

# SpendLog System

## 1. Teams

- นาย ทณพงษ์ รักษาภูมิ รหัสนักศึกษา 65116949
- นาย ภูมิพัฒน์ ม่วงทอง รหัสนักศึกษา 65106692
- นาย ธนภัทร ขุนเพชร รหัสนักศึกษา 65126930

## 2. Introduction (5 คะแนน, -0.25)

### a. Problem Statment

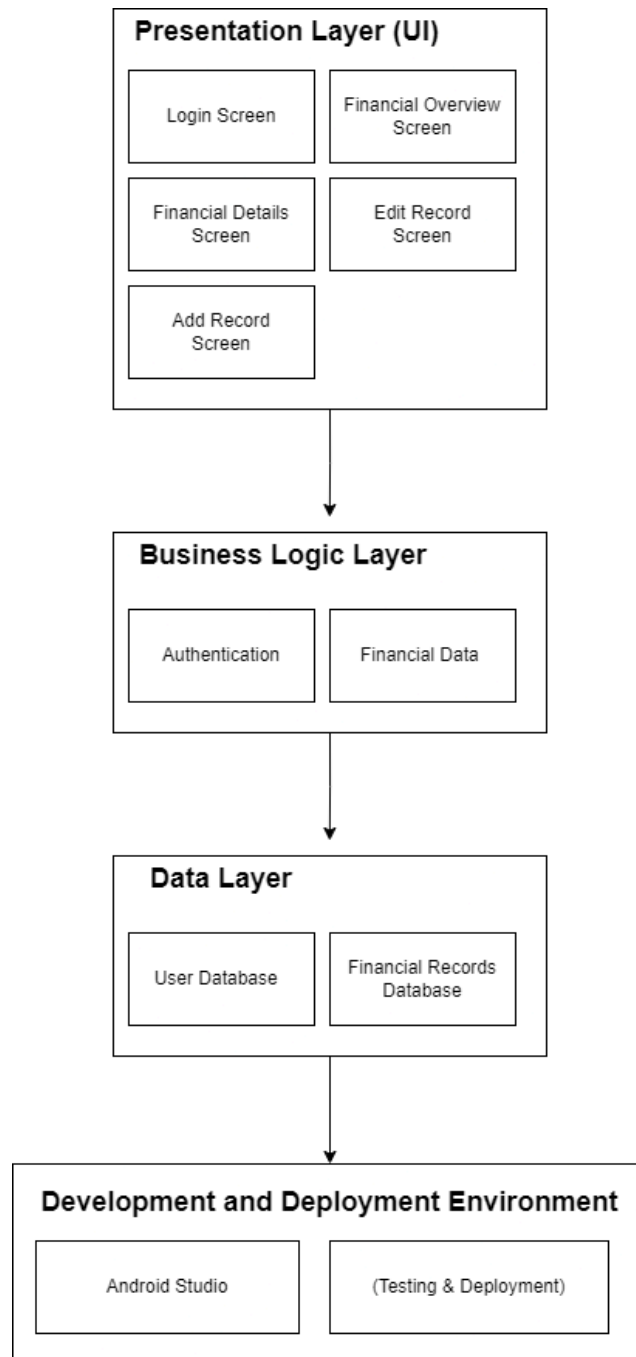
การจัดการการเงินส่วนบุคคลในปัจจุบันเป็นเรื่องที่สำคัญและท้าทายสำหรับหลาย ๆ คน เนื่องจากต้องการเครื่องมือที่สามารถช่วยติดตามรายรับ รายจ่าย และการออมได้อย่างมีประสิทธิภาพ แอปพลิเคชันจัดการการเงินที่ออกแบบมาเพื่อช่วยให้ผู้ใช้สามารถตรวจสอบสถานะการเงินของตนเองได้ง่ายและรวดเร็ว

### b. Objectives

1. พัฒนาแอปพลิเคชันจัดการการเงินส่วนบุคคล ที่สามารถติดตามรายรับ, รายจ่าย, และการออมได้อย่างสะดวกและรวดเร็ว เพื่อช่วยให้ผู้ใช้สามารถวางแผนการเงินได้อย่างมีประสิทธิภาพ

## 3. Design

### a. System Architecture (5 คะแนน, -0.25)



## 1. Presentation Layer (Frontend)

**หน้าจอเข้าสู่ระบบ (Login Screen)** เป็นหน้าจอแรกที่ใช้ต้องใส่ข้อมูลรับรองของตนเอง (อีเมลและรหัสผ่าน) เพื่อเข้าถึงแอปพลิเคชัน โดยสามารถเลือกเข้าสู่ระบบหรือสมัครสมาชิกได้

**หน้าจอสรุปการเงิน (Financial Overview Screen)** แสดงภาพรวมของข้อมูลการเงินของผู้ใช้ เช่น รายรับ, รายจ่าย, และการออม โดยจะแสดงเป็นกราฟหรือภาพเชิงลึกให้ผู้ใช้ดูง่ายขึ้น

**หน้าจอรายละเอียดการเงิน (Financial Details Screen)** แสดงรายละเอียดเกี่ยวกับรายการทางการเงิน เช่น รายรับ, รายจ่าย, และการออม ผู้ใช้สามารถเข้าถึงการแก้ไขหรือเพิ่มรายการจากหน้าจอนี้ได้

**หน้าจอแก้ไข (Edit Record Screen)** อนุญาตให้ผู้ใช้แก้ไขหรือลบรายการทางการเงินที่มีอยู่

**หน้าจอเพิ่ม (Add Record Screen)** ช่วยให้ผู้ใช้สามารถเพิ่มข้อมูลทางการเงินใหม่ได้ โดยกรอกรายละเอียด, ราคา, และประเภท (รายรับ, รายจ่าย, การออม)

## 2. Business Logic Layer (Application Layer)

**การยืนยันตัวตน (Authentication)** จัดการกับการเข้าสู่ระบบ, การลงทะเบียน, และการจัดการเซสชัน โดยจะตรวจสอบข้อมูลรับรองของผู้ใช้กับฐานข้อมูลผู้ใช้

**ข้อมูลการเงิน (Financial Data)** จัดการกับการดำเนินการที่เกี่ยวข้องกับข้อมูลการเงินทั้งหมด เช่น การคำนวณ, การตรวจสอบความถูกต้อง และการโต้ตอบกับชั้นข้อมูลเพื่อดึงหรืออัปเดตรายการ

## 3. Data Layer

**ฐานข้อมูลผู้ใช้ (User Database)** เก็บข้อมูลที่เกี่ยวข้องกับผู้ใช้ เช่น อีเมล, รหัสผ่าน, และข้อมูลโปรไฟล์อื่น ๆ

**ฐานข้อมูลบันทึกการเงิน (Financial Records Database)** เก็บบันทึกการรายละเอียดทางการเงินสำหรับผู้ใช้แต่ละราย เช่น รายรับ, รายจ่าย, และการออม

## 4. Development and Deployment Environment

**Android Studio** สภาพแวดล้อมการพัฒนาแบบบูรณาการ (IDE) ที่ใช้ในการพัฒนาแอปพลิเคชัน ซึ่งเฉพาะสำหรับ Android

**การทดสอบและปรับใช้ (Testing & Deployment)** กระบวนการและเครื่องมือที่ใช้ในการทดสอบแอปพลิเคชัน

## b. Software Architecture (5 คะแนน, -0.25)

### 1. UI Components

**การจัดการส่วนต่าง ๆ ปุ่ม (Buttons), ช่องกรอกข้อมูล (TextFields), และรายการ (Lists)** ถูกออกแบบให้ใช้งานง่ายและวางตำแหน่งชัดเจนในแอป เช่น หน้าจอการเงินและการแก้ไขข้อมูล

**ปรับแต่งและนำมาใช้ซ้ำ** UI Components ถูกออกแบบให้ปรับแต่งได้ง่ายและสามารถนำมาใช้ซ้ำในหลายหน้าจอของแอป ทำให้การพัฒนาเป็นไปอย่างรวดเร็วและมีประสิทธิภาพ

## 2. Data Management

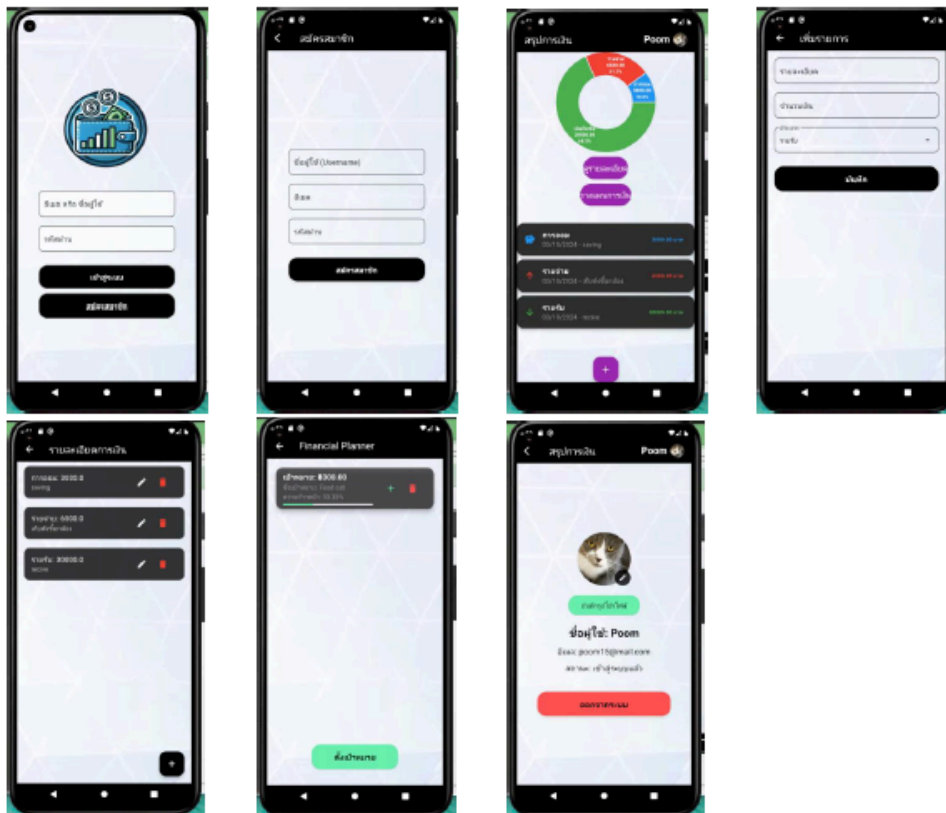
**การจัดการข้อมูลและ State** ใช้ฐานข้อมูลในเครื่อง (เช่น SQLite) จัดการข้อมูลผู้ใช้และการเงิน โดยมีการแยกแหล่งข้อมูล (Repository Pattern) เพื่อให้การจัดการข้อมูลมีความยืดหยุ่นและง่ายต่อการบำรุงรักษา

**เชื่อมต่อและแยกตรรกะ** ข้อมูลถูกจัดการแยกจากตรรกะการใช้งาน ทำให้การเชื่อมต่อกับฐานข้อมูลและการทำงานของแอปเป็นไปอย่างราบรื่น

## 3. Networking

**การเชื่อมต่อเครือข่าย** ในการเรียก API และจัดการการดึงข้อมูลแบบเรียลไทม์ ทำให้ข้อมูลในแอปพลิเคชันถูกอัปเดตอยู่เสมอและมีประสิทธิภาพในการทำงานทั้งออนไลน์และออฟไลน์

### c. UX/UI Design (5 คะแนน, -0.25)



#### หน้า Member Login

ส่วนประกอบ: มีช่องกรอกข้อมูลสำหรับอีเมลและรหัสผ่าน, ปุ่ม "Login", และลิงก์สำหรับผู้ที่ยังไม่มีบัญชี (เชื่อมโยงไปยังหน้า Member Register)

ฟังก์ชันการทำงาน: ผู้ใช้กรอกข้อมูลอีเมลและรหัสผ่านเพื่อเข้าสู่ระบบ เมื่อกดปุ่ม "Login" จะเข้าสู่หน้าจอแสดงข้อมูลทางการเงิน (Financial Details)

#### หน้า Member Register

ส่วนประกอบ: มีช่องกรอกข้อมูลสำหรับอีเมลและรหัสผ่าน, ปุ่ม "Register"

ฟังก์ชันการทำงาน: สำหรับผู้ที่ยังไม่มีบัญชี เมื่อกรอกข้อมูลอีเมลและรหัสผ่านแล้ว กดปุ่ม "Register" จะบันทึกข้อมูลและสามารถกลับไปหน้า Member Login เพื่อเข้าสู่ระบบ

#### หน้า Financial Details

ส่วนประกอบ: มีกราฟวงกลมที่แสดงการกระจายของรายได้, รายจ่าย, และเงินออม, แสดงข้อมูลรายได้ รายจ่าย และการออมในรูปแบบตัวเลข, ปุ่ม "Financial Details" และ "Sign out"

ฟังก์ชันการทำงาน: หน้านี้จะแสดงสรุปข้อมูลการเงินของผู้ใช้ทั้งหมด สามารถดูรายละเอียดเพิ่มเติมในหน้า Financial Details หรือออกจากระบบด้วยปุ่ม "Sign out"

### หน้า Financial Details (รายละเอียดทางการเงิน)

ส่วนประกอบ: ตารางที่แสดงรายละเอียดของแต่ละรายการการเงิน เช่น รายละเอียดของรายการ, จำนวนเงิน, ประเภทของรายการ (เช่น รายรับ, รายจ่าย, เงินออม), ปุ่มลบรายการ (รูปไอคอนถังขยะ), ปุ่มสำหรับเพิ่มรายการใหม่ (รูปไอคอนเครื่องหมายบวก), และปุ่ม "Sign out"

ฟังก์ชันการทำงาน: หน้านี้แสดงรายการรายละเอียดทางการเงินทั้งหมดของผู้ใช้ สามารถแก้ไขหรือลบรายการต่าง ๆ ได้ และยังสามารถเพิ่มรายการใหม่ได้ผ่านทางปุ่มไอคอนเครื่องหมายบวก

### หน้า Edit Record

ส่วนประกอบ: มีช่องกรอกข้อมูลสำหรับรายละเอียดของรายการ, จำนวนเงิน, และประเภทของรายการ (เช่น รายรับ, รายจ่าย, เงินออม), ปุ่ม "Confirm" และ "Delete"

ฟังก์ชันการทำงาน: หน้านี้ใช้สำหรับแก้ไขหรืออัปเดตรายละเอียดของรายการทางการเงินที่มีอยู่ หรือสามารถลบรายการนั้นออกได้

### หน้า Add Record

ส่วนประกอบ: มีช่องกรอกข้อมูลสำหรับรายละเอียดของรายการ, จำนวนเงิน, และประเภทของรายการ (เช่น รายรับ, รายจ่าย, เงินออม), ปุ่ม "Confirm"

ฟังก์ชันการทำงาน: หน้านี้ใช้สำหรับเพิ่มรายการทางการเงินใหม่ โดยกรอกข้อมูลรายละเอียดที่เกี่ยวข้องแล้วกด "Confirm" เพื่อบันทึกรายการ

## d. Testing Design (5 คะแนน, -0.25)

### 1. หน้าจอลงทะเบียน (Register Screen)

กรณีทดสอบ: ตรวจสอบการลงทะเบียนสำเร็จ

ข้อมูลนำเข้า: อีเมลและรหัสผ่านที่ถูกต้อง

ผลลัพธ์ที่คาดหวัง: ผู้ใช้ถูกลงทะเบียนสำเร็จและเปลี่ยนเส้นทางไปยังหน้าจอ Login

กรณีทดสอบ: ตรวจสอบการลงทะเบียนด้วยข้อมูลที่ไม่ถูกต้อง

ข้อมูลนำเข้า: อีเมลที่ไม่ถูกต้อง (เช่น ไม่มี "@") หรือรหัสผ่านไม่เป็นไปตามข้อกำหนด (เช่น รหัสผ่านสั้นเกินไป)

ผลลัพธ์ที่คาดหวัง: แสดงข้อความแสดงข้อผิดพลาดและผู้ใช้ยังคงอยู่ที่หน้าจอ Register

กรณีทดสอบ: ตรวจสอบการลงทะเบียนด้วยฟิลด์ว่าง

ข้อมูลนำเข้า: ทั้งฟิลด์อีเมลหรือรหัสผ่านว่างไว้

ผลลัพธ์ที่คาดหวัง: แสดงข้อความแสดงข้อผิดพลาดและผู้ใช้ยังคงอยู่ที่หน้าจอ Register

### 2. หน้าจอ Login

กรณีทดสอบ: ตรวจสอบการล็อกอินสำเร็จ

ข้อมูลนำเข้า: อีเมลและรหัสผ่านที่ถูกต้อง

ผลลัพธ์ที่คาดหวัง: ผู้ใช้ถูกเปลี่ยนเส้นทางไปยังหน้าหลัก (Dashboard) ของระบบ

กรณีทดสอบ: ตรวจสอบการล็อกอินด้วยข้อมูลที่ไม่ถูกต้อง

ข้อมูลนำเข้า: อีเมลหรือรหัสผ่านที่ไม่ถูกต้อง

ผลลัพธ์ที่คาดหวัง: แสดงข้อความแสดงข้อผิดพลาด และผู้ใช้ยังคงอยู่ที่หน้าจอ Login

กรณีทดสอบ: ตรวจสอบการล็อกอินเมื่อมีฟิลด์ว่าง

ข้อมูลนำเข้า: ทั้งฟิลด์อีเมลหรือรหัสผ่านว่างไว้

ผลลัพธ์ที่คาดหวัง: แสดงข้อความแสดงข้อผิดพลาด และผู้ใช้ยังคงอยู่ที่หน้าจอ Login

### 3. หน้าจอหลัก (Dashboard)

กรณีทดสอบ: ตรวจสอบการแสดงผลข้อมูล

ข้อมูลนำเข้า: ผู้ใช้ที่มีบันทึกทางการเงิน

ผลลัพธ์ที่คาดหวัง: แสดงจำนวนรายรับ รายจ่าย และการออมได้อย่างถูกต้อง

กรณีทดสอบ: ตรวจสอบการแสดงผลกราฟวงกลม (Pie Chart)

ข้อมูลนำเข้า: ผู้ใช้ที่มีบันทึกทางการเงินหลากหลายประเภท (รายรับ, รายจ่าย, การออม)

ผลลัพธ์ที่คาดหวัง: กราฟวงกลมแสดงสัดส่วนของแต่ละประเภทได้อย่างถูกต้อง

### 4. หน้าจอรายละเอียดทางการเงิน (Financial Details)

กรณีทดสอบ: ตรวจสอบการแสดงผลบันทึกทางการเงิน

ข้อมูลนำเข้า: ผู้ใช้ที่มีบันทึกหลายรายการ

ผลลัพธ์ที่คาดหวัง: แสดงบันทึกทางการเงินทั้งหมดด้วยรายละเอียดที่ถูกต้อง

กรณีทดสอบ: ตรวจสอบการแก้ไขบันทึก

ข้อมูลนำเข้า: เลือกรายการที่มีอยู่แล้วและเลือกแก้ไข

ผลลัพธ์ที่คาดหวัง: รายละเอียดของบันทึกถูกเติมเต็มในหน้าจอแก้ไขอย่างถูกต้อง

กรณีทดสอบ: ตรวจสอบการลบบันทึก

ข้อมูลนำเข้า: เลือกรายการแล้วเลือกที่จะลบ

ผลลัพธ์ที่คาดหวัง: บันทึกนั้นถูกลบออกจากรายการและยอดรวมถูกอัปเดตอย่างถูกต้อง

### 5. หน้าจอแก้ไขบันทึก (Edit Record)

กรณีทดสอบ: ตรวจสอบการแก้ไขบันทึกที่มีอยู่แล้ว

ข้อมูลนำเข้า: ปรับปรุงรายละเอียด ราคา และประเภทของบันทึก

ผลลัพธ์ที่คาดหวัง: รายละเอียดที่อัปเดตถูกบันทึกและแสดงผลอย่างถูกต้องในหน้าจอรายละเอียดทางการเงิน

กรณีทดสอบ: ตรวจสอบการตรวจสอบข้อมูล (Validation)

ข้อมูลนำเข้า: ทั้งฟิลด์ว่างหรือใส่ข้อมูลที่ไม่ถูกต้อง

ผลลัพธ์ที่คาดหวัง: แสดงข้อความแสดงข้อผิดพลาดและผู้ใช้ถูกขอให้แก้ไขข้อมูล

## 6. หน้าจอเพิ่มบันทึก (Add Record)

กรณีทดสอบ: ตรวจสอบการเพิ่มบันทึกใหม่

ข้อมูลนำเข้า: ป้อนรายละเอียด ราคา และประเภทสำหรับบันทึกใหม่

ผลลัพธ์ที่คาดหวัง: บันทึกนั้นถูกเพิ่มในรายการและยอดรวมถูก

อัปเดตอย่างถูกต้อง

กรณีทดสอบ: ตรวจสอบการตรวจสอบข้อมูล (Validation)

ข้อมูลนำเข้า: ทิ้งฟิลด์ว่างหรือใส่ข้อมูลที่ไม่ถูกต้อง

ผลลัพธ์ที่คาดหวัง: แสดงข้อความแสดงข้อผิดพลาดและผู้ใช้ถูกขอให้แก้ไขข้อมูล

## 7. การทดสอบฟังก์ชันทั่วไป

กรณีทดสอบ: ตรวจสอบฟังก์ชันการออกจากระบบ (Sign out)

ข้อมูลนำเข้า: คลิกปุ่ม "Sign out"

ผลลัพธ์ที่คาดหวัง: ผู้ใช้ถูกเปลี่ยนเส้นทางกลับไปหน้าจอ Login และสิ้นสุดเซสชัน