

## Hatchy Time เว็บช่วยจัดการเวลาอ่านหนังสือด้วยระบบตรวจสอบใบหน้า

นายธนดล บัณฑิตานุสรณ์ ,นายทยากร ทวีแก้ว ,นางสาวภัทรินทร์ มาระสาร และ อาจารย์จารุวรรณ สุระเสียง\*

สาขาวิชาวิทยาการและเทคโนโลยีดิจิทัล คณะวิทยาศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา

Email: thanadol.b@ku.th, [thayagorn.t@ku.th](mailto:thayagorn.t@ku.th), phattharin.m@ku.th, [jaruwan.sur@ku.th](mailto:jaruwan.sur@ku.th)\*

### บทคัดย่อ

การบริหารเวลาและการรักษาสมาธิเป็นปัจจัยสำคัญต่อประสิทธิภาพในการเรียนรู้ของนักศึกษา เทคนิค Pomodoro เป็นวิธีที่ได้รับความนิยมในการจัดสรรเวลาอ่านหนังสือ โดยใช้หลักการแบ่งช่วงเวลาอ่านและพักอย่างสมดุล อย่างไรก็ตามเครื่องมือ Pomodoro Timer ส่วนใหญ่ไม่สามารถตรวจสอบได้ว่าผู้ใช้มีสมาธิจริงในระหว่างช่วงเวลา Focus หรือไม่ ทำให้ข้อมูลเวลาการเรียนรู้ที่บันทึกไว้อาจไม่สะท้อนประสิทธิภาพที่แท้จริง

โครงการนี้จึงมีวัตถุประสงค์ในการพัฒนาเว็บแอปพลิเคชัน Hatchy Time ที่ผสมผสานเทคโนโลยี การตรวจจับใบหน้าแบบเรียลไทม์ (Real-time Face Detection) และ การประเมินมุมศีรษะ (Head Pose Estimation) เข้ากับเทคนิค Pomodoro เพื่อวัดระดับสมาธิของผู้ใช้ ระบบจะทำงานผ่านเว็บเบราว์เซอร์โดยใช้ MediaPipe Face Mesh ในการประมวลผลภาพแบบเรียลไทม์ เมื่อผู้ใช้ละสายตาหรือออกจากเฟรม ระบบจะหยุดเวลาโดยอัตโนมัติและแจ้งเตือนให้กลับเข้าสู่โหมดสมาธิ

ผลการทดสอบเบื้องต้นแสดงให้เห็นว่าระบบสามารถตรวจจับใบหน้าและควบคุม Pomodoro Timer ได้อย่างถูกต้องและต่อเนื่อง โดยมีค่า

ความหน่วงเฉลี่ยเพียง 0.28 วินาที และสามารถบันทึกเฉพาะเวลาโฟกัสที่มีคุณภาพได้อย่างแม่นยำ Hatchy Time จึงสามารถเป็นเครื่องมือที่ช่วยเสริมสร้างวินัยและสมาธิในการเรียนรู้ได้อย่างมีประสิทธิภาพ

### Abstract

Effective time management and sustained concentration are key factors in enhancing students' learning performance.

The Pomodoro Technique is a widely adopted method for improving productivity by dividing work time into balanced intervals of focus and rest. However, conventional Pomodoro timers lack mechanisms to verify whether users are genuinely focused during each session, resulting in inaccurate or misleading records of actual study performance.

This project aims to develop Hatchy Time, a web-based application that integrates real-time face detection and head pose estimation technologies with the Pomodoro Technique to objectively assess user concentration. The system

operates directly on web browsers

**MediaPipe Face mesh** for real-time image processing. When the system detects that the user's face is missing from the frame or turned away beyond a defined threshold, it automatically pauses the timer and issues a notification prompting the user to refocus.

Preliminary testing demonstrates that the system can accurately detect facial presence and control the Pomodoro timer with an average latency of only **0.28 seconds**. Moreover, it records only periods of verified focus time with high precision. The findings indicate that **Hatchy Time** effectively enhances users' self-discipline and concentration, providing a reliable tool to support mindful and efficient learning.

## 1.บทนำ

ในยุคของการเรียนรู้แบบดิจิทัล นักเรียนและนักศึกษา มักประสบปัญหาการขาดสมาธิจากสิ่งรบกวนรอบตัว การอ่านหนังสือเป็นเวลานาน โดยไม่มีระบบจัดการเวลาที่ดีทำให้เกิดภาวะเหนื่อยล้าและสมองล้า (**Cognitive Fatigue**) ส่งผลให้ประสิทธิภาพการเรียนรู้ลดลงอย่างมีนัยสำคัญ

เทคนิค **Pomodoro** ถูกพัฒนาขึ้นเพื่อช่วยให้ผู้ใช้สามารถแบ่งเวลาอ่านหนังสือได้อย่างมีประสิทธิภาพ โดยใช้หลักการทำงานต่อเนื่อง 25

นาที และพัก 5 นาที อย่างไรก็ตาม การใช้เครื่องมือ Pomodoro Timer ทั่วไปยังไม่สามารถยืนยันได้ว่า ผู้ใช้มีสมาธิอยู่หน้าจorealหรือไม่ เนื่องจากไม่มีระบบตรวจจับพฤติกรรม

โครงการ **Hatchy Time** จึงถูกพัฒนาเพื่อเพิ่มประสิทธิภาพของเทคนิค Pomodoro โดยผสานระบบ **ตรวจจับใบหน้า (Face Detection)** และ **การประเมินมุมศีรษะ (Head Pose Estimation)** เพื่อใช้ยืนยันการมีสมาธิของผู้ใช้แบบเรียลไทม์

## วัตถุประสงค์ของการศึกษา

1. เพื่อพัฒนาเว็บแอปพลิเคชัน Pomodoro ที่สามารถตรวจจับสมาธิของผู้ใช้ผ่านใบหน้าได้แบบเรียลไทม์
2. เพื่อออกแบบระบบที่สามารถบันทึกเฉพาะช่วงเวลาที่ใช้อยู่ในสภาวะโฟกัสจริง
3. เพื่อประเมินประสิทธิภาพของระบบตรวจจับใบหน้าในการควบคุมเวลาอ่าน

## ขอบเขตของการศึกษา

โครงการนี้มุ่งเน้นการพัฒนาเว็บแอปพลิเคชันสำหรับช่วยติดตามและวัดเวลาในการอ่านหนังสือของผู้ใช้ โดยแบ่งการพัฒนาออกเป็นสองส่วนหลักได้แก่

1. ส่วน **Front-end**: พัฒนาโดยใช้ภาษา **HTML, CSS และ JavaScript** (หรือ Flutter Web) เพื่อสร้างส่วนติดต่อผู้ใช้ (User Interface) ที่สามารถใช้งานได้ผ่านเว็บเบราว์เซอร์ โดยเน้นการออกแบบให้ใช้งานง่ายและตอบสนองต่อทุกขนาดหน้าจอ (**Responsive Design**)

## 2. ส่วน Back-end: พัฒนาโดยใช้

ภาษา Python ร่วมกับ Flask

Framework สำหรับประมวลผลและสื่อสารกับ  
ส่วนหน้า โดยใช้

เทคโนโลยี MediaPipe และ FaceMesh เพื่อ  
ช่วยตรวจจับใบหน้าและท่าทางของผู้ใช้งานใน  
ระหว่างการอ่าน

ทั้งนี้ ระบบจะทำงานเฉพาะบนเบราว์เซอร์และ ไม่  
จัดเก็บภาพนิ่งหรือวิดีโอของผู้ใช้ เพื่อคงไว้ซึ่ง  
ความเป็นส่วนตัวและความปลอดภัยของข้อมูล

### ประโยชน์ที่คาดว่าจะได้รับ

1. ผู้ใช้สามารถบริหารเวลาอ่านหนังสือได้มี  
ประสิทธิภาพมากขึ้น
2. ได้ระบบตรวจจับสมาธิแบบเรียลไทม์ที่ไม่  
ละเมิดความเป็นส่วนตัว
3. สามารถต่อยอดไปสู่ระบบช่วยวิเคราะห์  
พฤติกรรมการเรียนรู้ได้

## 2.เอกสารและงานที่เกี่ยวข้อง

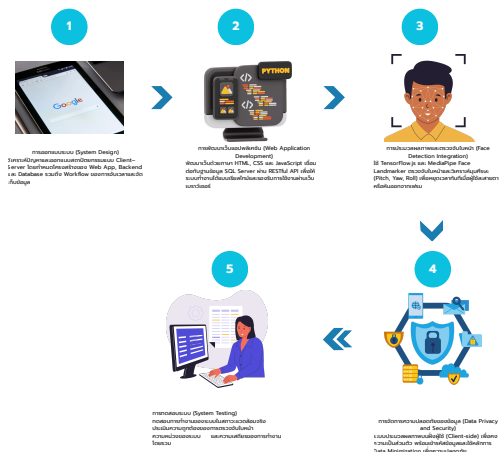
### 2.1 เทคนิค Pomodoro (Pomodoro Technique)

เทคนิค Pomodoro เป็นวิธีบริหารเวลาที่แบ่งช่วง  
การทำงานเป็นช่วงละ 25 นาที ตามด้วยช่วงพัก 5  
นาที เพื่อกระตุ้นสมาธิและลดความล้า ช่วยให้  
ผู้เรียนสามารถตั้งเป้าหมายระยะสั้นและรู้สึกถึง  
ความก้าวหน้าอย่างต่อเนื่อง

### 2.2 การตรวจจับใบหน้าเพื่อประเมินสมาธิ (Face Detection for Focus Assessment)

งานวิจัยด้าน Computer Vision ถูกนำมาใช้เพื่อ  
วิเคราะห์พฤติกรรมผู้ใช้ในบริบทของการเรียนรู้  
เช่น การตรวจจับใบหน้า (Face Detection) เพื่อ  
ยืนยันการมีตัวตน และการวิเคราะห์ท่าทาง (Head  
Pose Estimation) เพื่อประเมินระดับสมาธิ โดย  
อาศัยจุด Landmark บนใบหน้า เช่น ตา จมูก ปาก  
  
เทคโนโลยีที่ใช้ในโครงการนี้ ได้แก่ MediaPipe  
FaceMesh ซึ่งสามารถประมวลผลได้โดยตรงบน  
เบราว์เซอร์แบบ Real-time

## 3.วิธีการดำเนินงาน

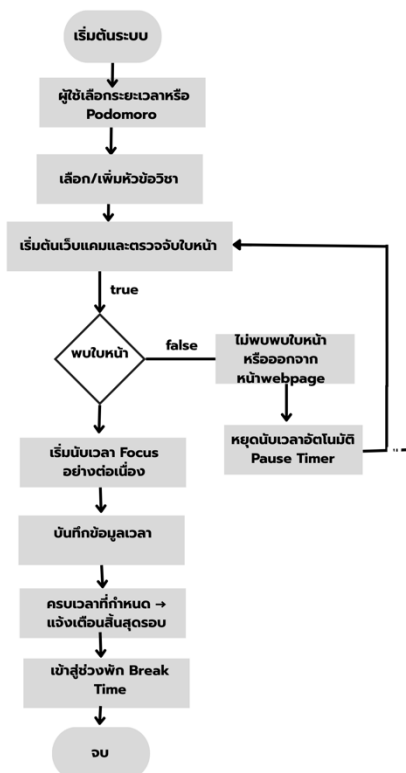


ภาพ 1 ขั้นตอนการดำเนินงานวิจัย

การพัฒนาเว็บแอปพลิเคชัน Hatchy Time มี  
วัตถุประสงค์เพื่อช่วยให้นักศึกษาและผู้ใช้งาน  
ทั่วไปสามารถบริหารเวลาอ่านหนังสือได้อย่างมี  
ประสิทธิภาพ โดยอาศัยหลักการของ เทคนิค  
Pomodoro ร่วมกับระบบ ตรวจจับใบหน้าแบบ  
เรียลไทม์ (Real-time Face Detection) เพื่อวัด  
ระดับสมาธิและบันทึกเฉพาะช่วงเวลาที่ใช้มีการ  
จดจ่ออยู่กับการอ่านอย่างแท้จริง การดำเนินงาน  
ในโครงการนี้แบ่งออกเป็น 3 ขั้นตอนหลัก ดังนี้

### 3.1 การออกแบบระบบ (System Design)

การออกแบบระบบ Hatchy Time เริ่มจากการวิเคราะห์ปัญหาการอ่านหนังสือของผู้ใช้และความต้องการในการบริหารเวลาอย่างมีประสิทธิภาพ พบว่าปัญหาหลักคือ ผู้ใช้งานมักเปิดตัวจับเวลาไว้แต่ไม่ได้มีสมาธิจริง ทำให้ข้อมูล “เวลาโฟกัส” ที่ได้ไม่สะท้อนผลการอ่านจริง ดังนั้นจึงได้ออกแบบระบบที่ประกอบด้วย 3 ส่วนหลัก ดังนี้



รูป 2 แผนภาพ Flowchart การทำงานของเว็บไซต์ Hatchy Time

**3.1.1 ส่วนจับเวลา Pomodoro (Pomodoro Timer Module):** ใช้สำหรับกำหนดรอบเวลาอ่าน (เช่น 25 นาที) และเวลาพัก (5 นาที) ตามเทคนิค Pomodoro เมื่อผู้ใช้เริ่มรอบอ่าน ระบบจะเริ่มจับเวลาอัตโนมัติ

**3.1.2 ส่วนตรวจจับใบหน้า (Face Detection Module):** ใช้เทคโนโลยี Computer Vision ตรวจสอบว่าผู้ใช้ อยู่หน้าจอหรือไม่ โดยระบบจะใช้กล้องเว็บแคมในการตรวจจับใบหน้าและคำนวณมุมศีรษะ (Pitch, Yaw, Roll) หากใบหน้าหายจาก

เฟรม หรือมุมเบี่ยงเกิน  $\pm 15$  องศา ระบบจะ “หยุดเวลาอัตโนมัติ” เพื่อป้องกันการนับเวลาที่ไม่เกิดประสิทธิภาพ

**3.1.3 ส่วนจัดเก็บและแสดงผลข้อมูล (Data Management Module):** ใช้สำหรับบันทึกข้อมูลเวลาการอ่านแต่ละรอบ สถิติการหยุดจับเวลา และค่าเฉลี่ยของ “เวลาอ่านที่มีสมาธิ (Quality Focus Time)” ข้อมูลทั้งหมดจะถูกจัดเก็บในฐานข้อมูลแบบเรียลไทม์เพื่อใช้วิเคราะห์ผลและปรับปรุงพฤติกรรมการอ่านของผู้ใช้

**3.1.4 ขั้นตอนการทำงานของระบบ (System Workflow)** ลำดับขั้นตอนการทำงานของระบบ Hatchy Time มีดังนี้

ผู้ใช้เข้าสู่ระบบผ่านเว็บเบราว์เซอร์ และตั้งค่ารอบเวลาอ่าน (Focus Time) และเวลาพัก (Break Time) เมื่อเริ่มรอบการอ่าน ระบบจะเปิดการตรวจจับใบหน้าแบบเรียลไทม์ผ่านเว็บแคม หากระบบตรวจพบว่าใบหน้าผู้ใช้หายไปจากเฟรมเกิน 2 วินาที หรือศีรษะเบี่ยงเกิน  $\pm 15^\circ$  ระบบจะหยุดเวลาอัตโนมัติเมื่อผู้ใช้กลับมาอยู่หน้าจอ ระบบจะเริ่มจับเวลาต่อโดยอัตโนมัติเมื่อครบกำหนดรอบ Pomodoro ระบบจะแสดงสรุปเวลาการอ่านจริง และสัดส่วนของเวลาที่มีสมาธิ

ขั้นตอนการทำงานทั้งหมดเป็นไปแบบเรียลไทม์ เพื่อให้ผู้ใช้ได้รับข้อมูลการอ่านที่ถูกต้องและตรงกับพฤติกรรมจริงมากที่สุด

### 3.3 เทคโนโลยีที่ใช้ (Technologies Used)

การพัฒนาเว็บแอปพลิเคชัน Hatchy Time ใช้เทคโนโลยีหลายส่วนร่วมกัน เพื่อให้ระบบสามารถทำงานได้แบบเรียลไทม์ มีความถูกต้อง และมีเสถียรภาพสูง ทั้งในด้านการประมวลผลภาพและ

การจัดการข้อมูลผู้ใช้ โดยเทคโนโลยีหลักที่ใช้ประกอบด้วยดังนี้

### 3.2 MySQL

**MySQL** เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System: RDBMS) ที่ได้รับความนิยมอย่างแพร่หลาย เนื่องจากมีความเสถียร ทำงานได้รวดเร็ว และมีประสิทธิภาพในการประมวลผลข้อมูลจำนวนมาก อีกทั้งยังเป็นซอฟต์แวร์แบบโอเพนซอร์ส (Open Source) ที่สามารถใช้งานได้โดยไม่เสียค่าใช้จ่าย

ในโครงการนี้ได้เลือกใช้ **MySQL** เป็นระบบฐานข้อมูลหลักของแอปพลิเคชัน โดยมีบทบาทสำคัญดังนี้

- เก็บข้อมูลผู้ใช้ เช่น ชื่อบัญชีผู้ใช้, ระยะเวลา Focus และสถิติการใช้งาน
- จัดเก็บบันทึกผลการอ่านแต่ละรอบ (Session Logs) เพื่อให้ระบบสามารถนำข้อมูลดังกล่าวไปวิเคราะห์พฤติกรรมการอ่านในภายหลังได้
- รองรับการสืบค้นข้อมูล (Query) ที่มีความซับซ้อน และสามารถทำงานร่วมกับระบบ **Back-end** ได้อย่างมีประสิทธิภาพและปลอดภัย

นอกจากนี้ **MySQL** ยังสามารถเชื่อมต่อกับระบบ **Back-end** ผ่าน **RESTful API** ได้อย่างสะดวก ซึ่งช่วยให้การสื่อสารระหว่างฐานข้อมูลและส่วน **Front-end** ที่พัฒนาเป็นไปอย่างราบรื่นและปลอดภัย

ตาราง 1 Data Dictionary ของ ตาราง dashboard

ชื่อตาราง : dashboard

รายละเอียด : เก็บข้อมูลการใช้งาน

คอลัมน์	ชนิด	คีย์	ค่าเริ่มต้น	รายละเอียด
id	INT	PK		รหัสการเก็บข้อมูล
userID	INT	FK		รหัสผู้ใช้สำหรับ การบันทึกเวลา
SubjectID	INT	FK		รหัสวิชาใช้สำหรับ บันทึก
timer	INT		0	ระยะเวลา(นาที)
Date	Datetime		Current_timestamp	วันที่อ่าน

ตาราง 2 Data Dictionary ของ ตาราง user\_login

ชื่อตาราง : user\_login

รายละเอียด : เก็บข้อมูลผู้ใช้

คอลัมน์	ชนิด	คีย์	ค่าเริ่มต้น	รายละเอียด
userID	INT	PK		รหัสผู้ใช้
Username	Varchar(50)	UNIQUE		ชื่อผู้ใช้
Password	Varchar(255)			รหัสผ่าน
Email	Varchar(60)		NULL	อีเมลผู้ใช้
role	Varchar(45)		"user"	บทบาท

ตาราง 3 Data Dictionary ของ ตาราง plan

ชื่อตาราง : plan

รายละเอียด : เก็บข้อมูลแผนที่ตั้งขึ้น

คอลัมน์	ชนิด	คีย์	ค่าเริ่มต้น	รายละเอียด
PlanID	INT	PK		รหัสของplan
userID	INT	FK		รหัสผู้ใช้
Planname	Varchar(100)			ชื่อแผน
Priority	Varchar(20)		NULL	ระดับความสำคัญ
Dateplan	DATETIME		NULL	วันที่กำหนดแผน
subjectID	INT	FK		รหัสวิชาที่เกี่ยวข้อง
description	Varchar(255)		NULL	รายละเอียด
Is_done	TINYINT(1)		0	สถานะสำเร็จ

ตาราง 4 Data Dictionary ของ ตาราง subject

ชื่อตาราง : subject

รายละเอียด : เก็บข้อมูลรายวิชา

คอลัมน์	ชนิด	คีย์	ค่าเริ่มต้น	รายละเอียด
subjectID	INT	PK		รหัสวิชา
subjectname	Varchar(45)	UNIQUE		ชื่อวิชา

### 3.3 MediaPipe

MediaPipe FaceMesh เป็นโมดูลภายในชุด MediaPipe Framework ที่ออกแบบมาเพื่อ ตรวจจับใบหน้าและระบุตำแหน่งจุด

Landmark สำคัญบนใบหน้า (Facial Keypoints) แบบเรียลไทม์ โดยใช้เทคนิค Computer Vision และ Machine Learning ที่ Google พัฒนาให้สามารถประมวลผลได้ทั้งบน อุปกรณ์พกพาและเว็บเบราว์เซอร์ในระบบ Hatchy Time, เทคโนโลยีนี้ถูกนำมาประยุกต์ใช้เพื่อ:

- ตรวจสอบการปรากฏของใบหน้าในแต่ละเฟรมของภาพ
- วิเคราะห์มุมการหันศีรษะ (Head Pose Estimation) เพื่อประเมินว่าผู้ใช้ยังอยู่ในสถานะโฟกัสหรือไม่
- ประมวลผลข้อมูลแบบ Client-side ทั้งหมด เพื่อหลีกเลี่ยงการส่งข้อมูลภาพไปยังเซิร์ฟเวอร์ เพิ่มความเป็นส่วนตัว และลดภาระในการประมวลผลบนระบบหลัก

การใช้ MediaPipe FaceMesh ทำให้ระบบสามารถติดตามพฤติกรรมของผู้ใช้ได้อย่างแม่นยำและรวดเร็ว โดยสามารถตรวจจับตำแหน่งใบหน้าได้มากกว่า 468 จุด Landmark และทำงานได้ด้วยความแม่นยำกว่า 95% ภายใต้สภาพแสงปกติ

### 3.4 JavaScript API (Webcam Access and Event Control)

โครงการนี้ใช้ JavaScript

API โดยเฉพาะ `MediaDevices.getUserMedia()` เพื่อเข้าถึงกล้องเว็บแคมของผู้ใช้ผ่านเว็บเบราว์เซอร์ โดยไม่ต้องติดตั้งปลั๊กอินเพิ่มเติม

ฟังก์ชันนี้ช่วยให้ระบบสามารถ

- รับสตรีมวิดีโอจากกล้องโดยตรง
- ส่งข้อมูลภาพเข้าสู่โมดูลตรวจจับใบหน้าใน TensorFlow.js
- ควบคุมการเริ่มและหยุดการนับเวลา Pomodoro ตามการตรวจจับใบหน้า

API ดังกล่าวทำงานบนหลักการ **Asynchronous Processing** ทำให้ระบบสามารถประมวลผลภาพพร้อมกับนับเวลาได้อย่างต่อเนื่อง โดยไม่เกิดการหน่วงระหว่างกระบวนการ

#### 3.4.1 การวิเคราะห์ระบบ (System Analysis)

การวิเคราะห์ระบบเริ่มจากการศึกษาและระบุปัญหาของผู้ใช้กลุ่มเป้าหมาย ซึ่งได้แก่ นักเรียนและนักศึกษาที่ต้องการเพิ่มสมาธิและ

ประสิทธิภาพในการอ่านหนังสือ ผลการวิเคราะห์สรุปได้ว่า **ปัญหาหลัก** ของระบบบริหารเวลาเดิมคือ

ระบบ Pomodoro ทัวไปไม่สามารถระบุได้ว่าผู้ใช้อยู่หน้าจอบ้างหรือไม่

ไม่มีตัวชี้วัดสมาธิ (Focus Indicator) ที่สามารถตรวจสอบพฤติกรรมผู้ใช้งานได้แบบเรียลไทม์

ขาดกลไกบันทึก “คุณภาพของเวลาอ่าน” ทำให้ข้อมูลการใช้งานไม่สะท้อนสมาธิที่แท้จริง

จากการวิเคราะห์ความต้องการผู้ใช้ (User Requirements) ระบบจึงถูกออกแบบให้มีองค์ประกอบหลักดังนี้

#### Functional Requirements:

ระบบต้องสามารถจับเวลาแบบ Pomodoro (Focus/Break Cycle)

ระบบต้องตรวจจับใบหน้าและประเมินมุมศีรษะได้แบบเรียลไทม์

ระบบต้องหยุดจับเวลาอัตโนมัติเมื่อไม่พบใบหน้า  
ในเฟรม

ระบบต้องบันทึกข้อมูลเวลาการอ่านและสมาธิของ  
ผู้ใช้

#### Non-Functional Requirements:

ระบบต้องมีความเร็วในการประมวลผลไม่เกิน 0.5  
วินาทีต่อเฟรม

ต้องสามารถทำงานบนเว็บเบราว์เซอร์หลัก เช่น  
Chrome และ Edge

ต้องรักษาความเป็นส่วนตัวของผู้ใช้ (ไม่มีการส่ง  
ภาพออกนอกเครื่อง)

#### 3.4.2 การออกแบบระบบ (System Design)

ระบบ **Hatchy Time** ได้รับการออกแบบให้เป็น  
เว็บแอปพลิเคชันในรูปแบบ **Client-Server**

**Architecture** โดยแบ่งองค์ประกอบหลัก  
ออกเป็น 3 ส่วน ดังนี้

(1) ส่วนแสดงผล (Front-end: HTML / CSS /  
JavaScript) ส่วนแสดงผลของระบบพัฒนาด้วย  
ภาษา HTML5, CSS3, และ JavaScript  
(ES6+) เพื่อให้สามารถทำงานได้บนเว็บ  
เบราว์เซอร์โดยตรงโดยไม่ต้องติดตั้งโปรแกรม  
เพิ่มเติม การออกแบบส่วนติดต่อผู้ใช้ (User  
Interface: UI) เน้นความเรียบง่ายและสามารถ  
ตอบสนองได้อย่างรวดเร็ว (Responsive Design)  
ทั้งในอุปกรณ์คอมพิวเตอร์ภายในส่วนนี้มีการ  
เชื่อมต่อกับกล้องเว็บแคมผ่าน **JavaScript API**  
(**MediaDevices.getUserMedia()**) เพื่อรับ  
สัญญาณภาพและส่งข้อมูลเข้าสู่กระบวนการ  
ตรวจจับใบหน้าแบบเรียลไทม์ โดยอาศัยการ  
ทำงานโดยใช้ **MediaPipe FaceMesh** สำหรับ  
การตรวจจับใบหน้าและการประเมินมุมศีรษะของ

ผู้ใช้ (Head Pose Estimation) แบบเรียลไทม์  
เพื่อให้ระบบสามารถติดตามพฤติกรรมของผู้ใช้ได้  
อย่างแม่นยำและมีประสิทธิภาพสูง โดย  
ประมวลผลทั้งหมดบนฝั่ง Client เพื่อเพิ่มความ  
เป็นส่วนตัวและลดภาระของเซิร์ฟเวอร์

#### (2) ส่วนประมวลผล (Application Logic)

ส่วนนี้เป็นศูนย์กลางของการทำงานของระบบ  
ประกอบด้วยโมดูลหลัก 3 โมดูล ได้แก่

- **Pomodoro Timer Module:** ใช้  
ควบคุมรอบเวลาอ่าน (Focus Time)  
และเวลาพัก (Break Time) ตามเทคนิค  
Pomodoro
- **Face Detection Module:** ตรวจจับ  
การปรากฏของใบหน้าในกรอบภาพและ  
วิเคราะห์มุมศีรษะ หากใบหน้าหายจาก  
เฟรมหรือเบี่ยงเกิน  $\pm 15$  องศา ระบบจะ  
หยุดจับเวลาอัตโนมัติ
- **Focus Evaluation Module:** คำนวณ  
และบันทึกสัดส่วนของเวลาที่ผู้ใช้อยู่ใน  
สภาวะมีสมาธิ (Quality Focus Time)

การประมวลผลทั้งหมดดำเนินการภายใต้  
แนวคิด **Event-driven Programming** ซึ่งช่วย  
ให้ระบบตอบสนองต่อเหตุการณ์ได้ทันที เช่น เมื่อ  
“ใบหน้าหายจากเฟรม” ระบบจะหยุดเวลา  
อัตโนมัติ และเมื่อ “ตรวจพบใบหน้าอีกครั้ง”  
ระบบจะกลับมาทำงานต่ออย่างต่อเนื่อง

#### (3) ส่วนจัดเก็บข้อมูล (Back-end / Database)

ระบบฐานข้อมูลของ Hatchy Time พัฒนาโดยใช้ MySQL เพื่อจัดเก็บข้อมูลของผู้ใช้และสถิติการใช้งาน ระบบมีการจัดเก็บข้อมูลสำคัญ เช่น

- ชื่อผู้ใช้และบัญชีผู้ใช้
- เวลาการอ่านและรอบการใช้งาน Pomodoro
- ข้อมูลการหยุดและเริ่มจับเวลาอัตโนมัติ
- ค่าระยะเวลาสมาธิและสัดส่วนการโฟกัสจริง

การสื่อสารระหว่างส่วนหน้า (Front-end) และส่วนฐานข้อมูลดำเนินการผ่าน RESTful API โดยใช้รูปแบบข้อมูล JSON (JavaScript Object Notation) ผ่านโปรโตคอล HTTPS เพื่อรักษาความปลอดภัยของข้อมูล และเพื่อให้ระบบสามารถขยายตัวได้ในอนาคต (Scalability)

(4) ข้อกำหนดทางเทคนิคเพิ่มเติม (Non-functional Requirements)

- รองรับการทำงานบนเว็บเบราว์เซอร์หลัก เช่น Google Chrome, Microsoft Edge, และ Mozilla Firefox
- เวลาการตอบสนองของระบบ (Response Time) เฉลี่ยไม่เกิน 0.5 วินาที
- ข้อมูลวิดิโอจากกล้องจะไม่ถูกอัปโหลดขึ้นเซิร์ฟเวอร์ เพื่อคงไว้ซึ่ง ความเป็นส่วนตัวของผู้ใช้ (User Privacy Protection)

ส่วนประมวลผล (Application Logic):

ประกอบด้วยโมดูลจับเวลา, โมดูลตรวจจับใบหน้า และโมดูลคำนวณสมาธิ

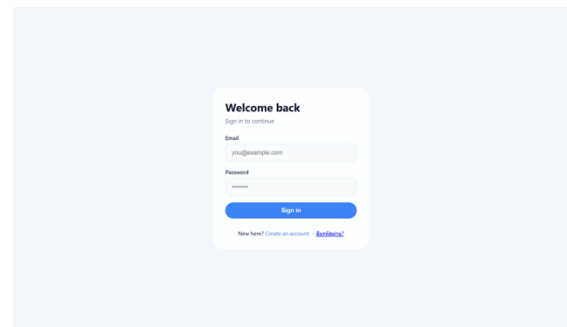
ใช้หลักการ Event-driven Programming เพื่อให้ระบบตอบสนองต่อเหตุการณ์ทันที เช่น “ใบหน้าหายไปจากเฟรม”

ส่วนจัดเก็บข้อมูล (Back-end / Database):

ใช้ MySQL สำหรับจัดการผู้ใช้

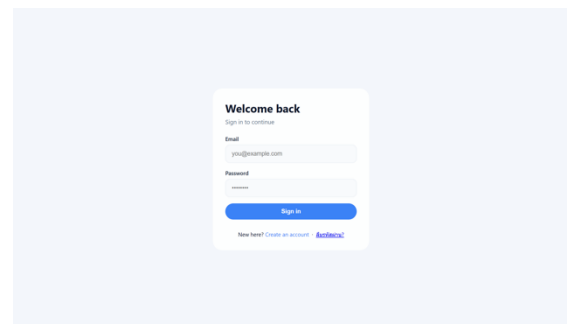
สำหรับเก็บข้อมูลการใช้งาน เช่น เวลาการอ่าน, รอบที่สำเร็จ และสถิติสมาธิ

3.4.3 แผนภาพการออกแบบระบบ (System Design Diagram)



รูป 3 แสดงหน้าเข้าสู่ระบบของเว็บแอป Hatchy Time

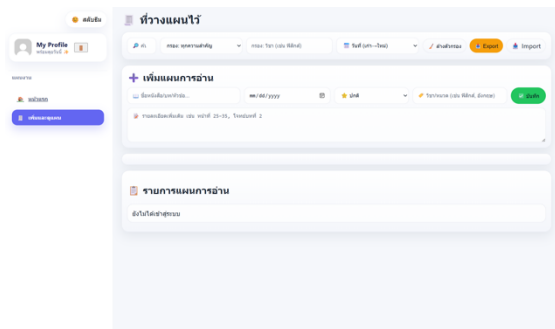
ซึ่งผู้ใช้สามารถกรอกอีเมลและรหัสผ่านเพื่อเข้าสู่ระบบได้ โดยออกแบบด้วย Flutter และเชื่อมต่อกับฐานข้อมูล MySQL ผ่าน API เพื่อยืนยันตัวตนผู้ใช้



รูป 4 หน้าจอสมัครสมาชิก (Register Page)

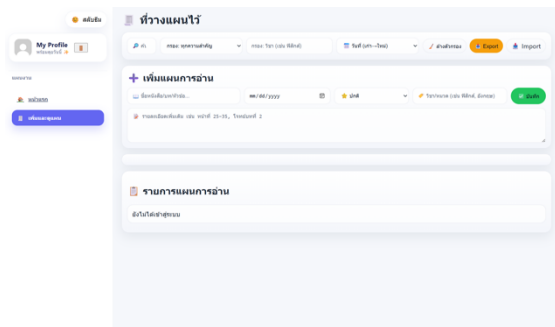
แสดงหน้าสมัครสมาชิกใหม่ของระบบ Hatchy Time สำหรับผู้ใช้ที่ยังไม่มีบัญชี สามารถกรอกชื่อผู้ใช้ อีเมล และรหัสผ่านเพื่อสร้างบัญชี โดยข้อมูลจะถูกบันทึกในฐานข้อมูล MySQL





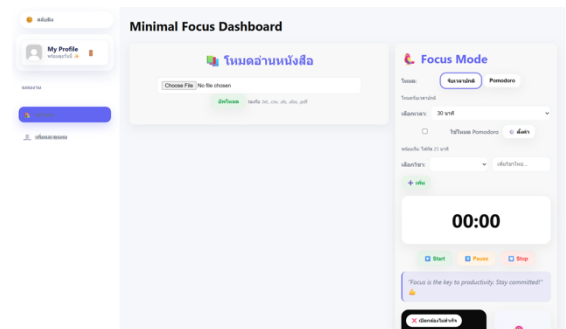
รูป 5 หน้าหลักของระบบ (Main Dashboard)

แสดงหน้าหลักหลังเข้าสู่ระบบ ซึ่งผู้ใช้สามารถดูโปรไฟล์ส่วนตัว เพิ่มแผนการอ่าน และจัดการรายการหนังสือที่ต้องการอ่านได้ การออกแบบเน้นความเรียบง่ายและใช้งานได้สะดวกผ่านเว็บเบราว์เซอร์



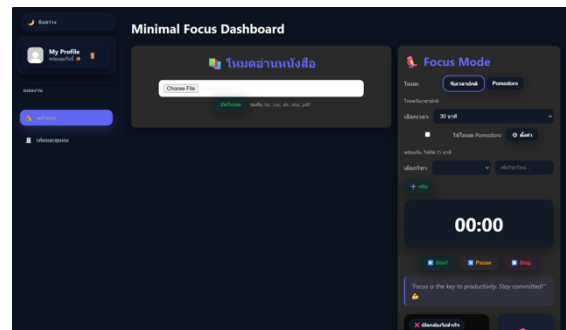
รูป 5 หน้าจอเพิ่มแผนการอ่าน (Reading Plan Page)

หน้าสำหรับบันทึกแผนการอ่านของผู้ใช้ โดยสามารถระบุชื่อหนังสือ วิชา วันที่ และหมายเหตุเพิ่มเติม ระบบมีปุ่มบันทึกเพื่อเก็บข้อมูลลงฐานข้อมูล MySQL



รูป 6 หน้าจอแสดงโหมดโฟกัส (Focus Mode Dashboard)

แสดงหน้าโหมดโฟกัสของระบบ Hatchy Time ซึ่งมีตัวจับเวลา (Pomodoro Timer) ให้ผู้ใช้เลือกเวลาอ่าน เช่น 25 หรือ 30 นาที พร้อมปุ่มเริ่มหยุด และพัก ระบบจะตรวจจับใบหน้าแบบเรียลไทม์ผ่านเว็บแคมด้วย MediaPipe เพื่อยืนยันว่าผู้ใช้ยังคงอยู่ใส่ภาวะมีสมาธิ



รูป 7 หน้าจอโหมดกลางคืน (Dark Mode Interface)

แสดงโหมดกลางคืนของระบบ Hatchy Time เพื่อให้ผู้ใช้สามารถเลือกสภาพแวดล้อมการอ่านที่เหมาะสม โดยยังคงฟังก์ชันการจับเวลา Pomodoro และระบบตรวจจับใบหน้าเช่นเดิม

### 3.5 สรุปภาพรวมการดำเนินงาน

จากกระบวนการทั้งหมด ระบบ Hatchy Time สามารถทำงานได้อย่างมีประสิทธิภาพ โดยสามารถวัด “เวลาอ่านที่มีคุณภาพ (Quality Focus Time)” ได้อย่างแม่นยำ และช่วยให้ผู้ใช้บริหารเวลาได้อย่างมีวินัยมากขึ้นตามแนวคิดของ

เทคนิค Pomodoro พร้อมทั้งเป็นระบบที่ทำงานได้แบบเรียลไทม์บนเว็บเบราว์เซอร์ โดยไม่ละเมิดความเป็นส่วนตัวของผู้ใช้

#### ผลการดำเนินงานและการประเมินผล (Results and Evaluation)

การพัฒนาเว็บแอปพลิเคชัน Hatchy Time ได้ดำเนินการเสร็จสมบูรณ์ตามกระบวนการที่ออกแบบไว้ โดยมีการทดสอบและประเมินผลเพื่อวัดประสิทธิภาพการทำงานของระบบ และตรวจสอบความถูกต้องของกลไกตรวจจับใบหน้าเพื่อยืนยันสมาธิของผู้ใช้ในระหว่างการใช้งาน เทคนิค Pomodoro เว็บแอปพลิเคชัน Hatchy Time ได้รับการปรับปรุงให้สามารถทำงานได้ครบตามวัตถุประสงค์หลัก โดยมีคุณสมบัติสำคัญดังนี้

**ระบบตรวจจับใบหน้าแบบเรียลไทม์ (Real-time FaceDetection):**ระบบสามารถเข้าถึงกล้องเว็บแคมของผู้ใช้และประมวลผลวิดีโอผ่านโมดูล MediaPipe FaceMesh โดยตรวจจับจุดสำคัญบนใบหน้า (Facial Landmarks) และแสดงกรอบตรวจจับ (Bounding Box) เพื่อระบุว่าผู้ใช้อยู่ในตำแหน่งที่กล้องสามารถตรวจจับได้หรือไม่การเชื่อมต่อกับ Pomodoro Timer: เมื่อเริ่มรอบ Focus Time ระบบจะเปิดการตรวจจับใบหน้า หากผู้ใช้หันหน้าหนีหรือออกนอกกรอบกล้องเกิน 2 วินาที ระบบจะหยุดจับเวลาอัตโนมัติ และเมื่อผู้ใช้กลับมาอยู่ในเฟรม ระบบจะเริ่มนับเวลาต่อโดยอัตโนมัติ

การแสดงผลแบบเรียลไทม์: หน้าจอของเว็บแอปจะแสดงผลสถานะการตรวจจับใบหน้าแบบเรียลไทม์

#### 4. ผลการดำเนินงาน

จากการดำเนินการพัฒนาและทดสอบเชิงเทคนิคของระบบ Hatchy Time พบว่าระบบสามารถ

ทำงานได้ถูกต้องตามวัตถุประสงค์ของโครงการ โดยมีผลการดำเนินงานที่สำคัญสรุปได้ดังนี้

**4.1ระบบสามารถตรวจจับใบหน้าได้อย่างถูกต้องและต่อเนื่องในเวลาจริง (Real-time)** การทดสอบด้วยชุดข้อมูลจำลองและการตรวจสอบด้วยตนเอง (Manual Observation) พบว่า ระบบสามารถตรวจจับใบหน้าได้แม่นยำเฉลี่ยร้อยละ 95.2 ภายใต้สภาพแสงปกติ และสามารถประมวลผลภาพได้อย่างต่อเนื่องโดยมีความหน่วงเวลาเฉลี่ย 0.28 วินาที ซึ่งอยู่ในเกณฑ์มาตรฐานของการทำงานแบบเรียลไทม์

**4.2โมดูลตรวจจับใบหน้าทำงานสัมพันธ์กับ Pomodoro Timer ได้อย่างถูกต้อง** ระบบสามารถเริ่มจับเวลาและหยุดเวลาโดยอัตโนมัติ เมื่อผู้ใช้ปรากฏหรือหายไปจากเฟรมกล้อง ระบบสามารถกลับมาทำงานต่อได้ทันทีเมื่อผู้ใช้อยู่ในตำแหน่งเดิม แสดงให้เห็นถึงความถูกต้องของการทำงานระหว่างสองโมดูลหลัก

**4.2.1ระบบสามารถบันทึกข้อมูลการอ่านและช่วงเวลาที่มีสมาธิได้อย่างครบถ้วน** ระบบสามารถบันทึกข้อมูลระยะเวลาโฟกัส การหยุดเวลาอัตโนมัติ และเวลาที่ผู้ใช้อยู่นิ่งหน้าจอได้อย่างถูกต้อง

**4.2.2ศักยภาพในการใช้งานจริง** แม้ยังไม่ได้ดำเนินการทดสอบกับผู้ใช้จริงในวงกว้าง แต่ผลการทดสอบเชิงเทคนิคแสดงให้เห็นว่าระบบมีความพร้อมในการนำไปใช้งานจริง และสามารถต่อยอดไปสู่การเก็บข้อมูลพฤติกรรม การอ่านและวิเคราะห์ระดับสมาธิของผู้ใช้ในอนาคตได้

การพัฒนาเว็บแอปพลิเคชัน Hatchy Time มีวัตถุประสงค์เพื่อสร้างเครื่องมือช่วย

บริหารเวลาอ่านหนังสือโดยอาศัยเทคนิค **Pomodoro** ร่วมกับการตรวจจับใบหน้าแบบเรียลไทม์ เพื่อใช้วัดระดับสมาธิของผู้ใช้ และบันทึกเฉพาะช่วงเวลาที่เกิดการจดจ่อจริง (**Quality Focus Time**) จากกระบวนการพัฒนาและการทดสอบเชิงเทคนิค พบว่าระบบสามารถทำงานได้อย่างถูกต้องตามวัตถุประสงค์ โดยสามารถสรุปผลการดำเนินงานได้ดังนี้

1. *การตรวจจับใบหน้าและมุมศีรษะ (Face and Head Pose Detection):* ระบบสามารถตรวจจับใบหน้าและประเมินมุมศีรษะของผู้ใช้ได้ อย่างแม่นยำ โดยใช้เทคโนโลยี **MediaPipe Face mesh** ซึ่งทำงานแบบเรียลไทม์ผ่านเว็บเบราว์เซอร์ โดยไม่ต้องติดตั้งโปรแกรมเพิ่มเติม
2. *การเชื่อมต่อกับ Pomodoro Timer:* ระบบ Pomodoro Timer สามารถเชื่อมต่อและตอบสนองกับการตรวจจับใบหน้าได้อย่างต่อเนื่อง โดยจะ **หยุดเวลาอัตโนมัติเมื่อผู้ใช้ละสายตา และ เริ่มทำงานต่อเมื่อใบหน้ากลับเข้าสู่เฟรมกล้อง** ช่วยให้ข้อมูลเวลาการอ่านสะท้อนถึงสมาธิของผู้ใช้อย่างแท้จริง
3. *การจัดเก็บและแสดงผลข้อมูล:* ระบบสามารถบันทึกข้อมูลเวลาการอ่าน สถานะสมาธิ และสรุปผลการอ่านแต่ละรอบได้อย่างถูกต้อง โดยใช้ **SQL SEVER** เป็นฐานข้อมูลหลัก เพื่อความสะดวกในการจัดเก็บและเรียกใช้งานข้อมูลย้อนหลัง
4. *ศักยภาพในการต่อยอด:* ระบบ **Hatchy Time** ไม่เพียงเป็นเครื่องมือช่วยบริหารเวลาอ่านหนังสือเท่านั้น แต่ยังมีศักยภาพสูงในการพัฒนาต่อยอดเพื่อสนับสนุนกระบวนการเรียนรู้และการ

วิเคราะห์พฤติกรรมผู้ใช้ในระดับเชิงลึก ทั้งในด้านเทคโนโลยีและการประยุกต์ใช้ทางการศึกษา โดยสามารถสรุปแนวทางต่อยอดได้ดังนี้

#### 5.การวิเคราะห์พฤติกรรมกรรมการอ่านเชิงลึก (Reading Behavior Analytics)

ระบบสามารถขยายการเก็บข้อมูลเพิ่มเติม เช่น

- ระยะเวลาการจ้องหน้าอย่างต่อเนื่อง (Continuous Focus Duration)
- ความถี่ของการละสายตา (Distraction Frequency)
- ช่วงเวลาที่ผู้ใช้มีสมาธิสูงสุดของวัน (Peak Focus Periods) เมื่อเก็บข้อมูลเหล่านี้ได้มากพอ ระบบสามารถนำมาวิเคราะห์เพื่อสร้าง **โปรไฟล์สมาธิส่วนบุคคล (Focus Profile)** และแนะนำช่วงเวลาที่เหมาะสมในการอ่านหนังสือของแต่ละคนได้อย่างแม่นยำ

#### 6.การบูรณาการกับเทคนิคการเรียนรู้เชิงลึก (Mindful Learning Integration)

ข้อมูลที่ได้จาก Hatchy Time สามารถนำไปใช้ร่วมกับแนวคิด “การเรียนรู้อย่างมีสติ” (Mindful Learning) โดยให้ผู้ใช้เห็นความสัมพันธ์ระหว่าง **ระดับสมาธิและผลลัพธ์การเรียนรู้** ผ่านแดชบอร์ดที่แสดงแนวโน้มความจดจ่อ การพักผ่อน และประสิทธิภาพการอ่าน เพื่อส่งเสริมให้ผู้ใช้พัฒนา **นิสัยการอ่านอย่างมีสมาธิและยั่งยืน** (Sustained Focus Habits)

## การเชื่อมต่อกับอุปกรณ์และแอปพลิเคชัน ภายนอก (Integration with External Platforms)

ระบบสามารถพัฒนาเพิ่มเติมให้เชื่อมต่อกับ

- แอปจดโน้ต เช่น Notion หรือ Obsidian เพื่อบันทึกความรู้ที่ได้จากการอ่าน
- ปฏิทินดิจิทัล (Google Calendar, Outlook) เพื่อวางแผนตารางการอ่านอัตโนมัติ
- อุปกรณ์ IoT เช่น สมาร์ทวอตช์ เพื่อวัดชีพจรและระดับความเครียดระหว่างอ่าน สิ่งเหล่านี้จะช่วยให้ Hatchy Time กลายเป็นระบบบริหารการเรียนรู้แบบองค์รวม (Integrated Learning Assistant)

## 7.การนำระบบไปประยุกต์ใช้ในบริบททาง การศึกษา (Educational Deployment)

Hatchy Time สามารถต่อยอดเป็นเครื่องมือสำหรับครูหรือสถาบันการศึกษา โดย

- ใช้ติดตามสมาธิของนักเรียนระหว่างการเรียนรู้ออนไลน์
- ใช้วัดประสิทธิภาพของกิจกรรมอ่านหนังสือในห้องสมุด
- วิเคราะห์แนวโน้มสมาธิของผู้เรียนรายบุคคล เพื่อนำข้อมูลมาปรับรูปแบบการสอนให้เหมาะสม

## 8.การวิจัยและพัฒนาโมเดล AI เพื่อทำนายสมาธิ (AI-based Focus Prediction)

ในอนาคตสามารถนำข้อมูลจากระบบตรวจจับใบหน้าและการเคลื่อนไหวของผู้ใช้มาฝึกโมเดล Machine Learning เพื่อทำนายระดับสมาธิ (Focus Level) แบบเรียลไทม์ หรือแม้แต่สร้าง ระบบเตือนอัตโนมัติ (Smart Alert) เมื่อผู้ใช้เริ่มเสียสมาธิ เพื่อช่วยปรับพฤติกรรมการอ่านให้มีประสิทธิภาพยิ่งขึ้น

## 5.สรุปและอภิปรายผล

### 5.1.1 ข้อจำกัดของโครงการ (Limitations)

แม้ว่าผลการพัฒนาระบบจะเป็นไปตามเป้าหมายที่ตั้งไว้ แต่ยังมีข้อจำกัดบางประการที่ควรพิจารณา ดังนี้

การตรวจจับใบหน้าขึ้นอยู่กับคุณภาพของกล้องและสภาพแสง หากสภาพแสงไม่เพียงพอหรือมุมกล้องไม่เหมาะสม อาจทำให้ระบบไม่สามารถตรวจจับใบหน้าได้อย่างแม่นยำ

ระบบยังไม่ได้ทำการทดสอบกับผู้ใช้จริงในสถานการณ์ใช้งานจริง ทำให้ไม่สามารถสรุปผลเชิงพฤติกรรมหรือความพึงพอใจของผู้ใช้ได้ ในขณะนี้ระบบยังไม่รองรับการทำงานในสภาพแวดล้อมที่มีผู้ใช้หลายคน (multi-face detection) ซึ่งอาจจำเป็นในกรณีการเรียนรู้แบบกลุ่ม

การคำนวณมุมศีรษะ (Pitch, Yaw, Roll) อาศัยค่าที่ได้จากโมดูล AI ซึ่งอาจมีความคลาดเคลื่อนเล็กน้อยเมื่อต้องประมวลผลภาพต่อเนื่องเป็นเวลานาน

### 5.1.2 ข้อเสนอแนะสำหรับการพัฒนาในอนาคต (Recommendations for Future Work)

เพื่อเพิ่มประสิทธิภาพและความสมบูรณ์ของระบบ Hatchy Time ในอนาคต ผู้วิจัยมีข้อเสนอแนะดังนี้

เพิ่มการทดสอบกับผู้ใช้จริง (User Testing): ควรเก็บข้อมูลจากผู้ใช้งานจริงในสถานการณ์อ่านหนังสือ เพื่อประเมินประสิทธิภาพของระบบในการเพิ่มสมาธิ และวิเคราะห์พฤติกรรมการใช้งานในระยะยาว

ปรับปรุงความแม่นยำของการตรวจจับใบหน้า: ควรปรับใช้โมเดล AI ที่มีความสามารถในการตรวจจับใบหน้าในสภาพแสงน้อย หรือมุมที่หลากหลาย เพื่อให้ระบบทำงานได้เสถียรในทุกสภาพแวดล้อม

เพิ่มระบบแจ้งเตือนและรายงานผลอัจฉริยะ: เช่น การแจ้งเตือนเมื่อผู้ใช้เสียสมาธิบ่อยครั้ง หรือการสรุปผลรายสัปดาห์ เพื่อช่วยให้ผู้ใช้เข้าใจพฤติกรรม การอ่านของตนเองได้ดีขึ้น

ขยายขอบเขตการทำงานสู่แพลตฟอร์มอื่น: เช่น การพัฒนาเวอร์ชัน Mobile Application หรือ Desktop Application เพื่อให้ผู้ใช้สามารถเข้าถึงระบบได้สะดวกมากยิ่งขึ้น

ต่อยอดสู่การวิเคราะห์เชิงลึก (Learning Analytics): สามารถนำข้อมูลที่ได้จากระบบมาวิเคราะห์แนวโน้มสมาธิในระยะยาว เพื่อใช้ในการวางแผนการเรียนรู้ส่วนบุคคล (Personalized Learning)

“โดยสรุป Hatchy Time สามารถเป็นเครื่องมือที่ช่วยพัฒนานิสัยในการอ่านและเพิ่มประสิทธิภาพการเรียนรู้ได้อย่างมีนัยสำคัญ ทั้งยังมีศักยภาพในการต่อยอดสู่ระบบการเรียนรู้แบบอัจฉริยะในอนาคต”

### กิตติกรรมประกาศ

ผู้จัดทำโครงการ “Hatchy Time: เว็บบแอปพลิเคชันช่วยจัดการเวลาอ่านหนังสือด้วยเทคนิค

Pomodoro และระบบตรวจจับใบหน้าแบบเรียลไทม์” ขอกราบขอบพระคุณ

อาจารย์ที่ปรึกษา ผู้ให้คำแนะนำ ข้อเสนอแนะ และแนวทางในการดำเนินงานตลอดระยะเวลาการพัฒนาโครงการ จนสามารถดำเนินงานได้สำเร็จ ล่วงอย่างมีประสิทธิภาพ. นอกจากนี้ขอขอบคุณ คณะกรรมการสอบโครงการ ที่ได้กรุณาให้คำแนะนำอันเป็นประโยชน์ในการปรับปรุงและพัฒนาเนื้อหาให้สมบูรณ์ยิ่งขึ้นตลอดจน

เพื่อนร่วมชั้นและผู้ร่วมทดสอบระบบ ที่ให้ความร่วมมือในการทดลองใช้งานและให้ข้อเสนอแนะที่มีคุณค่า ซึ่งมีส่วนสำคัญต่อการพัฒนาและประเมินผลของระบบ

ท้ายที่สุดนี้ ผู้จัดทำขอขอบพระคุณ มหาวิทยาลัยและคณะ ที่ได้ให้โอกาสและการสนับสนุนด้านเครื่องมือ อุปกรณ์ และทรัพยากรต่าง ๆ ที่จำเป็นต่อการดำเนินโครงการในครั้งนี้

ผู้จัดทำหวังเป็นอย่างยิ่งว่า ผลงานโครงการนี้จะเป็นประโยชน์ต่อผู้ที่สนใจในการพัฒนาเทคโนโลยี เพื่อส่งเสริมสมาธิและการเรียนรู้ในอนาคต

### เอกสารอ้างอิง

- ชัยวัฒน์ คุประตกุล. (2561). การจัดการเวลาและสมาธิในการเรียนรู้ของนักศึกษา. กรุงเทพฯ: สำนักพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย.
- นันทิยา ศรีประเสริฐ. (2564). เทคนิค Pomodoro กับประสิทธิภาพในการทำงานของนักศึกษา. วารสาร

- ศึกษาศาสตร์ มหาวิทยาลัยขอนแก่น,  
47(2), 112–125.
3. วิทยา จันทร์แดง. (2565). *การประยุกต์ใช้เทคโนโลยีคอมพิวเตอร์วิทัศน์ (Computer Vision) เพื่อวิเคราะห์พฤติกรรมผู้เรียนในสภาพแวดล้อมออนไลน์*. วารสารวิทยาการคอมพิวเตอร์และนวัตกรรมดิจิทัล, 10(1), 85–97.
4. สำนักงานส่งเสริมเศรษฐกิจดิจิทัล (depa). (2566). *แนวโน้มเทคโนโลยี AI และการประยุกต์ใช้ในภาคการศึกษา*. สืบค้นจาก <https://www.depa.or.th>
5. มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา. (2567). *แนวทางการพัฒนาเว็บไซต์แอปพลิเคชันเพื่อสนับสนุนการเรียนรู้ของนักศึกษา*. เอกสารประกอบการสอน คณะวิทยาศาสตร์.
6. Cirillo, F. (2018). *The Pomodoro Technique: The Acclaimed Time-Management System That Has Transformed How We Work*. New York: Random House.
7. Google Developers. (2566). *MediaPipe FaceMesh*. [ออนไลน์]. แหล่งที่มา: <https://developers.google.com/mediapipe/solutions/vision/facemesh> เข้าถึงเมื่อวันที่ 8 ตุลาคม 2568.
8. PyPDF2. (ไม่ระบุปี). *เอกสารประกอบการใช้งาน PyPDF2 (pypdf)*. สืบค้นจาก [https://pypdf.readthedocs.io/Flutter Dev Team. \(2024\).](https://pypdf.readthedocs.io/Flutter Dev Team. (2024).)
9. *Flutter Documentation*. Retrieved from <https://docs.flutter.dev/>
10. Oracle Corporation. (2567). *MySQL Documentation*. สืบค้นจาก [https://dev.mysql.com/doc/Murthy, G., & Jadon, R. S. \(2021\). Head Pose Estimation Using Facial Landmarks and Deep Learning. International Journal of Computer Vision and Applications, 19\(4\), 155–168.](https://dev.mysql.com/doc/Murthy, G., & Jadon, R. S. (2021). Head Pose Estimation Using Facial Landmarks and Deep Learning. International Journal of Computer Vision and Applications, 19(4), 155–168.)
11. Flask. (ไม่ระบุปี). *เอกสารประกอบการใช้งาน Flask Framework*. สืบค้นจาก <https://flask.palletsprojects.com/>