ระบบจัดการสินค้า

**Product Management System**

**ระบบจัดการสินค้าและหมวดหมู่แบบ Full-stack โดยใช้ Vue.js และ Express.js + MongoDB**

# 1. Quick Start (เริ่มต้นอย่างรวดเร็ว)

สำหรับผู้ที่มี Node.js และ MongoDB ติดตั้งในเครื่องแล้ว:

### 1. Clone โปรเจค

git clone <repository-url>

cd Pd Management

### 2. ติดตั้ง Backend

cd backend

npm install

copy .env.example .env

npm run seed

npm run dev

Backend จะรันที่ http://localhost:5000

### 3. ติดตั้ง Frontend

cd frontend

npm install

npm run dev

Frontend จะรันที่ http://localhost:5173

# 2. เทคโนโลยีที่ใช้

## Backend

* Node.js + Express.js
* MongoDB (Database)
* TypeScript
* Mongoose (ODM)

## Frontend

* Vue.js 3 (Composition API)
* TypeScript
* Vue Router
* Axios (HTTP Client)

# 3. ฟีเจอร์

## สินค้า (Products)

* แสดงรายการสินค้าในรูปแบบตาราง
* เพิ่มสินค้าใหม่พร้อมเลือกหมวดหมู่
* แก้ไขข้อมูลสินค้า
* ลบสินค้า
* ค้นหาสินค้า
* Pagination

## หมวดหมู่ (Categories)

* แสดงรายการหมวดหมู่ในรูปแบบตาราง
* เพิ่มหมวดหมู่ใหม่
* แก้ไขข้อมูลหมวดหมู่
* ลบหมวดหมู่
* ค้นหาหมวดหมู่
* Pagination

## อื่นๆ

* Form Validation (ตรวจสอบข้อมูลก่อนบันทึก)
* Responsive Design
* User-friendly Interface

# 4. ข้อกำหนดเบื้องต้น

คุณต้องติดตั้งโปรแกรมเหล่านี้ในเครื่องก่อน:

* **Node.js** (แนะนำ v18.0.0 ขึ้นไป)
* **MongoDB** (v6.0 ขึ้นไป) - ติดตั้งในเครื่อง หรือใช้ MongoDB Atlas (Cloud)
* **npm** หรือ **yarn** (มากับ Node.js)
* **Git** (สำหรับ clone โปรเจค)

## ตรวจสอบเวอร์ชัน

node --version

npm --version

# 5. คู่มือการติดตั้งแบบละเอียด

## ขั้นตอนที่ 1: ติดตั้ง Node.js

1. ไปที่ https://nodejs.org/
2. ดาวน์โหลด LTS version (แนะนำ v18 ขึ้นไป)
3. รันไฟล์ติดตั้งและทำตามขั้นตอน
4. เปิด Terminal/Command Prompt และตรวจสอบ

## ขั้นตอนที่ 2: ติดตั้ง MongoDB

**เลือกวิธีใดวิธีหนึ่ง:**

### วิธีที่ 1: MongoDB Local (ติดตั้งในเครื่อง)

**Windows:**

1. ไปที่ https://www.mongodb.com/try/download/community
2. ดาวน์โหลด MSI Installer
3. ติดตั้งแบบ Complete และเลือกติดตั้งเป็น Service
4. MongoDB จะทำงานอัตโนมัติเมื่อเปิดเครื่อง

**macOS:**

brew tap mongodb/brew

brew install mongodb-community

brew services start mongodb-community

### วิธีที่ 2: MongoDB Atlas (Cloud)

1. ไปที่ https://www.mongodb.com/cloud/atlas และสมัครสมาชิกฟรี
2. เข้าสู่ระบบและสร้าง Project
3. สร้าง Cluster (เลือก M0 Free - ฟรี 512MB)
4. สร้าง Database User และจดบันทึก Username และ Password
5. ตั้งค่า Network Access: Allow Access from Anywhere (0.0.0.0/0)
6. รับ Connection String และแทนที่ username/password

# 6. การตั้งค่า Backend

## ติดตั้ง Dependencies

cd backend

npm install

## สร้างไฟล์ .env

Windows (Command Prompt):

copy .env.example .env

macOS / Linux:

cp .env.example .env

## แก้ไขไฟล์ .env

**ถ้าใช้ MongoDB Local:**

PORT=5000

MONGODB\_URI=mongodb://localhost:27017/product-management

NODE\_ENV=development

**ถ้าใช้ MongoDB Atlas:**

PORT=5000

MONGODB\_URI=mongodb+srv://YOUR\_USERNAME:YOUR\_PASSWORD@...

NODE\_ENV=development

## Seed ข้อมูลตัวอย่าง

npm run seed

## รัน Backend

npm run dev

**สำคัญ: ปล่อย Terminal นี้ไว้ทำงาน อย่าปิด!**

# 7. การตั้งค่า Frontend

**เปิด Terminal/Command Prompt หน้าต่างใหม่**

cd frontend

npm install

npm run dev

Frontend จะรันที่:

http://localhost:5173

**เปิดเบราว์เซอร์ไปที่ URL ด้านบน คุณจะเห็นหน้าเว็บระบบจัดการสินค้า!**

# 8. API Endpoints

| **Method** | **Endpoint** |
| --- | --- |
| **GET** | /api/products - ดึงรายการสินค้าทั้งหมด |
| **GET** | /api/product/:id - ดึงข้อมูลสินค้าตาม ID |
| **POST** | /api/product - เพิ่มสินค้าใหม่ |
| **PUT** | /api/product/:id - แก้ไขข้อมูลสินค้า |
| **DELETE** | /api/product/:id - ลบสินค้า |
| **GET** | /api/categories - ดึงรายการหมวดหมู่ทั้งหมด |
| **GET** | /api/category/:id - ดึงข้อมูลหมวดหมู่ตาม ID |
| **POST** | /api/category - เพิ่มหมวดหมู่ใหม่ |
| **PUT** | /api/category/:id - แก้ไขข้อมูลหมวดหมู่ |
| **DELETE** | /api/category/:id - ลบหมวดหมู่ |

# 9. การแก้ไขปัญหาที่พบบ่อย

## ปัญหา: npm install ล้มเหลว

**วิธีแก้:**

npm cache clean --force

rm -rf node\_modules package-lock.json

npm install

## ปัญหา: Backend ไม่สามารถเชื่อมต่อ MongoDB

**วิธีแก้:**

* ตรวจสอบว่า MongoDB service กำลังทำงาน
* ตรวจสอบ username/password ในไฟล์ .env
* ตรวจสอบ Connection String

## ปัญหา: Port 5000 ถูกใช้งานแล้ว

**วิธีแก้:**

Windows - หา Process ที่ใช้ port:

netstat -ano | findstr :5000

taskkill /PID <PID> /F

macOS/Linux:

lsof -i :5000

kill -9 <PID>

## ปัญหา: Frontend ไม่สามารถเรียก API ได้

**วิธีแก้:**

1. ตรวจสอบว่า Backend กำลังทำงาน
2. ตรวจสอบ CORS ในไฟล์ backend/server.ts
3. ตรวจสอบ API URL ในไฟล์ frontend/src/services/api.ts
4. Clear browser cache และ reload

# 10. คำสั่งที่ใช้บ่อย

## Backend Commands

npm install # ติดตั้ง dependencies

npm run dev # รัน development mode

npm run build # Build สำหรับ production

npm start # รัน production

npm run seed # Seed ข้อมูลตัวอย่าง

## Frontend Commands

npm install # ติดตั้ง dependencies

npm run dev # รัน development mode

npm run build # Build สำหรับ production

npm run preview # Preview production build

npm run type-check # ตรวจสอบ TypeScript errors

# 11. โครงสร้างโปรเจค

Pd Management/

├── backend/

│ ├── src/

│ │ ├── config/ # การตั้งค่าฐานข้อมูล

│ │ ├── controllers/ # Controllers

│ │ ├── models/ # Mongoose Models

│ │ ├── routes/ # API Routes

│ │ ├── services/ # Business Logic

│ │ └── scripts/ # Seed scripts

│ ├── server.ts # Entry point

│ ├── .env # Config file

│ └── package.json

│

└── frontend/

├── src/

│ ├── components/ # Vue Components

│ ├── services/ # API Services

│ ├── views/ # Vue Views/Pages

│ ├── router/ # Vue Router

│ ├── App.vue # Root Component

│ └── main.ts # Entry point

└── package.json

# 12. ทรัพยากรเพิ่มเติม

* Vue.js Documentation - https://vuejs.org/
* Express.js Guide - https://expressjs.com/
* MongoDB Manual - https://docs.mongodb.com/
* TypeScript Handbook - https://www.typescriptlang.org/docs/
* Mongoose Documentation - https://mongoosejs.com/

**--- สิ้นสุดเอกสาร ---**