

ทก.01

## แบบเสนอโครงการพิเศษ (ปริญญานิพนธ์)

สาขาวิชาวิศวกรรมสารสนเทศและเครือข่าย ภาควิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีและการจัดการอุตสาหกรรม

---

### 1. ข้อมูลขั้นต้นของโครงการ

#### 1.1 ชื่อโครงการ

(ภาษาไทย)    สมาร์ทฟิต รูทีน:

(ภาษาอังกฤษ) :

#### 1.2 ชื่อนักศึกษาผู้ทำโครงการ

1) นางสาวลลนา สุขรักษ์

#### 1.3 ชื่ออาจารย์ที่ปรึกษา

1) อาจารย์สุภาพร ชิมเจริญ

### 2. รายละเอียดของโครงการ

#### 2.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันผู้คนหันมาใส่ใจสุขภาพและการออกกำลังกายมากขึ้น เนื่องจากวิถีชีวิตที่เร่งรีบ และการทำงานที่ต้องนั่งอยู่หน้าจคอมพิวเตอร์เป็นเวลานาน ทำให้การเคลื่อนไหวของร่างกายลดลงส่งผลให้เกิดปัญหาสุขภาพ เช่น โรคอ้วน โรคหัวใจ และปัญหาทางจิตใจต่าง ๆ การออกกำลังกายเป็นวิธีการหนึ่งที่สามารถช่วยลดความเสี่ยงต่อการเกิดโรคเหล่านี้ได้ อีกทั้งยังช่วยเสริมสร้างสมรรถภาพร่างกายและจิตใจ แต่หลายคนยังขาดความรู้และความเข้าใจในการวางแผนการออกกำลังกายที่เหมาะสมสำหรับตัวเอง ซึ่งทำให้เกิดการออกกำลังกายที่ไม่ต่อเนื่องหรือไม่มีประสิทธิภาพ

ด้วยเหตุนี้ การสร้างแอปพลิเคชันที่สามารถช่วยให้ผู้ใช้งานสร้างแผนการออกกำลังกายรายสัปดาห์ที่เหมาะสมกับเป้าหมายของตนเองจึงเป็นเรื่องที่มีความสำคัญ "Simple Fitness Routine Planner" เป็นเว็บแอปพลิเคชันที่ถูกพัฒนาขึ้นเพื่อให้ผู้ใช้งานสามารถออกแบบแผนการออกกำลังกายของตนเองได้อย่างง่ายดาย แอปพลิเคชันนี้จะมีระบบการแนะนำท่าออกกำลังกายต่าง ๆ ซึ่งจะถูกจัดกลุ่มตามประเภทของการออกกำลังกาย เช่น การฝึกความแข็งแรง การฝึกคาร์ดิโอ หรือการยืดกล้ามเนื้อ โดยทุกท่าจะมีคำอธิบายที่เข้าใจง่ายและวิดีโอสาธิตที่ช่วยให้ผู้ใช้สามารถทำตามได้อย่างถูกต้อง

นอกจากนี้ แอปพลิเคชันนี้ยังสามารถแนะนำท่าออกกำลังกายให้กับผู้ใช้ตามความต้องการส่วนบุคคล โดยระบบจะดึงข้อมูลจากฐานข้อมูลที่ถูกจัดทำอย่างละเอียด ฐานข้อมูลนี้ประกอบด้วยข้อมูลสำคัญ เช่น ชื่อท่าออกกำลังกาย คำอธิบาย วิดีโอสาธิต และข้อมูลทางเทคนิคที่เกี่ยวข้อง ส่วนการออกแบบ UI จะเน้นให้ใช้งานง่ายและเข้าถึงได้สะดวก เพื่อให้ผู้ใช้ทุกคนสามารถสร้างและปรับแต่งแผนการออกกำลังกายได้ตามต้องการอย่างง่ายดาย

## 2.2 วัตถุประสงค์ของการจัดทำโครงการพิเศษ

2.2.1 เพื่อพัฒนาเว็บแอปพลิเคชันที่ช่วยให้ผู้ใช้งานสร้างแผนการออกกำลังกายรายสัปดาห์ได้อย่างง่ายดายและเหมาะสมกับเป้าหมายของตนเอง

2.2.2 เพื่อให้คำแนะนำท่าออกกำลังกายที่เหมาะสมกับความต้องการของผู้ใช้ โดยอ้างอิงจากฐานข้อมูลที่มีรายละเอียดครบถ้วน

2.2.3 เพื่อออกแบบอินเตอร์เฟซที่ใช้งานง่ายและสะดวกต่อการเข้าถึงข้อมูลและการจัดการแผนการออกกำลังกายของผู้ใช้

## 2.3 ขอบเขตของการทำโครงการพิเศษ ( Scope of Special Project )

2.3.1 การพัฒนาเว็บแอปพลิเคชันสมาร์ตฟิต รูทีน ผู้ใช้สามารถลงทะเบียนเพื่อสร้างบัญชีผู้ใช้ใหม่ โดยกรอกข้อมูลส่วนตัว

2.3.1.1 ชื่อ-สกุล

2.3.1.2 อีเมล

2.3.1.3 ชื่อผู้ใช้ (Username)

2.3.1.4 รหัสผ่าน (Password)

หลังออกทะเบียน ผู้ใช้งานสามารถลงชื่อเข้าใช้งานระบบและยังสามารถเข้าถึงฟีเจอร์ต่าง ๆ ภายในเว็บแอปพลิเคชันได้

2.3.2 การสร้างและจัดการกับแผนการการออกกำลังกายจะถูกแบ่งออกเป็น 2 ข้อ ได้แก่

2.3.2.1 ผู้ใช้สามารถสร้างแผนการออกกำลังกายส่วนตัวโดยเลือกท่าทางการออกกำลังกายจากฐานข้อมูล ซึ่งแต่ละท่าจะมีรายละเอียด ได้แก่

2.3.2.1.1 ชื่อท่า

2.3.2.1.2 คำอธิบาย

2.3.2.1.3 วิดีโอสาธิต

2.3.2.2 ผู้ใช้สามารถบันทึกแผนการออกกำลังกายที่ตนเองสร้างขึ้นไว้ในระบบ และสามารถเรียกดูหรือปรับแก้แผนได้ตามต้องการ

2.3.3 ระบบสามารถแนะนำการออกกำลังกายที่เข้ากับเป้าหมายของผู้ใช้งานได้

2.3.4 เว็บแอปพลิเคชันจะมีการแจ้งเตือนผู้ใช้งานเกี่ยวกับการออกกำลังกาย หรือแจ้งเตือนเกี่ยวกับคำแนะนำใหม่ ๆ ที่เหมาะสมกับผู้ใช้

2.3.5 ฐานข้อมูลจะถูกจัดเตรียมเพื่อจัดเก็บข้อมูลสำคัญ ได้แก่

2.3.5.1 ข้อมูลผู้ใช้ (User Information)

2.3.5.2 ข้อมูลทำการออกกำลังกาย (Exercise Details)

2.3.5.3 ข้อมูลการติดตามความก้าวหน้าของผู้ใช้ (Progress Tracking)

## 2.4 รายละเอียดทฤษฎีที่ใช้ในการจัดทำปริญญานิพนธ์

### 2.4.1 ทฤษฎีที่เกี่ยวกับ Web Application

1 Client-Server Architectures เป็นรูปแบบของการออกแบบระบบคอมพิวเตอร์ที่แบ่งหน้าที่และภาระงานระหว่างคอมพิวเตอร์ที่เรียกว่าไคลเอนท์ (Clients) และคอมพิวเตอร์ที่เรียกว่าเซิร์ฟเวอร์ (Servers) เพื่อให้การทำงานและการประมวลผลเกิดขึ้นอย่างมีประสิทธิภาพและมีความสมดุลย์ สิ่งที่ทำให้แบบไคลเอนท์เซิร์ฟเวอร์มีประโยชน์หลัก 4 ประการได้แก่:

1.1 สามารถปรับเพิ่มลดได้ (Scalable): แบบไคลเอนท์เซิร์ฟเวอร์อนุญาตให้เพิ่มหรือลดทรัพยากรที่ใช้งาน เช่น พื้นที่จัดเก็บข้อมูลและความสามารถในการประมวลผล เพื่อให้รองรับการเติบโตของผู้ใช้และการให้บริการที่มากขึ้น

1.2 สนับสนุนความหลากหลายรูปแบบ: แบบไคลเอนท์เซิร์ฟเวอร์สามารถรองรับการทำงานร่วมกันของคอมพิวเตอร์ที่มีระบบปฏิบัติการและสมบัติที่แตกต่างกัน ทำให้ผู้ใช้สามารถเลือกใช้งานอุปกรณ์ต่างๆ ได้ตามความต้องการ

1.3 การแยกแยะและการปรับปรุงง่าย: แบบไคลเอนท์เซิร์ฟเวอร์ช่วยแยกแยะการแสดงผลการประมวลผล และการเข้าถึงข้อมูล ทำให้การออกแบบและการปรับปรุงระบบง่ายขึ้น และไม่กระทบต่อส่วนอื่นๆ

1.4 ความเสถียรและความยืดหยุ่นในการเปลี่ยนแปลง: แบบไคลเอนท์เซิร์ฟเวอร์ช่วยลดความเสี่ยงในกรณีเซิร์ฟเวอร์ล่ม และเมื่อเซิร์ฟเวอร์ต้องการปรับปรุงหรือเปลี่ยนแปลง สามารถทำได้โดยไม่ต้องกระทบต่อไคลเอนท์

แต่สถาปัตยกรรมแบบไคลเอนท์เซิร์ฟเวอร์ก็ยังมีข้อจำกัด เช่น ความซับซ้อนในการพัฒนาและบำรุงรักษาซอฟต์แวร์สองส่วน การจัดการการเชื่อมต่อระหว่างไคลเอนท์และเซิร์ฟเวอร์ และการรับมือกับปัญหาความไม่เสถียรของเครือข่ายหรือเซิร์ฟเวอร์

2 HTTP (Hypertext Transfer Protocol) คือโปรโตคอลการสื่อสารระหว่างคอมพิวเตอร์ที่ใช้กันบนเว็บไซต์และบนอินเทอร์เน็ตเพื่อแลกเปลี่ยนข้อมูลและแสดงผลในรูปแบบของเอกสารเรียกว่า "เซิร์ฟเล็ต" หรือ "เว็บเพจ" ซึ่งเป็นข้อมูลที่สามารถประมวลผลและแสดงผลในเว็บเบราว์เซอร์ได้ โปรโตคอลนี้ถูกใช้งานอย่างแพร่หลายเพื่อการรับส่งข้อมูลระหว่างเว็บเซิร์ฟเวอร์และเว็บไคลเอนท์ (เว็บเบราว์เซอร์หรือแอปพลิเคชันบนอุปกรณ์ผู้ใช้งาน) ซึ่งช่วยให้เนื้อหาและข้อมูลบนเว็บเป็นไปอย่างรวดเร็วและมีความเป็นระบบ

HTTP ถูกออกแบบมาให้ใช้งานร่วมกับรูปแบบของเอกสารที่เรียกว่า HTML (Hypertext Markup Language) ซึ่งเป็นรูปแบบการระบุโครงสร้างและเนื้อหาของเว็บเพจ ด้วยการใช้ HTTP เราสามารถเรียกดูเว็บเพจต่างๆ ดาวน์โหลด์รูปภาพ ส่งคำขอ (Request) เพื่อรับข้อมูลจากเซิร์ฟเวอร์ และรับข้อมูลที่เซิร์ฟเวอร์ส่งกลับมาให้เราได้ผ่านเว็บเบราว์เซอร์หรือแอปพลิเคชันบนอุปกรณ์ของเรา

HTTP มีหลายเวอร์ชันที่พัฒนามาตามความต้องการของเทคโนโลยีและการใช้งาน เวอร์ชันที่ได้รับความนิยมมากในปัจจุบันคือ HTTP/1.1 และเวอร์ชันใหม่ๆ อย่าง HTTP/2 และ HTTP/3 ก็ได้ถูกพัฒนาขึ้นเพื่อเพิ่มประสิทธิภาพในการรับส่งข้อมูลและเพิ่มความปลอดภัยในการสื่อสารบนเว็บไซต์และแอปพลิเคชันออนไลน์

3 REST (Representational State Transfer) เป็นแนวคิดทางสถาปัตยกรรมในการสื่อสารระหว่างระบบคอมพิวเตอร์บนเว็บ ที่มีจุดประสงค์เพื่อให้ระบบสามารถสื่อสารกันได้อย่างมีประสิทธิภาพและเป็นระเบียบ โดย REST มีลักษณะที่เน้นความสำคัญในการแยกแยะระหว่างฝั่งลูกค้าและเซิร์ฟเวอร์ พร้อมกำหนดหลักการและข้อจำกัดที่ช่วยให้การสื่อสารเกิดขึ้นได้อย่างมีประสิทธิภาพ โดยสรุปคุณสมบัติและการทำงานของ REST การสื่อสารระหว่างลูกค้าและเซิร์ฟเวอร์ได้ดังนี้:

3.1 การแยกแยะฝั่งลูกค้าและเซิร์ฟเวอร์: ใน REST ระบบที่ออกแบบแบบ RESTful ทำให้การพัฒนาและแก้ไขโค้ดในฝั่งลูกค้าและเซิร์ฟเวอร์สามารถทำได้อิสระต่อกัน ไม่จำเป็นต้องมีความรู้หรือการร่วมมือกันมาก โดยระบบสามารถทำงานได้โดยไม่รู้จักความสัมพันธ์หรือสถานะกัน

3.2 ความเสถียรและประสิทธิภาพ: การทำงานแบบ stateless ทำให้ระบบ REST ไม่จำเป็นต้องเก็บข้อมูลเกี่ยวกับสถานะของลูกค้าหรือเซิร์ฟเวอร์ ซึ่งทำให้สามารถกระจายการทำงานและการขยายขนาดระบบได้ง่ายขึ้น ระบบสามารถรับและส่งข้อมูลได้โดยไม่ต้องรู้ถึงข้อมูลก่อนหน้านี้

3.3 การใช้เนรมิตทรัพยากร: REST ใช้รูปแบบของทรัพยากร (Resource) ที่เป็นนามธรรมของสิ่งต่าง ๆ ที่เราต้องการจัดเก็บหรือส่งในระบบ เช่น ข้อมูล, ไฟล์, ภาพ และอื่น ๆ โดยสิ่งนี้ช่วยให้ระบบเป็นระเบียบและเป็นลำดับ

4 การใช้ HTTP Verbs ในการกระทำ: REST ใช้เมธอด (HTTP Verbs) เพื่อให้กำหนดการกระทำต่าง ๆ กับทรัพยากร เช่น

4.1 GET: ดึงข้อมูลทรัพยากรหรือคอลเลกชันของทรัพยากร

4.2 POST: สร้างทรัพยากรใหม่

4.3 PUT: อัปเดตข้อมูลทรัพยากรเฉพาะ

4.4 DELETE: ลบทรัพยากรที่ระบุ

5 การใช้ Headers และพารามิเตอร์: REST ใช้ Headers เพื่อระบุประเภทของเนื้อหาที่ลูกค้าต้องการรับจากเซิร์ฟเวอร์ และใช้พารามิเตอร์เพื่อระบุรายละเอียดเพิ่มเติมในการทำงาน เช่น ระบุ ID ของทรัพยากรที่ต้องการดำเนินการ

6 การส่งคำตอบและรหัสสถานะ: เซิร์ฟเวอร์ส่งข้อมูลกลับไปยังลูกค้าในรูปแบบข้อมูลที่กำหนดใน Headers ด้วย Content-Type ที่ระบุว่าข้อมูลมีรูปแบบอะไร เซิร์ฟเวอร์ยังส่งรหัสสถานะเพื่อแสดงผลสถานะการดำเนินการ เช่น 200 (OK) หรือ 404 (NOT FOUND)

2.4.2 รายงานการค้นคว้า การศึกษา หรือการวิจัยที่เกี่ยวข้อง \*\*\*\* สอง - สาม งาน(กำลังดูและเลือก รายงานที่เหมาะสมกับงานมากที่สุด)

## 2.5 วิธีการดำเนินงานจัดทำโครงการพิเศษ

ภาคการศึกษาที่ 1/2566

ภาคการศึกษาที่ 2/2566

## 2.6 แผนกิจกรรมและตารางเวลาในการจัดทำ

2.6.1 แผนกิจกรรมหลักและระยะเวลา

	ภาคการศึกษาที่ 1
--	------------------



## 2.7 ทรัพยากรที่ต้องใช้ในการจัดทำโครงการพิเศษ

### 2.7.1 เครื่องมือในการจัดทำโครงการพิเศษ

#### 2.7.1.1 Software

##### 2.7.1.1.1 MySQL

##### 2.7.1.1.2 NLP

##### 2.7.1.1.3 Python

##### 2.7.1.1.4 Power BI

#### 2.7.1.2 Hardware

##### 2.7.1.2.1 คอมพิวเตอร์/โน้ตบุ๊ก

##### 2.7.1.2.2 โทรศัพท์

### 2.7.2 งบประมาณที่ใช้ในการจัดทำ

- ค่าจัดทำปฏิญญานิพนธ์	1,000 บาท
- ค่าใช้จ่ายอื่น ๆ	500 บาท
- รวมเป็นเงิน	1,500 บาท

## 2.8 ผลที่คาดว่าจะได้รับ **ปรับตามวัตถุประสงค์**

2.8.1 ผู้ใช้สามารถหาเมนูอาหารที่ตรงตามความต้องการได้

2.8.2 Chatbot สามารถที่จะตอบโต้และแนะนำเมนูอาหารได้

2.8.3 ผู้ใช้ได้รับคำแนะนำอย่างเหมาะสมในด้านการรับประทานเมนูอาหาร

## 2.9 เอกสารอ้างอิง \*\* ใช้ Zotero เลือกเป็น APA7th

## 2.10 ภาคผนวก \*\* ใช้ figma หรือ wik

ลงชื่อ.....ผู้เสนอโครงการ

(นายพรเทพ )

ลงชื่อ.....ผู้เสนอโครงการ

(นางสาวลลนา สุขรักษ์)

วันที่ยื่นเสนอโครงการ...../...../.....

ค ว า ม เ ห้ น อ า จ า ร ย ์ ท ี่ ป ร ี ก ษ า โ ค ร ง ง าน

.....  
.....  
.....

ลงชื่อ.....อาจารย์ที่ปรึกษา

(.....)

วันที่...../...../.....

สาขาวิชา / ภาควิชาที่ได้รับแบบเสนอโครงการวันที่ .....

ผ ล ก า ร พ ิ จ า ร ญ าน

.....  
.....  
.....

ลงชื่อ.....ประธาน

(.....)

วันที่...../...../.....

ลงชื่อ.....กรรมการ

(.....)

วันที่...../...../.....

ลงชื่อ.....กรรมการ

(.....)

วันที่...../...../.....

ลงชื่อ.....กรรมการ

(.....)

วันที่...../...../.....