

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

จากการที่ผู้จัดทำได้มีโอกาสทำงานกับระบบสารสนเทศเพื่อการจัดการแข่งขันกีฬาพุทธรักษาเกมส์ ในช่วงเริ่มต้นของการจัดงาน พบว่าระบบเดิมที่ใช้ฐานข้อมูล Google Sheet มีข้อจำกัดในการจัดการข้อมูลจำนวนมากและมีปัญหาเรื่องความเร็วในการประมวลผล ซึ่งส่งผลกระทบต่อประสิทธิภาพและความราบรื่นของการจัดการแข่งขัน ด้วยเหตุนี้ ผู้จัดทำจึงเล็งเห็นถึงความจำเป็นในการพัฒนาระบบใหม่ที่สามารถรองรับปริมาณข้อมูลที่มากขึ้น มีความเสถียร และทำงานได้อย่างรวดเร็ว จึงได้ตัดสินใจนำความรู้และประสบการณ์ด้านการพัฒนาเว็บไซต์มาประยุกต์ใช้ โดยเลือกใช้ฐานข้อมูล MySQL ที่มีประสิทธิภาพสูงกว่า และพัฒนาส่วนหน้าของระบบด้วย React Framework ซึ่งเป็นเทคโนโลยีที่ทันสมัยและมีความสามารถในการสร้างส่วนต่อประสานผู้ใช้ (User Interface) ที่ใช้งานง่ายและตอบสนองได้ดี การพัฒนาระบบในครั้งนี้ไม่เพียงแต่เป็นการแก้ไขปัญหาที่เกิดขึ้นกับระบบเดิม แต่ยังเป็นการต่อยอดผลงานและเพิ่มขีดความสามารถของระบบให้สามารถรองรับการจัดการแข่งขันกีฬาในระดับที่ใหญ่ขึ้นได้อย่างมีประสิทธิภาพ โดยระบบใหม่นี้จะช่วยให้การจัดการข้อมูลนักกีฬา การลงทะเบียน การจัดการตารางการแข่งขัน การประกาศผล และอื่นๆ เป็นไปอย่างเป็นระบบ รวดเร็ว และถูกต้องแม่นยำ ซึ่งจะส่งผลดีต่อการพัฒนางานกีฬารวมของประเทศในภาพรวม

นอกจากนี้ การพัฒนาระบบในครั้งนี้ยังเป็นโอกาสอันดีในการเรียนรู้และพัฒนาทักษะของตนเองในด้านการพัฒนาเว็บไซต์ โดยได้นำความรู้ที่เรียนมาประยุกต์ใช้ในการทำงานจริง และได้เรียนรู้เทคนิคใหม่ๆ ในการพัฒนาเว็บไซต์ด้วย React Framework ซึ่งจะเป็นประโยชน์อย่างยิ่งต่อการทำงานในอนาคต

1.2 วัตถุประสงค์

3.1 ระบบสารสนเทศสามารถลงทะเบียน ค้นหา และพิมพ์ข้อมูลของนักศึกษาที่ลงแข่งขัน เจ้าหน้าที่ผู้คุมนักกีฬา องค์การนักศึกษา และกรรมการตัดสิน

3.2 ระบบสารสนเทศสามารถแสดงผลการแข่งขันของแต่ละกีฬา และพิมพ์เกียรติบัตรของผู้เข้าร่วมงาน

3.3 ผู้ดูแลระบบสามารถเพิ่ม ลบ และแก้ไขข้อมูลต่าง ๆ ที่เกี่ยวข้องกับการแข่งขันได้อย่างสะดวก.

3.4 สามารถเพิ่มสิทธิการจัดการข้อมูลต่างให้กับ คณะกรรมการ สามารถแก้ไขข้อมูลผลการแข่งขัน ของกีฬาต่างๆได้

1.3 ขอบเขตของโครงการ

การพัฒนาระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรีฑา กีฬา พุทธรักษาเกมส์ ที่พัฒนาขึ้นในโครงการนี้มีขอบเขต ความสามารถดังต่อไปนี้

1.3.1 การออกแบบระบบสารสนเทศ เพื่อตอบโจทย์การทำงาน ของงานพุทธรักษาเกมส์ที่มากยิ่งขึ้น

1.3.2 การพัฒนาระบบเว็บไซต์ด้วย react framework ด้วยภาษา javascript

1.3.3 สร้างระบบจัดการข้อมูลผู้สมัครโดนมีแอดมินและกรรมการ สามารถแก้ไขข้อมูลได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 สามารถลงทะเบียน ค้นหา และพิมพ์ข้อมูลของนักกีฬา เจ้าหน้าที่ องค์การนักศึกษา และกรรมการตัดสินได้อย่างสะดวกและรวดเร็ว

1.4.2 ระบบสามารถแสดงผลการแข่งขันของแต่ละกีฬาแบบเรียลไทม์ และพิมพ์เกียรติบัตรสำหรับผู้เข้าร่วมได้ทันที

1.4.3 ผู้ดูแลระบบสามารถตรวจสอบข้อมูล และจัดการข้อมูลต่างๆของระบบสารสนเทศได้

1.4.4 คณะกรรมการสามารถเข้าถึงและแก้ไขข้อมูลผลการแข่งขันได้ตามสิทธิที่ได้รับ ทำให้การจัดการข้อมูลมีความถูกต้องและเชื่อถือได้

1.5 อุปกรณ์เครื่องมือที่ใช้ในโครงการ

1.5.1 ฮาร์ดแวร์ ประกอบด้วย

- 1) คอมพิวเตอร์โน้ตบุ๊ค huawei matebook
- 2) หน่วยประมวลผล CPU intel(R) Core(TM) i5-1135G7 ความเร็ว 2.40 GHz
- 3) หน่วยความจำ RAM 8 GB
- 4) หน่วยความสำรอง SSD 512 GB
- 5) ระบบปฏิบัติการ Windows 11

1.5.2 ซอฟต์แวร์ ประกอบด้วย

- 1) โปรแกรม Visual Studio Code ใช้ในการพัฒนาเว็บไซต์และเขียนโค้ด
- 2) react framework ใช้ในการพัฒนาหน้าตาของเว็บไซต์
- 3) ภาษาที่ใช้ในการทำงาน javascript

4) ใช้ docker ในการจำลอง server กับ เผยแพร่งาน

5) โปรแกรม Database (Mysql) ใช้ในการเชื่อมต่อและจัดการฐานข้อมูล

1.6 นิยามศัพท์ที่ใช้ในโครงการ

1.6.1 ระบบสารสนเทศ (Information System): องค์ประกอบต่างๆที่มีความเกี่ยวข้องและทำงานประสานกันในการเก็บรวบรวม บันทึก ประมวลผล จัดเก็บและแจกจ่ายสารสนเทศ เพื่อสนับสนุนการตัดสินใจและหน้าที่ทางการบริหาร ซึ่งได้แก่ การวางแผน การจัดองค์กร การประสานงาน การควบคุมและการสื่อสารภายในองค์กร

1.6.2 ระบบฐานข้อมูล (Database System): การออกแบบและการจัดเก็บข้อมูลในรูปแบบที่สามารถเข้าถึงและใช้งานได้อย่างมีประสิทธิภาพ เพื่อให้สามารถจัดการข้อมูลผู้ใช้และข้อมูลทางการแพทย์ได้อย่างเหมาะสม

1.6.3 รีแอคเฟรมเวิร์ค (React framework): React เป็นไลบรารี JavaScript ที่พัฒนาโดย Facebook เพื่อช่วยสร้าง User Interface (UI) ที่มีประสิทธิภาพ โดย React มุ่งเน้นการสร้าง Component ซึ่งเป็นส่วนประกอบของ UI ที่สามารถใช้ซ้ำได้ และแต่ละ Component สามารถเก็บสถานะ (state) และเมทอด (methods) ต่างๆ เพื่อการจัดการกับข้อมูลและการแสดงผล

1.6.4 Docker คือแพลตฟอร์มซอฟต์แวร์ที่ช่วยให้คุณสร้าง ทดสอบ และติดตั้งแอปพลิเคชันให้จริงได้อย่างรวดเร็ว docker จะบรรจุซอฟต์แวร์ลงไปเป็นหน่วยที่เป็นมาตรฐานเรียกว่า คอนเทนเนอร์ ซึ่งจะมีทุกสิ่งที่ซอฟต์แวร์ต้องใช้ในการเรียกใช้งาน ซึ่งรวมถึงไลบรารี เครื่องมือสำหรับระบบ โค้ด และรันไทม์ เมื่อใช้ docker คุณสามารถปรับใช้และปรับขนาดแอปพลิเคชันลงในทุกสภาพแวดล้อมและทราบว่าโค้ดของคุณจะเรียกใช้ได้อย่างอย่างรวดเร็ว

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

การพัฒนาระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรณีศึกษา พุทธรักษาเกมส์ ผู้วิจัย ได้มีการศึกษาทฤษฎีที่เกี่ยวข้องต่าง ๆ และนำทฤษฎีเหล่านั้นมาประยุกต์ใช้ เพื่อระบบที่พัฒนาขึ้น มีความถูกต้องตรงกับความต้องการผู้ใช้ และทำให้การใช้งานระบบนี้เป็นไปอย่างสะดวก และมีประสิทธิภาพมากที่สุด โดยทฤษฎีต่าง ๆ ที่เกี่ยวข้องในการจัดทำโครงการครั้งนี้ มีรายละเอียดของทฤษฎีดังต่อไปนี้

2.1 ทฤษฎีและความรู้ที่เกี่ยวข้อง

2.1.1 ทฤษฎีเกี่ยวกับ React

2.1.2 ทฤษฎีเกี่ยวกับ Nodejs

2.1.3 ทฤษฎีเกี่ยวกับ Docker

2.1.4 ทฤษฎีเกี่ยวกับ SQL

2.1.5 ทฤษฎีเกี่ยวกับ Javascript

2.2 งานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีและความรู้ที่เกี่ยวข้อง

2.1.1 React คือไลบรารีจาวาสคริปต์ที่ถูกมองว่าเป็นตัวช่วยให้สามารถสร้าง UI (User Interface หรือองค์ประกอบของเว็บที่เชื่อมต่อกับผู้ใช้งานโดยตรง) ได้แม่นยำและรวดเร็วมากยิ่งขึ้น และส่งผลให้การแสดงผลมีความเป็นระบบคงเส้นคงวามากขึ้นไปพร้อมๆ กัน สิ่งนี้ถูกนำมาใช้กันอย่างแพร่หลายในการเขียนโค้ดสำหรับเว็บไซต์และแอปพลิเคชันมีประโยชน์กับคนที่ทำงานในฐานะ Front-end จึงนับว่าเป็นหนึ่งในสิ่งที่คนทำงานในสายนักพัฒนาจำเป็นต้องเรียนรู้



ภาพที่ 1 แสดงสัญลักษณ์ React

ข้อดี

1. React สามารถนำ ตัวโค้ดส่วนต่างๆ นำกลับมาใช้ใหม่ได้ทำให้ง่ายใช้งานง่ายขึ้นและลดการเขียนซ้ำของโค้ดได้
2. React จะอัปเดตเฉพาะส่วนที่จำเป็นเท่านั้น ทำให้แอปพลิเคชันของคุณทำงานได้เร็วขึ้น
3. React มีชุมชนขนาดใหญ่และเอกสารประกอบที่ครอบคลุม ทำให้ง่ายต่อการเรียนและแก้ไขปัญหา

ข้อเสียของ React

1. ยากต่อการเรียนรู้ต้องมีพื้นเรื่อง jsx มาก่อน ทั้งเรื่อง api ติดต่อกับหลังบ้านที่ต้องเรียนรู้
2. ไลบรารี React มีขนาดค่อนข้างใหญ่ ซึ่งอาจส่งผลต่อประสิทธิภาพของแอปพลิเคชันของคุณ
3. React มีการอัปเดตอย่างรวดเร็ว ซึ่งอาจทำให้คุณต้องเรียนรู้สิ่งใหม่ ๆ อยู่เสมอ

2.1.2 NodeJS คือแพลตฟอร์มที่มีขนาดเล็กและมีประสิทธิภาพ เหมาะสำหรับการพัฒนาแอปพลิเคชันที่ต้องการความเร็ว และสร้างขึ้นบน Runtime ของ JavaScript บน Chrome ซึ่ง NodeJS เหมาะสำหรับแอปพลิเคชันที่ใช้ข้อมูลจำนวนมาก ซึ่งสามารถทำงานบนอุปกรณ์ได้หลายรูปแบบ โดยคุณสามารถติดตั้งและใช้งานบน Open Landscape Cloud



ภาพที่ 2 แสดงสัญลักษณ์ Nodejs

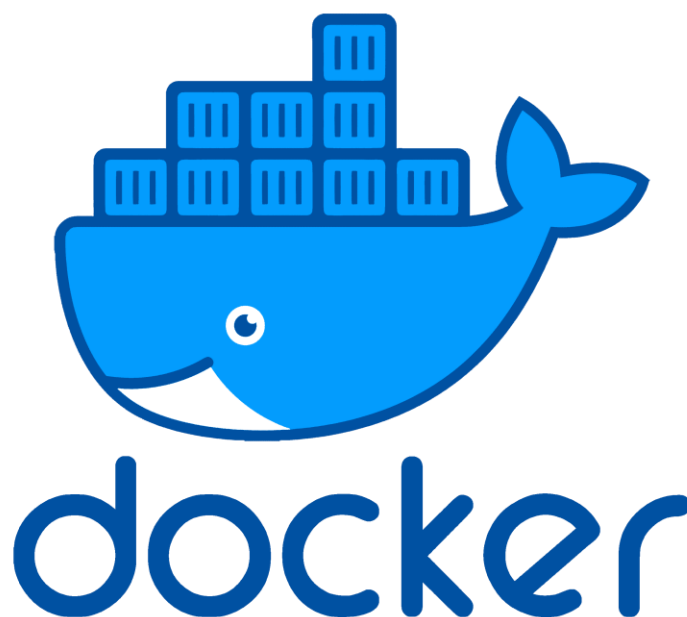
ข้อดีของ Nodejs

1. ประสิทธิภาพสูงในการจัดการงานแบบ non-blocking I/O ทำให้รองรับผู้ใช้งานจำนวนมากได้ดี
2. มี package manager (npm) ที่มี library และ module จำนวนมาก ช่วยพัฒนาได้รวดเร็ว
3. เหมาะกับการพัฒนา real-time applications เช่น chat, game หรือ streaming เพราะจัดการการเชื่อมต่อได้ดี
4. มีชุมชนนักพัฒนาขนาดใหญ่ ทำให้หาความช่วยเหลือและแหล่งเรียนรู้ได้ง่าย

ข้อเสีย

1. ไม่เหมาะกับงานที่ต้องประมวลผลหนัก ๆ เพราะ JavaScript เป็น single-threaded
2. Callback hell อาจเกิดขึ้นได้ถ้าไม่จัดการ code ให้ดี ทำให้อ่านและดูแลยาก
3. Library บางตัวอาจไม่เสถียรหรือไม่ได้รับการพัฒนาต่อ เพราะระบบนิเวศเปิดกว้าง
4. ประสิทธิภาพอาจด้อยกว่าภาษาที่ compile แบบ static เช่น Go หรือ Java ในบางกรณี

2.1.3 Docker คือแพลตฟอร์มซอฟต์แวร์ที่ช่วยให้คุณสร้าง ทดสอบ และติดตั้งแอปพลิเคชันใช้จริงได้อย่างรวดเร็ว Docker จะบรรจุซอฟต์แวร์ลงในหน่วยที่เป็นมาตรฐานเรียกว่า คอนเทนเนอร์ ซึ่งจะมีทุกสิ่งที่ซอฟต์แวร์ต้องใช้ในการเรียกใช้งาน ซึ่งรวมถึงไลบรารี เครื่องมือสำหรับระบบโค้ด และรันไทม์ เมื่อใช้ Docker คุณสามารถปรับใช้และปรับขนาดแอปพลิเคชันลงในทุกสภาพแวดล้อมและทราบว่าโค้ดของคุณจะเรียกใช้ได้อย่างอย่างรวดเร็ว



ภาพที่ 3 แสดงสัญลักษณ์ ภาษา Docker

ข้อดีของ Docker

1. รับประกันความสม่ำเสมอของสภาพแวดล้อมการทำงาน – แอปพลิเคชันจะทำงานเหมือนกันทุกที่ ไม่ว่าจะเป็นเครื่องพัฒนา เครื่องทดสอบ หรือเซิร์ฟเวอร์จริง
2. ทำให้ตั้งค่าและติดตั้งระบบได้เร็วขึ้นมาก เพราะไม่ต้องติดตั้งโปรแกรมและตั้งค่าทีละอย่าง สามารถใช้ image ที่กำหนดไว้แล้ว
3. ประหยัดทรัพยากรมากกว่า Virtual Machine เพราะใช้ kernel ร่วมกับระบบปฏิบัติการหลัก
4. แยกส่วนการทำงานได้ดี (Isolation) ทำให้แต่ละบริการไม่รบกวนกัน และจัดการได้ง่าย
5. สามารถ scale ขึ้นลงได้ง่าย เหมาะกับระบบที่ต้องการความยืดหยุ่นสูง

ข้อเสียของ Docker

1. มีความซับซ้อนในการเรียนรู้และตั้งค่าเบื้องต้น โดยเฉพาะสำหรับทีมที่ไม่คุ้นเคย
2. Performance อาจด้อยกว่าการรันโปรแกรมบนเครื่องโดยตรงเล็กน้อย เพราะมี overhead จากการทำ containerization
3. ความปลอดภัยต้องได้รับการดูแลเป็นพิเศษ เพราะ container ที่มีช่องโหว่อาจส่งผลกระทบต่อระบบหลักได้

2.1.4 SQL Structured Query Language (SQL) เป็นภาษาโปรแกรมสำหรับจัดเก็บและประมวลผลข้อมูลในฐานข้อมูลแบบเชิงสัมพันธ์ ฐานข้อมูลแบบเชิงสัมพันธ์เก็บข้อมูลในรูปแบบตารางที่มีแถวและคอลัมน์ที่เป็นตัวแทนของหมวดข้อมูลที่แตกต่างกันและความสัมพันธ์ต่างๆ ระหว่างค่าข้อมูล สามารถใช้คำสั่ง SQL ในการจัดเก็บ ปรับปรุง ลบ ค้นหา และดึงข้อมูลจากฐานข้อมูล นอกจากนี้ยังสามารถใช้ SQL ในการรักษาและเพิ่มประสิทธิภาพการทำงานของฐานข้อมูล (Database) โดย SQL ช่วยให้นักพัฒนาสามารถเข้าถึงข้อมูล, ค้นหาข้อมูล, เพิ่ม, ปรับปรุงข้อมูลได้อย่างมีประสิทธิภาพ ช่วยให้ผู้ใช้สามารถดำเนินการจัดการฐานข้อมูลได้หลากหลายรูปแบบ เช่น การสร้างฐานข้อมูลใหม่, การสร้างตาราง, การเพิ่มข้อมูล, การแก้ไขข้อมูล, และการลบข้อมูล การเชื่อมต่อกับ MySQL/MariaDB เป็นที่นิยมใช้บ่อยในทุกประเภทของแอปพลิเคชัน นักวิเคราะห์ข้อมูลและนักพัฒนาเรียนรู้และใช้ SQL เพราะมันบูรณาการได้ดีกับการเขียนโปรแกรมภาษาที่แตกต่างกัน ตัวอย่างเช่น พวกเขาสามารถฝังแบบสอบถาม SQL

กับการเขียนโปรแกรมภาษา Java เพื่อสร้างประสิทธิภาพสูงโปรแกรมประมวลผลข้อมูลที่มีระบบฐานข้อมูล SQL ที่สำคัญเช่น Oracle หรือ MS SQL Server SQL ยังคงค่อนข้างง่ายที่จะเรียนรู้ในขณะที่มันใช้คำหลักภาษาอังกฤษทั่วไปในงบ



ภาพที่ 2 แสดงสัญลักษณ์ ภาษา SQL

ข้อดีของ SQL

1. SQL เป็นภาษาที่มีรูปแบบการเขียนที่ชัดเจนและอ่านเข้าใจได้ง่าย ทำให้ผู้ใช้สามารถเข้าใจคำสั่งได้โดยไม่ต้องมีพื้นฐานการเขียนโปรแกรมที่ซับซ้อน
2. SQL เป็นมาตรฐานที่ใช้กันอย่างแพร่หลายในการจัดการฐานข้อมูล ทำให้สามารถใช้คำสั่ง SQL ได้กับฐานข้อมูลหลายประเภท เช่น MySQL, PostgreSQL, Oracle และ SQL Server
3. SQL ช่วยให้ผู้ใช้สามารถสร้าง, อ่าน, อัปเดต, และลบข้อมูล (CRUD) ได้
4. SQL สามารถจัดการกับข้อมูลจำนวนมากได้ดี และมีฟังก์ชันในการทำงานกับข้อมูลที่ช่วยเพิ่มประสิทธิภาพในการจัดการข้อมูล
5. SQL มีความสามารถในการสร้างคำสั่งสืบค้นที่ซับซ้อนได้ เช่น การใช้ JOIN, GROUP BY, และการจัดการฟังก์ชันแบบ aggregate ซึ่งช่วยให้สามารถดึงข้อมูลตามที่ต้องการได้

ข้อเสียของ SQL

1. ความซับซ้อนในการจัดการข้อมูลที่ไม่เป็นระเบียบ
2. ผู้ใช้ที่ไม่มีพื้นฐานด้านการจัดการฐานข้อมูลหรือ SQL อาจมีความยากลำบากในการเริ่มต้นใช้งาน และต้องใช้เวลาในการเรียนรู้
3. สำหรับฐานข้อมูลที่มีข้อมูลจำนวนมาก การเขียนคำสั่ง SQL ที่ไม่เหมาะสมอาจทำให้ประสิทธิภาพในการสืบค้นข้อมูลลดลง
4. ความปลอดภัยของข้อมูลหากไม่ได้จัดการสิทธิ์ในการเข้าถึงฐานข้อมูลอย่างเหมาะสม อาจเสี่ยงต่อการถูกโจมตี เช่น SQL Injection ซึ่งอาจนำไปสู่การเข้าถึงข้อมูลที่ไม่ได้รับอนุญาต
5. ไม่มีฟังก์ชันสำหรับการจัดการข้อมูลในรูปแบบ NoSQLSQL ไม่รองรับการจัดการฐานข้อมูลที่เป็น NoSQL ซึ่งอาจจำเป็นสำหรับแอปพลิเคชันที่ต้องการจัดการข้อมูลที่ไม่เป็นระเบียบหรือมีการขยายตัวได้ง่าย

2.1.5 JavaScript เป็นภาษาโปรแกรมที่นักพัฒนาใช้ในการสร้างหน้าเว็บแบบอินเทอร์แอคทีฟ ตั้งแต่การรีเฟรชฟีดโซเชียลมีเดียไปจนถึงการแสดงผลภาพเคลื่อนไหวและแผนที่แบบอินเทอร์แอคทีฟ ฟังก์ชันของ JavaScript สามารถปรับปรุงประสบการณ์ที่ผู้ใช้จะได้รับจากการใช้งานเว็บไซต์ และในฐานะที่เป็นภาษาในการเขียนสคริปต์ฝั่งไคลเอนต์ จึงเป็นหนึ่งในเทคโนโลยีหลักของ World

Wide Web ยกตัวอย่างเช่น เมื่อคุณท่องเว็บแล้วเห็นภาพสไลด์ เมนูรื้อปดาวน์แบบคลิกให้แสดงผล หรือสิ่งประกอบที่เปลี่ยนแบบไดนามิกบนหน้าเว็บ นั่นคือคุณเห็นเอฟเฟกต์ของ JavaScript



ภาพที่ 3 แสดงสัญลักษณ์ ภาษา Javascript

ข้อดีของ JavaScript

1. เรียนรู้ได้ง่าย เป็นภาษาระดับสูงที่มี syntax ไม่ซับซ้อน เหมาะสำหรับผู้เริ่มต้น
2. ทำงานได้ทั้งฝั่ง Frontend และ Backend (ผ่าน Node.js) ทำให้นักพัฒนาสามารถใช้ภาษาเดียวตลอดทั้ง stack
3. มีชุมชนขนาดใหญ่ ทำให้หาแหล่งเรียนรู้ library และความช่วยเหลือได้ง่าย
4. ทำงานบนเว็บเบราว์เซอร์ได้โดยตรง ไม่ต้องคอมไพล์ ช่วยให้พัฒนาและทดสอบได้รวดเร็ว
5. มีความยืดหยุ่นสูง สามารถเขียนได้หลายรูปแบบ ทั้งแบบ functional, object-oriented หรือ procedural

ข้อเสียของ Javascript

1. Type System ไม่เข้มงวด ทำให้อาจเกิดข้อผิดพลาดที่ตรวจจับได้ยากในระหว่างการพัฒนา (แม้จะมี TypeScript มาช่วย)
2. การจัดการ memory ไม่มีประสิทธิภาพเท่าภาษาระดับต่ำ เพราะมี garbage collection อัตโนมัติ
3. ความไม่สอดคล้องระหว่างเบราว์เซอร์ต่างๆ อาจทำให้ต้องเขียนโค้ดเพิ่มเพื่อรองรับ
4. การทำงานแบบ asynchronous อาจทำให้เกิด callback hell ถ้าไม่จัดการให้ดี (แม้จะมี Promise และ async/await มาช่วย)

2.2 งานวิจัยที่เกี่ยวข้อง

น.ส.ธัญมน นิศามณีวงศ์. ระบบสารสนเทศสำหรับจัดการแข่งขันยิงธนู มหาวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย (2565) “ระบบสารสนเทศสำหรับจัดการแข่งขันยิงธนู” โดยมีแนวคิด และหลักการที่นำมาใช้ในการพัฒนาระบบ ได้แก่ แนวคิดในการพัฒนาเว็บแอปพลิเคชัน แนวคิดด้านคลังข้อมูล แนวคิดด้านการวิเคราะห์ข้อมูล และแนวคิดในการแข่งขันกีฬายิงธนู

สุนิสา ริมเจริญ. "ระบบการจัดการแข่งขันกีฬาสถาบันการพลศึกษาแห่งประเทศไทย." (2560) ระบบแข่งขันกีฬาสถาบันการพลศึกษาแห่งประเทศไทย ผู้จัดทำงานนิพนธ์พัฒนาระบบขึ้น เพื่อช่วยสนับสนุนการตัดสินใจของผู้บริหารในการพัฒนาศักยภาพทางด้านกีฬาแก่สถาบันการพลศึกษาและเพื่อสนับสนุนส่งเสริมให้นักกีฬาที่มีความสามารถได้เข้าแข่งขันในระดับกีฬามหาวิทยาลัย เพื่อให้ระบบมีประสิทธิภาพ

บทที่ 3

ขั้นตอนการดำเนินงาน

การศึกษาค้นคว้าเรื่อง ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรณีศึกษา พุทธรักษาเกมส์ ผู้ศึกษาได้ค้นคว้าทฤษฎีที่เกี่ยวข้อง ได้ข้อมูลที่จะนำมาออกแบบระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรณีศึกษา พุทธรักษาเกมส์ ได้ดังนี้

3.1 การวิเคราะห์และออกแบบระบบ

- 3.1.1 ยูสเคสไดอะแกรม (Use Case Diagram)
- 3.1.2 ซีควเอนซ์ไดอะแกรม (Sequence Diagram)
- 3.1.3 แอคทิวิตีไดอะแกรม (Activity Diagram)
- 3.1.4 อีอาร์ไดอะแกรม (E-R Diagram)
- 3.1.5 พจนานุกรมข้อมูล (Data Dictionary)
- 3.1.6 โฟลว์ชาร์ต (Flowchart)

3.2 การออกแบบระบบ

3.1 การวิเคราะห์ระบบ

โครงการพัฒนาระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรณีศึกษา พุทธรักษาเกมส์ เพื่อสร้างระบบสารสนเทศเพื่อต่อยอดเว็บไซต์ พุทธรักษาเกมส์ ในข้อผิดพลาดของระบบบางอย่างให้ใช้งานง่ายและสบายยิ่งขึ้น จึงออกแบบระบบดังนี้

3.1.1 Use case diagram

Use Case Diagram คือ แผนภาพที่มีไว้เพื่อแสดงการทำงานของผู้ใช้งานระบบ (Use) และความสัมพันธ์กับระบบย่อย (Sub system) ภายในระบบใหญ่ในการเขียน Use Case Diagram ผู้ใช้งานระบบ จะถูกกำหนดให้เป็น Actor และระบบย่อย คือ Use Case Diagram จุดประสงค์หลักในการเขียน Use Case Diagram เพื่อเล่าเรื่องราวทั้งหมดของระบบว่ามีการทำงานอะไรบ้าง เป็นการดึง Requirement หรือเรื่องราวต่างๆ ของระบบจากผู้ใช้งาน ซึ่งถือว่าเป็นจุดเริ่มต้นในการวิเคราะห์ และออกแบบระบบ นอกจากนั้น Use Case ทุกๆตัวจะต้องอยู่ภายในสี่เหลี่ยมเดียวกันซึ่งมีชื่อระบบระบุอยู่ด้วย

สัญลักษณ์ที่ใช้ใน Use Case Diagram คือ

1. ยูสเคส (Use Case) คือ แผนภาพที่ใช้แสดงให้ทราบว่าระบบทำงานหรือมีหน้าที่อะไรบ้าง โดยมีสัญลักษณ์รูปวงรีแทน Use Case และสัญลักษณ์รูปคน Actor สำหรับชื่อ Use Case นั้นซึ่งต้องใช้คำกริยาหรือกริยาวลีก็ได้

2. แอ็กเตอร์ (Actor) คือ ผู้ที่กระทำกับระบบ หรือผู้ที่เกี่ยวข้อง โดยจะเป็นคนหรือโม๊กก็ได้ซึ่งเป็นผู้แลกเปลี่ยนข้อมูลข่าวสารกับระบบที่ทำการพัฒนา โดยเราจะใช้สัญลักษณ์รูปคนแทนสัญลักษณ์ของ Actor นั้น

3. เส้นแสดงความสัมพันธ์ (Relationship) คือ เส้นแสดงความสัมพันธ์ระหว่างเส้น Actor และ Use Case Diagram เพื่อแสดงการใช้งานของ Use Case Diagram

ในส่วน Use Case Diagram ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรณีศึกษา พุทธรักษาเกมส์ เป็นดังภาพที่ 6



ภาพที่ 6 Use Case Diagram ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรีฑา กีฬา ฟุตบอล เกม

ตารางที่ 1 อธิบายเหตุการณ์ที่เกิดขึ้นใน Use Case Diagram เข้าสู่ระบบ

Use Case Id	1
Use Case Name	เข้าสู่ระบบ
Scenario	เข้าสู่ระบบ
Triggering Event	NONE
Primary Actors	admin
Related Use Case	NONE
Precondition	–
Post Condition	–
Flow of Events for Main Success Scenario:	
Main Flow	<ul style="list-style-type: none"> – กรอกข้อมูล username password – กรอกข้อมูลให้ถูกต้อง
Flow of Events for Extension (Alternate Scenario):	
Exceptional Flow	<ul style="list-style-type: none"> –กรณีกรอกข้อมูลไม่ครบ –กรณีกรอกข้อมูลไม่ถูกต้อง

ตารางที่ 2 อธิบายเหตุการณ์ที่เกิดขึ้นใน Use Case Diagram กำหนดสิทธิ์การเข้าถึง

Use Case Id	2
Use Case Name	กำหนดสิทธิ์การเข้าถึง
Scenario	กำหนดสิทธิ์การเข้าถึงข้อมูล
Triggering Event	NONE
Primary Actors	admin
Related Use Case	NONE
Precondition	–
Post Condition	–
Flow of Events for Main Success Scenario:	
Main Flow	<ul style="list-style-type: none"> – เพิ่มข้อมูล username password ให้กับกรรมการ – กำหนดขอบเขตการเข้าถึงข้อมูลของ กรรมการ
Flow of Events for Extension (Alternate Scenario):	
Exceptional Flow	–กรณีการออกข้อมูลไม่ครบ

ตารางที่ 3 อธิบายเหตุการณ์ที่เกิดขึ้นใน Use Case Diagram จัดการข้อมูลการแข่งขัน	
Use Case Id	3
Use Case Name	จัดการข้อมูลการแข่งขัน
Scenario	จัดการข้อมูลการแข่งขัน
Triggering Event	NONE
Primary Actors	admin,director
Related Use Case	NONE
Precondition	เข้าสู่ระบบ
Post Condition	-
Flow of Events for Main Success Scenario:	
Main Flow	- สามารถจัดข้อมูลเกี่ยวกับการแข่งขันของแต่ละประเภท ก็ทำได้
Flow of Events for Extension (Alternate Scenario):	
Exceptional Flow	-กรณีการออกข้อมูลไม่ครบ -การออกข้อมูลผิดประเภท

ตารางที่ 4 อธิบายเหตุการณ์ที่เกิดขึ้นใน Use Case Diagram อนุมัติข้อมูลผู้สมัคร	
Use Case Id	4
Use Case Name	อนุมัติข้อมูลผู้สมัคร
Scenario	อนุมัติข้อมูลผู้สมัคร
Triggering Event	NONE
Primary Actors	admin
Related Use Case	NONE
Precondition	เข้าสู่ระบบ
Post Condition	–
Flow of Events for Main Success Scenario:	
Main Flow	<ul style="list-style-type: none"> – สามารถดูข้อมูลผู้สมัครได้ เมื่อพิจารณาข้อมูลว่าถูกต้องตามเงื่อนไขสมัคร สามารถอนุมัติได้ – ถ้าข้อมูลไม่ถูกต้องสามารถไม่อนุมัติ ข้อมูลผู้สมัครได้
Flow of Events for Extension (Alternate Scenario):	
Exceptional Flow	–หาข้อมูลไม่พบกรณีไม่มีข้อมูลสมัคร

ตารางที่ 5 อธิบายเหตุการณ์ที่เกิดขึ้นใน Use Case Diagram จัดการข้อมูลผู้สมัคร

Use Case Id	5
Use Case Name	จัดการข้อมูลผู้สมัคร
Scenario	จัดการข้อมูลผู้สมัคร
Triggering Event	NONE
Primary Actors	admin
Related Use Case	NONE
Precondition	เข้าสู่ระบบ
Post Condition	-
Flow of Events for Main Success Scenario:	
Main Flow	- สามารถจัดการข้อมูลจัดการข้อมูลผู้สมัคร
Flow of Events for Extension (Alternate Scenario):	
Exceptional Flow	- เลือกข้อมูลผิดประเภท

ตารางที่ 6 อธิบายเหตุการณ์ที่เกิดขึ้นใน Use Case Diagram จัดการข้อมูลข่าวสารกีฬา

Use Case Id	6
Use Case Name	จัดการข้อมูลข่าวสารกีฬา
Scenario	จัดการข้อมูลข่าวสารกีฬา
Triggering Event	NONE
Primary Actors	admin
Related Use Case	NONE
Precondition	เข้าสู่ระบบ
Post Condition	-
Flow of Events for Main Success Scenario:	
Main Flow	- สามารถจัดข้อมูลข่าวสารเกี่ยวกับงานกีฬาได้
Flow of Events for Extension (Alternate Scenario):	
Exceptional Flow	- กรอกข้อมูลไม่ครบ
	- กรอกข้อมูลผิดประเภท

ตารางที่ 7 อธิบายเหตุการณ์ที่เกิดขึ้นใน Use Case Diagram ลงทะเบียน

Use Case Id	7
Use Case Name	ลงทะเบียน
Scenario	ลงทะเบียนผู้สมัคร
Triggering Event	NONE
Primary Actors	user,director
Related Use Case	NONE
Precondition	-
Post Condition	-
Flow of Events for Main Success Scenario:	
Main Flow	- สามารถจัดข้อมูลข่าวสารเกี่ยวกับงานกีฬาได้
Flow of Events for Extension (Alternate Scenario):	
Exceptional Flow	- กรอกข้อมูลไม่ครบ
	- กรอกข้อมูลผิดประเภท

ตารางที่ 8 อธิบายเหตุการณ์ที่เกิดขึ้นใน Use Case Diagram พิมพ์บัตรประจำตัว

Use Case Id	8
Use Case Name	พิมพ์บัตรประจำตัว
Scenario	พิมพ์บัตรประจำตัวผู้สมัคร
Triggering Event	NONE
Primary Actors	user
Related Use Case	NONE
Precondition	-
Post Condition	-
Flow of Events for Main Success Scenario:	
Main Flow	<ul style="list-style-type: none"> - สามารถค้นหาข้อมูลของผู้สมัครตามประเภทต่างๆที่ลงทะเบียนไว้ได้ - เมื่อค้นหาเจอสามารถกดปุ่ม พิมพ์บัตรประจำตัวเพื่อให้พิมพ์บัตรประจำตัวได้
Flow of Events for Extension (Alternate Scenario):	
Exceptional Flow	<ul style="list-style-type: none"> - กรอกข้อมูลไม่ครบ - กรอกข้อมูลผิดประเภท

ตารางที่ 9 อธิบายเหตุการณ์ที่เกิดขึ้นใน Use Case Diagram พิมพ์เกียรติบัตร

Use Case Id	9
Use Case Name	พิมพ์เกียรติบัตร
Scenario	พิมพ์เกียรติบัตร
Triggering Event	NONE
Primary Actors	user
Related Use Case	NONE
Precondition	-
Post Condition	-
Flow of Events for Main Success Scenario:	
Main Flow	<ul style="list-style-type: none"> - สามารถค้นหาข้อมูลของผู้สมัครตามประเภทต่างๆที่ลงทะเบียนไว้ได้ - เมื่อค้นหาเจอสามารถกดปุ่ม พิมพ์เกียรติบัตรเป็น pdf ได้
Flow of Events for Extension (Alternate Scenario):	
Exceptional Flow	<ul style="list-style-type: none"> - กรอกข้อมูลไม่ครบ - กรอกข้อมูลผิดประเภท

ตารางที่ 10 อธิบายเหตุการณ์ที่เกิดขึ้นใน Use Case Diagram คู่มือการแข่งขันและผลสรุปเหรียญ

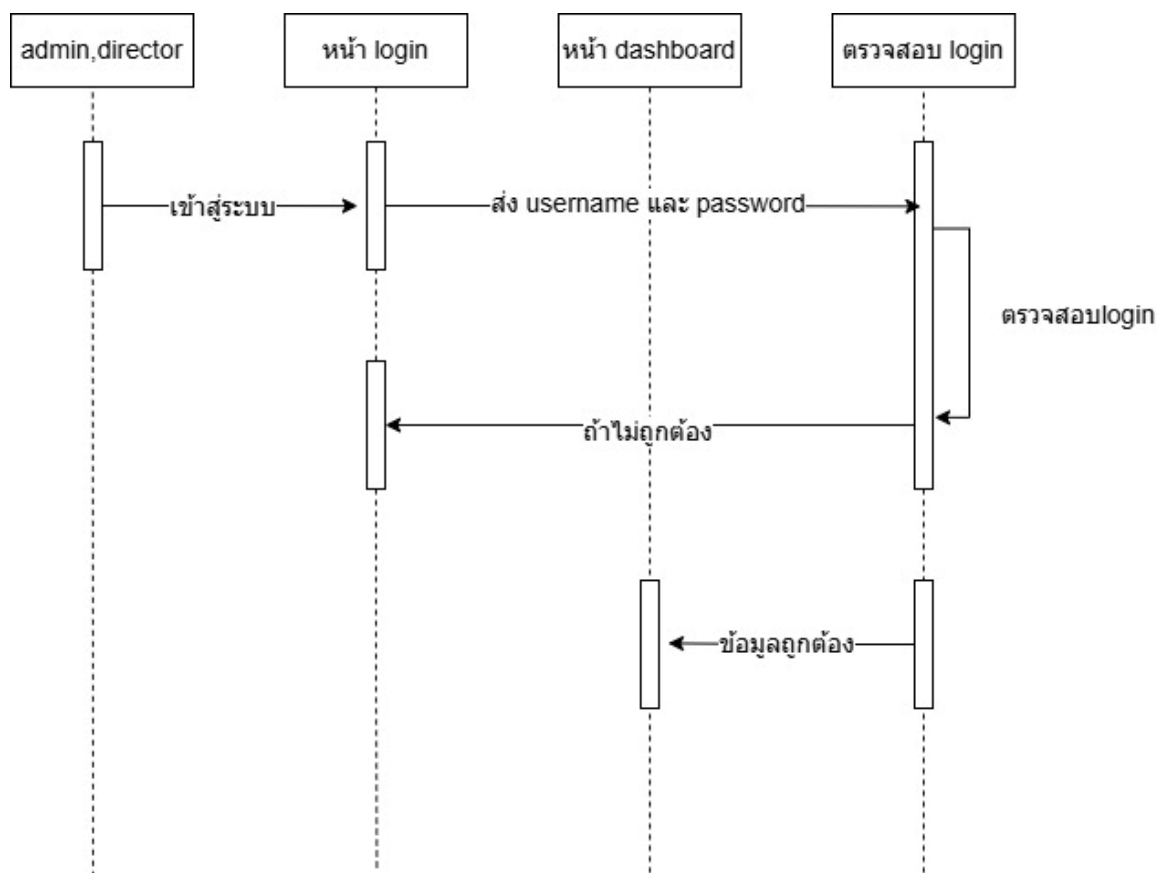
Use Case Id	10
Use Case Name	คู่มือการแข่งขันและผลสรุปเหรียญ
Scenario	คู่มือการแข่งขันและผลสรุปเหรียญ
Triggering Event	NONE
Primary Actors	user
Related Use Case	NONE
Precondition	-
Post Condition	-
Flow of Events for Main Success Scenario:	
Main Flow	<ul style="list-style-type: none"> - สามารถค้นหาข้อมูลผลการแข่งขันของกีฬาแต่ละประเภทได้ - สามารถดูผลการแข่งขันของกีฬาแต่ละประเภทได้ - สามารถดูจำนวนเหรียญรางวัลของแต่ละมหาลัยได้
Flow of Events for Extension (Alternate Scenario):	
Exceptional Flow	ไม่มี

3.1.2 ซีควเอนซ์ไดอะแกรม (Sequence Diagram)

ซีควเอนซ์ไดอะแกรม (Sequence Diagram) คือ การสร้างแบบจำลองเชิงกิจกรรม (Dynamic Model) หรือ (Behavioral Model) ซึ่งก็คือการจำลองกระบวนการที่ทำให้เกิดกิจกรรมของระบบ เกิดจากชุดกิจกรรม ซึ่งกิจกรรมหนึ่งๆนั้นเกิดจากการที่ Object หนึ่งตอบโต้กับอีก Object หนึ่ง Sequence Diagram เป็น Diagram ที่ประกอบด้วย Class หรือ Diagram ภายใน Sequence Diagram จะใช้สี่เหลี่ยม Class หรือ Object ซึ่งภายในกรอบสี่เหลี่ยมจะมีชื่อของ Object หรือ Class ประกอบอยู่ Object : Class กิจกรรมที่เกิดขึ้นจะแทนด้วยลูกศรแนวนอนที่ชี้จาก Class หรือ Object หนึ่งไปยัง Class หรือ Object ต่อไป การระบุชื่อกิจกรรมนั้นจะอยู่ในรูปแบบ [Condition] ฟังก์ชัน ชื่อของกิจกรรมจะต้องเป็นฟังก์ชันที่มีอยู่ใน Class หรือ Object ที่ลูกศรชี้ไป เส้นแสดงเวลาจะแทนด้วยเส้นตรงประแนวตั้ง โดยเวลาจะเดินจากด้านบนลงมาสู่ด้านล่าง นั่นหมายถึงว่า ถ้าหากกิจกรรมที่เกิดขึ้นเกิดอยู่ด้านบนสุดกิจกรรมนั้นเป็นกิจกรรมแรก และกิจกรรมที่อยู่บริเวณด้านล่างมาจะเป็นกิจกรรมที่เกิดต่อจากนั้น

Sequence Diagram เป็นแผนภาพที่ใช้อธิบายการทำงานของ Use Case เพื่อแสดงถึง ขั้นตอนการทำงานและลำดับของการสื่อสาร (Message) ระหว่าง Object ที่ตอบโต้กัน จะแสดงอยู่ในรูปแบบ 2 มิติ โดยเส้นประแนวตั้ง (Lifeline) จะนำเสนอในด้านเวลา ส่วนเส้นแนวนอน (Message) จะนำเสนอเกี่ยวกับการโต้ตอบกันระหว่าง Object หรือ Class ต่างๆ ประโยชน์ Sequence Diagram

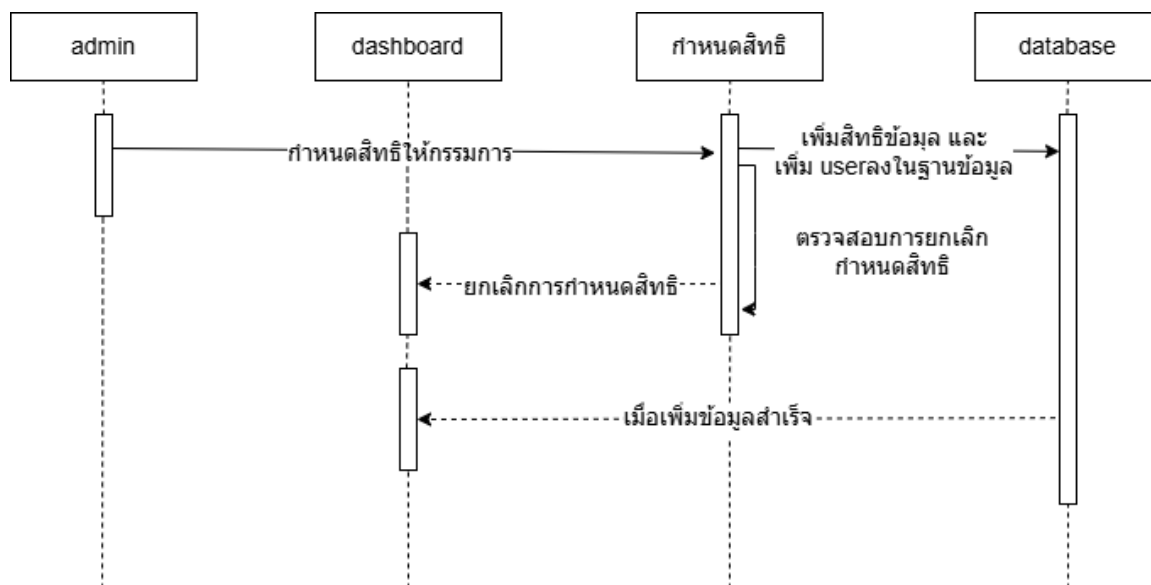
1. ช่วยในการพิจารณาว่าในคลาสไดอะแกรม (Class Diagram) ที่สร้างขึ้นนั้นมีฟังก์ชันใดขาดหายไป หรือควรเพิ่มเติมเข้าไปอีกหรือไม่
2. ทำคลาสดังๆ ที่สร้างขึ้นมีความสมบูรณ์มากยิ่งขึ้น



ภาพที่ 7 ซีควেনซ์ไดอะแกรม (Sequence Diagram) การเข้าสู่ระบบ (Login)

ตารางที่ 12 อธิบายเหตุการณ์ที่เกิดขึ้นใน ซีควেনซ์ไดอะแกรม (Sequence Diagram) การเข้าสู่ระบบ (Login) ระบบสารสนเทศเพื่อการแข่งขันกีฬากีฬา วิทยาลัยการศึกษามหาวิทยาลัยราชภัฏวไลยอลงกรณ์

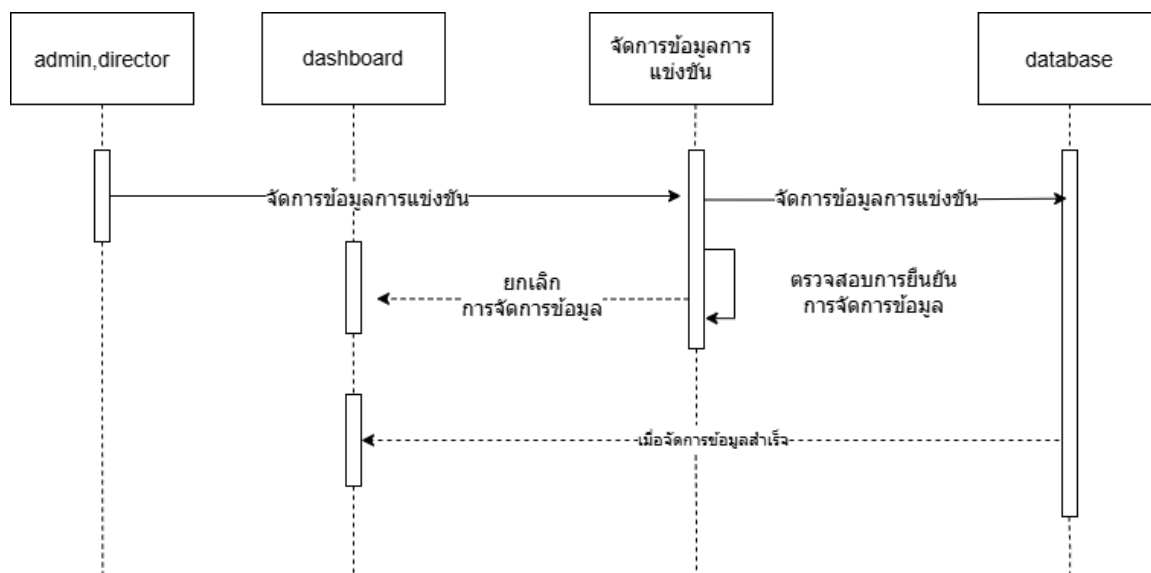
เหตุการณ์ที่เกิดขึ้น	คำอธิบาย
เข้าสู่ระบบ	ผู้ใช้งานระบบกรอก username และ Password
ตรวจสอบการ Login	ตรวจสอบข้อมูลที่ผู้ใช้งานกรอกเข้ามา
กรณี ข้อมูลที่ถูกต้อง	จะเข้าไปที่หน้าหลัก
กรณี ข้อมูลไม่ถูกต้อง	จะเข้าสู่หน้า dashboard



ภาพที่ 8 ซีควนซ์ไดอะแกรม (Sequence Diagram) กำหนดสิทธิ์ผู้สมัคร

ตารางที่ 13 อธิบายเหตุการณ์ที่เกิดขึ้นใน ซีควนซ์ไดอะแกรม (Sequence Diagram) กำหนดสิทธิ์ผู้สมัคร ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรีฑาศึกษา พุทธรักษา

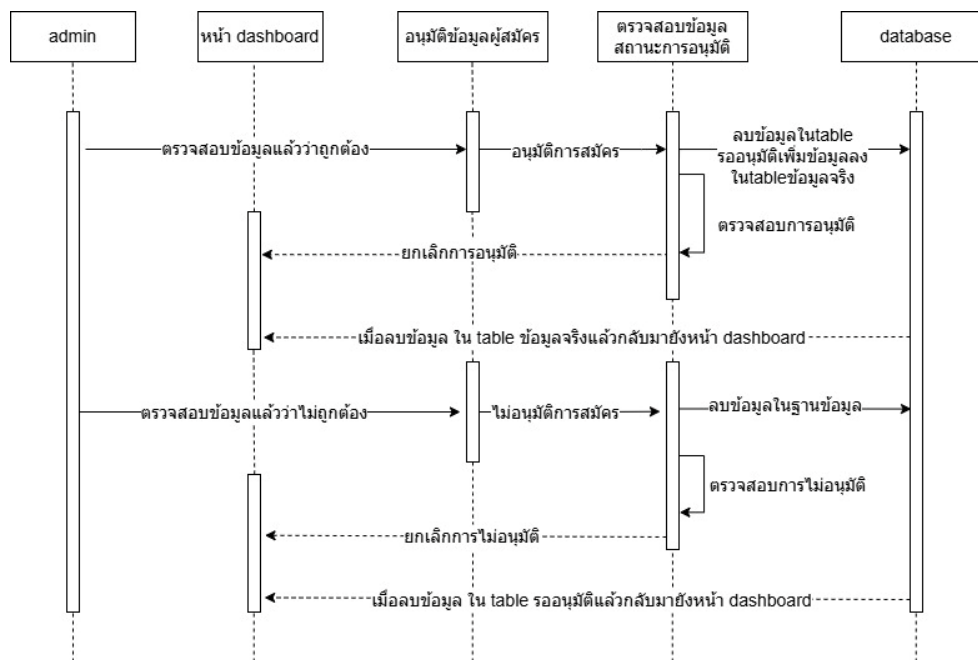
เหตุการณ์ที่เกิดขึ้น	คำอธิบาย
กำหนดสิทธิ์	กำหนดสิทธิ์ให้กรรมการ มีอยู่ 2 กรณี กรณีแรก เลือกเพิ่มสิทธิ์ให้กรรมการสามารถ กำหนดสิทธิ์การเข้าถึงข้อมูล และเพิ่ม username และ password ให้กรรมการลงในฐานข้อมูลสามารถ login เพื่อแก้ไขข้อมูลตามขอบเขตได้กรณีที่2 ยกเลิกกำหนดสิทธิ์จะกลับไปยังหน้า dashboard
ฐานข้อมูล	เมื่อเพิ่มข้อมูล username password ของการกำหนดสิทธิ์เสร็จจะกลับยังหน้า dashboard



ภาพที่ 9 ซีควนซ์ไดอะแกรม (Sequence Diagram) จัดการข้อมูลการแข่งขัน

ตารางที่ 14 อธิบายเหตุการณ์ที่เกิดขึ้นใน ซีควนซ์ไดอะแกรม (Sequence Diagram) จัดการข้อมูลการแข่งขัน ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรณีศึกษา พุทธรักษา

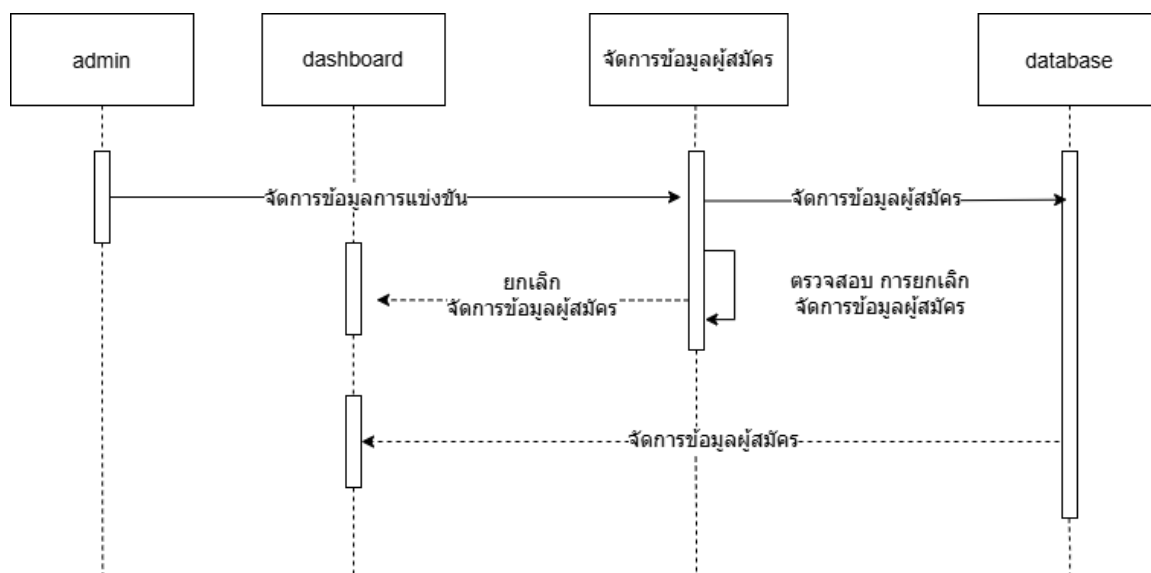
เหตุการณ์ที่เกิดขึ้น	คำอธิบาย
จัดการข้อมูลการแข่งขัน	มี 2 กรณีเมื่อต้องการจัดการข้อมูล จะจัดการข้อมูลในฐานข้อมูล อีกกรณีเมื่อ ยกเลิกการจัดการข้อมูลการแข่งขัน จะกลับไปยังหน้า dashboard
ฐานข้อมูล	เมื่อจัดการข้อมูลสำเร็จจะกลับไปยังหน้า dashboard



ภาพที่ 10 ซีควเอนซ์ไดอะแกรม (Sequence Diagram) อนุมัติข้อมูลผู้สมัคร

ตารางที่ 15 อธิบายเหตุการณ์ที่เกิดขึ้นใน ซีควเอนซ์ไดอะแกรม (Sequence Diagram) อนุมัติข้อมูลผู้สมัคร ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรีฑาศึกษา พุทธรักษา

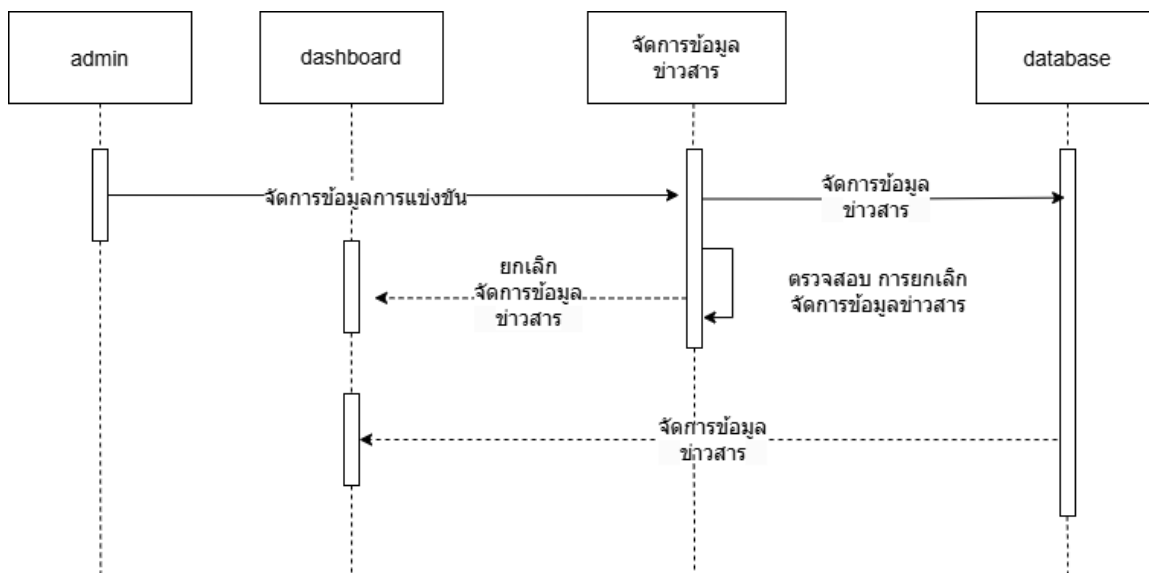
เหตุการณ์ที่เกิดขึ้น	คำอธิบาย
อนุมัติข้อมูลสมัคร	มีอยู่ 2 กรณี คือเราอนุมัติการสมัครกับ ไม่อนุมัติการสมัคร
ตรวจสอบข้อมูลสถานะการอนุมัติ	ในกรณีที่เรา อนุมัติการสมัคร เราสามารถเลือกได้ว่า จะอนุมัติใหม่หรือยกเลิกกรณีอนุมัติเราจะเพิ่มข้อมูลลงในtable ข้อมูลจริง แล้วลบข้อมูล ในtable รออนุมัติ กรณีที่เรา ยกเลิกการอนุมัติจะกลับไปยังหน้าdashboard กรณี เราไม่อนุมัติมีอยู่2กรณี ถ้าเราไม่อนุมัติจะลบข้อมูลในฐานข้อมูลรออนุมัติ ส่วนถ้าเรายกเลิกการอนุมัติจะกลับไปหน้าdashboard
ฐานข้อมูล	มี 2 กรณีกรณีแรกอนุมัติเมื่อเพิ่มข้อมูลลงฐานข้อมูลจริงจะกลับไปหน้าdashboard อีกกรณีคือไม่อนุมัติลบข้อมูล เมื่อลบข้อมูลเสร็จก็ไปยัง หน้า dashboard



ภาพที่ 11 ซีควอนซ์ไดอะแกรม (Sequence Diagram) จัดการข้อมูลผู้สมัคร

ตารางที่ 16 อธิบายเหตุการณ์ที่เกิดขึ้นใน ซีควอนซ์ไดอะแกรม (Sequence Diagram) จัดการข้อมูลผู้สมัคร ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา ករណីศึกษา พุทธรักษา

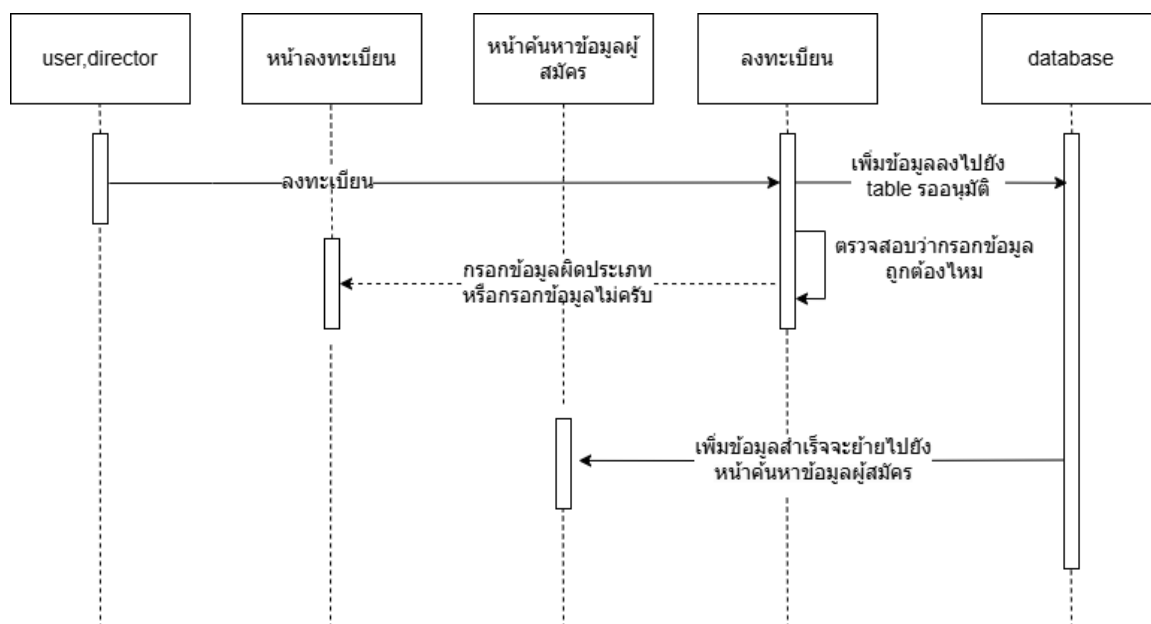
เหตุการณ์ที่เกิดขึ้น	คำอธิบาย
จัดการข้อมูลผู้สมัคร	มี 2 กรณีเมื่อต้องการจัดการข้อมูล จะจัดการข้อมูลผู้สมัครในฐานข้อมูล อีกกรณีเมื่อ ยกเลิกการจัดการข้อมูลผู้สมัคร จะกลับไปยังหน้า dashboard
ฐานข้อมูล	เมื่อจัดการข้อมูลผู้สมัครสำเร็จจะกลับไปยังหน้า dashboard



ภาพที่ 12 ซีควেনซ์ไดอะแกรม (Sequence Diagram) จัดการข้อมูลข่าวสาร

ตารางที่ 17 อธิบายเหตุการณ์ที่เกิดขึ้นใน ซีควেনซ์ไดอะแกรม (Sequence Diagram) จัดการข้อมูลข่าวสาร ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา วิทยาลัยการศึกษ พุทธจักร

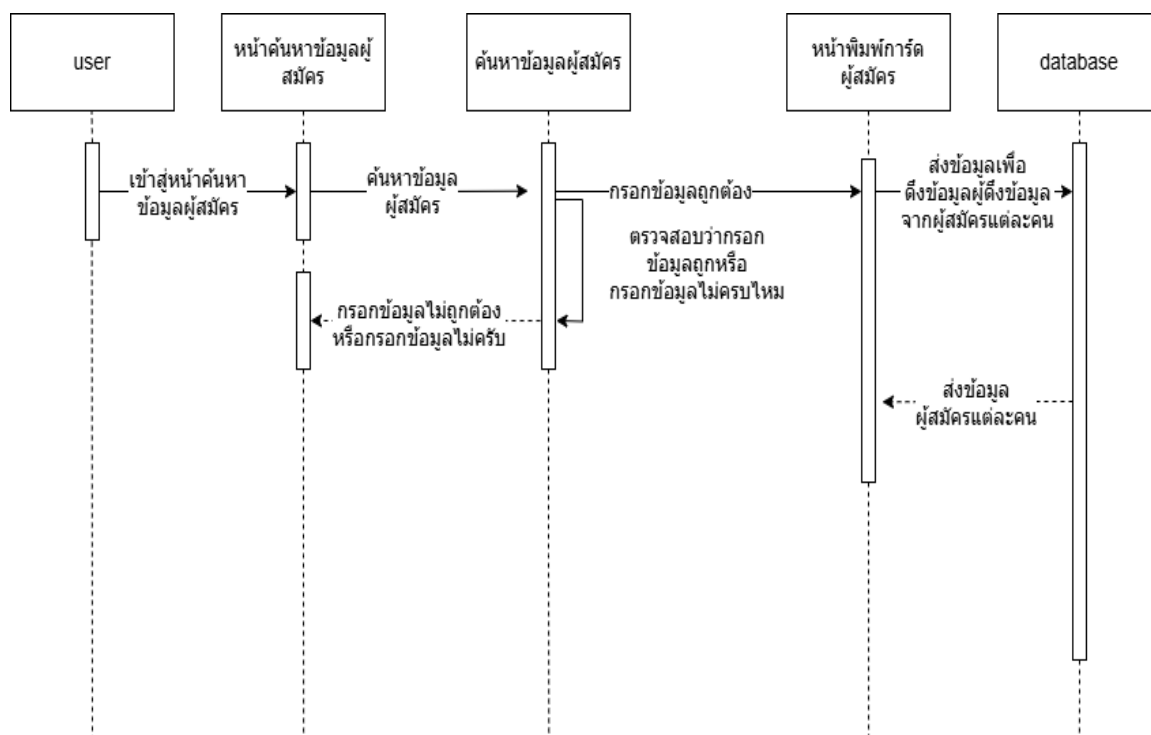
เหตุการณ์ที่เกิดขึ้น	คำอธิบาย
จัดการข้อมูลผู้สมัคร	มี 2 กรณีเมื่อต้องการจัดการข้อมูล จะจัดการข้อมูลข่าวสารในฐานะข้อมูล อีกกรณีเมื่อ ยกเลิกการจัดการข้อมูลข่าวสาร จะกลับไปยังหน้า dashboard
ฐานข้อมูล	เมื่อจัดการข้อมูลข่าวสารสำเร็จจะกลับไปยังหน้า dashboard



ภาพที่ 13 ซีควนซ์ไดอะแกรม (Sequence Diagram) ลงทะเบียน

ตารางที่ 18 อธิบายเหตุการณ์ที่เกิดขึ้นใน ซีควนซ์ไดอะแกรม (Sequence Diagram) ลงทะเบียนระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรณีศึกษา พุทธรักษา

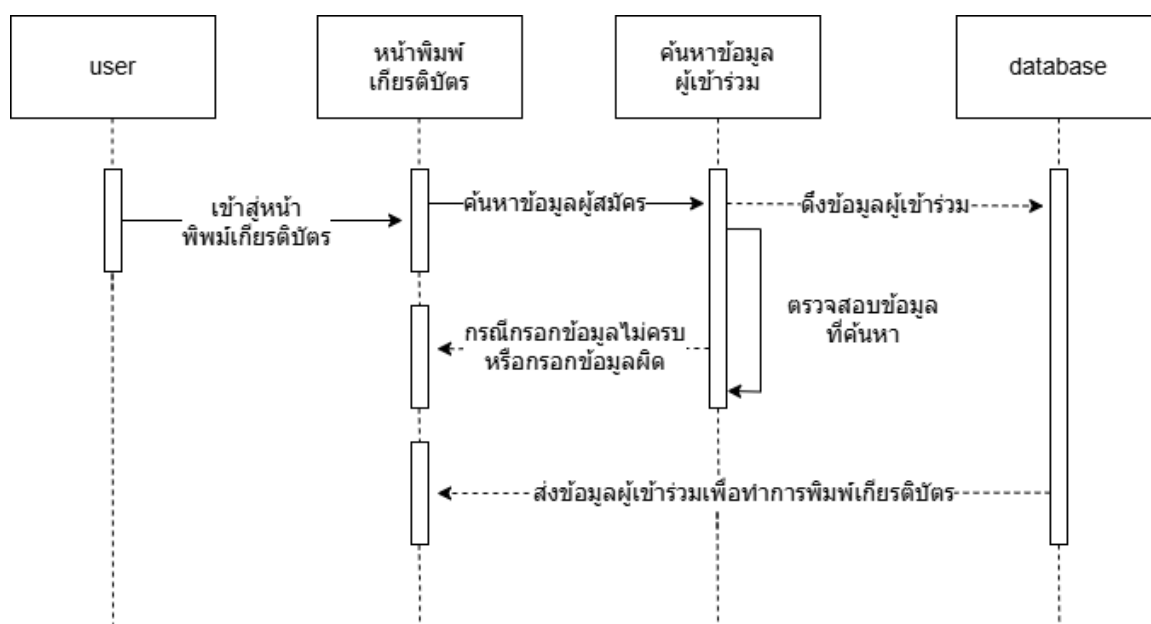
เหตุการณ์ที่เกิดขึ้น	คำอธิบาย
จัดการข้อมูลผู้สมัคร	มี 2 กรณีเมื่อลงทะเบียนกรอกข้อมูลถูกประเภท ไม่มีข้อมูลว่างจะเพิ่มข้อมูลลงไปยัง table รอนุมัติ อีกกรณีเมื่อ กรอกข้อมูลไม่ครบหรือผิดประเภทจะขึ้นแจ้งเตือนความผิดพลาดจะยังคงอยู่หน้าลงทะเบียน
ฐานข้อมูล	กรณีลงทะเบียนเพิ่มข้อมูลลงใน table รอนุมัติสำเร็จจะย้ายไปยังหน้าค้นหาข้อมูลผู้สมัคร



ภาพที่ 14 ซีควেনซ์ไดอะแกรม (Sequence Diagram) พิมพ์บัตรประจำตัว

ตารางที่ 19 อธิบายเหตุการณ์ที่เกิดขึ้นใน ซีควেনซ์ไดอะแกรม (Sequence Diagram) พิมพ์บัตรประจำตัว ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรีฑา กีฬา พุทธรักษา

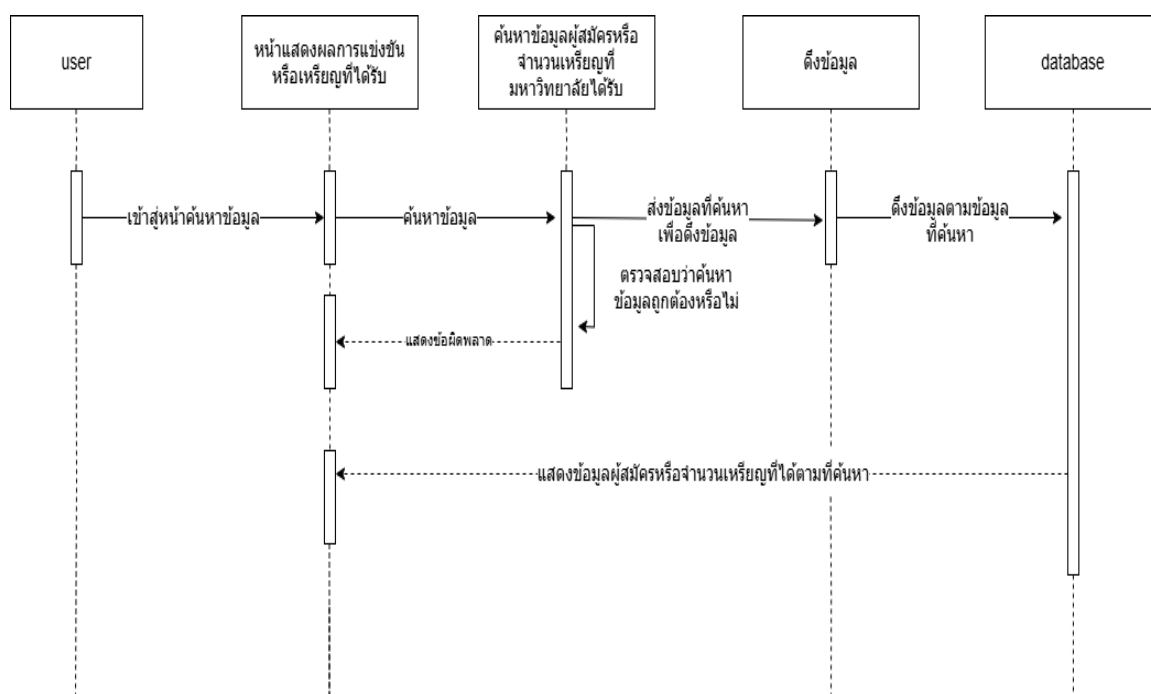
เหตุการณ์ที่เกิดขึ้น	คำอธิบาย
ค้นหาข้อมูล	มี 2 กรณีเมื่อค้นหาข้อมูลถูกต้องจะสามารถกดปุ่มเพื่อไปยังหน้าพิมพ์การ์ดได้กรณีข้อมูลไม่ถูกต้องหรือกรอกข้อมูลไม่ครบจะกลับไปยังหน้าค้นหาข้อมูลผู้สมัคร
ดึงข้อมูลผู้สมัคร	เมื่อไปยังหน้าพิมพ์การ์ดผู้สมัครจะดึงข้อมูลตรงกับข้อมูลที่เราค้นหาใช้งานสามารถพิมพ์ใบสมัครได้
ฐานข้อมูล	จะส่งข้อมูลผู้สมัครตามข้อมูลที่เราค้นหาไปยังหน้าพิมพ์การ์ดผู้สมัคร



ภาพที่ 15 ซีควเรนซ์ไดอะแกรม (Sequence Diagram) พิมพ์เกียรติบัตร

ตารางที่ 20 อธิบายเหตุการณ์ที่เกิดขึ้นใน ซีควเรนซ์ไดอะแกรม (Sequence Diagram) พิมพ์เกียรติบัตร ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา การศึกษา พุทธศึกษา

เหตุการณ์ที่เกิดขึ้น	คำอธิบาย
ค้นหาข้อมูลผู้เข้าร่วม	มี 2 กรณีเมื่อค้นหาข้อมูลผู้สมัครถูกต้องจะดึงข้อมูลจากฐานข้อมูลอีกกรณีเมื่อค้นหาข้อมูลแล้วกรอกข้อมูลไม่ครบหรือผิดพลาดจะแจ้งเตือนข้อผิดพลาดและกลับไปยังหน้าพิมพ์เกียรติบัตร
ดึงข้อมูลผู้เข้าร่วม	เมื่อค้นหาข้อมูลถูกต้องจะดึงข้อมูลจากฐานข้อมูลตามที่เราค้นหา
ฐานข้อมูล	จะส่งข้อมูลผู้สมัครตามข้อมูลผู้สมัครเพื่อนำไปพิมพ์เพื่อยึดตามที่ข้อมูลเราค้นหา



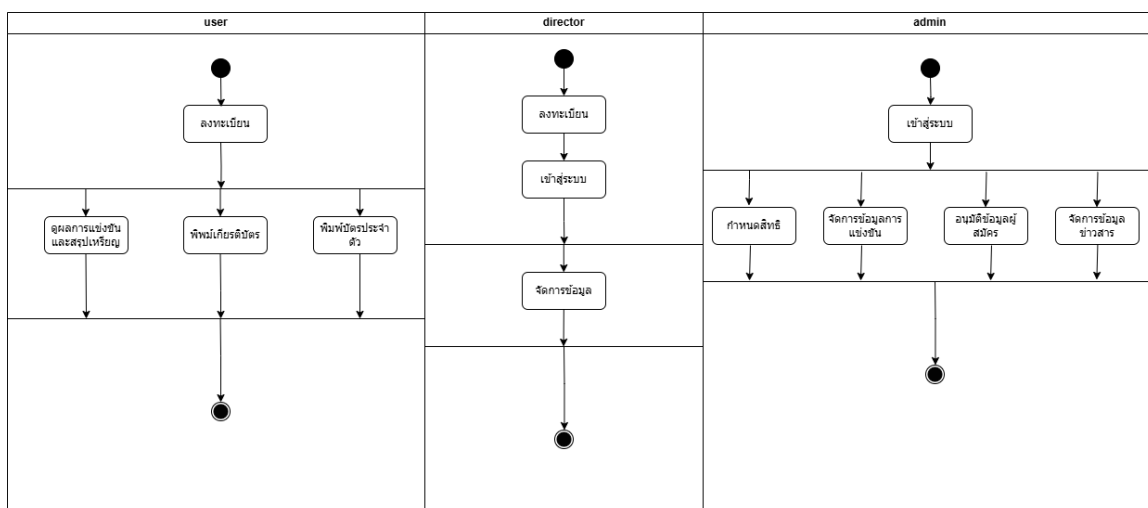
ภาพที่ 16 ซีควেনซ์ไดอะแกรม (Sequence Diagram) ดูผลการแข่งขันและผลสรุปเหรียญ
 ตารางที่ 21 อธิบายเหตุการณ์ที่เกิดขึ้นใน ซีควেনซ์ไดอะแกรม (Sequence Diagram) ดูผลการ
 แข่งขันและผลสรุปเหรียญ ระบบสารสนเทศเพื่อการจัดแข่งขันกีฬา กรณีศึกษา พุทธรักษา

เหตุการณ์ที่เกิดขึ้น	คำอธิบาย
ค้นหาข้อมูลผู้สมัครหรือจำนวนเหรียญที่มหาวิทยาลัยได้รับ	มี 2 กรณีเมื่อค้นหาข้อมูลถูกต้องจะส่งข้อมูลเพื่อไปดึงข้อมูลตามสิ่งที่ค้นหาอีกกรณีเมื่อค้นหาไม่ถูกต้องจะแสดงข้อผิดพลาด
ดึงข้อมูล	เมื่อค้นหาข้อมูลถูกต้องจะดึงข้อมูลจากฐานข้อมูลตามที่เราค้นหา
ฐานข้อมูล	จะส่งข้อมูลผู้สมัครหรือจำนวนเหรียญที่มหาวิทยาลัยได้รับตามที่เราค้นหา

3.1.3 แอคทิวิตีไดอะแกรม (Activity Diagram)

เป็นแผนภาพที่ใช้ที่แสดงขั้นตอนการทำงานของ Use Case (เช่นเดียวกับ Sequence Diagram และ Collaboration Diagram) แต่จะเน้นไปที่งานย่อยของวัตถุ โดยจะมีกระบวนการทำงานคล้ายกับ Flowchart

Activity Diagram บางครั้งมีลักษณะคล้าย Swim lane โดยจะแบ่งกลุ่มกิจกรรมที่เกิดขึ้นเป็นช่อง โดยกำกับแต่ละช่องด้วยชื่อของ Object แต่ละ Swim lane แสดงถึงกิจกรรมที่เกิดขึ้นกับ Object นั้นๆ Activity Diagram แสดงลำดับกิจกรรมของการทำงาน (Work Flow) สามารถแสดงทางเลือกที่เกิดขึ้นได้ Activity Diagram จะแสดงขั้นตอนการทำงานในการปฏิบัติการ โดยประกอบไปด้วยสถานะต่างๆที่เกิดขึ้นระหว่างการทำงาน และผลจากการทำงานขั้นตอนต่างๆ



ภาพที่ 17 แอคทิวิตีไดอะแกรม (Activity Diagram) ระบบสารสนเทศเพื่อการจัดการแข่งขันกีฬา
พุทธรักษาเกมส์

3.2 การออกแบบหน้าจอ