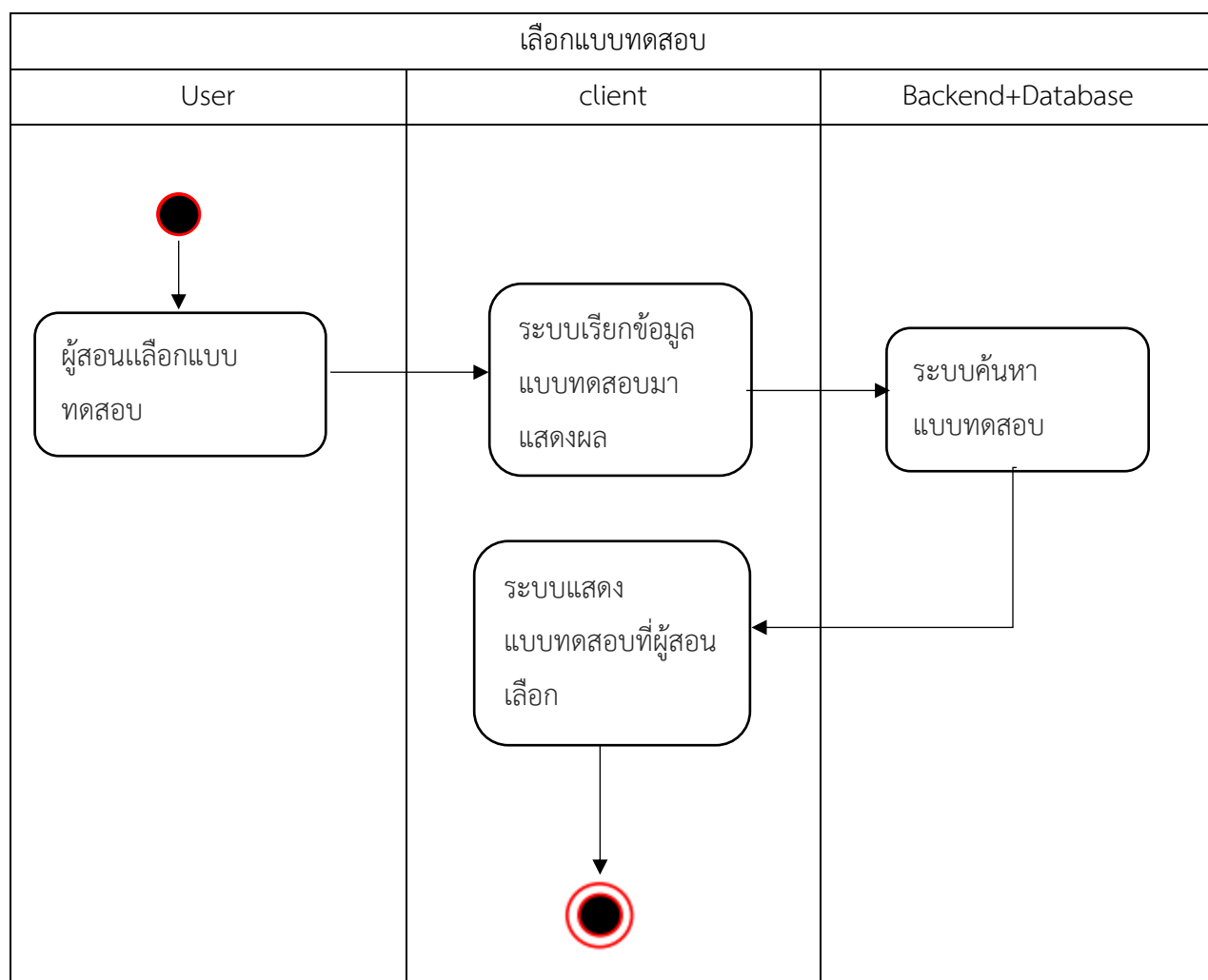
ภาพที่ 3.5 activity diagram ทำแบบทดสอบ



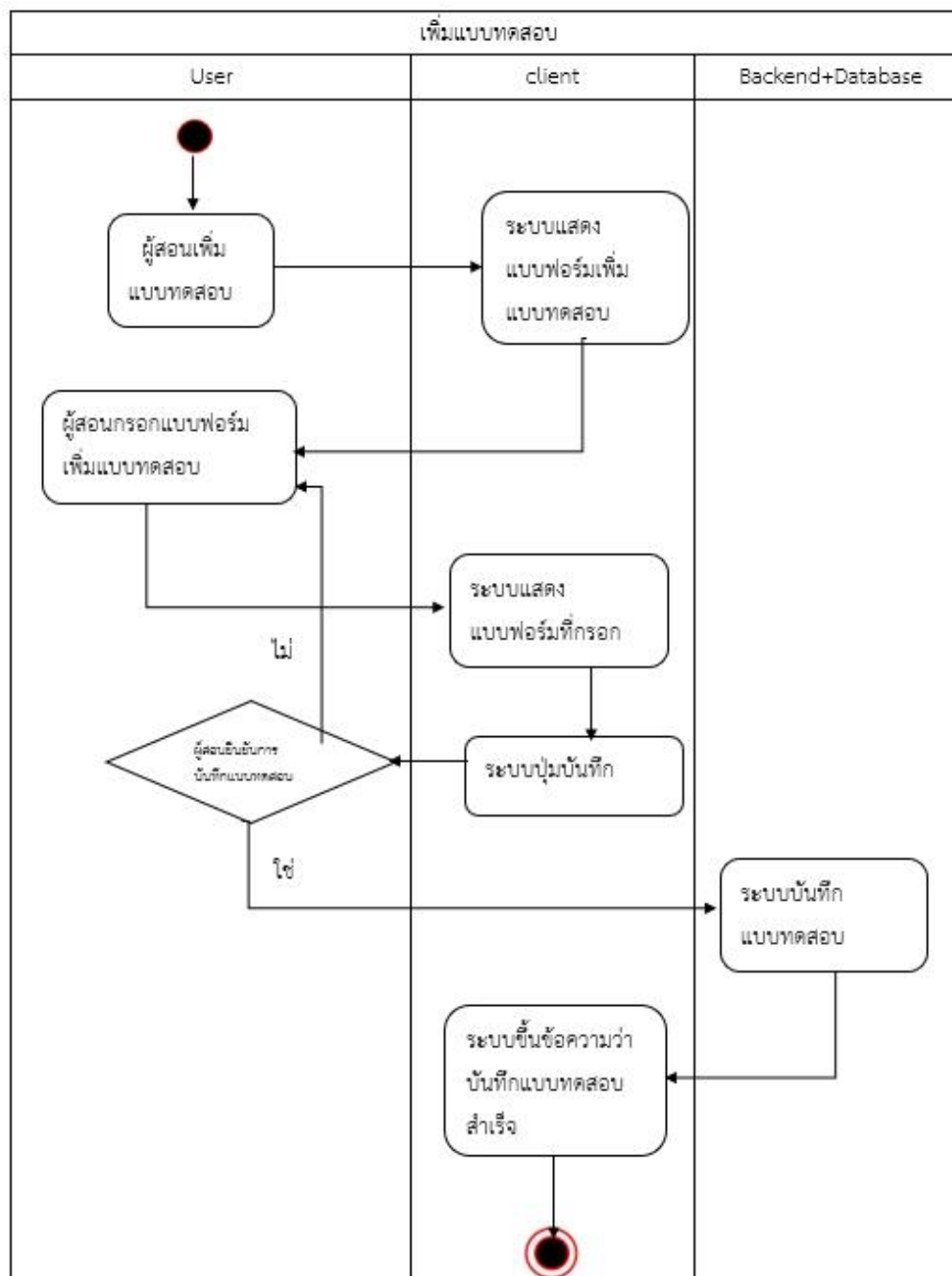
ภาพที่ 3.6 activity diagram ส่งแบบทดสอบ



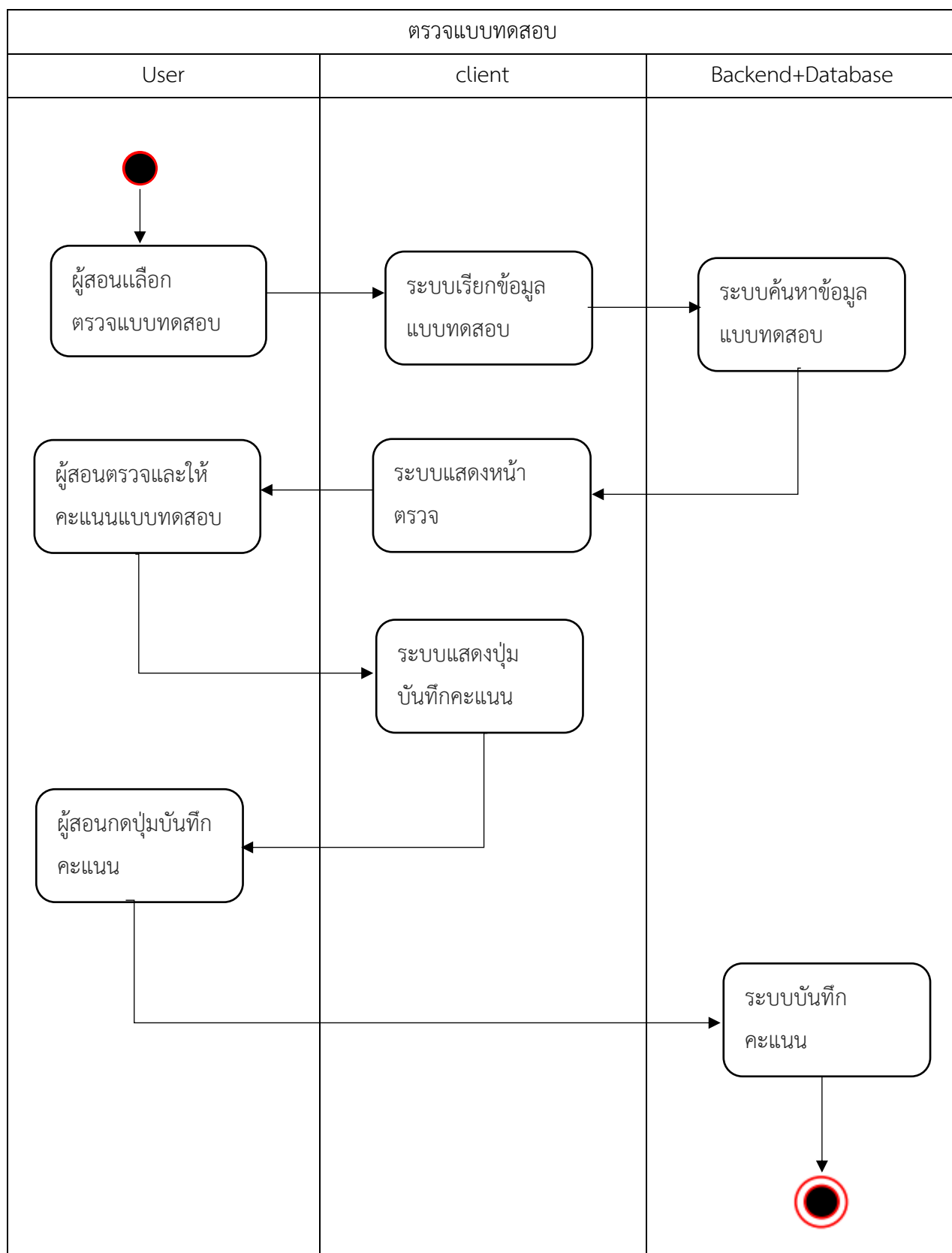
ภาพที่ 3.7 activity diagram ดูผลการทดสอบ



ภาพที่ 3.8 activity diagram เลือกแบบทดสอบ

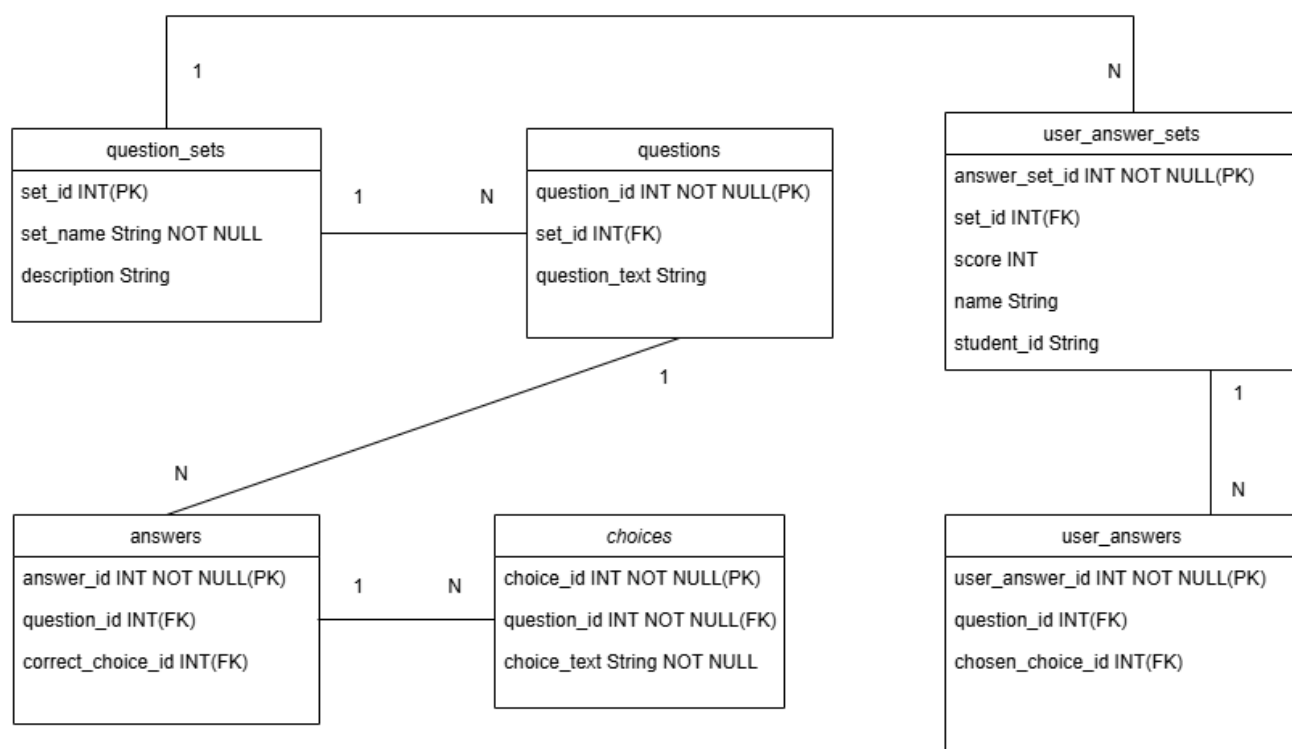


ภาพที่ 3.9 activity diagram เพิ่มแบบทดสอบ



ภาพที่ 3.10 activity diagram ตรวจแบบทดสอบ

3.4 การออกแบบฐานข้อมูลของระบบ (Entity-Relationship Diagrams)



ภาพที่ 3.11 Entity-Relationship Diagrams

ตาราง 3.10 question_sets ประกอบด้วยลำดับข้อมูลในตาราง ดังนี้

No.	Attribute Name	Data Type		Constraint	Description
1	set_id	int	not null	PK	รหัสนักศึกษา
2	set_name	string	not null		ชื่อ
3	description	string			คำอธิบาย

ตาราง 3.11 questions ประกอบด้วยลำดับข้อมูลในตาราง ดังนี้

No.	Attribute Name	Data Type		Constraint	Description
1	question_id	int	not null	PK	คำถาม
2	set_id	int	not null	FK	ชุดคำถาม
3	question_text	string	not null		คำถาม

ตาราง 3.12 answers ประกอบด้วยลำดับข้อมูลในตาราง ดังนี้

No.	Attribute Name	Data Type		Constraint	Description
1	answer_id	int	not null	PK	คำตอบ
2	question_id	int	not null	FK	คำถาม
3	correct_choice_id	int	not null	FK	คำตอบที่ถูกต้อง

ตาราง 3.13 choices ประกอบด้วยลำดับข้อมูลในตาราง ดังนี้

No.	Attribute Name	Data Type		Constraint	Description
1	choice_id	int	not null	PK	ตัวเลือก
2	question_id	int	not null	FK	คำถาม
3	choice_text	string	not null		ตัวเลือก

ตาราง 3.14 user_answer_sets ประกอบด้วยลำดับข้อมูลในตาราง ดังนี้

No.	Attribute Name	Data Type		Constraint	Description
1	answer_set_id	int	not null	PK	คำตอบ
2	set_id	int	not null	FK	ชุด id
3	score	int			คะแนน
4	name	string			ชื่อ
5	student_id	string			รหัสนักศึกษา

ตาราง 3.15 user_answers ประกอบด้วยลำดับข้อมูลในตาราง ดังนี้

No.	Attribute Name	Data Type		Constraint	Description
1	user_answer_id	int	not null	PK	คำตอบ
2	question_id	int	not null	FK	คำถาม
3	chosen_choice_id	int	not null	FK	ตัวเลือก

3.3 ประเด็นที่น่าสนใจและสิ่งที่ท้าทาย

1. สร้างแอนิเมชันที่ลื่นไหลและมีประสิทธิภาพ
2. ใช้เครื่องมือและเทคนิคที่หลากหลาย เช่น GSAP และ SVG
3. รองรับการทำงานบนอุปกรณ์หลายรูปแบบ
4. ออกแบบแอนิเมชันให้เหมาะสมกับประสบการณ์ผู้ใช้
5. จัดการเวลาและลำดับของแอนิเมชันที่ซับซ้อน
6. ทดสอบและปรับปรุงแอนิเมชันให้ตอบโจทย์ทุกสถานการณ์
7. ศึกษาและประยุกต์ใช้เทคโนโลยีใหม่อย่างเหมาะสม
8. บริหารเวลาการพัฒนาให้เสร็จตามแผนงาน

3.4 ผลลัพธ์ที่คาดหวัง

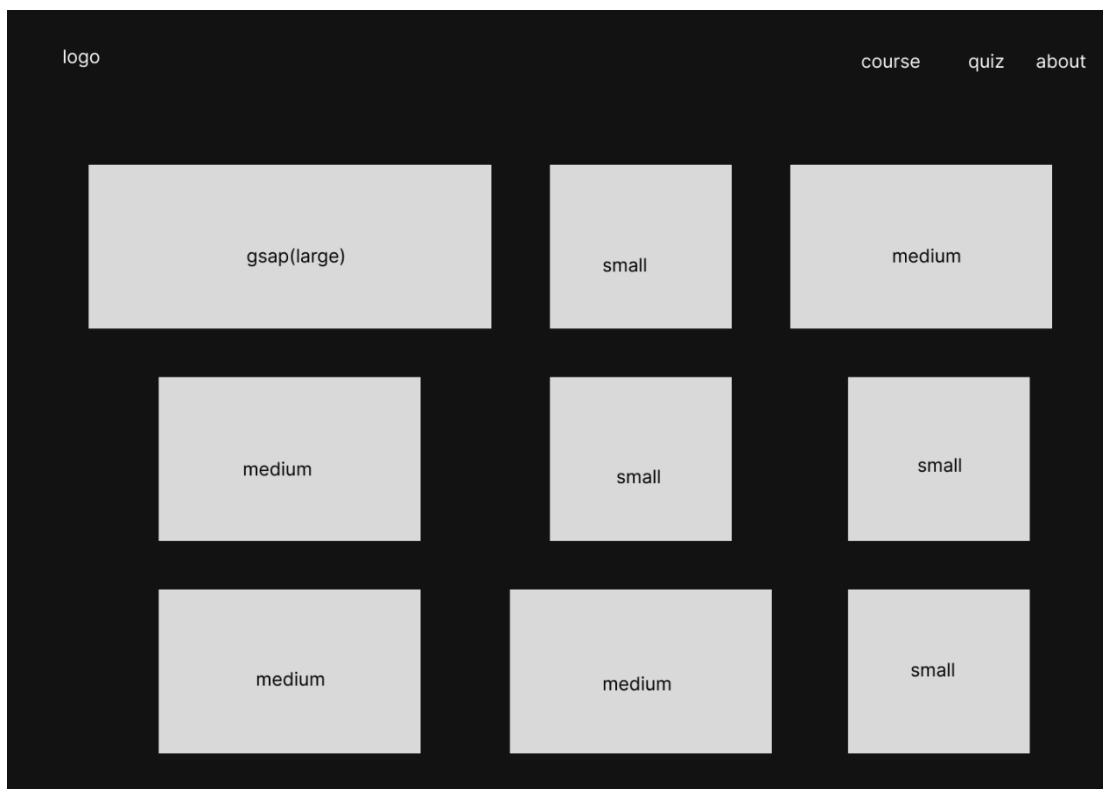
เว็บไซต์การแนะนำการพัฒนาการแสดงผลระบบส่วนหน้า (frontend)

เพื่อการแสดงผลอย่างมีประสิทธิภาพ เป็นระบบที่ถูกออกแบบให้ตรงตามวัตถุประสงค์ในบทที่

1 โดยมุ่งเน้นให้ผู้เรียนสามารถศึกษาและทดลองสร้างแอนิเมชันด้วยเทคนิคต่าง ๆ ได้แก่ CSS Animation, JavaScript Animation, SVG, 3D และ Font Animation ระบบนี้สามารถใช้เป็นต้นแบบสำหรับพัฒนาเนื้อหาการเรียนรู้แบบเชิงโต้ตอบในระดับที่สูงขึ้น โดยผลลัพธ์ที่คาดหวังมีรายละเอียดดังนี้

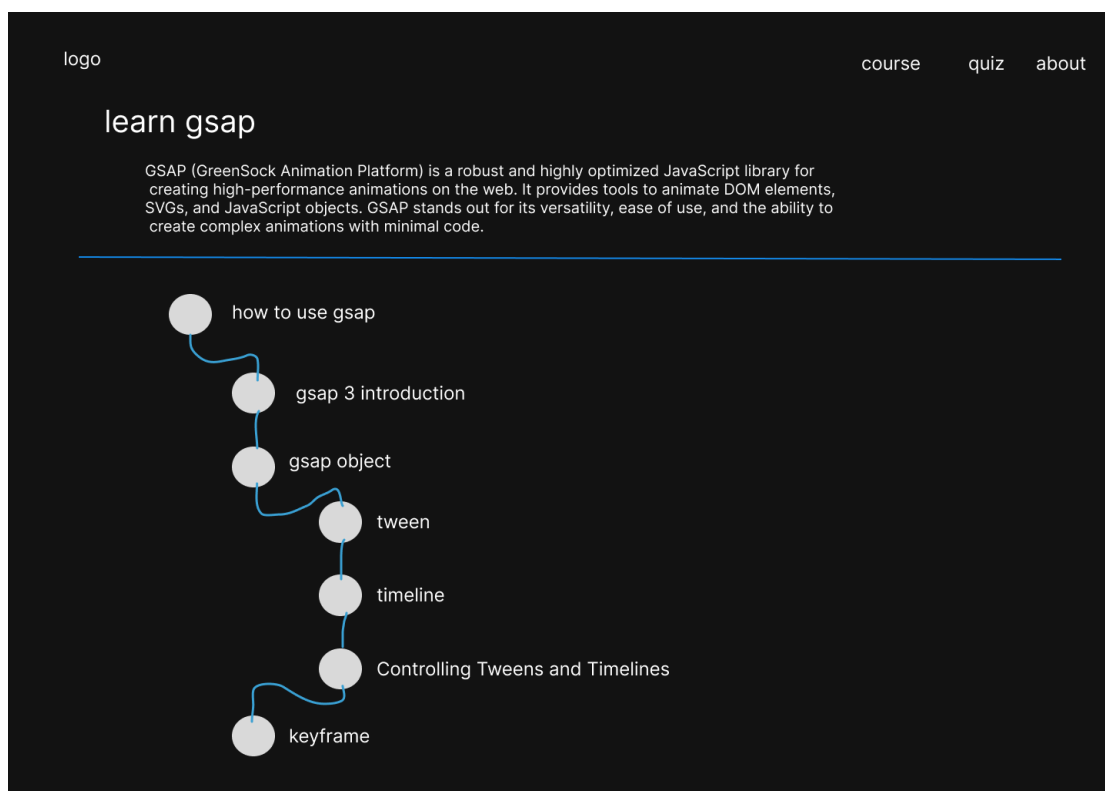
1. ผู้เรียนสามารถสร้างแอนิเมชันเว็บที่มีความลื่นไหลและใช้งานได้จริง
2. ระบบสามารถแสดงผลการทดลองโค้ดแบบโต้ตอบ ช่วยเพิ่มประสิทธิภาพการเรียนรู้
3. เนื้อหาที่สอนครอบคลุมตั้งแต่ระดับกลางจนถึงระดับสูงในแต่ละหัวข้อ
4. ผู้เรียนมีความเข้าใจในการเลือกใช้เทคนิคแอนิเมชันให้เหมาะสมกับบริบทของงาน
5. ระบบสามารถพัฒนาและต่อยอดให้เป็นแพลตฟอร์มสื่อการสอนในรูปแบบกว้างขวางมากขึ้นในอนาคต

3.5 ระบบต้นแบบและผลลัพธ์เบื้องต้น



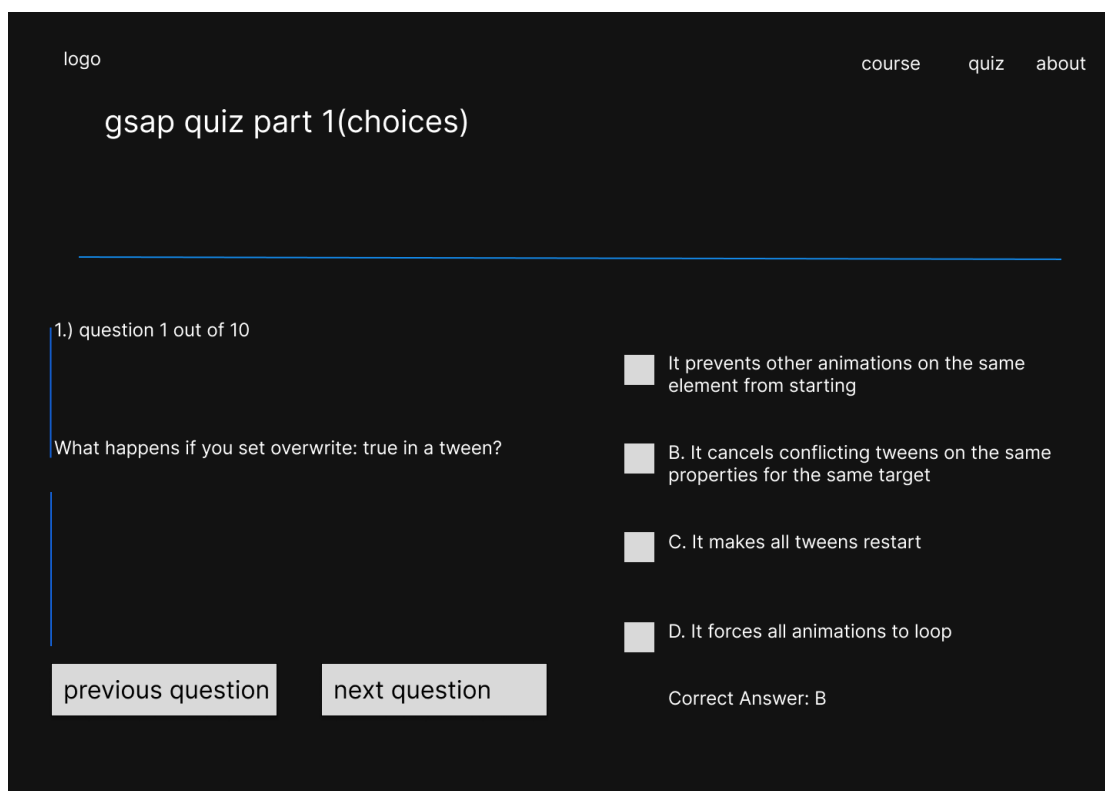
ภาพที่ 3.12 หน้า course ของระบบ

จากภาพที่ 3.12 แสดงภาพหน้า course ของระบบการพัฒนาการแสดงผลระบบ ส่วนหน้า (frontend) ในเว็บไซต์เพื่อการแสดงผลอย่างมีประสิทธิภาพ โดยผู้ใช้งานสามารถเลือกบทเรียนได้ในหน้านี้



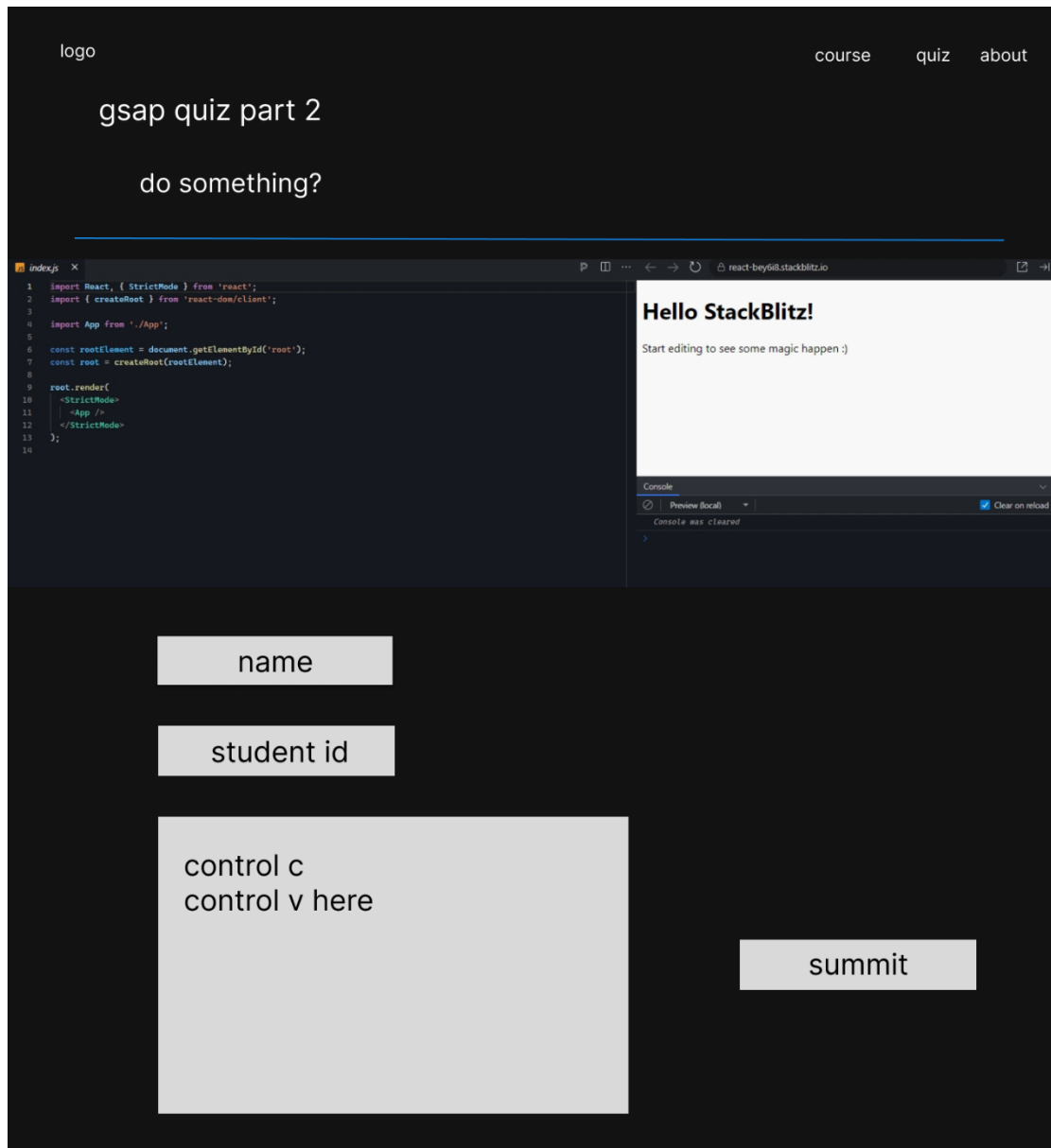
ภาพที่ 3.13 หน้า บทเรียนตัวอย่าง (gsap)

จากภาพที่ 3.13 แสดงภาพหน้า บทเรียนตัวอย่าง (gsap)



ภาพที่ 3.14 หน้า quiz ส่วนปรนัย

จากภาพที่ 3.14 แสดงภาพหน้า quiz ส่วนปรนัย ของระบบการพัฒนาการ
แสดงผลระบบส่วนหน้า (frontend) ในเว็บไซต์เพื่อการแสดงผลอย่างมีประสิทธิภาพ โดยผู้จัดทำ
แบบทดสอบเพื่อประเมินความเข้าใจ ได้ในหน้านี้



ภาพที่ 3.15 หน้า quiz ส่วนอตัณย

จากภาพที่ 3.15 แสดงภาพหน้า quiz ส่วนอตัณย ของระบบการพัฒนาการ
แสดงผลระบบส่วนหน้า (frontend) ในเว็บไซต์เพื่อการแสดงผลอย่างมีประสิทธิภาพ โดยผู้จัดทำ
แบบทดสอบเพื่อประเมินความเข้าใจ ได้ในหน้านี้

บทที่ 4

ทรัพยากรและแผนการดำเนินงาน

4.1 การจัดเตรียมฮาร์ดแวร์และซอฟต์แวร์

4.1.1 ภาษาที่ใช้ในการพัฒนา

- 1.) Javascript
- 2.) Typescript
- 3.) CSS

4.1.2 โปรแกรมที่ใช้ในการพัฒนา

- 1.) Visual Studio Code

4.1.3 คอมพิวเตอร์ที่ใช้ในการพัฒนา

คอมพิวเตอร์ตั้งโต๊ะ

4.1.4 Computer specification

Processor: AMD Ryzen 7 2700X

Memory:16 GB

Graphic card: NVIDIA GeForce RTX 2060

Storage: WD HDD 4TB 5400rpm SATA III 64MB Blue

Os:Window10 64bit

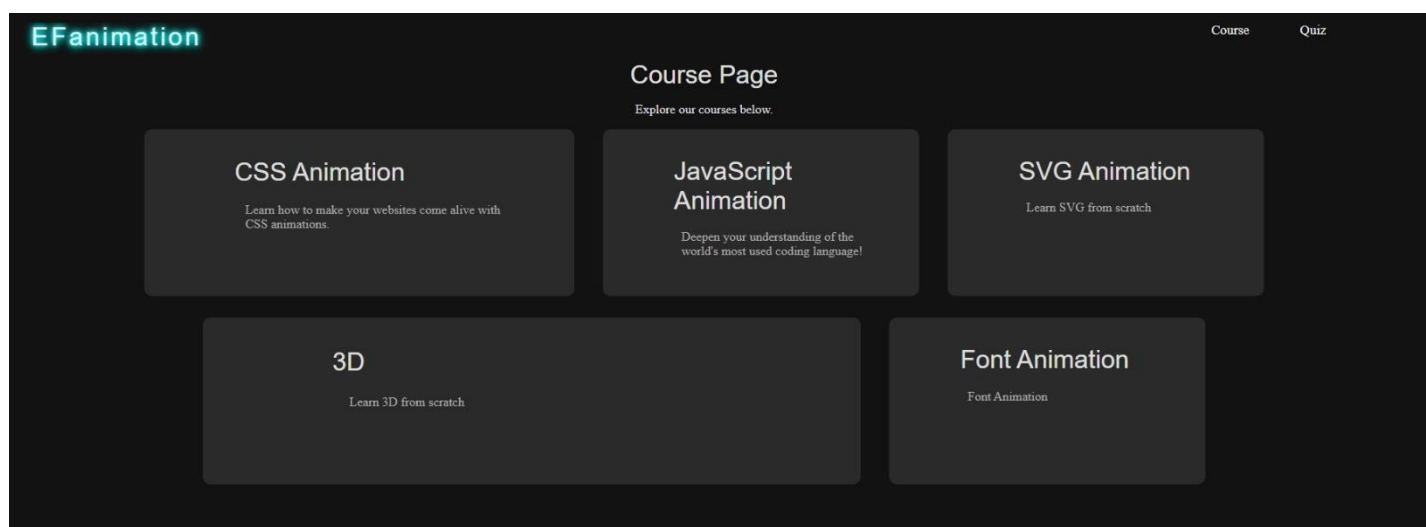
4.3 ภาพการทำงานของระบบ

ระบบที่ออกแบบในโครงงานนี้เป็นเว็บแอปพลิเคชันที่มุ่งเน้นการนำเสนอเนื้อหาการเรียนรู้เกี่ยวกับแอนิเมชันสำหรับเว็บไซต์ โดยแสดงภาพการทำงานของระบบผ่านส่วนติดต่อผู้ใช้ที่สามารถโต้ตอบได้ทันที ผู้เรียนสามารถทดลองโค้ดและเห็นผลลัพธ์แบบเรียลไทม์ ซึ่งช่วยให้เข้าใจการทำงานของ CSS, JavaScript และไลบรารี GSAP ได้อย่างชัดเจน แนวทางการพัฒนาให้ความสำคัญกับการเพิ่มประสิทธิภาพด้านการแสดงผล เช่น การใช้เทคนิค GPU acceleration การจัดการ DOM อย่างเหมาะสม และการโหลดข้อมูลแบบไม่ซ้ำซ้อน เพื่อให้สามารถใช้งานได้อย่างลื่นไหลแม้บนอุปกรณ์ที่มีข้อจำกัดด้านทรัพยากร ทั้งหมดนี้สะท้อนแนวคิดของการออกแบบระบบส่วนหน้าอย่างมีประสิทธิภาพและรองรับการขยายตัวในอนาคตของเนื้อหาและฟังก์ชันต่าง ๆ ได้อย่างยั่งยืน

4.3.1 ภาพการทำงานของระบบและการแนะนำการพัฒนาการแสดงผลระบบส่วนหน้า (Frontend) เพื่อการแสดงผลอย่างมีประสิทธิภาพ

4.3.1.1 หน้า course

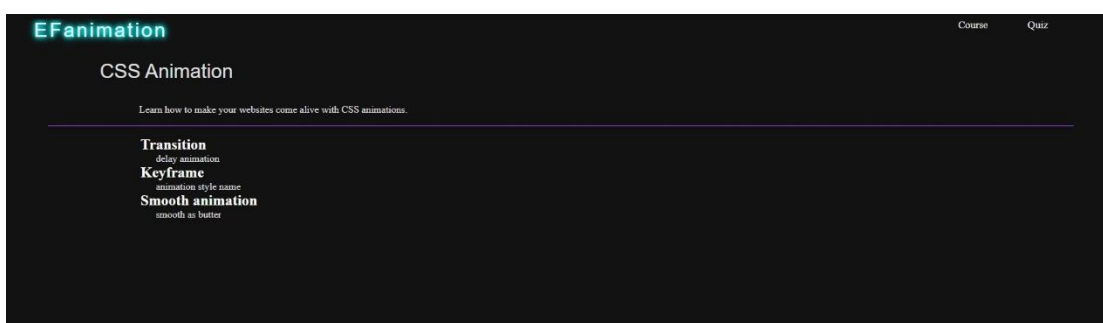
เมื่อผู้ใช้เข้าสู่หน้าเว็บ <https://efanimation.pages.dev/> จะได้หน้าแสดงผล(หน้า home page) หรือ หน้า course ดังนี้



ภาพที่ 4.1 หน้า course

4.3.1.2 หน้า lesson

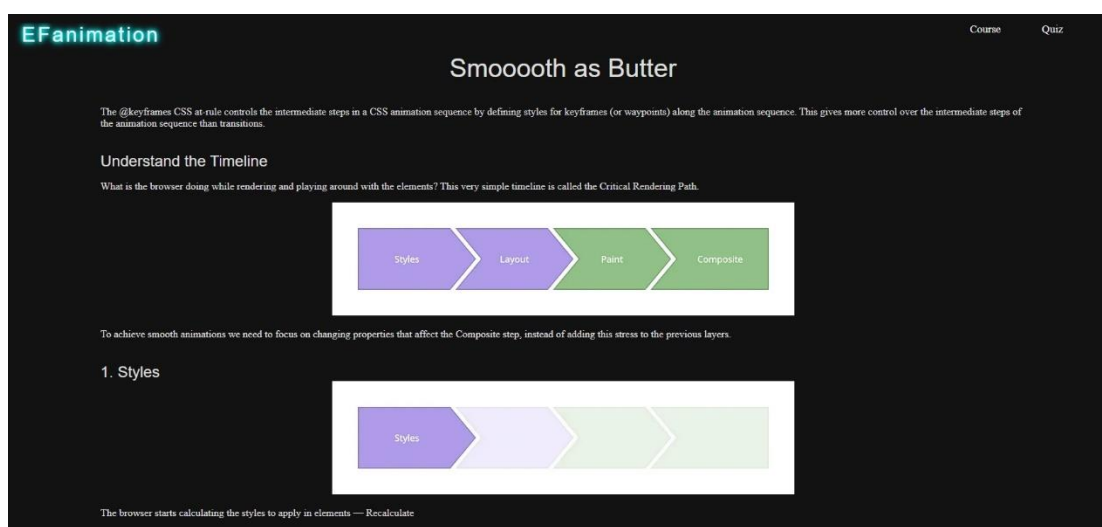
เมื่อผู้ใช้เลือกบทเรียนจากหน้า course ก็จะเข้าหน้า lesson เพื่อเลือกบทเรียนย่อยของบทที่ผู้ใช้เลือก



ภาพที่ 4.2 หน้า lesson

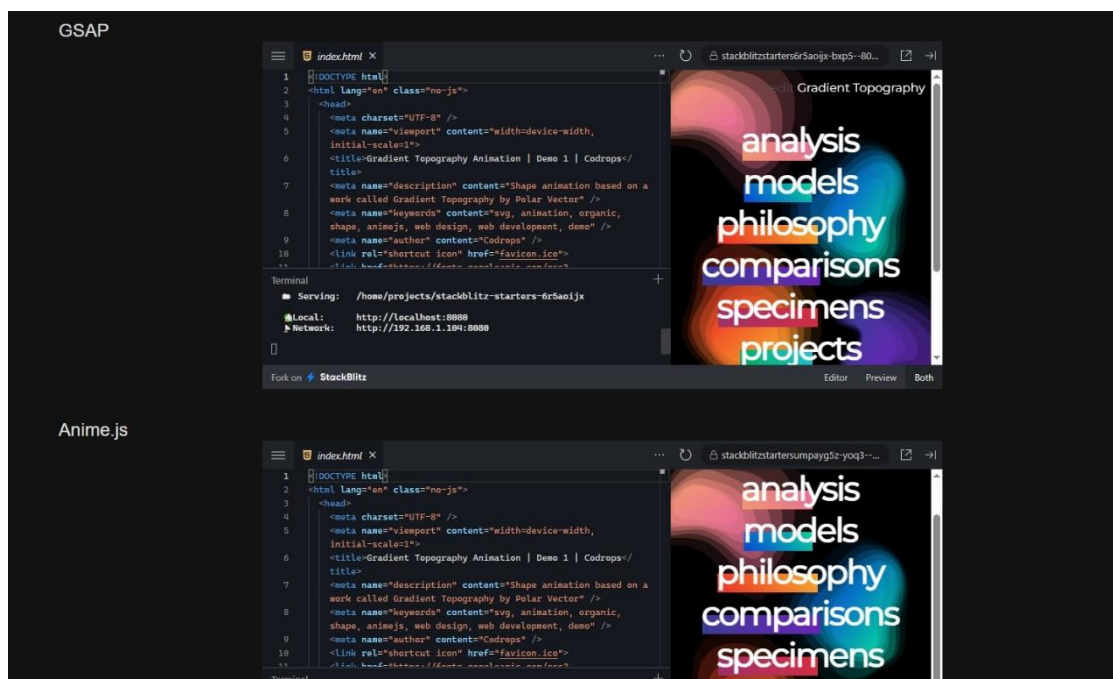
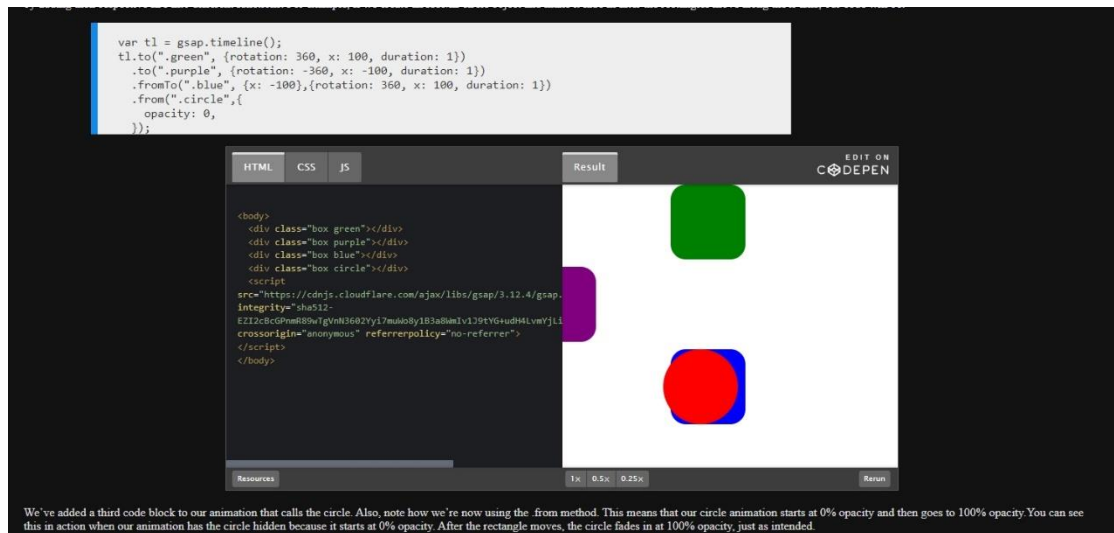
4.3.1.3 หน้า sublesson

เมื่อผู้ใช้เลือกบทเรียนย่อยจากหน้า lesson ก็จะเข้าสู่เนื้อหาที่ผู้ใช้เลือก



ภาพที่ 4.3 หน้า sublesson

โดยที่เนื้อหาจะมีภาพประกอบและมีโค้ดตัวอย่างที่ผู้ใช้สามารถแก้ไขได้ด้วย



ภาพที่ 4.4 และ 4.5 หน้า sublesson

ผู้ใช้สามารถแก้ไขโค้ดในกล่องได้ด้วยการกดละพิมพ์ข้อความลงไปกล่องได้เลย

4.3.1.4 หน้า quiz

ตัวอย่างคำถามและโจทย์ที่เกี่ยวข้องจากเนื้อหาของบริษัท

<https://medium.com/outsystems-experts/how-to-achieve-60-fps-animations-with-css3-db7b98610108>

โดย José Rosário

Transition ใน CSS เป็นเทคนิคที่ช่วยให้การเปลี่ยนแปลงคุณสมบัติขององค์ประกอบบนหน้าเว็บเกิดขึ้นอย่างราบรื่น ไม่กระตุก โดยสามารถใช้คำสั่งแบบ shorthand คือ transition เพื่อกำหนดการเปลี่ยนแปลงทั้งหมดได้อย่างสะดวกภายในบรรทัดเดียว ซึ่งวิธีนี้ช่วยลดความซับซ้อนของโค้ด และลดความผิดพลาดจากการกำหนดค่าที่ไม่สอดคล้องกัน

คำสั่ง transition ประกอบด้วยคุณสมบัติย่อยหลัก 4 ส่วน ได้แก่

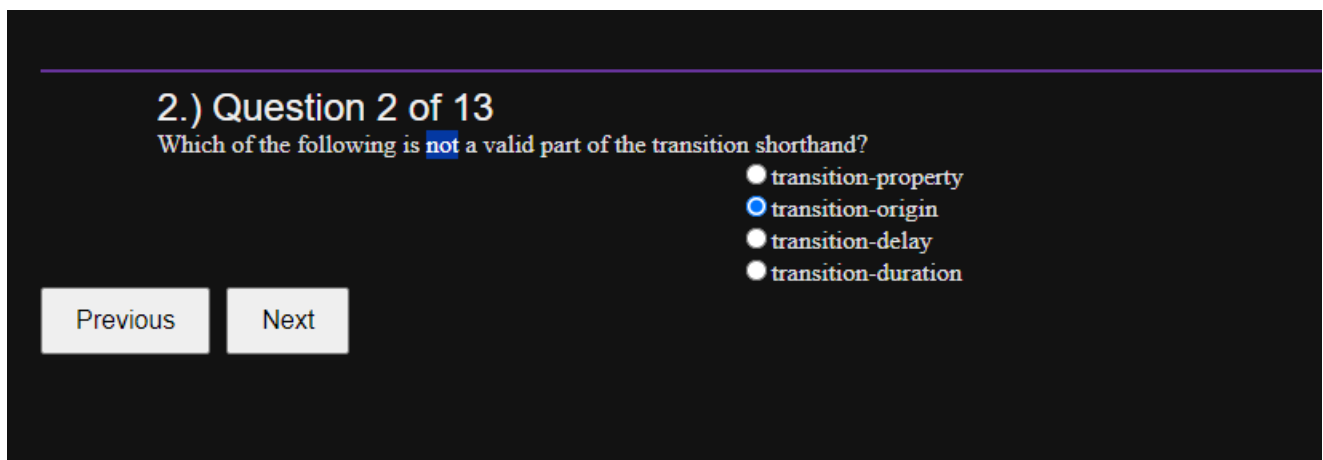
1. transition-property: ระบุชื่อของคุณสมบัติ (CSS property) ที่ต้องการให้มีการเปลี่ยนแปลงแบบ transition
2. transition-duration: กำหนดระยะเวลาในการเปลี่ยนแปลง (เช่น 0.3s, 1s)
3. transition-timing-function: ระบุรูปแบบของการเปลี่ยนค่า เช่น linear, ease-in, ease-out หรือกำหนดเองด้วย cubic-bezier
4. transition-delay: กำหนดระยะเวลาที่รอก่อนจะเริ่มการเปลี่ยนแปลง

Defining transitions

CSS Transitions are controlled using the shorthand transition property. This is the best way to configure transitions, as it makes it easier to avoid out of sync parameters, with time debugging in CSS. You can control the individual components of the transition with the following sub-properties: transition-property Specifies the name or names of properties to be animated during transitions; changes to all other properties occur instantaneously as usual. transition-duration Specifies the duration of the transition. A single duration that applies to all properties during the transition, or multiple values to allow each property to transition over a different period of time. transition-timing-function Intermediate values for properties are computed. Easing functions determine how intermediate values of the transition are calculated. Most easing functions can be specified by four points defining a cubic bezier. You can also choose easing from Easing functions cheat sheet. transition-delay Defines how long to wait before the transition actually begins. The transition shorthand CSS syntax is written as follows:

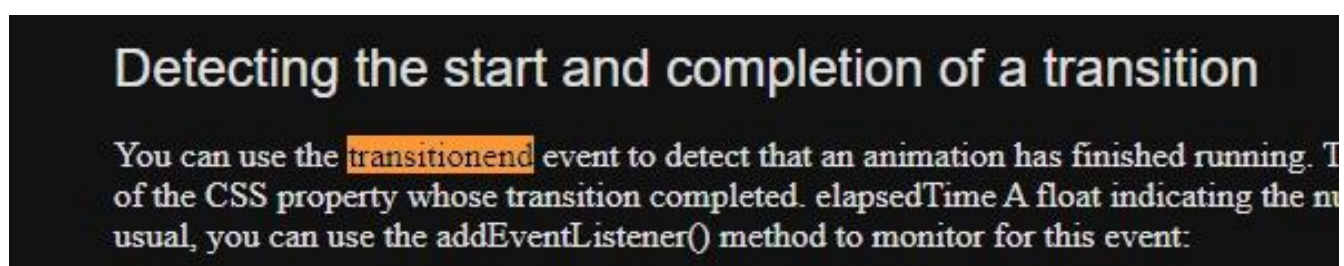
```
div transition: <property> <duration> <timing-function> <delay>;
```

ภาพที่ 4.22 หน้า transition shorthand



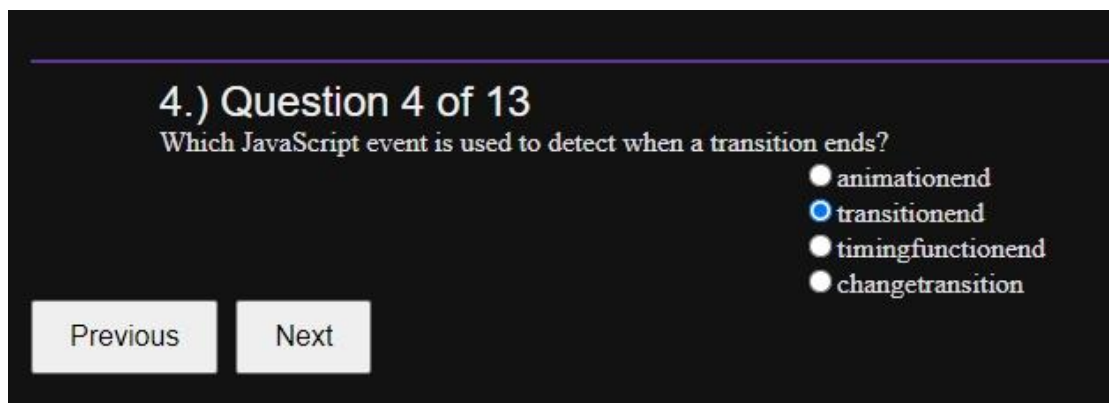
ภาพที่ 4.23 หน้าคำถาม transition shorthand

เนื่องจากถามว่าวิธีไหนไม่ใช่ transition shorthand จึงตอบ transition-origin เพราะไม่ได้กล่าวไว้ในเนื้อหา



ภาพที่ 4.24 หน้า transitionend

จากเนื้อหาที่ highlight ไว้ “You can use the transitionend event to detect that an animation has finished running.” ว่าจะเห็นว่า detect ด้วย transitionend

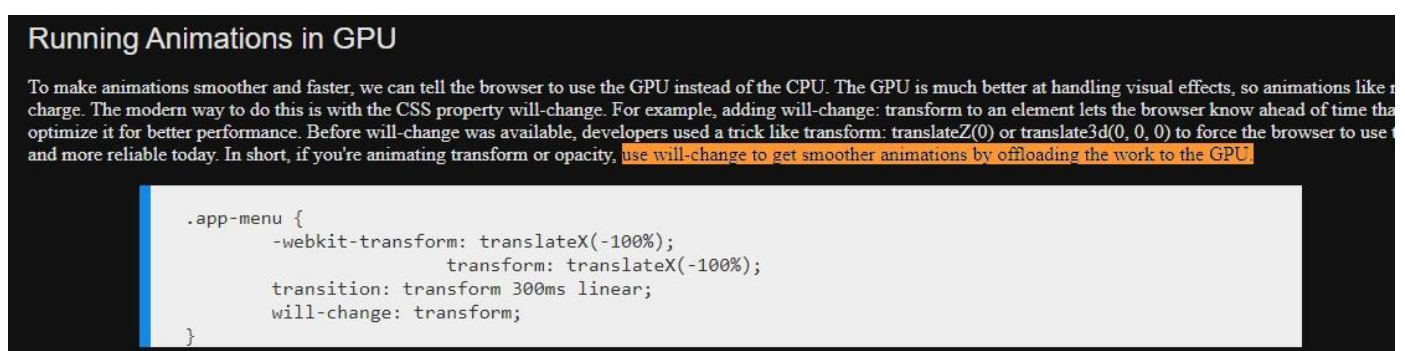


ภาพที่ 4.25 หน้าคำถาม transitionend

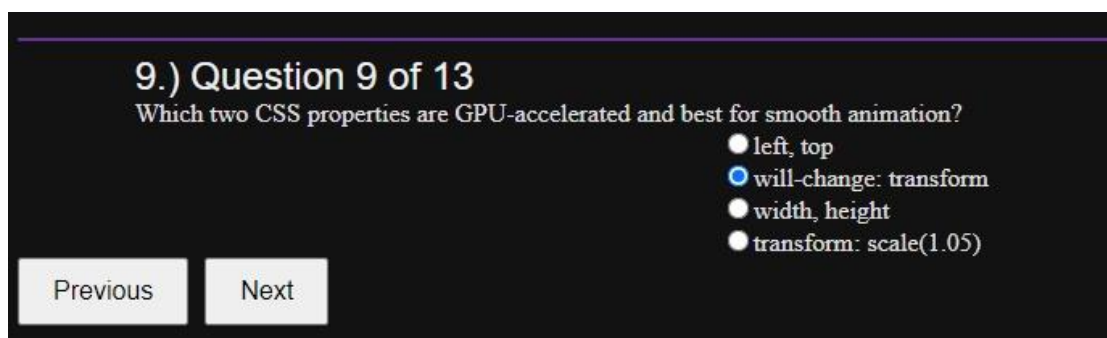
ดังนั้นจึงตอบ detect ด้วย transitionend

การทำ Animation บน GPU (Running Animations in GPU)

โดยทั่วไปแล้วเบราว์เซอร์จะใช้ CPU ในการประมวลผลโครงสร้างและการจัดวางหน้าเว็บ (layout) แต่สำหรับแอนิเมชันที่เกี่ยวข้องกับการเปลี่ยนแปลงเชิงภาพ เช่น transform หรือ opacity เราสามารถสั่งให้เบราว์เซอร์เตรียมใช้ GPU ล่วงหน้าได้ ด้วยการใช้คำสั่ง will-change



ภาพที่ 4.26 หน้า Animation บน GPU



ภาพที่ 4.27 หน้าคำถาม Animation บน GPU

ดังนั้นจึงตอบ will-change: transform

4.3.2 การวัดประสิทธิภาพของแอนิเมชันด้วย DevTools Performance

หลังจากที่เราใช้เทคนิคต่าง ๆ เช่น will-change หรือ translateZ(0) เพื่อให้แอนิเมชันทำงานผ่าน GPU ขั้นตอนถัดไปที่สำคัญคือ การตรวจสอบและวัดประสิทธิภาพการทำงานจริง ของเว็บไซต์ เพื่อให้มั่นใจว่าแอนิเมชันลื่นไหล และไม่มีปัญหากระตุกหรือเฟรมเรตตก

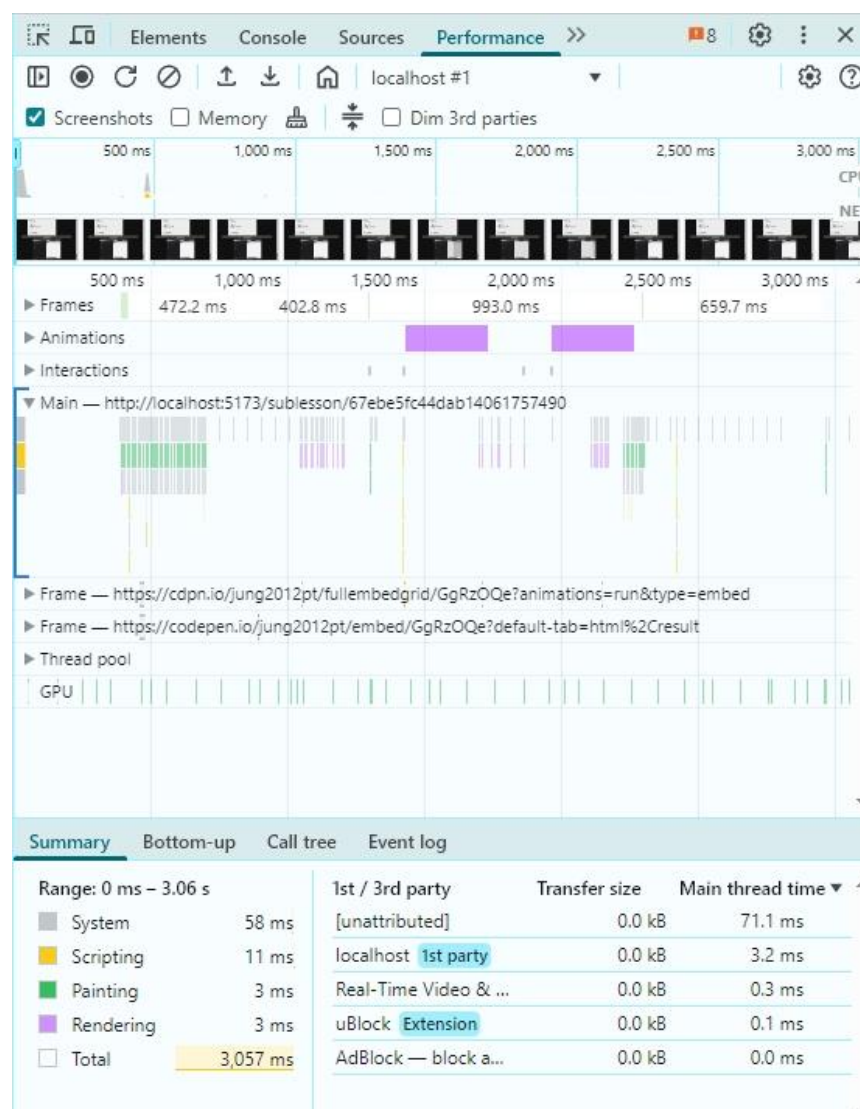
เครื่องมือที่นิยมใช้มากที่สุดคือ **Chrome DevTools** โดยเฉพาะในแท็บ **Performance (หรือชื่อเดิมคือ Timeline)** ซึ่งช่วยให้เราตรวจสอบรายละเอียดเกี่ยวกับการเรนเดอร์ของเบราว์เซอร์ได้อย่างแม่นยำ

ขั้นตอนการใช้งาน DevTools Performance

1. เปิดเว็บไซต์ที่ต้องการวิเคราะห์ใน Google Chrome
2. คลิกขวาบนหน้าเว็บ เลือก **Inspect** หรือกด F12
3. ไปที่แท็บ **Performance**
4. กดปุ่ม **Record** แล้วดำเนินการแอนิเมชันที่ต้องการวัด (เช่น เลื่อนเมาส์, คลิก หรือรอให้แอนิเมชันทำงาน)
5. กด **Stop** เพื่อหยุดการบันทึกข้อมูล

เกณฑ์วัดที่สามารถวัดผลได้

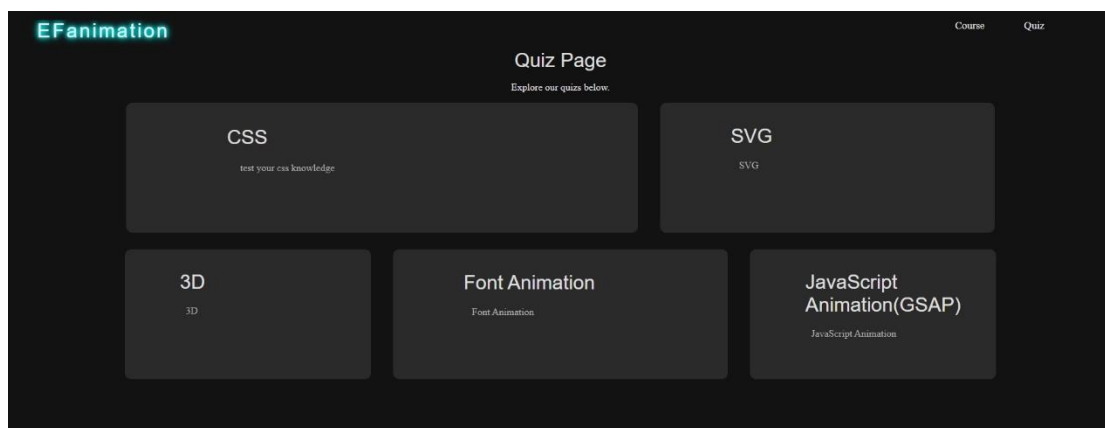
1. **Frames per second (FPS):** ควรอยู่ใกล้ 60 FPS เพื่อความลื่นไหล
2. **Compositing Layers:** ดูว่าองค์ประกอบใดถูกเรนเดอร์โดย GPU
3. **Paint และ Layout:** หากมีการคำนวณ layout หรือ repaint บ่อยๆ อาจต้องพิจารณาแก้ไข CSS หรือ DOM ที่เกี่ยวข้อง
4. **Long Tasks:** สังเกตว่าเฟรมใดใช้เวลาหนานผิดปกติ ซึ่งอาจเป็นสาเหตุของความหน่วง



ภาพที่ 4.28 หน้า Performance

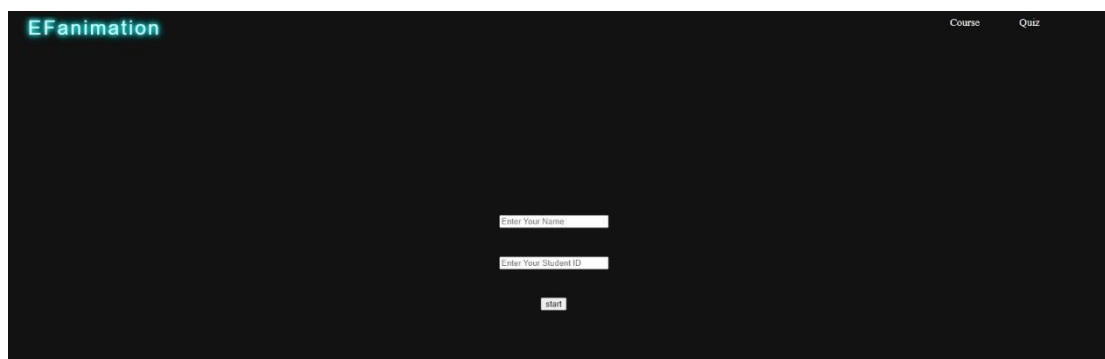
4.3.3 วิธีการเข้าถึง quiz

ผู้ใช้งานสามารถเลือกบทที่จะทดสอบได้ด้วยการกด card ของบทเรียน



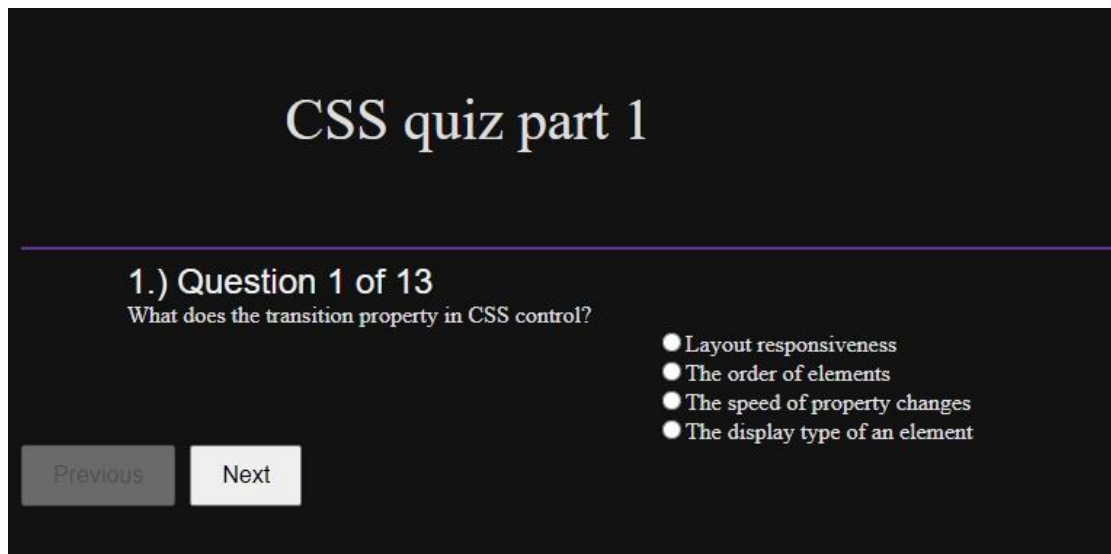
ภาพที่ 4.6 หน้า quiz

เมื่อกดเข้าไปในบททดสอบจะต้องกรอก ชื่อ และรหัสนักศึกษา



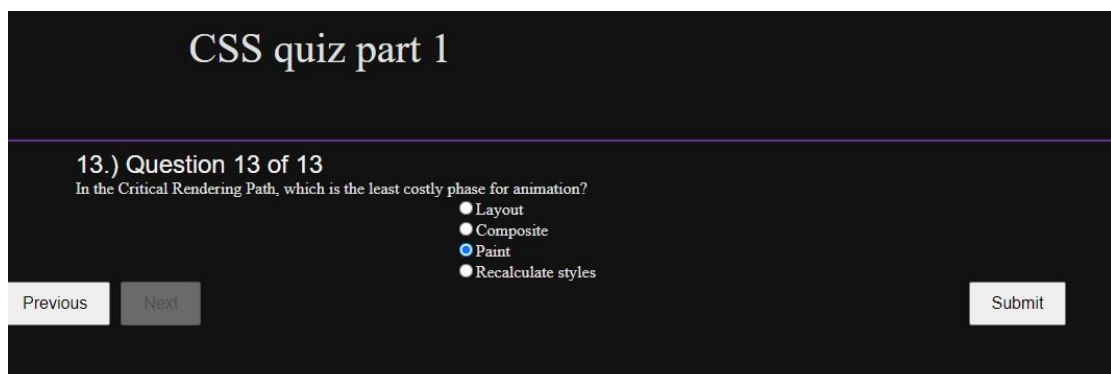
ภาพที่ 4.7 หน้า กรอก ชื่อ และรหัสนักศึกษา

เมื่อกด start ก็จะสามารถทำบททดสอบได้

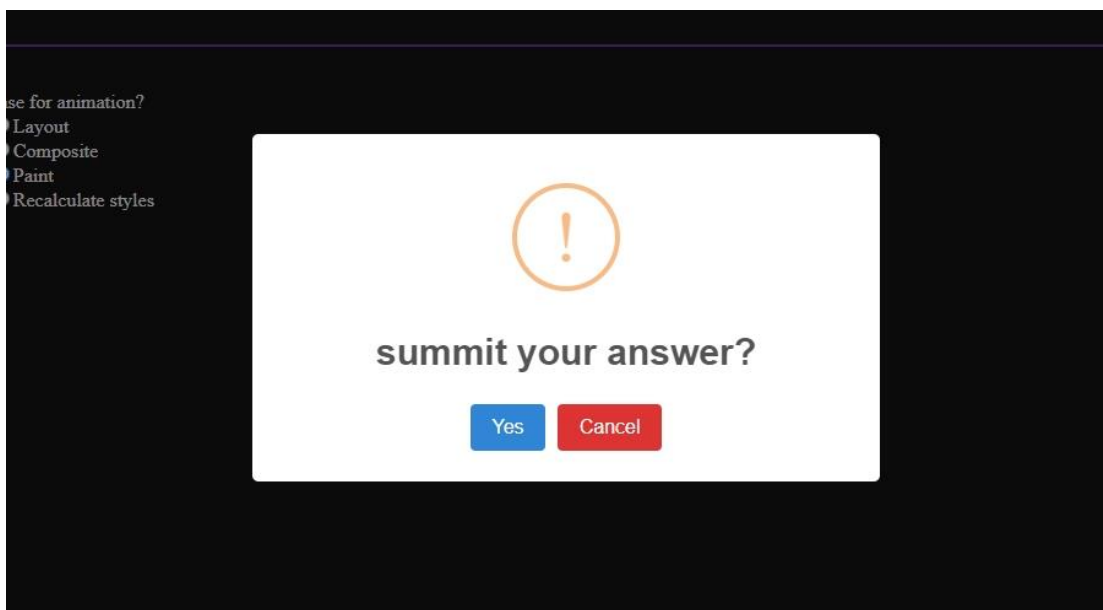


ภาพที่ 4.8 หน้า quiz

เมื่อทำครบทุกข้อระบบจะแสดงปุ่ม submit

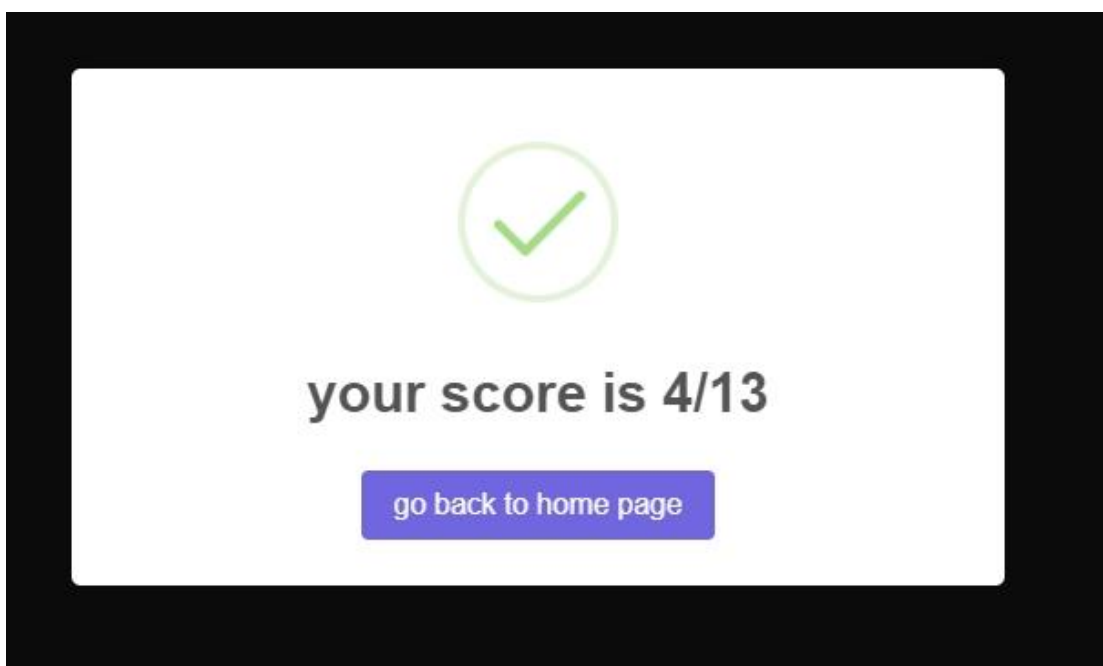


ภาพที่ 4.9 หน้า submit



ภาพที่ 4.10 หน้า ยืนยัน

เมื่อกดปุ่ม submit จะมี pop up ยืนยันอีกที



ภาพที่ 4.11 หน้า แสดงคะแนน

เมื่อกด yes ระบบจะส่งคำตอบเข้า database และแสดงข้อมูลให้ผู้ดู

หลังจากนั้นถ้ากดปุ่ม go back to home page ระบบจะกลับไปหน้าแรก

4.4 การทดสอบระบบส่วนเนื้อหา

การวัดความเข้าใจของบทเรียนจากผู้ใช้งานสามารถใช้คะแนน quiz ของแต่ละบทวัดความเข้าใจได้เลย

โดยในการประเมินความเข้าใจของผู้ใช้งานจากบทเรียนนี้ สามารถดำเนินการได้โดยใช้คะแนนแบบทดสอบ (Quiz) ของแต่ละบทเรียนเป็นตัวชี้วัดหลัก โดยผู้เรียนจะทำแบบทดสอบหลังจากศึกษาบทเรียนจบ เพื่อประเมินว่ามีความเข้าใจเนื้อหาในระดับใด

โดยกำหนดเกณฑ์การประเมินเบื้องต้นดังนี้:

- คะแนนต่ำกว่า 50% : ไม่ผ่าน – ควรกลับไปทบทวนบทเรียนอีกครั้ง
- คะแนน 50%–69% : ผ่าน – มีความเข้าใจในระดับพื้นฐาน
- คะแนน 70%–89% : เข้าใจดี – มีความเข้าใจเนื้อหาในระดับดี
- คะแนน 90% ขึ้นไป : เข้าใจอย่างดีเยี่ยม – เข้าใจเนื้อหาได้อย่างครบถ้วน

ผู้ใช้คนที่ 1

หัวข้อ Quiz	คะแนนของผู้ใช้
Quiz ของบท CSS	6/13
Quiz ของบท SVG	3/6
Quiz ของบท 3D	3/7
Quiz ของบท Font Animation	3/4
Quiz ของบท Javascript Animation	2/5
คะแนนรวม	17/35

ผู้ใช้คนที่ 2

หัวข้อ Quiz	คะแนนของผู้ใช้
Quiz ของบท CSS	9/13
Quiz ของบท SVG	-
Quiz ของบท 3D	-
Quiz ของบท Font Animation	-
Quiz ของบท Javascript Animation	-
คะแนนรวม	9/13

ผู้ใช้คนที่ 3

หัวข้อ Quiz	คะแนนของผู้ใช้
Quiz ของบท CSS	8/13
Quiz ของบท SVG	-
Quiz ของบท 3D	-
Quiz ของบท Font Animation	-
Quiz ของบท Javascript Animation	-
คะแนนรวม	8/13

ผู้ใช้คนที่ 4

หัวข้อ Quiz	คะแนนของผู้ใช้
Quiz ของบท CSS	11/13
Quiz ของบท SVG	6/6
Quiz ของบท 3D	7/7
Quiz ของบท Font Animation	4/4
Quiz ของบท Javascript Animation	5/5
คะแนนรวม	33/35

ผู้ใช้คนที่ 5

หัวข้อ Quiz	คะแนนของผู้ใช้
Quiz ของบท CSS	-
Quiz ของบท SVG	-
Quiz ของบท 3D	-
Quiz ของบท Font Animation	-
Quiz ของบท Javascript Animation	5/5
คะแนนรวม	5/5

$$\text{คะแนนเฉลี่ย} = \frac{17 + 9 + 8 + 33 + 5}{35 + 13 + 13 + 35 + 5} = 0.7128712871$$

เมื่อแปลงเป็นเปอร์เซ็นต์: $0.71287 \times 100 = 71.29\%$

ดังนั้น ผู้เรียนได้คะแนนเฉลี่ย **71.29%** ซึ่งถือว่า **ผ่านเกณฑ์** ตามเงื่อนไขที่กำหนดไว้ในระบบ

โดยกำหนดเกณฑ์การประเมินเบื้องต้นดังนี้:

- **คะแนนต่ำกว่า 50% :** ไม่ผ่าน – ควรกลับไปทบทวนบทเรียนอีกครั้ง
- **คะแนน 50%–69% :** ผ่าน – มีความเข้าใจในระดับพื้นฐาน
- **คะแนน 70%–89% :** เข้าใจดี – มีความเข้าใจเนื้อหาในระดับดี
- **คะแนน 90% ขึ้นไป :** เข้าใจอย่างดีเยี่ยม – เข้าใจเนื้อหาได้อย่างครบถ้วน

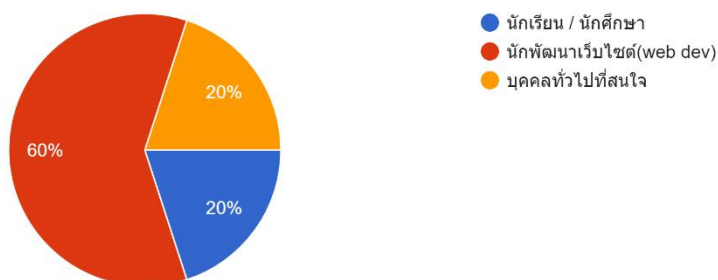
ภาพที่ 4.29 หน้า เกณฑ์การประเมินเบื้องต้น

4.5 การทดสอบระบบโดยรวม

ผู้พัฒนาได้ดำเนินการทดสอบระบบโดยรวมของเว็บแอปพลิเคชันผ่านแบบสอบถาม Google Form โดยมีวัตถุประสงค์เพื่อประเมินความเหมาะสมของเนื้อหา ความเข้าใจของผู้ใช้งาน และความพึงพอใจในการใช้งาน ผลการตอบแบบสอบถามจะนำมาใช้ประกอบการพิจารณาว่าระบบสามารถตอบโจทย์การเรียนรู้ได้ตรงตามวัตถุประสงค์หรือไม่ และมีจุดใดที่ควรปรับปรุงเพิ่มเติม

ข้อมูลและกราฟการประเมินระบบโดยรวม :

สถานะของคุณ
5 responses

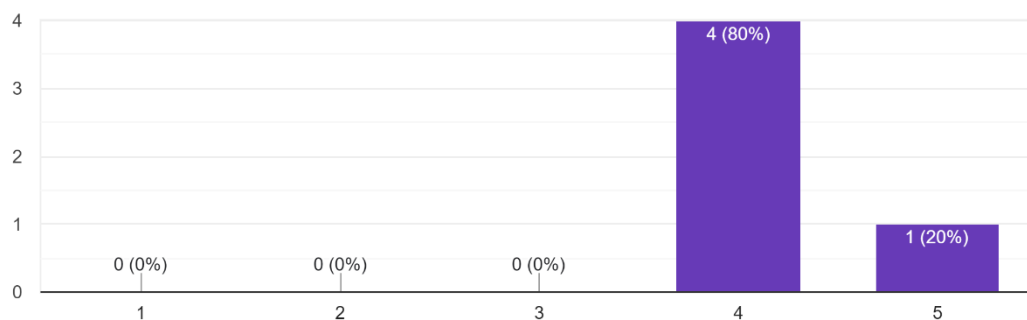


ภาพที่ 4.12 หน้า กราฟสถานะของผู้ใช้

กราฟนี้แสดงสัดส่วนของผู้ตอบแบบสอบถามตามสถานะ เช่น นักเรียน/นักศึกษา, นักพัฒนาเว็บไซต์ (Web Developer), และบุคคลทั่วไปที่สนใจเพื่อให้สามารถวัดความเข้าใจได้ละเอียดมากขึ้น

จากข้อมูลในแบบสอบถาม ผู้ตอบส่วนใหญ่เป็นนักพัฒนาเว็บไซต์ แสดงว่าแอปนี้สามารถเข้าถึงกลุ่มเป้าหมายที่มีความสนใจและความรู้ในด้านนี้อยู่แล้ว(ระดับ intermediate) ในขณะที่ยังสามารถเข้าถึงกลุ่มนักศึกษาหรือผู้เริ่มต้นได้ด้วย

หลังจากใช้แอปนี้ คุณคิดว่าความเข้าใจของคุณเกี่ยวกับการพัฒนา frontend ดีขึ้นหรือไม่ 1 = ไม่ดีขึ้นเลย, 5 = ดีขึ้นมาก
5 responses

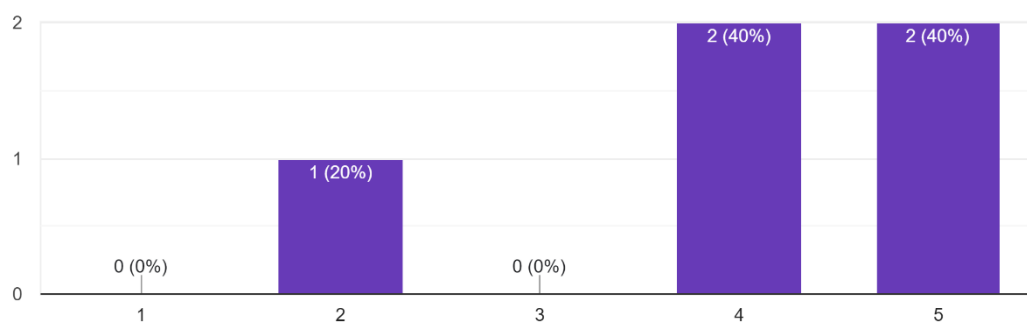


ภาพที่ 4.13 หน้า กราฟประเมินความเข้าใจหลังจากใช้งานแอป

กราฟนี้บอกว่า หลังจากใช้แอปแล้ว ผู้ใช้งานรู้สึกว่าคุณเข้าใจของตนเองดีขึ้นแค่ไหน เนื่องจากคะแนนโดยรวมสูง แปลว่าแอปช่วยให้ผู้ใช้เข้าใจดีขึ้นจริง สามารถชี้วัดได้ว่าแอปนี้มีประโยชน์ต่อการเรียนรู้ด้าน frontend

แบบทดสอบหรือ Quiz ในแอปช่วยให้คุณประเมินความเข้าใจของตัวเองได้หรือไม่ 1 = ไม่ช่วยเลย , 5 = ช่วยให้เข้าใจมากขึ้น

5 responses

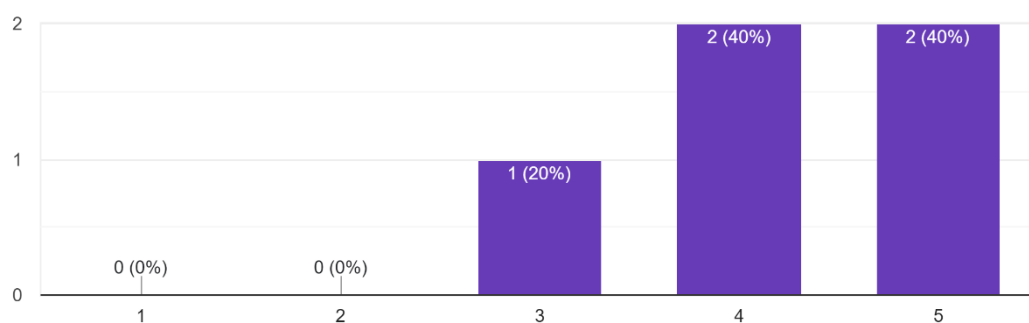


ภาพที่ 4.14 หน้า กราฟประเมิน quiz

กราฟนี้ใช้วัดว่าความเข้าใจของผู้ใช้งานดีขึ้นหรือไม่หลังจากใช้งานแอป โดยเปรียบเทียบกับความเข้าใจในตอนเริ่มต้น เนื่องจาก คะแนนเฉลี่ยของข้อนี้อยู่ในระดับสูง สะท้อนว่าแบบทดสอบในแอปสามารถช่วยให้ผู้เรียนประเมินความเข้าใจได้อย่างชัดเจน

คุณสามารถนำความรู้จากแอปนี้ไปใช้ในงานจริงหรือโปรเจกต์ส่วนตัวได้มากน้อยเพียงใด 1 = ไม่สามารถนำไปใช้ได้เลย , 5 = สามารถนำไปใช้งานได้จริง

5 responses

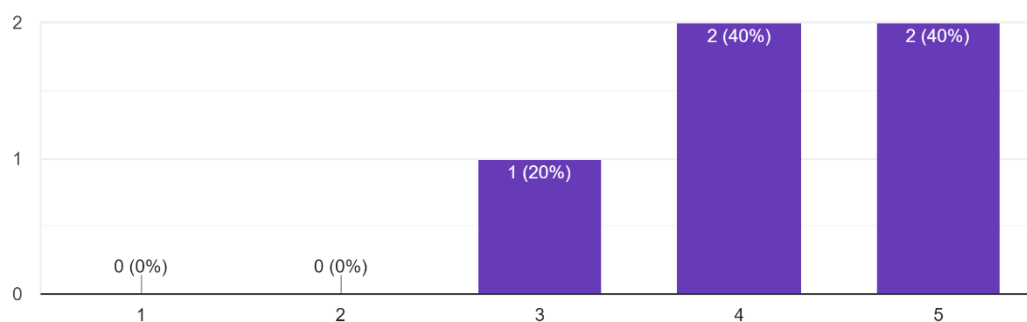


ภาพที่ 4.15 หน้า กราฟการนำความรู้ไปใช้งานจริง

กราฟนี้ชี้วัดว่าผู้ใช้งานสามารถนำความรู้ที่ได้จากแอปไปใช้จริงในชีวิตหรือในงานพัฒนาเว็บไซต์ได้มากน้อยแค่ไหน เนื่องจากคะแนนเฉลี่ยระดับสูง แสดงว่าผู้ใช้ส่วนใหญ่มองว่าเนื้อหาในแอปสามารถนำไปใช้ได้จริง ไม่ว่าจะเป็นการทำโปรเจกต์ หรือการทำงานที่เกี่ยวข้องกับ frontend

คุณเข้าใจแนวคิดพื้นฐานของ frontend (เช่น HTML, CSS, JavaScript) มากน้อยเพียงใดก่อนใช้แอปนี้ 1 = ไม่เข้าใจเลย, 5 = เข้าใจดีมาก

5 responses



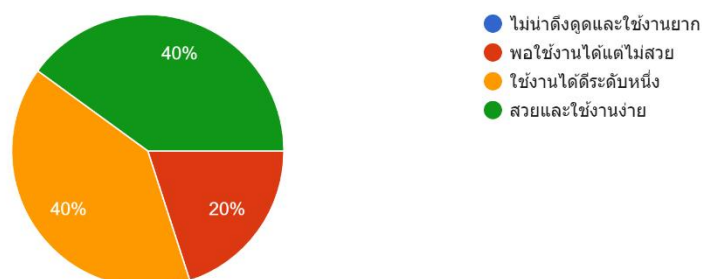
ภาพที่ 4.16 หน้า กราฟประเมินความเข้าใจพื้นฐานก่อนใช้งานแอป

กราฟนี้ใช้วัดระดับความเข้าใจเกี่ยวกับการพัฒนา frontend ของผู้ใช้งานก่อนที่จะเริ่มใช้แอป เพื่อประเมินว่าผู้ใช้งานเริ่มต้นด้วยพื้นฐานในระดับใด

ผู้ใช้งานส่วนใหญ่มีความเข้าใจพื้นฐานในระดับพอใช้ถึงค่อนข้างดี หรืออยู่ในช่วงระดับ beginner ถึง intermediate อย่างชัดเจน แสดงให้เห็นว่าแอปเข้าถึงกลุ่มเป้าหมายที่มีความรู้เบื้องต้นอยู่แล้ว

การออกแบบหน้าตาของแอป (UI) เป็นมิตรและใช้งานง่ายเพียงใด

5 responses

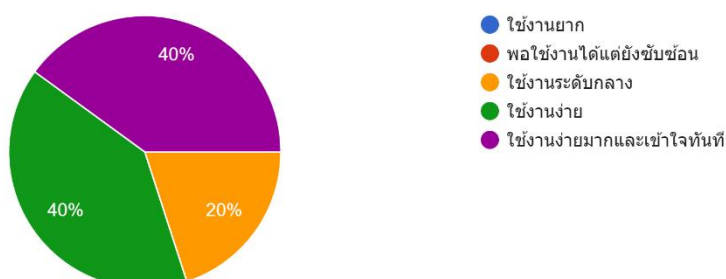


ภาพที่ 4.17 หน้า กราฟความคิดเห็นต่อการออกแบบ UI

กราฟนี้ใช้วัดว่าผู้ใช้งานคิดเห็นอย่างไรเกี่ยวกับหน้าตาและการออกแบบส่วนติดต่อผู้ใช้ (User Interface) ของแอป เนื่องจากคะแนนความเห็นเกี่ยวกับการออกแบบ UI อยู่ในระดับดีถึงดีมาก ซึ่งสะท้อนว่าผู้ใช่มองว่าแอปมีหน้าตาที่สวยงาม ใช้งานง่าย

แอปนี้ใช้งานง่ายและไม่ซับซ้อนจนเกินไปหรือไม่

5 responses

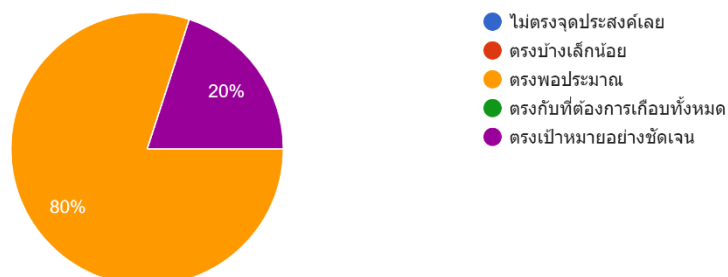


ภาพที่ 4.18 หน้า กราฟประเมินการใช้งาน

กราฟนี้ใช้วัดว่าผู้ใช้งานรู้สึกว่าการใช้งานง่ายหรือไม่ เช่น เมนูชัดเจน เข้าใจได้ทันที และไม่ซับซ้อน จากคะแนนโดยรวมที่สูง แสดงว่าผู้ใช้ส่วนใหญ่มองว่าแอปใช้งานง่าย ไม่ซับซ้อน และสามารถใช้งานได้โดยไม่ต้องใช้คู่มือ และไม่ต้องการอธิบายวิธีใช้งาน

คุณรู้สึกว่าคุณได้ตอบโจทย์เป้าหมายของคุณในการเรียนรู้การพัฒนา frontend หรือไม่

5 responses

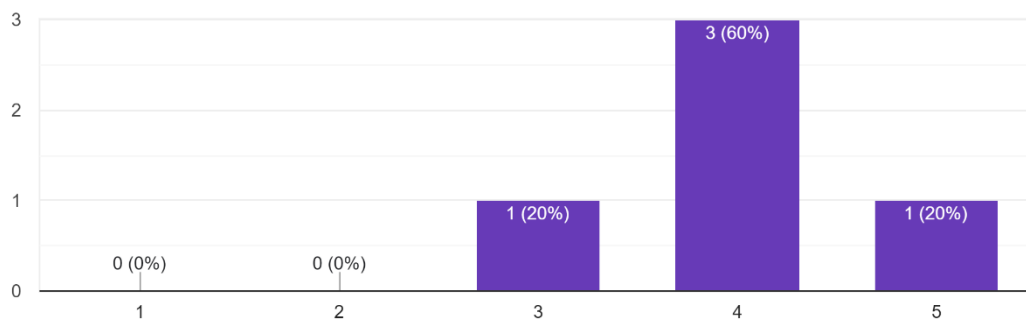


ภาพที่ 4.19 หน้า กราฟประเมินเป้าหมาย

กราฟนี้ใช้วัดว่าแอปสามารถตอบโจทย์ด้านการเรียนรู้ได้ตรงกับสิ่งที่ผู้ใช้คาดหวังไว้หรือไม่ เนื่องจากคะแนนที่ผู้ใช้ให้ในข้อนี้ในระดับสูง แสดงว่าแอปนี้สามารถตอบโจทย์ในการเรียนรู้ได้จริง โดยเนื้อหามีความตรงประเด็น ครอบคลุมหัวข้อสำคัญ

โดยรวมแล้ว คุณประเมินคุณภาพของแอปนี้อยู่ในระดับใด 1 = แย่มาก , 5 = ดีเยี่ยม
และอยากให้ออกแบบเพิ่มเติมต่อ

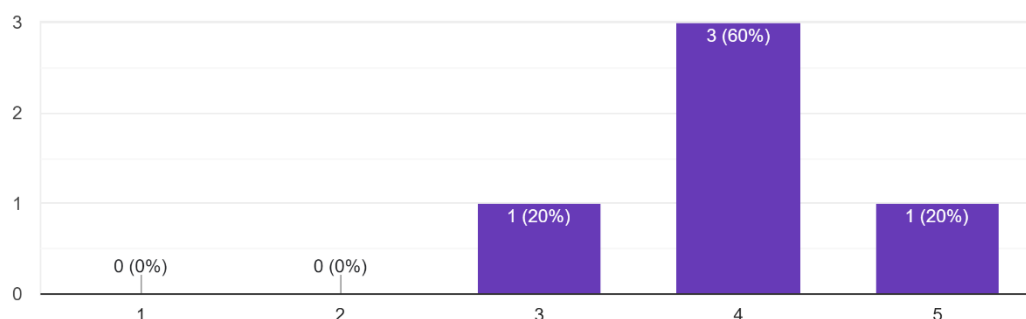
5 responses



ภาพที่ 4.20 หน้า กราฟประเมินคุณภาพ

กราฟนี้ใช้วัดภาพรวมของคุณภาพแอป ว่าคิดว่าแอปนี้มีคุณภาพดีหรือไม่
เนื่องจาก ผู้ใช้ให้คะแนนคุณภาพโดยรวมของแอปในระดับ 4-5 เป็นส่วนใหญ่ ซึ่งแสดงให้เห็นว่าแอป
มีคุณภาพดี ทั้งในแง่เนื้อหา การออกแบบ และประสบการณ์การใช้งาน

คุณมีความพึงพอใจต่อแอปนี้ในระดับใด 1 = ไม่พึงพอใจเลย , 5 = พึงพอใจอย่างยิ่ง
5 responses



ภาพที่ 4.21 หน้า กราฟประเมินความพึงพอใจ

กราฟนี้ใช้วัดระดับความพึงพอใจโดยรวมของผู้ใช้งานที่มีต่อแอป ไม่ว่าจะเป็นเนื้อหา การใช้งาน หรือภาพรวมของแอปทั้งหมด

จากผลการประเมิน ผู้ใช้ให้คะแนนเฉลี่ยอยู่ที่ 4.3 จาก 5 ซึ่งถือว่าอยู่ในระดับสูง สะท้อนว่าผู้ใช้งานมีความพึงพอใจโดยรวมต่อแอปในหลายด้าน ทั้งเนื้อหา ความเข้าใจง่าย และประสบการณ์การใช้งาน แสดงว่าผู้ใช้งานส่วนใหญ่รู้สึกดีต่อการใช้งานแอป ทั้งในแง่ของเนื้อหา ความเข้าใจง่าย และประสบการณ์การใช้งานโดยรวม

จากแบบสอบถามที่ผู้ใช้งาน พบว่าผู้ใช้งานส่วนใหญ่มีพื้นฐานมาบ้างอยู่แล้ว แต่พอได้ใช้แอป ความเข้าใจเรื่อง HTML, CSS, JavaScript ก็ดีขึ้นอย่างชัดเจน โดยเฉพาะแบบทดสอบในแอปที่หลายคนบอกว่าช่วยให้เข้าใจมากขึ้น

เรื่องการใช้งาน ผู้ใช้พบว่าแอปใช้งานง่าย หรืออย่างน้อยก็ไม่ซับซ้อนจนเกินไป ส่วนการออกแบบหน้าตาก็ถือว่าโอเค ดูเข้าใจง่าย และใช้งานได้จริง แต่ควรปรับปรุงเล็กน้อย

โดยรวมแล้ว คนที่ใช้ส่วนมากให้คะแนนดีถึงดีมาก และอยากให้แอปนี้มีการพัฒนาต่อไปอีก ถือว่าผ่านเกณฑ์การทดสอบ และช่วยยืนยันว่าแอปนี้ตอบโจทย์การเรียนรู้ Frontend ได้จริง

บทที่ 5 สรุป

5.1 สรุปผลการดำเนินงาน

ผู้พัฒนาได้ออกแบบเว็บแอปพลิเคชันระบบแนะนำเทคนิคการพัฒนาแอนิเมชันบนเว็บไซต์ โดยมีจุดประสงค์เพื่อช่วยให้นักพัฒนาเว็บสามารถเรียนรู้และเข้าใจการใช้งานเทคนิคแอนิเมชันที่มีประสิทธิภาพ เช่น CSS Animation, JavaScript Animation, SVG, 3D และ Font Animation ซึ่งเป็นองค์ความรู้สำคัญในการพัฒนาเว็บไซต์ยุคใหม่ที่เน้นประสบการณ์ผู้ใช้ (User Experience)

เว็บแอปพลิเคชันนี้ไม่เพียงแต่รวบรวมเนื้อหาแอนิเมชันอย่างเป็นระบบ แต่ยังสอนวิธีการนำเทคนิคที่เรียนรู้ไปประยุกต์ใช้กับการออกแบบเว็บไซต์จริง พร้อมทั้งมีระบบทดลองโค้ดแบบโต้ตอบ (Interactive Code Preview) ที่ช่วยให้ผู้เรียนสามารถแก้ไขและเห็นผลลัพธ์ได้ทันที ช่วยส่งเสริมการเรียนรู้เชิงปฏิบัติอย่างมีประสิทธิภาพ

นอกจากนี้ยังได้คัดเลือกเนื้อหาที่มักกระจุกกระจายหรือไม่มีในตำราเรียนทั่วไปมารวบรวมไว้ในที่เดียว เพื่อให้ผู้เรียนสามารถเข้าใจทั้งหลักการ พฤติกรรมของแอนิเมชัน และแนวทางการนำไปใช้จริงในบริบทของเว็บไซต์

ทั้งนี้ ผู้พัฒนาคาดหวังว่าเว็บแอปพลิเคชันนี้จะเป็เครื่องมือช่วยเพิ่มทักษะด้านการพัฒนาเว็บไซต์อย่างมีประสิทธิภาพ พร้อมเป็นแนวทางที่สามารถต่อยอดไปสู่การออกแบบระบบหรือเว็บไซต์ที่ตอบสนองต่อผู้ใช้งานได้อย่างแท้จริง

5.2 ปัญหาและอุปสรรคที่พบ

1. ผู้เขียนไม่ค่อยมีความรู้ด้าน backend จึงต้องใช้เวลาเรียนรู้เพิ่ม
2. ปัญหาการ deploy เว็บแอปพลิเคชันขึ้นสู่ cloudflare ซึ่งเมื่อขึ้น server แล้วจะมี
ปัญหาตอนเปลี่ยน css ของบทเรียน
3. ปัญหาเรื่อง type ของ Javascript

5.3 ข้อเสนอแนะเพิ่มเติม

แม้ว่าโครงงานนี้จะสามารถออกแบบระบบสื่อการเรียนรู้เกี่ยวกับการสร้างแอนิเมชันบนเว็บไซต์ได้ตรงตามวัตถุประสงค์ และครอบคลุมเนื้อหาในระดับกลางถึงสูง แต่จากการทดสอบระบบเบื้องต้นและการวิเคราะห์ข้อคิดเห็นจากผู้ใช้งาน พบว่ายังมีหลายประเด็นที่สามารถพัฒนาและปรับปรุงให้ดียิ่งขึ้น ทั้งในด้านเนื้อหา ระบบการใช้งาน และประสบการณ์ของผู้เรียน เพื่อให้ระบบมีความสมบูรณ์ ใช้งานได้กว้างขวาง และเกิดประโยชน์สูงสุด

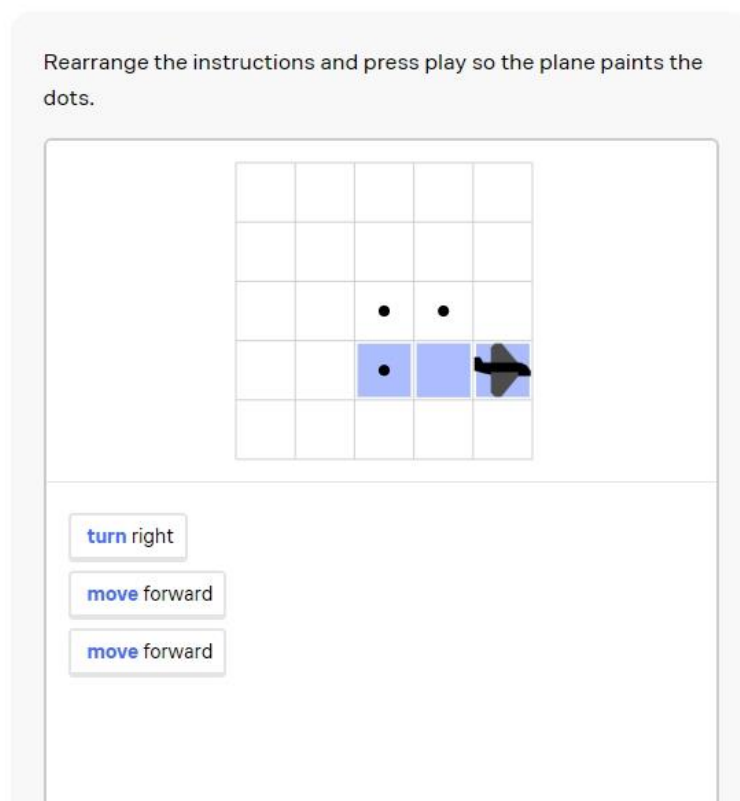
จึงได้จัดทำข้อเสนอแนะเพิ่มเติมดังนี้

1. แบ่งเวลาและวางแผนการทำงานให้ดีเพราะ เวลาการจัดทำระบบที่ครอบคลุมทั้งหมด อาจจะใช้เวลามากกว่าที่คาดการณ์ไว้
2. ถ้ามีเวลาและงบประมาณมากพอ สามารถเพิ่มบทเรียนแบบ interactive ได้อย่างเช่น

brilliant.org (<https://brilliant.org/>)

brilliant.org เป็นแพลตฟอร์มการเรียนรู้ออนไลน์ที่เน้นการเรียนรู้ผ่านการลงมือทำ (learn by doing) โดยเฉพาะในด้านคณิตศาสตร์ วิทยาศาสตร์ และวิทยาการคอมพิวเตอร์ สำหรับผู้ที่สนใจฝึกฝนการเขียนโปรแกรมด้วยแบบฝึกหัดที่มีปฏิสัมพันธ์ (interactive coding)

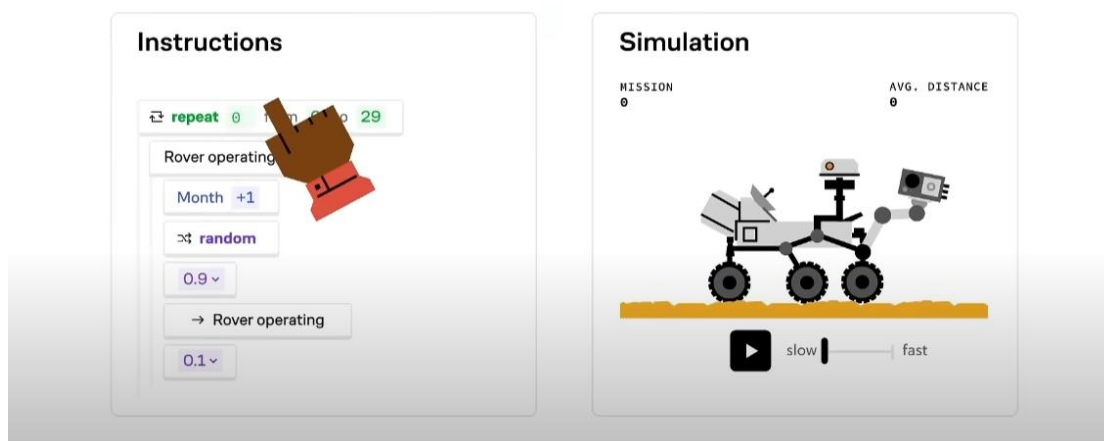
We can use code blocks to navigate a plane that paints as it moves.



ภาพที่ 5.1 หน้า แบบฝึกหัดตัวอย่าง

แบบฝึกหัดนี้มาจากเว็บไซต์ Brilliant.org โดยผู้เรียนจะเห็นฉากเป็นตารางกริด มีวัตถุ เช่น เครื่องบิน และจุดที่ต้องผ่านหรือระบายสี จากนั้นจะได้รับบล็อกคำสั่งพื้นฐาน เช่น move forward หรือ turn right เพื่อให้เรียงลำดับอย่างถูกต้อง แล้วกดปุ่ม play เพื่อให้เครื่องบินเคลื่อนที่ตามคำสั่ง หากยังไม่ถูกต้อง ผู้เรียนสามารถปรับคำสั่งและลองใหม่ได้ทันที โดยผู้เรียนต้องใช้เหตุผลในการเรียงลำดับคำสั่ง เพื่อให้เครื่องบินระบายจุดสีบนแผนที่ให้ตรงกับตำแหน่งเป้าหมาย

Simulations



ภาพที่ 5.2 หน้า แบบฝึกหัดตัวอย่าง

แบบฝึกหัดจำลองสถานการณ์ ที่ให้ผู้เรียนใช้บล็อกคำสั่งเพื่อควบคุมยานสำรวจ (Rover) โดยมีคำสั่งเช่น repeat, random, และตัวแปรต่าง ๆ ที่สามารถปรับค่าได้ ผู้เรียนจะต้องจัดคำสั่งให้ยานทำงานและดูผลลัพธ์ในแถบด้านขวาที่แสดงภาพเคลื่อนไหวของยานและระยะทางที่เดินทางได้

3. ผู้ใช้บางส่วนเข้าใช้ระบบด้วย smartphone หรือ safari browser สามารถต่อยอดโดยการทำให้ระบบรองรับผู้ใช้กลุ่มนี้ด้วย

รายการอ้างอิง

Kipakapron(2018) . เกร็ดความรู้ CSS คืออะไร สืบค้น 16/11/2567 จาก

<https://blog.sogoodweb.com/Article/Detail/79237/CSS%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3%E0%B8%A1%E0%B8%B5%E0%B8%9B%E0%B8%A3%E0%B8%B0%E0%B9%82%E0%B8%A2%E0%B8%8A%E0%B8%99%E0%B9%8C%E0%B8%AD%E0%B8%A2%E0%B9%88%E0%B8%B2%E0%B8%87%E0%B9%84%E0%B8%A3>

วรพงศ์(2020) . JavaScript สืบค้น 16/11/2567 จาก

<https://webdodee.com/what-is-javascript/>

developer.mozilla.org(2024) SVG: Scalable Vector Graphics สืบค้น
16/11/2567 จาก

<https://developer.mozilla.org/en-US/docs/Web/SVG>

ไขข้อสงสัย 3D Modeling ออกแบบสามมิติ คืออะไร มีกี่ประเภท และมีประโยชน์
อย่างไร? (2023) สืบค้น 16/11/2567 จาก

<https://www.tkk3dprinting.com/3dmodeling/>

W3Schools. (n.d.). *W3Schools Online Web Tutorials*.

สืบค้น 16/11/2567 จาก

<https://www.w3schools.com>

Mozilla Developer Network. (n.d.). *MDN Web Docs*.

สืบค้น 16/11/2567 จาก

<https://developer.mozilla.org/en-US/>

Smooth as Butter: Achieving 60 FPS Animations with CSS3(2016)

สืบค้น 16/11/2567 จาก

<https://medium.com/outsystems-experts/how-to-achieve-60-fps-animations-with-css3-db7b98610108>

ภาคผนวก

ภาคผนวก ก: แบบสอบถามการประเมินการใช้งานระบบ จัดทำโดยใช้ Google Form เพื่อวัดความพึงพอใจของผู้ใช้งานระบบด้านเนื้อหา การนำเสนอ และ ประสิทธิภาพใช้งาน โดยมีหัวข้อให้ประเมินในระดับคะแนน 1-5

<https://forms.gle/fd6jawmJSTrQPfGJ6>

ภาคผนวก ข: แผนผังการทำงานของระบบ แผนภาพแสดง flow การทำงานของระบบ ตั้งแต่การเข้าสู่ระบบ(ไม่ต้อง log in) > เลื่อกบทเรียน > ศึกษาเนื้อหา > ทำแบบทดสอบ > ดูผลคะแนน

ภาคผนวก ค: ลิงก์ระบบต้นแบบออนไลน์ ผู้พัฒนาจัดทำระบบต้นแบบให้ทดลองใช้งานจริงผ่านเว็บไซต์ที่อยู่ด้านล่าง

ลิงก์: <https://efanimation.pages.dev/>

ระบบสามารถทดลองเรียนบทเรียน ทำแบบทดสอบ และดูผลลัพธ์ได้ทันที

