

# Tech Stack Overview (.md version)

## Frontend

### React + React-Bootstrap + TailwindCSS

- ใช้ React-Bootstrap สร้าง layout ง่าย ๆ เช่น `<Row><Col>`
- ใช้ TailwindCSS ปรับแต่ง style ได้ละเอียด เช่น `p-4`, `text-center`

### qrcode

- ใช้แสดง QR code บนหน้า React ได้ เช่นใช้ `qrcode.react`

### Electron

- ทำให้ frontend เป็น Desktop App
- ใช้เชื่อมกับ native feature เช่น เครื่องพิมพ์, filesystem

---

## Backend

### Node.js + Express.js

- ใช้แยก layer เป็น `routes`, `controllers`, `services`, `middlewares`

### Prisma (ORM สำหรับ MySQL)

- ใช้ง่าย ปลอดภัย type-safe
- จัดการ schema/migration ง่าย
- ใช้ Prisma Client เชื่อม MySQL

### JWT (ใช้เมื่อมี Login)

- สำหรับยืนยันตัวตนระหว่าง client ↔ backend
- ถ้าเป็น app local-only ใช้ localStorage/session แทนได้

### Winston / Morgan

- **Morgan:** log HTTP request
- **Winston:** log ระบบ, การพิมพ์, error, ฯลฯ

### node-printer

- ใช้ร่วมกับ Electron เพื่อสั่งพิมพ์ไปยังเครื่องปริ้นในระบบ

## Database

### MySQL

- รองรับทั้ง local และ cloud เช่น PlanetScale, Railway
  - Prisma เชื่อม MySQL ได้ดี
- 

## แนะนำโครงสร้างโฟลเดอร์

```
my-desktop-app/  
├── backend/  
│   ├── prisma/  
│   │   ├── schema.prisma  
│   │   └── migrations/  
│   ├── src/  
│   │   ├── controllers/  
│   │   ├── routes/  
│   │   ├── services/  
│   │   ├── middlewares/  
│   │   └── index.ts  
│   └── frontend/  
│       ├── src/  
│       │   ├── components/  
│       │   ├── pages/  
│       │   ├── hooks/  
│       │   └── App.tsx  
│   └── electron/  
│       └── main.ts  
├── public/  
└── package.json
```

## สิ่งที่ต้องเพิ่มในอนาคต

- ระบบ Auto-update สำหรับ Electron
- ระบบ Sync/Backup ข้อมูล
- ระบบสิทธิ์ผู้ใช้ (Role/Permission)
- Offline mode (ใช้ Prisma SQLite แทน MySQL)