

แบบรายงานความก้าวหน้าในการทำโครงการ

โครงการ เว็บไซต์พลิเคชันสำหรับระบบติดตามสถานะนักศึกษาในระดับบัณฑิตศึกษา

รายงานครั้งที่ 3

1. ภาพรวม (Summary)

งานด้านเทคนิค

คะแนน (เต็ม 10)

(ลายมือชื่อ
อ.ที่ปรึกษา)

งาน (Task)	นายนิติภูมิ จันทร์อำ	นายภัทรพล พลกุล	ร้อยละ ความสำเร็จตาม แผนงาน
แก้ไข Diagram(ร้อยละ 100)	3 ชั่วโมง	3 ชั่วโมง	100
ทดลองการใช้งาน API ใน Postman(ร้อยละ 100)	3 ชั่วโมง	3 ชั่วโมง	100
ออกแบบหน้าต่างผู้ใช้งาน UX/UI ของเว็บไซต์ สำหรับระบบติดตามสถานะนักศึกษาในระดับ บัณฑิตศึกษา (ร้อยละ 100)	3 ชั่วโมง	3 ชั่วโมง	100
แก้ไขหน้าต่างผู้ใช้งาน UX/UI ของเว็บไซต์ สำหรับระบบติดตามสถานะนักศึกษาในระดับ บัณฑิตศึกษา (ร้อยละ 100)	3 ชั่วโมง	3 ชั่วโมง	100

งานทั่วไป

งาน (Task)	นายนิติภูมิ จันทร์อำ	นายภัทรพล พลกุล	ร้อยละ ความสำเร็จตาม แผนงาน
แก้ไข Diagram Database (ร้อยละ 100)	3 ชั่วโมง	3 ชั่วโมง	100
ศึกษาวิธีการเพิ่มและลบข้อมูลและอัปเดตข้อมูลผ่าน หน้าเว็บ (ร้อยละ 100)	3 ชั่วโมง	3 ชั่วโมง	100
การศึกษาการออกแบบ UX/UI (ร้อยละ 100)	3 ชั่วโมง	3 ชั่วโมง	100

2. ผลลัพธ์ที่สำคัญในสัปดาห์ที่ผ่านมา (Highlights)

1. เพิ่มและลบข้อมูลและอัปเดตข้อมูลผ่านหน้าเว็บ
2. เพิ่มข้อมูลและเรียกข้อมูลโดยใช้ภาษา python
3. ออกแบบหน้าตาต่างผู้ใช้งาน UX/UI ของเว็บแอปพลิเคชันสำหรับระบบติดตามสถานะนักศึกษาในระดับบัณฑิตศึกษา
4. แก้ไขหน้าตาต่างผู้ใช้งาน UX/UI ของเว็บแอปพลิเคชันสำหรับระบบติดตามสถานะนักศึกษาในระดับบัณฑิตศึกษาให้สมบูรณ์

3. รายละเอียดในการปฏิบัติงาน (Details)

3.1. รายละเอียดของทฤษฎีที่เกี่ยวข้องกับการพัฒนาเว็บแอปพลิเคชัน

- ทฤษฎีการพัฒนาเว็บแอปพลิเคชัน (Web Application Development) คือ เว็บแอปพลิเคชันที่พัฒนาสำหรับระบบติดตามนักศึกษาในระดับบัณฑิตศึกษาเป็นซอฟต์แวร์ที่สามารถทำงานบนเว็บเบราว์เซอร์ ทำให้นักศึกษาและเจ้าหน้าที่สามารถเข้าถึงและใช้งานระบบได้ผ่านอินเทอร์เน็ต ไม่ว่าจะอยู่ที่ใดก็ตาม ซึ่งช่วยลดความจำเป็นในการเดินทางมายังคณะเพื่อดำเนินการเอกสาร ระบบนี้มีการออกแบบให้ผู้ใช้สามารถกรอกแบบฟอร์มและติดตามสถานะของเอกสารได้อย่างสะดวกและรวดเร็ว

ในส่วนของ Front-End Development ซึ่งเป็นส่วนที่นักศึกษาหรือเจ้าหน้าที่จะได้ตอบกับระบบผ่านหน้าเว็บ UI ควรมีการใช้เทคโนโลยี HTML, CSS และ JavaScript เพื่อสร้างประสบการณ์ผู้ใช้ (User Experience, UX) ที่ดี โดยหน้าเว็บจะต้องใช้งานง่ายและเป็นมิตรต่อผู้ใช้ ในขณะที่ Back-End Development ใช้เฟรมเวิร์คอย่าง Django ที่ทำงานบนภาษา Python สำหรับจัดการข้อมูลและการทำงานทางธุรกิจ เช่น การประมวลผลข้อมูลการยื่นเอกสาร การอัปเดตสถานะ และการจัดการข้อมูลผู้ใช้

- เฟรมเวิร์ค Django และการพัฒนาระบบ Back-End คือ การพัฒนา Back-End ของระบบนี้ใช้ Django Framework ซึ่งเป็นเครื่องมือที่ทรงพลังสำหรับการพัฒนาเว็บแอปพลิเคชันในภาษา Python โดย Django มีระบบจัดการผู้ใช้ (User Authentication) ที่สามารถนำไปใช้กับการยืนยันตัวตนผ่านอีเมลของมหาวิทยาลัยได้ เพื่อให้สิทธิ์การเข้าถึงและจัดการข้อมูลระหว่างนักศึกษา เจ้าหน้าที่ และอาจารย์แตกต่างกันไป ระบบนี้ใช้โครงสร้าง Model-View-Template (MVT) ที่ช่วยจัดการข้อมูลในฐานข้อมูลและแสดงผลข้อมูลบนหน้าเว็บ การแยกตรรกะการทำงานออกเป็นส่วนๆ ทำให้การพัฒนาง่ายต่อการดูแลรักษา

ในกรณีของระบบติดตามเอกสาร การใช้ Django ช่วยให้การจัดการเอกสารทั้งหมดเป็นระบบดิจิทัล สามารถบันทึกและดึงข้อมูลจากฐานข้อมูลได้โดยตรง ทำให้เจ้าหน้าที่ไม่ต้องเสียเวลาบันทึกข้อมูลลงในไฟล์ Excel นอกจากนี้ Django ยังรองรับการเชื่อมต่อกับ API เพื่อรับและส่งข้อมูลระหว่างเว็บแอปพลิเคชันและโมบายแอปพลิเคชัน

- ทฤษฎีการออกแบบส่วนติดต่อผู้ใช้ (User Interface Design) และประสบการณ์ผู้ใช้ (User Experience) คือ ระบบนี้จำเป็นต้องมีการออกแบบส่วนติดต่อผู้ใช้ (UI) ที่เรียบง่ายและสะดวกในการใช้งาน เนื่องจากผู้ใช้เป็นทั้งนักศึกษา เจ้าหน้าที่ และอาจารย์ การออกแบบ Responsive Web Design เป็นสิ่งสำคัญเพื่อให้ระบบสามารถปรับการแสดงผลให้เหมาะสมกับหน้าจอขนาดต่างๆ ทั้งคอมพิวเตอร์และอุปกรณ์พกพา เทคโนโลยี HTML และ CSS ถูกใช้ในการสร้างโครงสร้างและการจัดวางองค์ประกอบหน้าเว็บ ขณะที่ JavaScript ใช้ในการเพิ่มความสามารถในการโต้ตอบและทำให้การแสดงผลข้อมูลในระบบเป็นแบบไดนามิก

- ฐานข้อมูลและการจัดการข้อมูล (Database Management) คือ เว็บแอปพลิเคชันนี้มีความต้องการในการจัดการข้อมูลเอกสารจำนวนมากจากนักศึกษาระดับบัณฑิตศึกษา เช่น ข้อมูลการยื่นคำร้องขออนุมัติหัวข้อวิทยานิพนธ์ และข้อมูลสถานะการอนุมัติ การจัดการข้อมูลจึงต้องการใช้ Relational Database Management System (RDBMS) อย่าง MySQL หรือ PostgreSQL ซึ่งจะช่วยให้การจัดการข้อมูลมีความเป็นระบบ สามารถค้นหาและอัปเดตข้อมูลได้อย่างรวดเร็ว ระบบนี้ออกแบบให้มีการจัดเก็บข้อมูลแบบดิจิทัล ทำให้เจ้าหน้าที่สามารถเข้าถึงและจัดการข้อมูลได้โดยไม่ต้องจัดการเอกสารในรูปแบบกระดาษ ซึ่งเป็นการลดภาระงานและข้อผิดพลาดที่เกิดจากการจัดเก็บข้อมูลแบบเดิม

- การเชื่อมต่อผ่าน API (Application Programming Interface) คือ ในระบบติดตามเอกสารนี้มีการพัฒนา API เพื่อเชื่อมต่อกับระบบภายนอก เช่น โมบายแอปพลิเคชันที่นักศึกษาจะใช้ในการตรวจสอบสถานะของเอกสาร การออกแบบ API ที่ดีช่วยให้ระบบสามารถสื่อสารและส่งข้อมูลกันได้อย่างมีประสิทธิภาพ โดย API จะถูกใช้ในการยืนยันตัวตน การดึงข้อมูลสถานะเอกสาร การอัปเดตสถานะ และการแจ้งเตือนเหตุการณ์สำคัญผ่านอีเมล นอกจากนี้ API ยังมีการรักษาความปลอดภัยข้อมูล โดยใช้การเข้ารหัสข้อมูลระหว่างการรับส่ง และการตรวจสอบสิทธิ์ผู้ใช้ในการเข้าถึงข้อมูล

3.2. รายละเอียด เนื้อหาทางทฤษฎี หลักการออกแบบ web framework Django โครงสร้างของระบบที่ได้จากการออกแบบ

- หลักการออกแบบเว็บแอปพลิเคชันด้วย Django เป็นหนึ่งในเว็บเฟรมเวิร์คที่พัฒนาด้วยภาษา Python มีจุดเด่นในการเป็น "Batteries Included" นั้นหมายความว่า Django มีฟังก์ชันและเครื่องมือที่จำเป็นสำหรับการ

พัฒนาเว็บแอปพลิเคชันแบบครบครัน เช่น ระบบจัดการผู้ใช้ ระบบยืนยันตัวตน ระบบฐานข้อมูล ระบบการจัดการไฟล์ และระบบการแสดงผลข้อมูล Django ช่วยลดเวลาในการพัฒนาแอปพลิเคชันด้วยการจัดเตรียมโครงสร้างพื้นฐานและส่วนประกอบสำคัญไว้ล่วงหน้า

- หลักการออกแบบที่ใช้ใน Django คือการแยกตรรกะการทำงานออกเป็นส่วน ๆ โดยแบ่งเป็น 3 ส่วนหลัก ประกอบด้วย

- Model - เป็นส่วนที่จัดการข้อมูลในฐานข้อมูล ซึ่งจะรับผิดชอบในการสร้าง แก้ไข ลบ และเรียกข้อมูลจากฐานข้อมูล Model ใน Django มีโครงสร้างที่สอดคล้องกับตารางข้อมูลในฐานข้อมูล (Relational Database) เช่น MySQL หรือ PostgreSQL ในระบบติดตามนักศึกษา ข้อมูลต่าง ๆ เช่น ข้อมูลนักศึกษา ข้อมูลเอกสาร และสถานะของเอกสาร จะถูกจัดเก็บในฐานข้อมูลผ่าน Model
- View - เป็นส่วนที่รับข้อมูลจาก Model และจัดการกับตรรกะทางธุรกิจ (Business Logic) ก่อนส่งข้อมูลไปยังผู้ใช้ผ่านส่วนติดต่อผู้ใช้ (User Interface) ในระบบนี้ View จะรับผิดชอบการดึงข้อมูลเอกสาร สถานะการอนุมัติ การยื่นเอกสาร และการอัปเดตสถานะต่าง ๆ ของเอกสารที่ส่งมาจากนักศึกษาและเจ้าหน้าที่
- Template - เป็นส่วนที่จัดการการแสดงผลข้อมูลหรือ UI (User Interface) ให้กับผู้ใช้ ซึ่งจะใช้เทคโนโลยี HTML, CSS และ JavaScript ในการออกแบบให้การแสดงผลสวยงาม ใช้งานง่าย ในระบบนี้หน้าแดชบอร์ดจะแสดงสถานะเอกสารต่าง ๆ เช่น วิทยานิพนธ์และคำร้องที่นักศึกษายื่นไป

- โครงสร้างของระบบตามการออกแบบ คือ การออกแบบโครงสร้างระบบติดตามนักศึกษาระดับบัณฑิตศึกษาผ่านเว็บแอปพลิเคชันนี้ มีโครงสร้างที่สอดคล้องกับสถาปัตยกรรม Model-View-Template (MVT) ซึ่งเป็นหลักการออกแบบระบบใน Django โดยสามารถแบ่งเป็น 3 ส่วนหลัก ดังนี้

- Frontend (User Interface): นักศึกษา เจ้าหน้าที่ และอาจารย์ สามารถโต้ตอบกับระบบผ่านหน้า UI ที่ออกแบบมาให้ใช้งานง่ายและเป็นมิตรกับผู้ใช้ (User-Friendly) เทคโนโลยีที่ใช้ในส่วนนี้ประกอบไปด้วย HTML, CSS และ JavaScript โดยที่หน้าแดชบอร์ดหลักจะแสดงข้อมูลสถานะเอกสารการอนุมัติของนักศึกษา ระบบจะทำการแจ้งเตือนเมื่อนักศึกษามีการเปลี่ยนแปลงสถานะเอกสาร หรือต้องมีการอัปเดตข้อมูล
- Backend (Server Side Logic): ส่วนนี้ใช้ Django ในการจัดการตรรกะทางธุรกิจ การประมวลผลข้อมูลจากฐานข้อมูล และการจัดการคำร้องขอจากผู้ใช้ เช่น การอัปเดตสถานะเอกสาร การจัดเก็บและดึงข้อมูล

จากฐานข้อมูล ระบบการยืนยันตัวตนผ่านอีเมลมหาวิทยาลัยและการให้สิทธิ์การเข้าถึงข้อมูลต่าง ๆ เช่น นักศึกษา เจ้าหน้าที่ และอาจารย์ ระบบ Backend จะส่งข้อมูลไปยัง Frontend เพื่อนำไปแสดงผลบนหน้าเว็บ

- Database Management: ในระบบนี้ใช้ฐานข้อมูลแบบ Relational Database ซึ่งเหมาะสมกับการจัดเก็บข้อมูลที่มีโครงสร้างเป็นตาราง เช่น ข้อมูลนักศึกษา ข้อมูลเอกสาร และสถานะต่าง ๆ ของเอกสาร โดยฐานข้อมูลนี้จะเชื่อมต่อกับ Django ผ่าน ORM (Object-Relational Mapping) ที่ช่วยให้การจัดการข้อมูลในฐานข้อมูลมีความสะดวกและปลอดภัยมากขึ้น Django ORM จะช่วยในการสร้างตารางข้อมูลจาก Model ที่ออกแบบไว้โดยไม่ต้องเขียน SQL เอง

- ความปลอดภัยของระบบ (Security) การพัฒนาระบบติดตามนักศึกษานี้ใช้ ระบบการยืนยันตัวตน (Authentication) ผ่านอีเมลของมหาวิทยาลัย ซึ่งเป็นระบบที่ปลอดภัยในการตรวจสอบสิทธิ์ของผู้ใช้ รวมถึงการจัดการสิทธิ์การเข้าถึงข้อมูล (Authorization) โดย Django มีระบบจัดการผู้ใช้ (User Management System) ในตัว ทำให้สามารถควบคุมการเข้าถึงข้อมูลของผู้ใช้ในแต่ละระดับ เช่น นักศึกษา อาจารย์ และเจ้าหน้าที่ ได้อย่างมีประสิทธิภาพ

- การเชื่อมต่อกับ API (Application Programming Interface) ระบบติดตามนักศึกษานี้ยังมีการออกแบบ API เพื่อใช้ในการเชื่อมต่อระหว่างเว็บแอปพลิเคชันกับโมบายแอปพลิเคชันของนักศึกษา ซึ่ง API จะทำหน้าที่เป็นสะพานกลางในการรับส่งข้อมูล เช่น การยืนยันตัวตน การอัปเดตสถานะเอกสาร หรือการแจ้งเตือนข้อมูลสำคัญ โดยการใช้ Django Rest Framework (DRF) ช่วยในการพัฒนา API ให้มีความยืดหยุ่นและประสิทธิภาพสูง

API ที่ถูกพัฒนาขึ้นจะช่วยให้โมบายแอปพลิเคชันสามารถดึงข้อมูลสถานะเอกสารของนักศึกษาหรือข้อมูลสำคัญอื่น ๆ ได้ทันทีผ่านอินเทอร์เน็ตที่ออกแบบไว้ ซึ่งช่วยเพิ่มความสะดวกในการติดตามเอกสารแบบเรียลไทม์ผ่านอุปกรณ์พกพา

3.3. หลักการออกแบบเว็บไซต์

3.3.1. การกำหนดขนาดเว็บเพจ

- ขนาดของเว็บไซต์ที่นิยมในปัจจุบันมี 2 ขนาด คือ ขนาดเว็บไซต์แบบ 800 X 600 pixels เป็นขนาดที่สามารถใช้ได้กับหน้าจอทุกขนาดในปัจจุบันเป็นขนาดของการออกแบบเว็บไซต์ที่ใช้ในอดีต เนื่องจากอดีตขนาดของจอคอมพิวเตอร์มีขนาดเล็ก

- ขนาดเว็บไซต์แบบ 1024 X 768 pixels เป็นขนาดที่นิยมในปัจจุบัน เนื่องจากผู้ใช้นิยมใช้จอคอมพิวเตอร์ขนาดใหญ่ขึ้นเนื่องมาจากราคาจอคอมพิวเตอร์ที่ถูกลง

3.3.2. รูปแบบการออกแบบเว็บไซต์

เว็บไซต์ในปัจจุบันจะมีการออกแบบที่แตกต่างกันไม่มากนัก ซึ่งการออกแบบหน้าตาเว็บไซต์ส่วนใหญ่จะมององค์ประกอบขององค์กร หน่วยงาน หรือเนื้อหาเรื่องที่น่าสนใจเป็นหลัก ซึ่งการออกแบบหน้าตาของเว็บไซต์มีอยู่ 3 แบบ คือ

- การออกแบบเว็บไซต์ที่เน้นการนำเสนอเนื้อหา

เป็นการออกแบบเว็บไซต์ที่เน้นการนำเสนอเนื้อหามากกว่ารูปภาพ โดยโครงสร้างใช้รูปแบบตารางเป็นหลัก มีการออกแบบหน้าตาแบบง่าย เช่น มีเมนูสารบัญ และเนื้อหา

- การออกแบบเว็บไซต์ที่เน้นภาพกราฟิก

เป็นการออกแบบเว็บไซต์ที่เน้นภาพกราฟิกที่สวยงาม ซึ่งอาจจะใช้โปรแกรม Photoshop สำหรับการตกแต่งภาพ ข้อดี สวยงาม น่าสนใจ ข้อเสีย อาจจะใช้เวลาในการโหลดเว็บนาน

- การออกแบบเว็บไซต์ที่มีทั้งภาพและเนื้อหา

เป็นการออกแบบเว็บที่นิยมในปัจจุบันซึ่งประกอบด้วยข้อความ รูปภาพ โดยมีการจัดองค์ประกอบต่าง ๆ เพื่อให้เว็บน่าสนใจ

3.3.3. การออกแบบโครงสร้างเว็บไซต์

การออกแบบโครงสร้างเว็บไซต์ คือ การวางแผนการจัดลำดับ เนื้อหาสาระของเว็บไซต์ ออกเป็นหมวดหมู่เพื่อจัดทำเป็นโครงสร้างในการจัดวางหน้าเว็บเพจทั้งหมด เปรียบเสมือนแผนที่ ที่ทำให้เห็นโครงสร้างทั้งหมดของ

เว็บไซต์ ช่วยในนักออกแบบเว็บไซต์ไม่ให้หลงทาง การจัดโครงสร้างของเว็บไซต์ มีจุดมุ่งหมายสำคัญคือ การที่จะทำให้ผู้เข้าเยี่ยมชม สามารถค้นหาข้อมูลในเว็บเพจได้อย่างเป็นระบบ ซึ่งถือว่าเป็นขั้นตอนที่สำคัญ ที่สามารถสร้างความสำเร็จให้กับผู้ที่ทำหน้าที่ในการออกแบบและพัฒนาเว็บไซต์ (Webmaster) การออกแบบโครงสร้างหรือจัดระเบียบของข้อมูลที่ชัดเจน แยกย่อยเนื้อหาออกเป็นส่วนต่าง ๆ ที่สัมพันธ์กันและให้อยู่ในมาตรฐานเดียวกัน จะช่วยให้การใช้งานและง่ายต่อการเข้าอ่านเนื้อหาของผู้ใช้เว็บไซต์

3.3.4. หลักในการออกแบบโครงสร้างเว็บไซต์ ควรพิจารณาดังนี้

- กำหนดวัตถุประสงค์ โดยพิจารณาว่าเป้าหมายของการสร้างเว็บไซต์นี้ทำเพื่ออะไร
- ศึกษาคุณลักษณะของผู้ที่เข้ามาใช้ว่ากลุ่มเป้าหมายใดที่ผู้สร้างต้องการสื่อสาร ข้อมูลอะไรที่พวกเขาต้องการโดยขั้นตอนนี้ควรปฏิบัติควบคู่ไปกับขั้นตอนที่หนึ่ง

- วางแผนเกี่ยวกับการจัดรูปแบบโครงสร้างเนื้อหาสาระ การออกแบบเว็บไซต์ต้องมีการจัดโครงสร้างหรือจัดระเบียบข้อมูลที่ชัดเจน การที่เนื้อหาไม่มีความต่อเนื่องไปไม่สิ้นสุดหรือกระจายมากเกินไป อาจทำให้เกิดความสับสนต่อผู้ใช้ได้ ฉะนั้นจึงควรออกแบบให้มีลักษณะที่ชัดเจนแยกย่อยออกเป็นส่วนตัวอย่าง ๆ จัดหมวดหมู่ในเรื่องที่สัมพันธ์กัน รวมทั้งอาจมีการแสดงให้ผู้ใช้งานเห็นแผนที่โครงสร้างเพื่อป้องกันความสับสนได้

- กำหนดรายละเอียดให้กับโครงสร้าง ซึ่งพิจารณาจากวัตถุประสงค์ที่ตั้งไว้ โดยตั้งเกณฑ์ในการใช้ เช่น ผู้ใช้ควรทำอะไรบ้าง จำนวนหน้าควรมีเท่าใด มีการเชื่อมโยง มากน้อยเพียงใด

- หลังจากนั้น จึงทำการสร้างเว็บไซต์แล้วนำไปทดลองเพื่อหาข้อผิดพลาดและทำการแก้ไขปรับปรุง แล้วจึงนำเข้าสู่อินเทอร์เน็ตเป็นขั้นสุดท้าย

3.3.5. องค์ประกอบที่ดีของการออกแบบเว็บไซต์

3.3.5.1. **โครงสร้างที่ชัดเจน** ผู้ออกแบบเว็บไซต์ควรจัดโครงสร้างหรือจัดระเบียบของข้อมูลที่ชัดเจน แยกย่อยเนื้อหาออกเป็นส่วนตัวอย่าง ๆ ที่สัมพันธ์กันและให้อยู่ในมาตรฐานเดียวกัน จะช่วยให้การใช้งานและง่ายต่อการอ่านเนื้อหาของผู้ใช้

3.3.5.2. **การใช้งานที่ง่าย** ลักษณะของเว็บที่มีการใช้งานง่ายจะช่วยให้ผู้ใช้รู้สึกสบายใจต่อการอ่านและสามารถทำความเข้าใจกับเนื้อหาได้อย่างเต็มที่ โดยไม่ต้องมาเสียเวลาอยู่กับการทำความเข้าใจ การใช้งานที่สับสนด้วยเหตุนี้ผู้ออกแบบจึงควรกำหนดปุ่มการใช้งานที่ชัดเจน เหมาะสม โดยเฉพาะปุ่มควบคุมเส้นทางการเข้าสู่เนื้อหา (Navigation) ไม่ว่าจะเป็นเดินหน้า ถอยหลัง หากเป็นเว็บไซต์ที่มีเว็บเพจจำนวนมาก ควรจะจัดทำแผนผังของเว็บไซต์ (Site Map) ที่ช่วยให้ผู้ใช้ทราบว่า ตอนนี้อยู่ ณ จุดใด หรือเครื่องมือสืบค้น (Search Engine) ที่ช่วยในการค้นหาหน้าที่ต้องการ

3.3.5.3. **การเชื่อมโยงที่ดี** ลักษณะไฮเปอร์เท็กซ์ที่ใช้ในการเชื่อมโยง ควรอยู่ในรูปแบบที่เป็นมาตรฐาน ทัวไปและต้องระวังเรื่องของตำแหน่งในการเชื่อมโยง การที่จำนวนการเชื่อมโยงมากและกระจัดกระจายอยู่ทั่วไปในหน้าอาจก่อให้เกิดความสับสน นอกจากนี้คำที่ใช้สำหรับการเชื่อมโยงจะต้องเข้าใจง่ายมีความชัดเจนและไม่สับสนเกินไป นอกจากนี้ในแต่ละเว็บเพจที่สร้างขึ้นควรมี จุดเชื่อมโยงกลับมายังหน้าแรกของเว็บไซต์ที่กำลังใช้งานอยู่ด้วย ทั้งนี้เพื่อว่าผู้ใช้เกิดหลงทาง และไม่ทราบว่าจะทำอย่างไรต่อไปจะได้มีหนทางกลับมาสู่จุดเริ่มต้นใหม่ ระวังอย่าให้มีหน้าที่ไม่มีการเชื่อมโยง (Orphan Page) เพราะจะทำให้ผู้ใช้ไม่รู้จะทำอย่างไรต่อไป

3.3.5.4. **ความเหมาะสมในหน้าจอ** เนื้อหาที่น่าเสนอในแต่ละหน้าจอควรสั้น กระชับ และทันสมัย หลีกเลี่ยงการใช้หน้าจอที่มีลักษณะการเลื่อนขึ้นลง (Scrolling) แต่ถ้าจำเป็นต้องมี ควรจะให้ข้อมูลที่มีความสำคัญอยู่บริเวณด้านบนสุดของหน้าจอ หลีกเลี่ยงการใช้กราฟิกด้านบนของหน้าจอ เพราะถึงแม้จะดูสวยงาม

แต่จะทำให้ผู้ใช้เสียเวลาในการได้รับข้อมูลที่ต้องการ แต่หากต้องมีการใช้ภาพประกอบก็ควรใช้เฉพาะที่มีความสัมพันธ์กับเนื้อหาเท่านั้น นอกจากนี้การใช้รูปภาพเพื่อเป็นพื้นหลัง (Background) ไม่ควรเน้นสีที่ฉูดฉาดมากนัก เพราะอาจจะไปลดความเด่นชัดของเนื้อหา ควรใช้ภาพที่มีสีอ่อน ๆ ไม่สว่างจนเกินไปรวมถึงการใช้เทคนิคต่าง ๆ เช่น ภาพเคลื่อนไหว หรือตัวอักษรวิ่ง (Marquees) ซึ่งอาจจะเกิดการรบกวนการอ่านได้ ควรใช้เฉพาะที่จำเป็นจริง ๆ เท่านั้นตัวอักษรที่นำมาแสดงบนจอภาพควรเลือกขนาดที่อ่านง่าย ไม่มีสีสันและลวดลายมากเกินไป

3.3.5.5. ความรวดเร็ว ความรวดเร็วเป็นสิ่งสำคัญประการหนึ่งที่ส่งผลต่อการเรียนรู้ ผู้ใช้จะเกิดอาการเบื่อหน่ายและหมดความสนใจกับเว็บที่ใช้เวลาในการแสดงผลนาน สาเหตุสำคัญที่จะทำให้การแสดงผลนานคือการใช้ภาพกราฟิกหรือภาพเคลื่อนไหว ซึ่งแม้ว่าจะช่วยดึงดูดความสนใจได้ดี ฉะนั้นในการออกแบบจึงควรหลีกเลี่ยงการใช้ภาพขนาดใหญ่ หรือภาพเคลื่อนไหวที่ไม่จำเป็น และพยายามใช้กราฟิกแทนตัวอักษรธรรมดาให้น้อยที่สุด โดยไม่ควรใช้มากเกินไปกว่า 2 – 3 บรรทัดในแต่ละหน้าจอ

3.3.6. โครงสร้างของเว็บไซต์

3.3.6.1. เว็บที่มีโครงสร้างแบบเรียงลำดับ (Sequential Structure)

เป็นโครงสร้างแบบธรรมดาที่ใช้กันมากที่สุดเนื่องจากง่ายต่อการจัดระบบข้อมูล ข้อมูลที่นิยม จัดด้วยโครงสร้างแบบนี้มักเป็นข้อมูลที่มีลักษณะเป็นเรื่องราวตามลำดับของเวลา เช่น การเรียงลำดับตามตัวอักษร ธรรมชาติ สารานุกรม หรืออภินิหารศัพท์ โครงสร้างแบบนี้ เหมาะกับเว็บไซต์ที่มีขนาดเล็ก เนื้อหาไม่ซับซ้อนใช้การลิงก์ (Link) ไปที่หน้า ทิศทางของการเข้าสู่เนื้อหา (Navigation) ภายในเว็บจะเป็นการดำเนินเรื่องในลักษณะเส้นตรง โดยมี ปุ่มเดินหน้า-ถอยหลังเป็นเครื่องมือหลักในการกำหนดทิศทาง ข้อเสียของโครงสร้างแบบนี้คือ ผู้ใช้ไม่สามารถกำหนดทิศทางการเข้าสู่เนื้อหาของตนเองได้ ทำให้เสียเวลาเข้าสู่เนื้อ

3.3.6.2. เว็บที่มีโครงสร้างแบบลำดับขั้น (Hierarchical Structure)

เป็นวิธีที่ดีที่สุดในหนึ่งในการจัดระบบโครงสร้างที่มีความซับซ้อนของข้อมูล โดยแบ่งเนื้อหาออกเป็นส่วนต่างๆ และมีรายละเอียดย่อยๆ ในแต่ละส่วนลดหลั่นกันมาในลักษณะแนวคิดเดียวกับ แผนภูมิองค์กร จึงเป็นการง่ายต่อการทำความเข้าใจกับโครงสร้างของเนื้อหาในเว็บลักษณะนี้ ลักษณะเด่นเฉพาะของ เว็บประเภทนี้คือการมีจุดเริ่มต้นที่จุดรวมจุดเดียว นั่นคือ โฮมเพจ (Homepage) และเชื่อมโยงไปสู่เนื้อหา ในลักษณะเป็นลำดับจากบนลงล่าง

3.3.6.3. เว็บที่มีโครงสร้างแบบตาราง (Grid Structure)

โครงสร้างรูปแบบนี้มีความซับซ้อนมากกว่ารูปแบบที่ผ่านมา การออกแบบเพิ่มความยืดหยุ่น ให้แก่การเข้าสู่เนื้อหาของผู้ใช้ โดยเพิ่มการเชื่อมโยงซึ่งกันและกันระหว่างเนื้อหาแต่ละส่วน เหมาะแก่ การแสดงให้เห็นความสัมพันธ์กันของเนื้อหา การเข้าสู่เนื้อหาของผู้ใช้จะไม่เป็นลักษณะเชิงเส้นตรง เนื่องจากผู้ใช้สามารถเปลี่ยนทิศทางการเข้าสู่เนื้อหาของตนเองได้ ในการจัดระบบโครงสร้างแบบนี้ เนื้อหาที่นำมาใช้แต่ละส่วนควรมีลักษณะที่เหมือนกัน และสามารถใช้รูปแบบร่วมกัน หลักการออกแบบคือนำหัวข้อทั้งหมดมาบรรจุลงในที่เดียวกันซึ่งโดยทั่วไป จะเป็นหน้าแผนภาพ (Map Page) ที่แสดงในลักษณะเดียวกับโครงสร้างของเว็บ เมื่อผู้ใช้คลิกเลือก หัวข้อใด ก็จะไปสู่หน้าเนื้อหา (Topic Page) ที่แสดงรายละเอียดของหัวข้อนั้นๆ และภายในหน้านั้น ก็จะมีการเชื่อมโยงไปยังหน้ารายละเอียดของหัวข้ออื่นที่เป็นเรื่องเดียวกัน นอกจากนี้ยังสามารถนำ โครงสร้างแบบเรียงลำดับและแบบลำดับขั้นมาใช้ร่วมกันได้อีกด้วย ถึงแม้โครงสร้างแบบนี้ อาจสร้างความยุ่งยากในการเข้าใจได้ และอาจเกิดปัญหาการคงค้าง ของหัวข้อ (Cognitive Overhead) ได้ แต่จะเป็นประโยชน์ที่สุดเมื่อผู้ใช้ได้เข้าใจถึงความสัมพันธ์ ระหว่างเนื้อหา ในส่วนของการออกแบบจำเป็นจะต้องมีการวางแผนที่ดี เนื่องจากการเชื่อมโยงที่เกิดขึ้น ได้หลายทิศทาง นอกจากนี้การปรับปรุงแก้ไขอาจเกิดความยุ่งยากเมื่อต้องเพิ่มเนื้อหาในภายหลัง

3.3.6.4. เว็บที่มีโครงสร้างแบบใยแมงมุม (Web Structure)

โครงสร้างประเภทนี้มีความยืดหยุ่นมากที่สุด ทุกหน้าในเว็บสามารถจะเชื่อมโยงไปถึงกัน ได้หมด เป็นการสร้างรูปแบบการเข้าสู่เนื้อหาที่เป็นอิสระ ผู้ใช้สามารถกำหนดวิธีการเข้าสู่เนื้อหาได้ด้วย ตนเอง การเชื่อมโยงเนื้อหาแต่ละหน้าอาศัยการโยงใยข้อความที่มีมโนทัศน์ (Concept) เหมือนกัน ของแต่ละหน้าในลักษณะของไฮเปอร์เท็กซ์หรือไฮเปอร์มีเดีย โครงสร้างลักษณะนี้จัดเป็นรูปแบบที่ ไม่มีโครงสร้างที่แน่นอนตายตัว (Unstructured) นอกจากนี้การเชื่อมโยงไม่ได้จำกัดเฉพาะเนื้อหา ภายในเว็บนั้นๆ แต่สามารถเชื่อมโยงออกไปสู่เนื้อหาจากเว็บภายนอกได้ ลักษณะการเชื่อมโยงในเว็บนั้น นอกเหนือจากการใช้ไฮเปอร์เท็กซ์หรือไฮเปอร์มีเดีย กับข้อความที่มีมโนทัศน์ (Concept) เหมือนกันของแต่ละหน้าแล้ว ยังสามารถใช้ลักษณะการเชื่อมโยง จากรายการที่รวบรวมชื่อหรือหัวข้อของเนื้อหาแต่ละหน้าไว้ ซึ่งรายการนี้จะปรากฏอยู่บริเวณใด บริเวณหนึ่งในหน้าจอ ผู้ใช้สามารถคลิกที่หัวข้อใดหัวข้อหนึ่งในรายการเพื่อเลือกที่จะเข้าไปสู่หน้าใดๆ ก็ได้ตามความต้องการ ข้อดีของรูปแบบนี้คือง่ายต่อผู้ใช้ในการท่องเที่ยวนเว็บ โดยผู้ใช้สามารถกำหนดทิศทาง การเข้าสู่เนื้อหาได้ด้วยตนเอง แต่ข้อเสียคือถ้ามีการเพิ่มเนื้อหาใหม่ๆ อยู่เสมอจะเป็นการยากในการ ปรับปรุง นอกจากนี้การเชื่อมโยงระหว่างข้อมูลที่มีมากมายนั้นอาจทำให้ผู้ใช้เกิดการสับสนและ เกิดปัญหาการคงค้างของหัวข้อ (Cognitive Overhead) ได้

3.3.7. องค์ประกอบของการออกแบบเว็บไซต์

3.3.7.1. ความเรียบง่าย (Simplicity)

หมายถึง การจำกัดองค์ประกอบเสริมให้เหลือเฉพาะองค์ประกอบหลัก กล่าวคือในการสื่อสารเนื้อหากับผู้ใช้นั้น เราต้องเลือกเสนอสิ่งที่เราต้องการนำเสนอจริง ๆ ออกมาในส่วนของกราฟิก สี สัน ตัวอักษรและภาพเคลื่อนไหว ต้องเลือกให้พอเหมาะ ถ้าหากมีมากเกินไปจะรบกวนสายตาและสร้างความรำคาญต่อผู้ใช้ตัวอย่างเว็บไซต์ที่ได้รับการออกแบบที่ดี ได้แก่ เว็บไซต์ของบริษัทใหญ่ ๆ อย่างเช่น Apple Adobe Microsoft หรือ Nokia ที่มีการออกแบบเว็บไซต์ในรูปแบบที่เรียบง่าย ไม่ซับซ้อน และใช้งานอย่างสะดวก

3.3.7.2. ความสม่ำเสมอ (Consistency)

หมายถึง การสร้างความสม่ำเสมอให้เกิดขึ้นตลอดทั้งเว็บไซต์ โดยอาจเลือกใช้รูปแบบเดียวกัน ตลอดทั้งเว็บไซต์ก็ได้ เพราะถ้าหากว่าแต่ละหน้าในเว็บไซต์นั้นมีความแตกต่างกันมากจนเกินไป อาจทำให้ผู้ใช้เกิดความสับสนและไม่แน่ใจว่ากำลังอยู่ในเว็บไซต์เดิมหรือไม่ เพราะฉะนั้นการออกแบบเว็บไซต์ในแต่ละหน้าควรมีรูปแบบ สไตล์ของกราฟิก ระบบเนวิเกชัน (Navigation) และโทนสีที่มีความคล้ายคลึงกันตลอดทั้งเว็บไซต์

3.3.7.3. ความเป็นเอกลักษณ์ (Identity)

ในการออกแบบเว็บไซต์ต้องคำนึงถึงลักษณะขององค์กรเป็นหลัก เนื่องจากเว็บไซต์จะสะท้อนถึงเอกลักษณ์และลักษณะขององค์กร การเลือกใช้ตัวอักษร ชุดสี รูปภาพหรือกราฟิก จะมีผลต่อรูปแบบของเว็บไซต์เป็นอย่างมาก ตัวอย่างเช่น ถ้าเราต้องออกแบบเว็บไซต์ของธนาคารแต่เรากลับเลือกสีส้มและกราฟิกมากมาย อาจทำให้ผู้ใช้คิดว่าเป็นเว็บไซต์ของสวนสนุกซึ่งส่งผลต่อความเชื่อถือขององค์กรได้

3.3.7.4. เนื้อหา (Useful Content)

ถือเป็นสิ่งสำคัญที่สุดในเว็บไซต์ เนื้อหาในเว็บไซต์ต้องสมบูรณ์และได้รับการปรับปรุงพัฒนาให้ทันสมัยอยู่เสมอ ผู้พัฒนาต้องเตรียมข้อมูลและเนื้อหาที่ผู้ใช้ต้องการให้ถูกต้องและสมบูรณ์ เนื้อหาที่สำคัญที่สุดคือเนื้อหาที่ทีมผู้พัฒนาสร้างสรรค์ขึ้นมาเอง และไม่ไปซ้ำกับเว็บอื่น เพราะจะถือเป็นสิ่งที่ดึงดูดผู้ใช้ให้เข้ามาเว็บไซต์ได้เสมอ แต่ถ้าเป็นเว็บที่ลิงค์ข้อมูลจากเว็บอื่น ๆ มาเมื่อใดก็ตามที่ผู้ใช้ทราบว่า ข้อมูลนั้นมาจากเว็บใด ผู้ใช้ก็ไม่จำเป็นต้องกลับมาใช้งานลิงค์เหล่านั้นอีก

3.3.7.5. ระบบเนวิเกชัน (User-Friendly Navigation)

เป็นส่วนประกอบที่มีความสำคัญต่อเว็บไซต์มาก เพราะจะช่วยให้ผู้ใช้เกิดความสับสนระหว่างดูเว็บไซต์ ระบบเนวิเกชันจึงเปรียบเสมือนป้ายบอกทาง ดังนั้นการออกแบบเนวิเกชัน จึงควรให้เข้าใจง่าย ใช้งานได้สะดวก ถ้ามีการใช้กราฟิกก็ควรสื่อความหมาย ตำแหน่งของการวางเนวิเกชันก็ควรวางให้สม่ำเสมอ เช่น อยู่

ตำแหน่งบนสุดของทุกหน้าเป็นต้น ซึ่งถ้าจะให้ดีเมื่อมีเมนูชั้นที่เป็นกราฟิกก็ควรเพิ่มระบบเมนูชั้นที่เป็นตัวอักษรไว้ส่วนล่างด้วย เพื่อช่วยอำนวยความสะดวกให้กับผู้ใช้ที่ยกเลิกการแสดงผลภาพกราฟิกบนเว็บเบราว์เซอร์

3.3.7.6. คุณภาพของสิ่งที่ปรากฏให้เห็นในเว็บไซต์ (Visual Appeal)

ลักษณะที่น่าสนใจของเว็บไซต์นั้น ขึ้นอยู่กับความชอบส่วนบุคคลเป็นสำคัญ แต่โดยรวมแล้วก็สามารถสรุปได้ว่าเว็บไซต์ที่น่าสนใจนั้นส่วนประกอบต่าง ๆ ควรมีคุณภาพ เช่น กราฟิกควรสมบูรณ์ไม่มีรอยหรือขอบขั้นบันไดให้เห็น ชนิดตัวอักษรอ่านง่ายสบายตา มีการเลือกใช้โทนสีที่เข้ากันอย่างสวยงาม เป็นต้น

3.3.7.7. ความสะดวกของการใช้ในสภาพต่าง ๆ (Compatibility)

การใช้งานของเว็บไซต์นั้นไม่ควรมีข้อจำกัด กล่าวคือ ต้องสามารถใช้งานได้ดีในสภาพแวดล้อมที่หลากหลาย ไม่มีการบังคับให้ผู้ใช้ต้องติดตั้งโปรแกรมอื่นใดเพิ่มเติม นอกเหนือจากเว็บเบราว์เซอร์ ควรเป็นเว็บที่แสดงผลได้ดีในทุกระบบปฏิบัติการ สามารถแสดงผลได้ในทุกความละเอียดหน้าจอ ซึ่งหากเป็นเว็บไซต์ที่มีผู้ใช้บริการมากและกลุ่มเป้าหมายหลากหลายควรให้ความสำคัญกับเรื่องนี้ให้มาก

3.3.7.8. ความคงที่ในการออกแบบ (Design Stability)

ถ้าต้องการให้ผู้ใช้งานรู้สึกว่าคุณภาพ ถูกต้อง และเชื่อถือได้ ควรให้ความสำคัญกับการออกแบบเว็บไซต์เป็นอย่างมาก ต้องออกแบบวางแผนและเรียบเรียงเนื้อหาอย่างรอบคอบ ถ้าเว็บที่จัดทำขึ้นอย่างลวก ๆ ไม่มีมาตรฐานการออกแบบและระบบการจัดการข้อมูล ถ้ามีปัญหามากขึ้นอาจส่งผลให้เกิดปัญหาและทำให้ผู้ใช้หมดความเชื่อถือ

3.3.7.9. ความคงที่ของการทำงาน (Function Stability)

ระบบการทำงานต่าง ๆ ในเว็บไซต์ควรมีความถูกต้องแน่นอน ซึ่งต้องได้รับการออกแบบสร้างสรรค์และตรวจสอบอยู่เสมอ ตัวอย่างเช่น ลิงค์ต่าง ๆ ในเว็บไซต์ ต้องตรวจสอบว่ายังสามารถลิงค์ข้อมูลได้ถูกต้องหรือไม่ เพราะเว็บไซต์อื่นอาจมีการเปลี่ยนแปลงได้ตลอดเวลา ปัญหาที่เกิดจากลิงค์ ก็คือ ลิงค์ขาด ซึ่งพบได้บ่อยเป็นปัญหาที่สร้างความรำคาญกับผู้ใช้เป็นอย่างมาก

3.3.8. ขั้นตอนการสร้างเว็บไซต์

3.3.8.1. ขั้นตอนที่ 1 การวางแผนการจัดทำเว็บไซต์

เป็นขั้นตอนแรกของการออกแบบเว็บ เนื่องจากเราต้องกำหนดชื่อเรื่อง เนื้อหา และรายละเอียดของเว็บที่เราจะจัดทำเพื่อให้เห็นมุมมองคร่าว ๆ ก่อนจะลงมือสร้างเว็บไซต์ นอกจากนี้เรายังต้องทำการแบ่งเนื้อหาเป็นหมวดหมู่ต่าง ๆ ตามลำดับก่อน-หลัง เพื่อให้ง่ายต่อการจัดทำโครงสร้างของเว็บ

3.3.8.2. ขั้นตอนที่ 2 การกำหนดโครงสร้างของเว็บ

เป็นขั้นตอนในการกำหนดผังของเว็บ เพื่อให้ทราบองค์ประกอบทั้งหมดของเว็บ

- หน้าแรกของเว็บ หรือโฮมเพจ จะต้องชื่อ index ส่วนนามสกุลให้ใส่ตามลักษณะของภาษาที่ใช้ในการสร้างเว็บ

- การตั้งชื่อเว็บเพจแต่ละหน้าเวลาให้กำหนดชื่อเป็นภาษาอังกฤษตามด้วยนามสกุลของภาษาที่เราสร้างเว็บ เช่น index.html, home.html, history.html เป็นต้น

3.3.8.3. ขั้นตอนที่ 3 การกำหนดการเชื่อมโยงเว็บเพจ

การกำหนดการเชื่อมเว็บเพจ เป็นการกำหนดความสัมพันธ์ของการเชื่อมโยงในแต่ละหน้าเว็บ เพื่อให้สามารถกลับไปกลับมาระหว่างหน้าต่าง ๆ ได้ โดยแต่ละไฟล์จะมีความสัมพันธ์กัน ตัวอย่างดังรูป

3.3.8.4. ขั้นตอนที่ 4 การตั้งชื่อไฟล์และโฟลเดอร์

การสร้างโฟลเดอร์

การสร้างโฟลเดอร์ให้สร้างเป็นชื่อหน่วยงาน / เรื่องนั้น ๆ ควรใช้ตัวอักษร ภาษาอังกฤษตัวพิมพ์เล็ก หรือผสมกับตัวเลข 0-9 เช่น swt คือ โรงเรียนเสริมงามวิทยาคม จากนั้นข้างในโฟลเดอร์ swt ให้เราสร้างโฟลเดอร์เก็บรูปภาพ พื้นหลัง ไฟล์เสียง ไฟล์วิดีโอ หรือโฟลเดอร์อื่นเป็นชื่อภาษาอังกฤษ เช่น pic คือโฟลเดอร์เก็บรูปภาพ, bg คือ โฟลเดอร์เก็บพื้นหลัง เป็นต้น

การตั้งชื่อไฟล์

การตั้งชื่อไฟล์ให้ตั้งชื่อและนามสกุลไฟล์เป็นตัวอักษร ภาษาอังกฤษตัวพิมพ์เล็ก หรือผสมกับตัวเลข 0-9 หรือเครื่องหมายขีดลบ/ขีดล่าง และตั้งชื่อไฟล์ให้ตรงกับเรื่องนั้น ๆ เช่น history.html คือ ประวัติของโรงเรียน, person.html คือ บุคลากรของโรงเรียน เป็นต้น

3.3.8.5. ขั้นตอนที่ 5 การออกแบบเว็บเพจแต่ละหน้าในเว็บไซต์

ส่วนประกอบของเว็บไซต์ที่ดี

โครงสร้างของเว็บไซต์โดยจะมีส่วนหลักๆอยู่ด้วยกันทั้งหมด 3 ส่วนหลัก ๆ ดังนี้

3.3.8.5.1. ส่วนหัวของหน้า (Page Header)

เป็นส่วนที่อยู่ตอนบนสุดของหน้า และเป็นส่วนที่สำคัญที่สุดของหน้า เพราะเป็นส่วนที่ดึงดูดผู้ชมให้ติดตามเนื้อหาภายในเว็บไซต์ มักใส่ภาพกราฟิกเพื่อสร้างความประทับใจ ส่วนใหญ่ประกอบด้วย

- โลโก้ (Logo) เป็นสิ่งที่เว็บไซต์ควรมี เป็นตัวแทนของเว็บไซต์ได้เป็นอย่างดี และยังทำให้เว็บน่าเชื่อถือ

- ชื่อเว็บไซต์
- เมนูหลักหรือลิงค์ (Navigation Bar) เป็นจุดเชื่อมโยงไปสู่เนื้อหาของเว็บไซต์

3.3.8.5.2. ส่วนของเนื้อหา (Page Body)

เป็นส่วนที่อยู่ตอนกลางของหน้า ใช้แสดงข้อมูลเนื้อหาของเว็บไซต์ ซึ่งประกอบด้วย ข้อความ, ตารางข้อมูล ภาพกราฟิก วิดีโอ และอื่นๆ และอาจมีเมนูหลัก หรือเมนูเฉพาะกลุ่มวางอยู่ในส่วนนี้ด้วย สำหรับส่วนเนื้อหาควรแสดงใจความสำคัญที่เป็นหัวเรื่องไว้บนสุด ข้อมูลมีความกระชับ ใช้รูปแบบตัวอักษรที่อ่านง่าย และจัด Layout ให้เหมาะสมและเป็นระเบียบ

3.3.8.5.3. ส่วนท้ายของหน้า (Page Footer)

เป็นส่วนที่อยู่ด้านล่างสุดของหน้า มักวางระบบนำทางที่เป็นลิงค์ข้อความง่าย ๆ และอาจแสดงข้อมูลเพิ่มเติมเกี่ยวกับเนื้อหาภายในเว็บไซต์ เช่น เจ้าของเว็บไซต์, ข้อความแสดงลิขสิทธิ์, วิธีการติดต่อกับผู้ดูแลเว็บไซต์, คำแนะนำการใช้เว็บไซต์ เป็นต้น โดยปกติส่วนหัวและส่วนท้ายมักแสดงเหมือนกันในทุกหน้าของเว็บเพจ

3.3.8.6. ขั้นตอนที่ 6 การสร้างเว็บเพจ

เมื่อจัดวางองค์ประกอบของเว็บแต่ละหน้าแล้ว ต่อไปคือขั้นตอนการเขียนเว็บด้วยโปรแกรมภาษา HTML เพื่อกำหนดให้แต่ละหน้าเว็บเพจนำเสนอข้อความ รูปภาพ วิดีโอ และเสียง ให้อยู่ในรูปแบบการที่ต้องการ

3.3.8.7. ขั้นตอนที่ 7 การลงทะเบียนขอพื้นที่เว็บไซต์

เมื่อทำการออกแบบและสร้างเว็บไซต์เสร็จแล้ว ขั้นตอนต่อไป คือ การเผยแพร่เว็บไซต์สู่โลกของอินเทอร์เน็ตให้คนอื่นเข้ามาเยี่ยมชม วิธีการ คือ การนำเว็บไซต์ไปฝากกับผู้ให้บริการพื้นที่เว็บไซต์ทั้งแบบเสียค่าใช้จ่ายหรือบริการ และแบบพื้นที่เว็บไซต์ฟรี

3.3.8.8. ขั้นตอนที่ 8 การอัปโหลดเว็บไซต์

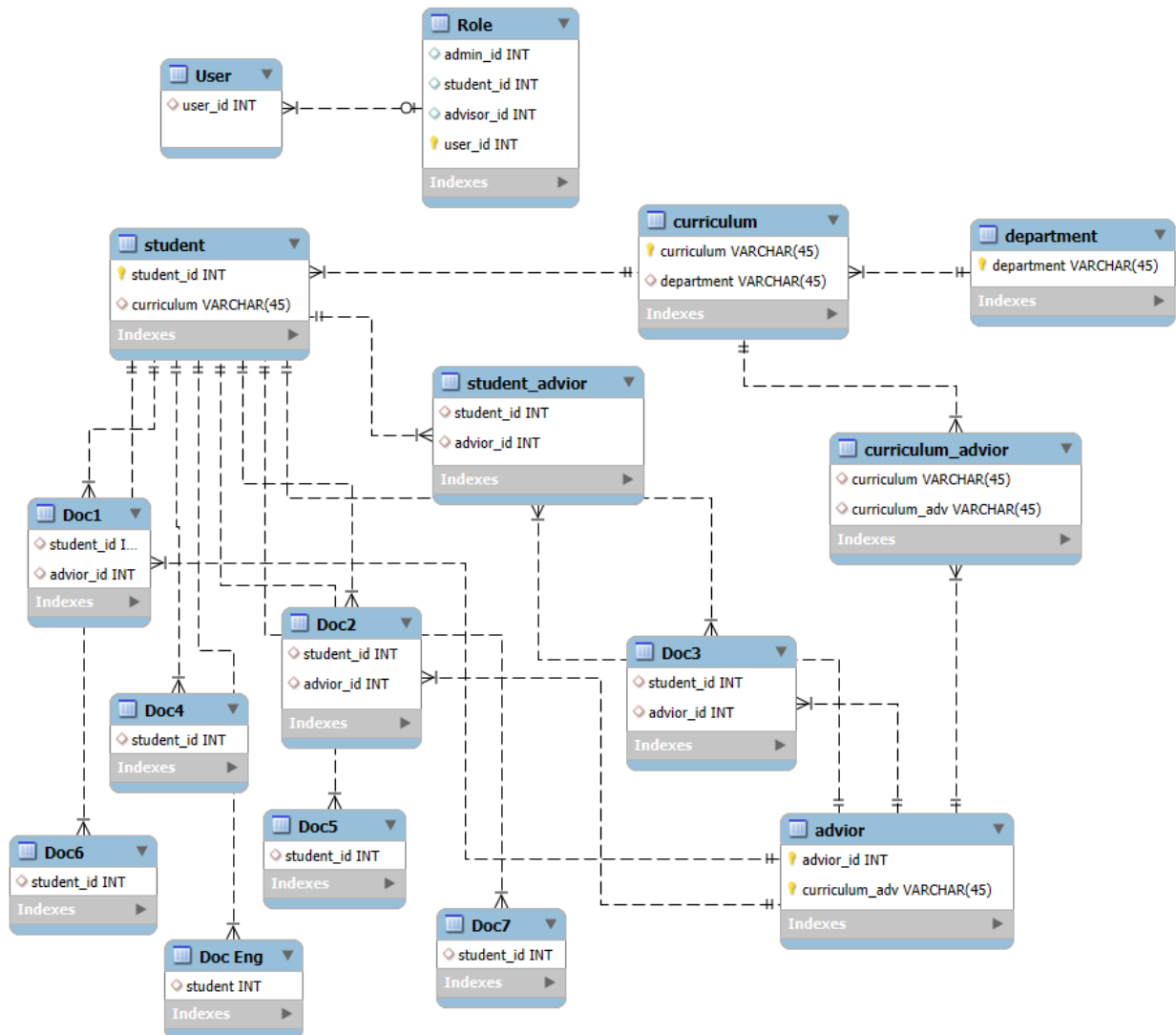
เมื่อเราทำการสมัครบริการพื้นที่ฝากเว็บแล้ว ขั้นตอนต่อไป คือการอัปโหลดไฟล์เว็บไซต์ของเราไปยังเว็บไซต์ที่ให้บริการพื้นที่ฝากเว็บซึ่งอาจจะทำการอัปโหลดผ่านเว็บเบราว์เซอร์เว็บที่ให้บริการ หรือการอัปโหลดด้วยโปรแกรม เช่น CuteFTP, Filezilla, WS_FTP เป็นต้น เพื่อให้ผู้คนได้เข้าเยี่ยมชมโดยสามารถดูในเว็บไซต์ของเราผ่านเครือข่ายอินเทอร์เน็ต

3.3.8.9. ขั้นตอนที่ 2 การเรียกดูเว็บไซต์

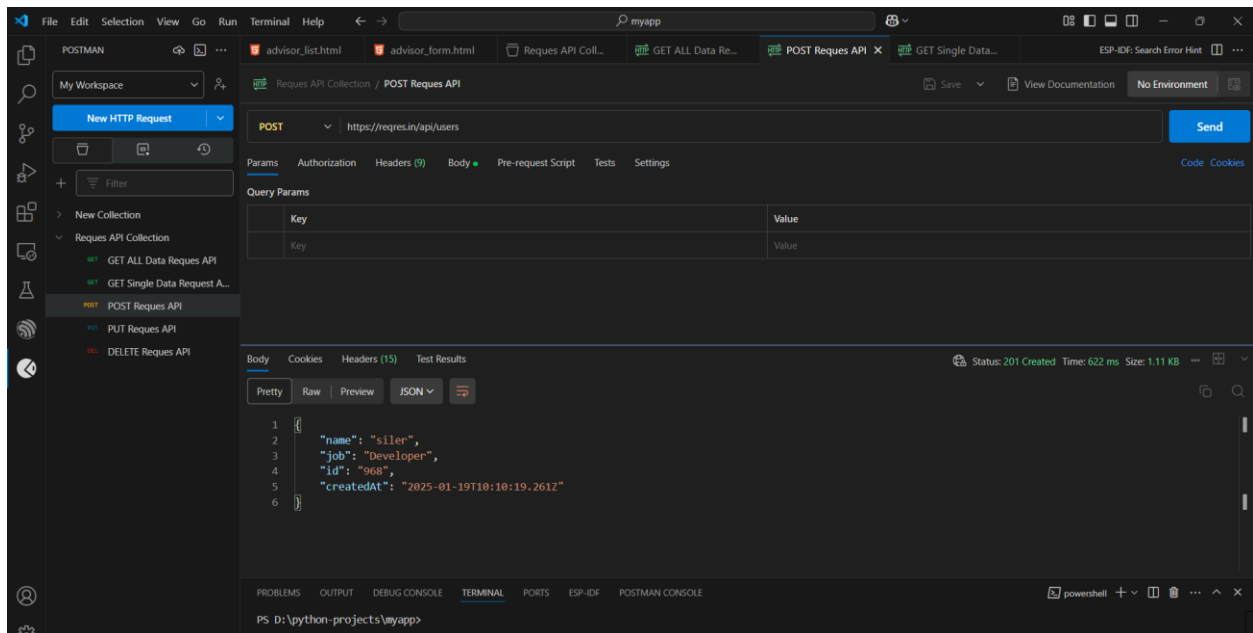
เมื่อเราทำการอัปโหลดไฟล์เว็บไซต์ของเราขึ้นบนเว็บไซต์ที่ให้บริการพื้นที่ฝากเว็บแล้ว เราสามารถเปิดดูเว็บไซต์ของเราผ่านโปรแกรมเว็บเบราว์เซอร์ เช่น Internet Explorer, Mozilla Firefox, Google Chrome เป็นต้น โดยการพิมพ์ที่อยู่เว็บไซต์ตรง Address Bar เป็นต้น

3.4. ส่วนการออกแบบฐานข้อมูลของเว็บแอปพลิเคชันติดตามสถานะนักศึกษาในระดับบัณฑิตศึกษา

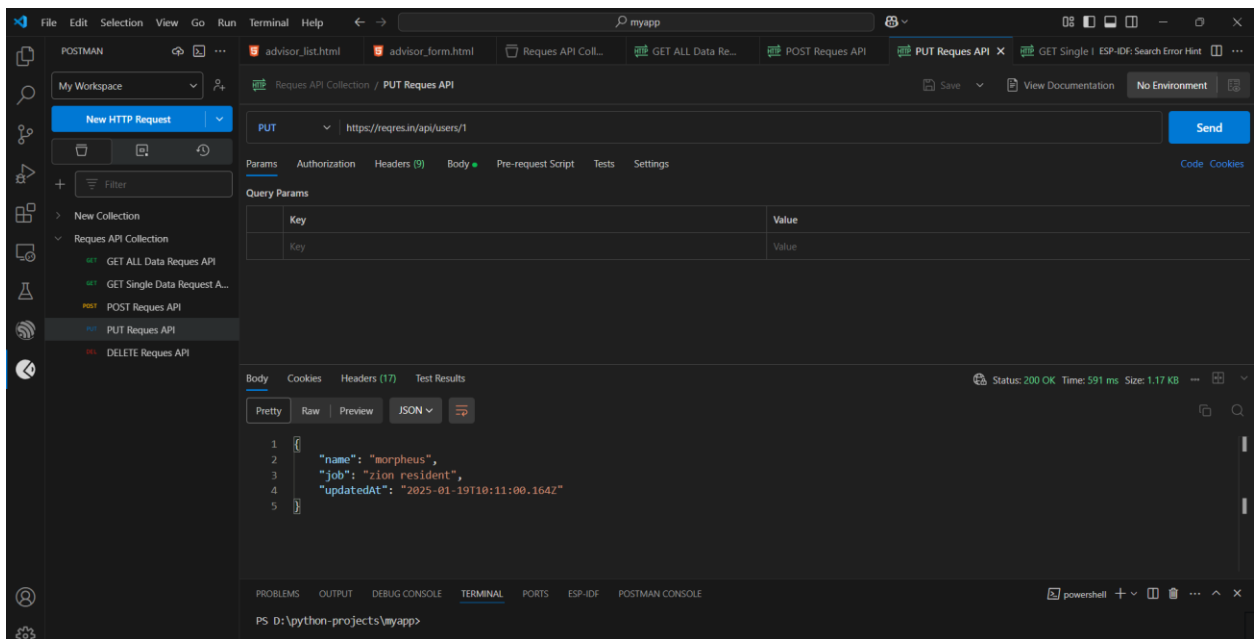
3.5. แก๊ไข Diagram Database



3.6 ทดลองการใช้งาน API ใน Postman



- ส่วนของแก้ไขข้อมูล



- ส่วนของการลบข้อมูล

POSTMAN

File Edit Selection View Go Run Terminal Help

myapp

My Workspace

New HTTP Request

Filter

New Collection

Reques API Collection

- GET GET ALL Data Reques API
- GET GET Single Data Request A...
- POST POST Reques API
- PUT PUT Reques API
- DELETE DELETE Reques API

Reques API Collection / DELETE Reques API

DELETE | https://reqres.in/api/users/1

Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Query Params

Key	Value
Key	Value

Body Cookies Headers (14) Test Results

Status: 204 No Content Time: 620 ms Size: 1010 B

Pretty Raw Preview Text

1

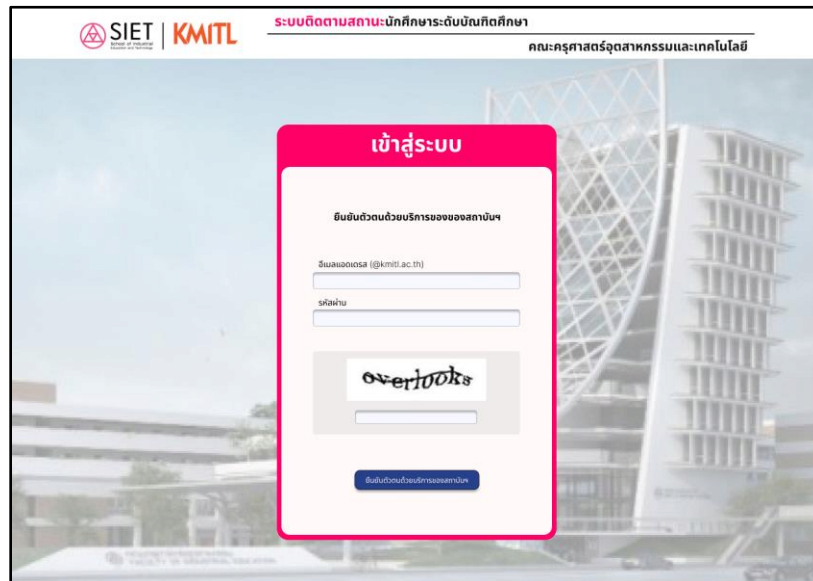
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS ESP-IDF POSTMAN CONSOLE

PS D:\python-projects\myapp>

3.5. การออกแบบ UX/UI ของเว็บไซต์ และหน้า Report ที่ให้ user ใช้งาน

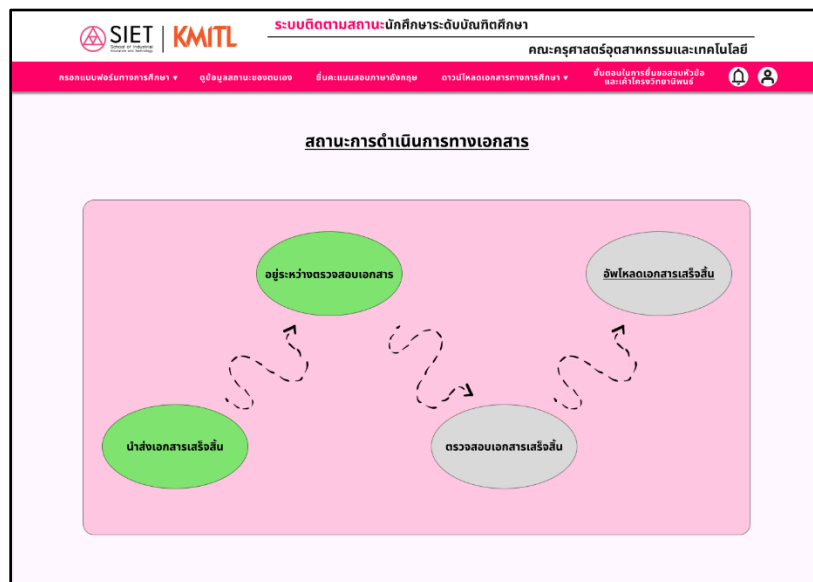
3.5.1. หน้าต่างการทำงานของนักศึกษาระดับบัณฑิตศึกษา

- หน้าต่างการล็อกอิน



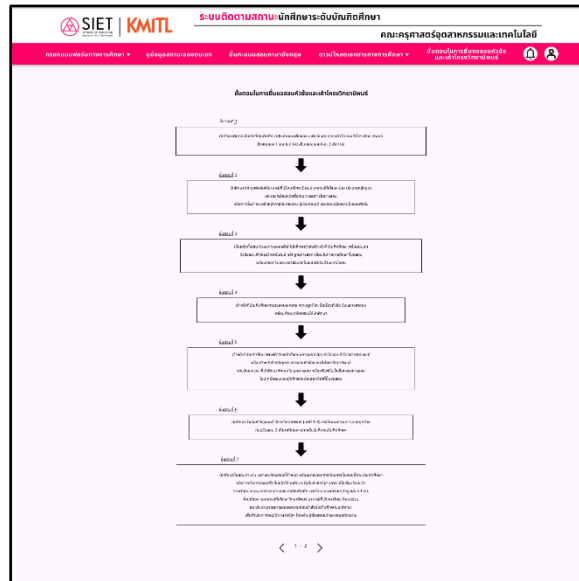
รูปที่ 1

- หน้าผู้ใช้งานหลัก



รูปที่ 2

- หน้าขั้นตอนในการยื่นสอบหัวข้อและเค้าโครงวิทยานิพนธ์



รูปที่ 3

- หน้าข้อมูลส่วนตัว

The screenshot shows a web interface with a header for "SIET | KMUTL" and "ระบบติดตามสถานะนักศึกษาในระดับบัณฑิตศึกษา" (Student Status Tracking System for Graduate Level). The main content area is titled "ข้อมูลส่วนตัว" (Personal Information) and contains a form with the following fields:

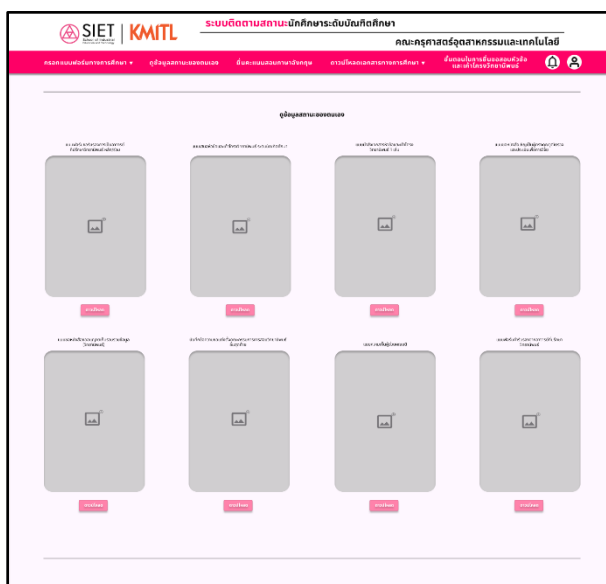
- รหัสประจำตัว (ID Number)
- รหัสประจำตัวประชาชน (National ID Number)
- คำนำหน้าชื่อ (Prefix)
- ชื่อ - สกุล (First and Last Name)
- คำนำหน้าชื่อ (ENI) (Prefix - ENI)
- ชื่อ - สกุล (ENI) (First and Last Name - ENI)
- วัน / เดือน / ปี เกิด (Date of Birth)
- เพศ (Gender)
- สถานภาพนักศึกษา (Student Status)
- ประเภทการรับเข้า (Admission Type)
- ปีเข้าศึกษา (Year of Admission)
- คณะ (Faculty)
- สาขาวิชา (Major)

รูปที่ 4

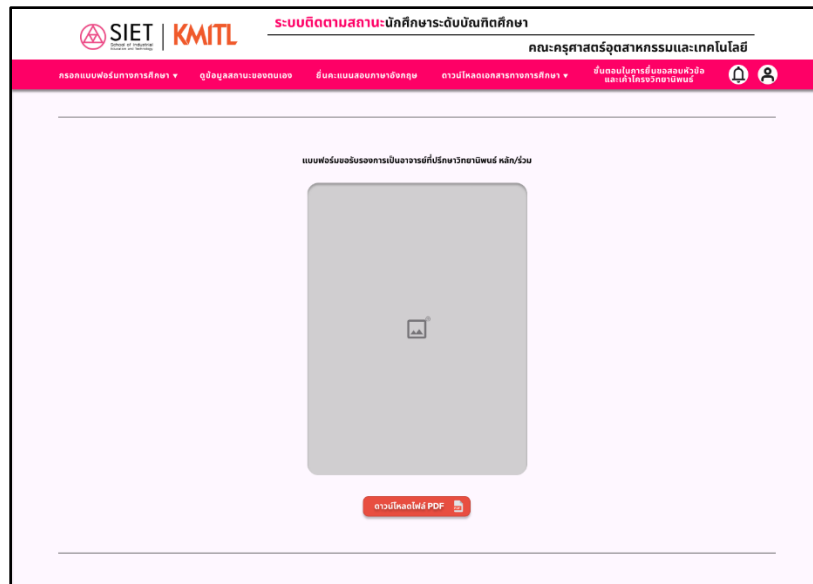
- ตัวอย่างหน้าการกรอกแบบฟอร์มทางการศึกษาแบบออนไลน์

รูปที่ 5

- ตัวอย่างหน้าการดาวน์โหลดเอกสารทางการศึกษาแบบสมบูรณ์



รูปที่ 6.1

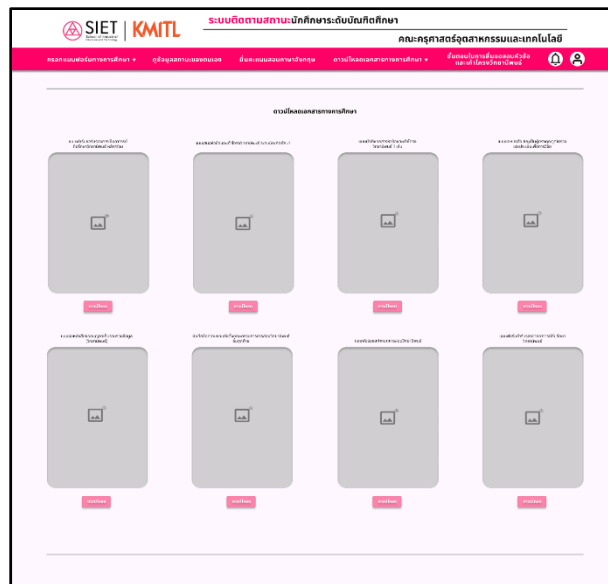


รูปที่ 6.2

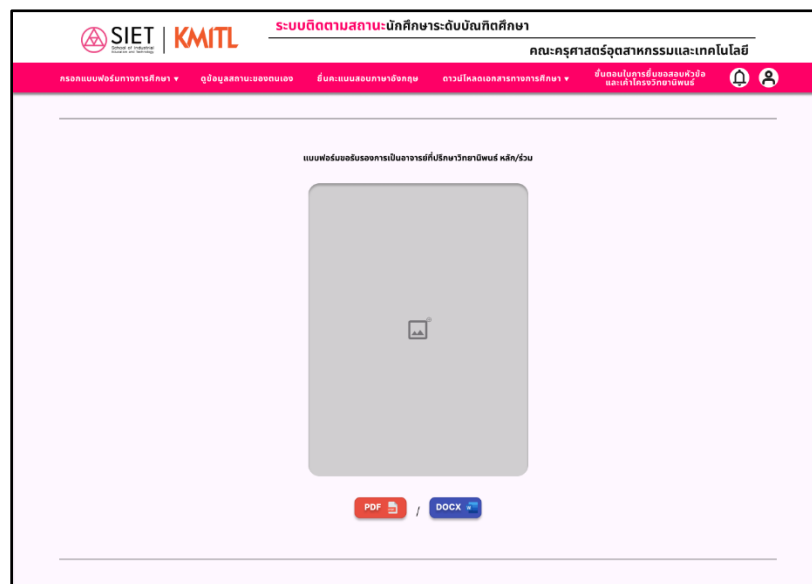
- หน้าต่างการกรอกแบบฟอร์มสอบวัดผลภาษาอังกฤษ

รูปที่ 7

- หน้าต่างการดาวน์โหลดเพิ่มแพลตฟอร์มเอกสารทางการศึกษา



รูปที่ 8.1



รูปที่ 8.2

- หน้าต่างการล๊อคอิน



รูปที่ 10

- หน้ารายชื่อนักศึกษาระดับบัณฑิตศึกษาที่รับผิดชอบ


รหัสผู้ศึกษา	ชื่อ-นามสกุล	ชื่อวิทยานิพนธ์	
65030130	นายณัฐภูมิ วัฒนศิริ	นวัตกรรมการพัฒนาระบบการจัดการเรียนการสอน...	
65030179	นายวิเศษ วัฒนศิริ	นวัตกรรมการพัฒนาระบบการจัดการเรียนการสอน...	
65030226	นายวิเศษ วัฒนศิริ	นวัตกรรมการพัฒนาระบบการจัดการเรียนการสอน...	
65030278	นายณัฐภูมิ วัฒนศิริ	นวัตกรรมการพัฒนาระบบการจัดการเรียนการสอน...	

รูปที่ 11.1


ชื่อผู้ศึกษา	ชื่ออาจารย์	สาขาวิชา	ปีการศึกษา	ชื่อวิทยานิพนธ์	ชื่ออาจารย์ที่ปรึกษา
นายณัฐภูมิ วัฒนศิริ	นายวิเศษ วัฒนศิริ	วิศวกรรมศาสตร์	2567	นวัตกรรมการพัฒนาระบบการจัดการเรียนการสอน...	นายวิเศษ วัฒนศิริ

รูปที่ 11.2

- หน้าข้อมูลการสอบวิทยานิพนธ์ขั้นสุดท้ายของนักศึกษาระดับบัณฑิตศึกษาที่รับผิดชอบ





SIET
SAKON NAKHON SIAM ENGINEERING TECHNOLOGY



ระบบติดตามสถานะนักศึกษาในระดับบัณฑิตศึกษา
 คณะครุศาสตร์อุตสาหกรรมและเทคโนโลยี

รายชื่อบัณฑิตที่รับต่อเนื่อง
 [ข้อมูลการสอบของบัณฑิตที่รับต่อเนื่อง](#)
[ดาวน์โหลดเอกสารทางการศึกษา](#)

ข้อมูลการสอบวิทยานิพนธ์ชั้นสูงทางของบัณฑิตที่รับต่อเนื่อง

รหัสบัณฑิต

 ชื่อ-นามสกุล

 มีที่งานอยู่

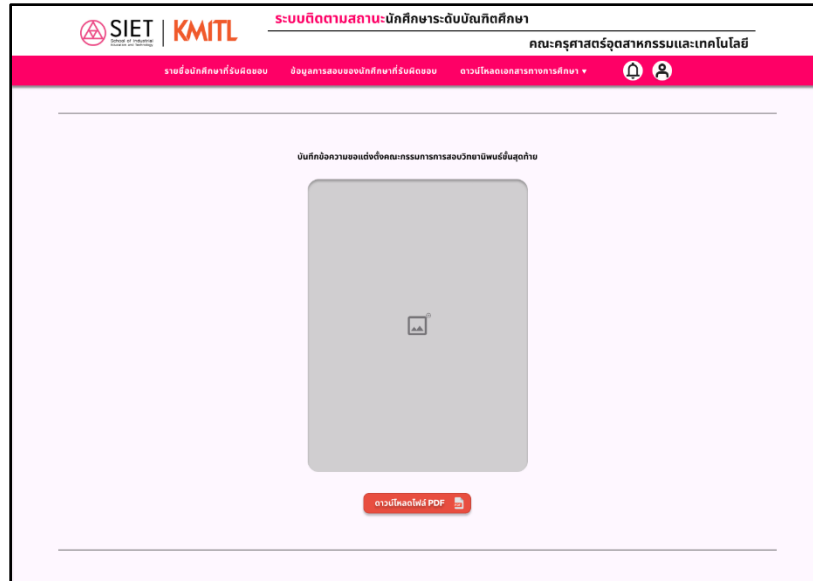
 หลักสูตร

รหัสบัณฑิต	ชื่อ-นามสกุล	มีที่งานอยู่	วันที่สอบ
65030130	นายณัฐชัย อินทร์คำ	ไม่พบข้อมูล	03 / พ.ค. / 67
65030179	นายศุภกร พลสุข	ไม่พบข้อมูล	07 / พ.ค. / 67
65030226	นายธีรธร น้อยพันธ์	ไม่พบข้อมูล	09 / พ.ค. / 67
65030276	นางธนพิชญ์ รัตนกิจ	ไม่พบข้อมูล	11 / พ.ค. / 67

รูปที่ 12.1

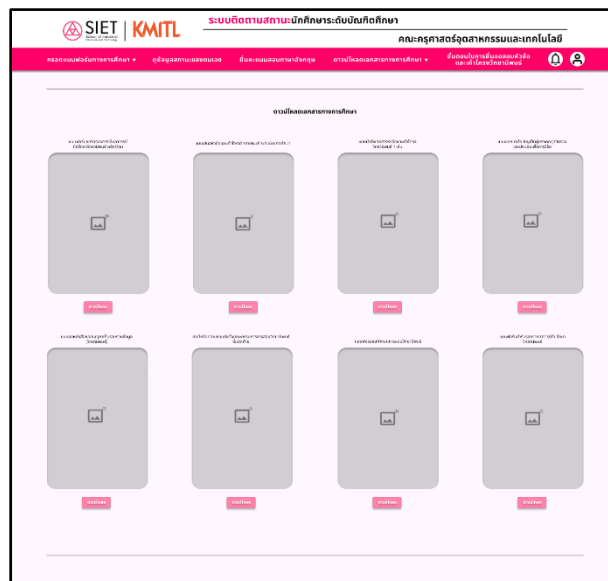
[illegible]

รูปที่ 12.2

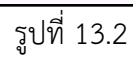


รูปที่ 12.3

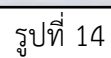
- หน้าต่างการดาวน์โหลดเพิ่มแฟ้มเอกสารทางการศึกษา



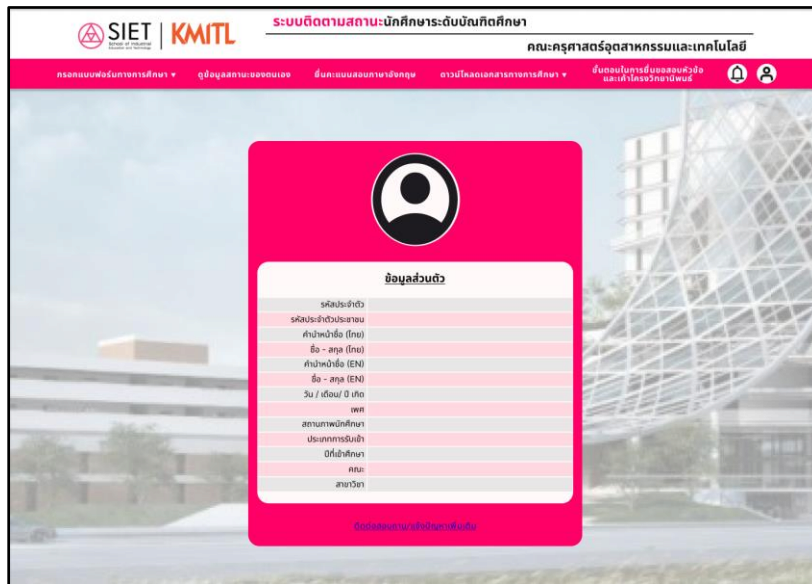
รูปที่ 13.1



- หน้าต่างการล๊อคอิน

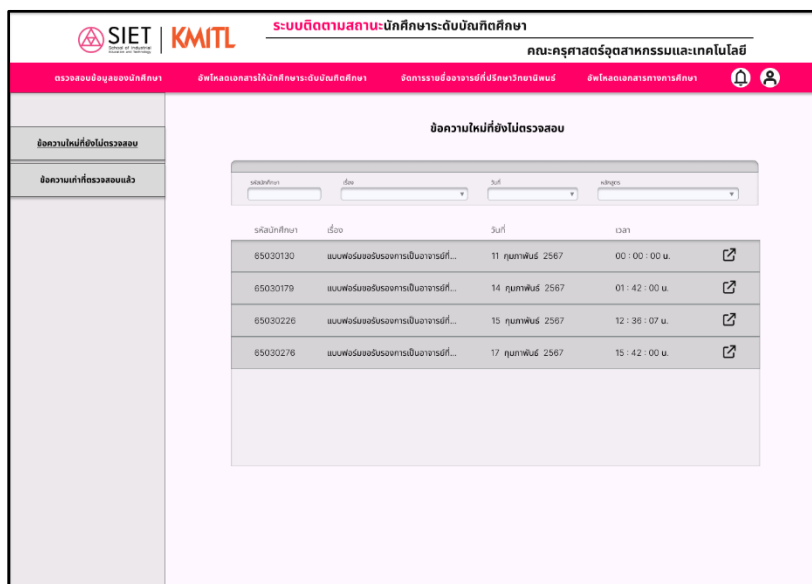


- หน้าข้อมูลส่วนตัว

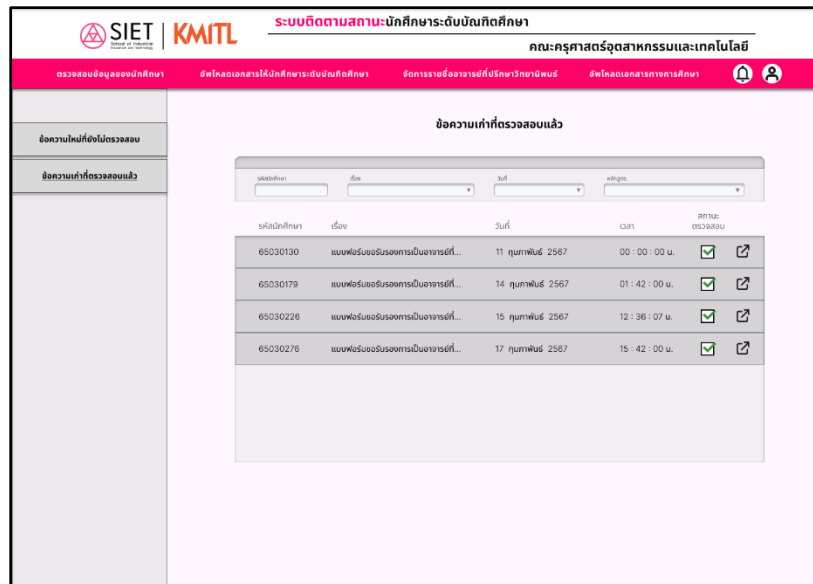


รูปที่ 15

- หน้าผู้ใช้งานหลัก

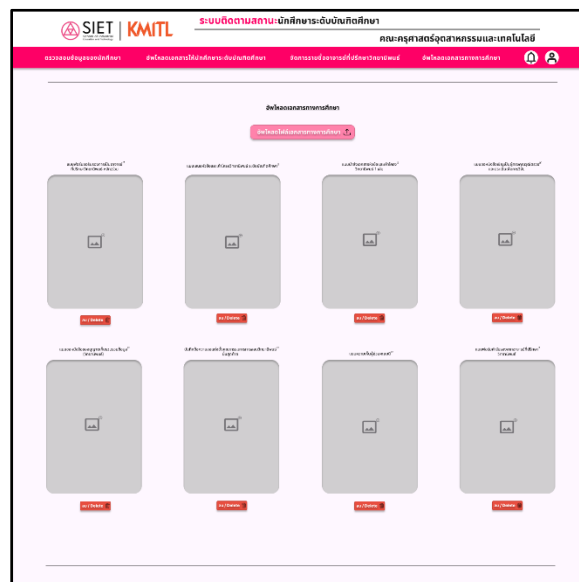


รูปที่ 16.1



รูปที่ 16.2

- หน้าอัปโหลดเพิ่มแฟลตเอกสารทางการศึกษา



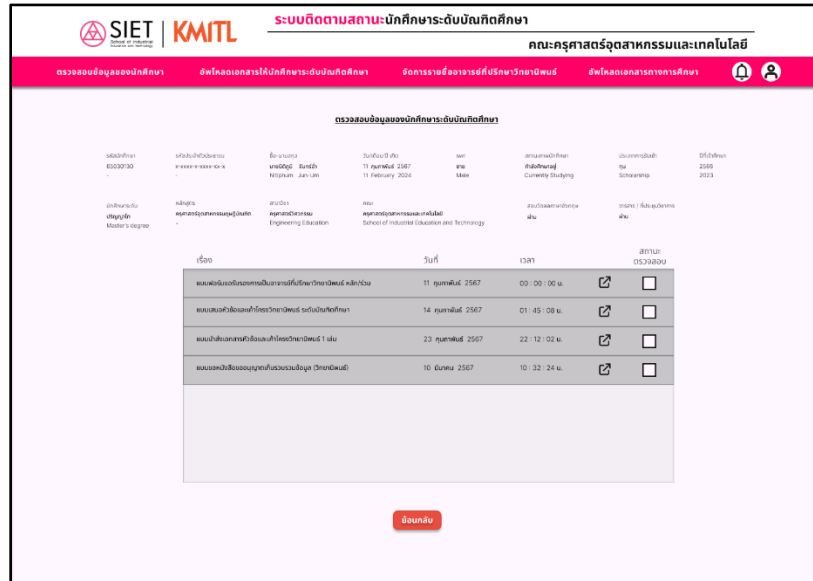
รูปที่ 17.1

รูปที่ 17.2

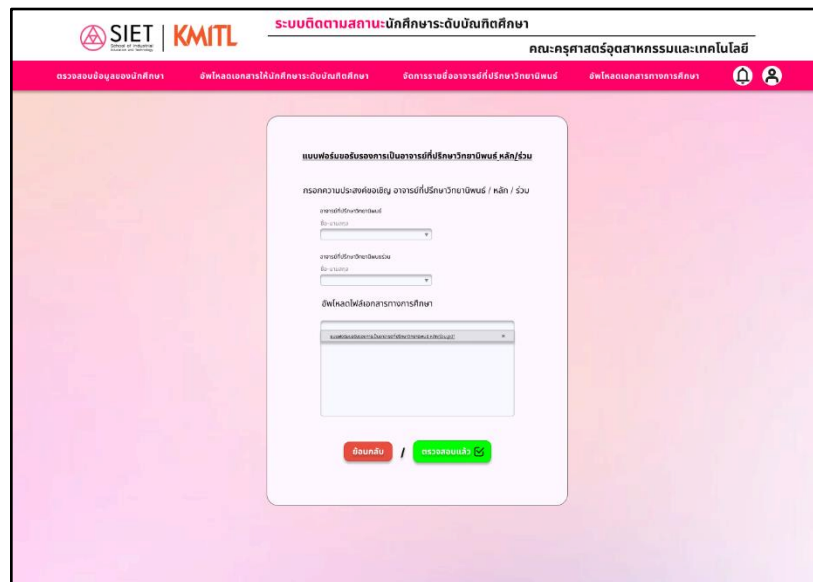
- หน้าตรวจสอบข้อมูลของนักศึกษาระดับบัณฑิตศึกษา

รหัสนักศึกษา	ชื่อ-นามสกุล	วันที่	เวลา	
65030130	นายธิติภูมิ จันทะวี	11 กุมภาพันธ์ 2567	00 : 00 : 00 น.	✎
65030179	นายทิชพล พูลสูง	14 กุมภาพันธ์ 2567	01 : 42 : 00 น.	✎
65030228	นายวิธกร น้อยพันธ์	15 กุมภาพันธ์ 2567	12 : 36 : 07 น.	✎
65030276	นายณพัทธ์ ชัยมณี	17 กุมภาพันธ์ 2567	15 : 42 : 00 น.	✎

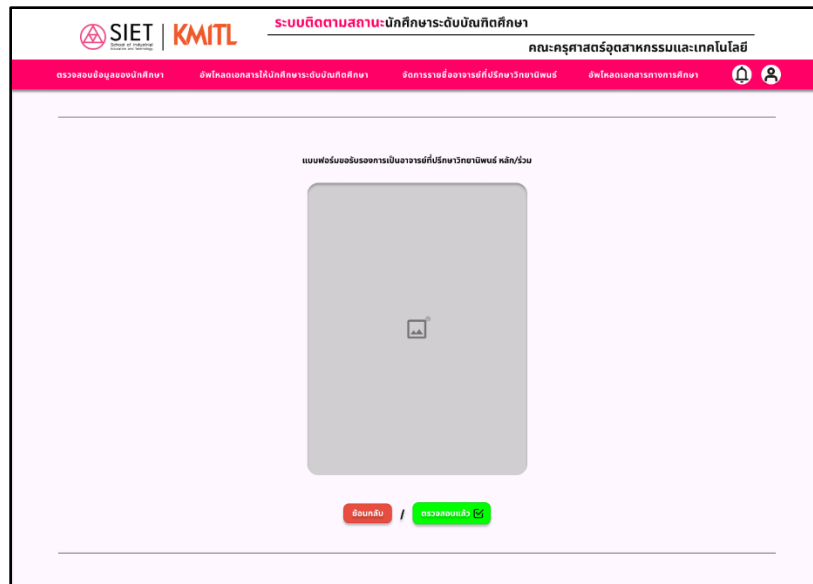
รูปที่ 18.1



รูปที่ 18.2

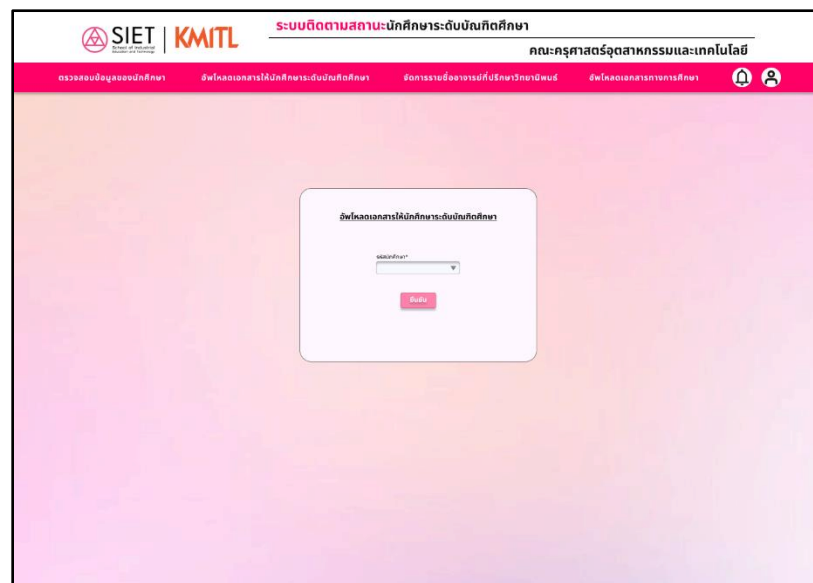


รูปที่ 18.3

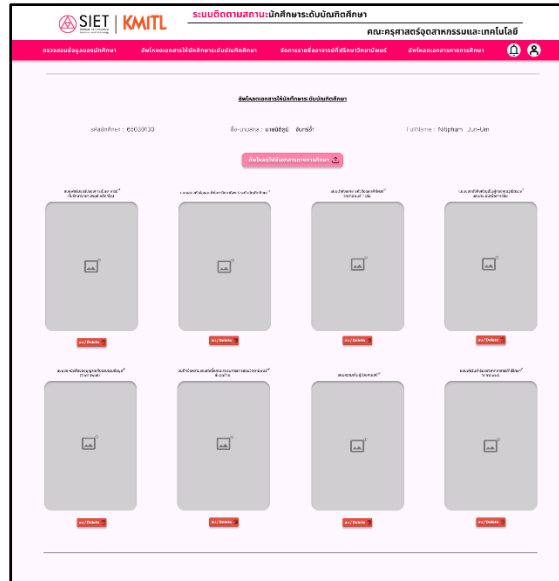


รูปที่ 18.4

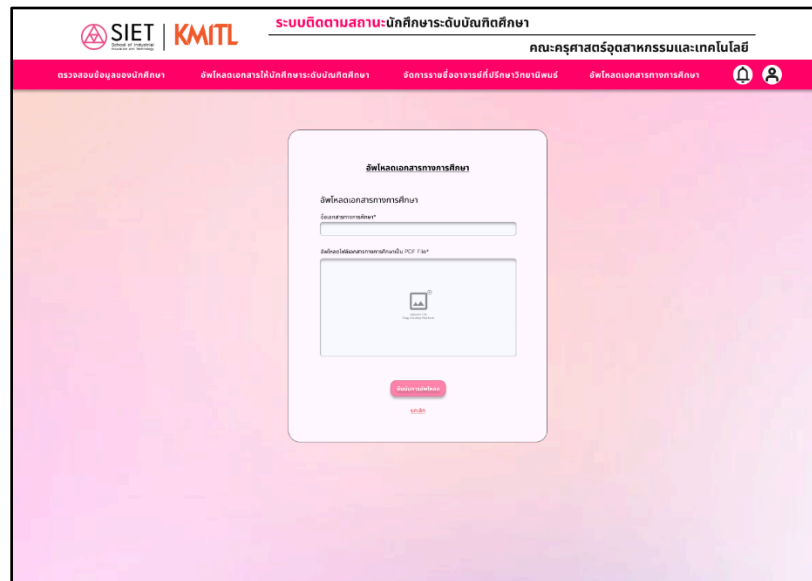
- หน้าตรวจสอบข้อมูลของนักศึกษาระดับบัณฑิตศึกษา



รูปที่ 19.1



รูปที่ 19.2



รูปที่ 19.3

The screenshot shows a web interface for SIET and KMITL. The main content area displays a date selection calendar for June 2024. The date '26' is highlighted. Below the calendar, there is a time selection interface showing '8:00 PM'. At the bottom, there is a button labeled 'เลือกเวลา' (Select Time).

รูปที่ 19.4

- หน้ารายชื่ออาจารย์ที่ปรึกษาวิทยานิพนธ์

The screenshot shows a web interface for SIET and KMITL. The main content area displays a list of advisors for a thesis. The list includes the following information for each advisor:

- ตำแหน่ง: ผศ.
- ชื่อ: สุระชัย
- นามสกุล: พิมพ์ลภี
- ประจำภาควิชา: ภาควิชาวิศวกรรม
- ที่ปรึกษาประจำหลักสูตรปริญญาโท: ภา.บ. คณะวิศวกรรมศาสตร์, ภา.บ. ภาควิชาวิศวกรรมและเทคโนโลยี
- ที่ปรึกษาประจำหลักสูตรปริญญาเอก: ภา.บ. คณะวิศวกรรมศาสตร์, ภา.บ. ภาควิชาวิศวกรรมและเทคโนโลยี
- รายชื่อภาควิชาในระดับบัณฑิตศึกษาที่รับผิดชอบ: 650.301.30 ภาควิชาวิศวกรรมเครื่องกล, 650.301.79 ภาควิชาวิศวกรรมเครื่องกล

รูปที่ 20

- หน้าการเพิ่มรายชื่ออาจารย์ที่ปรึกษาวิทยานิพนธ์

[illegible]

รูปที่ 21

- หน้าการแก้ไขรายชื่ออาจารย์ที่ปรึกษาวิทยานิพนธ์

[illegible]

รูปที่ 22

- หน้าการจัดการรายชื่อภาควิชา / หลักสูตร

The screenshot shows a web application interface for SIET KMITL. The header includes the SIET and KMITL logos, the text 'ระบบติดตามสถานะนักศึกษาในระดับบัณฑิตศึกษา' (Graduate Student Status Tracking System), and the user's name 'คณะครุศาสตร์อุตสาหกรรมและเทคโนโลยี' (Faculty of Industrial and Technological Education). The navigation bar has four items: 'ตรวจสอบข้อมูลของนักศึกษา' (Check Student Information), 'เพิ่ม/ลบ/แก้ไขภาควิชา-ระดับบัณฑิตศึกษา' (Add/Delete/Edit Department-Graduate), 'จัดการรายชื่อนักเรียนที่ปรึกษาวิทยานิพนธ์' (Manage Student Advisor List), and 'จัดโดยเอกสารทางการศึกษา' (Organized by Educational Documents). The sidebar on the left has five items: 'รายชื่อภาควิชาที่ศึกษาภาควิชา' (Department List), 'เพิ่มรายชื่อภาควิชาที่ศึกษาภาควิชา' (Add Department List), 'แก้ไขรายชื่อภาควิชาที่ศึกษาภาควิชา' (Edit Department List), and 'จัดการรายชื่อภาควิชา / หลักสูตร' (Manage Department / Course). The main area is titled 'เพิ่มรายชื่อภาควิชา / หลักสูตร' (Add Department / Course) and contains a form with fields for 'ภาควิชา' (Department), 'ระดับปริญญา' (Degree Level), and 'ชื่อหลักสูตร' (Course Name), with a 'เพิ่ม' (Add) button. The right panel shows a table of existing departments and courses.

ชื่อภาควิชา	หลักสูตร
ภาควิชาศึกษาศาสตร์ / ศึกษาศาสตร์	ปริญญาตรี ค.บ.บ. การศึกษาระดับปริญญาตรี / ค.บ.บ. การศึกษาระดับปริญญาตรี / ค.บ.บ. การศึกษาระดับปริญญาตรี / ค.บ.บ. การศึกษาระดับปริญญาตรี /
ภาควิชาศึกษาศาสตร์ / ศึกษาศาสตร์	ปริญญาตรี ค.บ.บ. การศึกษาระดับปริญญาตรี / ค.บ.บ. การศึกษาระดับปริญญาตรี / ค.บ.บ. การศึกษาระดับปริญญาตรี /
ภาควิชาศึกษาศาสตร์ / ศึกษาศาสตร์	ปริญญาตรี ค.บ.บ. การศึกษาระดับปริญญาตรี / ค.บ.บ. การศึกษาระดับปริญญาตรี / ค.บ.บ. การศึกษาระดับปริญญาตรี /
ภาควิชาศึกษาศาสตร์ / ศึกษาศาสตร์	ปริญญาตรี ค.บ.บ. การศึกษาระดับปริญญาตรี / ค.บ.บ. การศึกษาระดับปริญญาตรี / ค.บ.บ. การศึกษาระดับปริญญาตรี /

รูปที่ 23

- Figma ที่ใช้ในการออกแบบ UX/UI

<https://shorturl.asia/uejOm>

4. แผนการปฏิบัติงานในสัปดาห์ต่อไป (Plans)

- แก้ไขการออกแบบหน้าการแสดงผล UX/UI ของเว็บแอปพลิเคชันติดตามสถานะนักเรียนระดับ
บัณฑิตศึกษา

- แก้ไขการออกแบบฐานข้อมูลของระบบ
- เขียนโค้ดหน้าการแสดงผลตามที่ได้ออกแบบไว้
- ออกแบบ Rest API ที่ต้องใช้ในเว็บแอปพลิเคชันติดตามสถานะนักเรียนระดับบัณฑิตศึกษา

5. บันทึกเพิ่มเติม (Open Issues)

ลงชื่อ อาจารย์ที่ปรึกษา

วันที่