**TÀI LIỆU KIẾN TRÚC PHẦN MỀM (SAD)**

**Dự án:** PlanbookAI (PBA) – Cổng công cụ AI dành cho giáo viên THPT

**1. Giới thiệu**

**1.1 Mục đích**

* **Xác định vấn đề:** Giáo viên THPT phải đồng thời giảng dạy và thực hiện các công việc hành chính lặp đi lặp lại như soạn giáo án, tạo đề, chấm điểm.
* **Mục tiêu:** Tự động hóa lập kế hoạch bài giảng, tạo đề kiểm tra và chấm điểm; giảm thời gian công việc thủ công; cung cấp không gian lưu trữ tài liệu cá nhân.

**1.2 Phạm vi tài liệu**

* Bao gồm: Frontend web, backend API, cơ sở dữ liệu, tích hợp AI & OCR, triển khai Docker.
* Không bao gồm: Ứng dụng di động, dịch vụ kiểm thử bên thứ ba.

**1.3 Thuật ngữ & Viết tắt**

| **Viết tắt** | **Giải nghĩa** |
| --- | --- |
| PBA | PlanbookAI |
| SAD | Tài liệu Kiến trúc phần mềm |
| API | Giao diện lập trình ứng dụng |
| OCR | Nhận dạng ký tự quang học |
| JWT | JSON Web Token |
| RBAC | Phân quyền theo vai trò |

**2. Tổng quan Kiến trúc**

**2.1 Tóm tắt Kiến trúc**

Hệ thống theo mô hình **N-tier**:

1. **Presentation Layer:** Next.js (ReactJS)
2. **Application Layer:** Spring Boot RESTful API
3. **Data Layer:** Supabase (Postgres)
4. **AI & OCR Services:** Gemini AI, Google Vision API
5. **Infrastructure:** Docker containers trên VPS

[NextJS] ⇄ [Spring Boot API] ⇄ [Postgres]

⇅

[Gemini AI] [OCR Service]

**2.2 Ràng buộc & Quyết định thiết kế**

* **Next.js** để SSR và SEO.
* **Spring Boot** cho backend linh hoạt.
* **Supabase** cung cấp realtime và authentication.
* **Gemini AI** vì chất lượng sinh đề.
* **Google Vision API** cho OCR chính xác.
* **Docker** để container hoá, CI/CD qua GitHub Actions.

**3. Các thành phần hệ thống**

| **Thành phần** | **Công nghệ & Mô tả** |
| --- | --- |
| Frontend | Next.js (ReactJS): giao diện Admin và Manager |
| Backend | Spring Boot: xử lý nghiệp vụ, bảo mật JWT |
| Database | Supabase (Postgres): lưu users, questions, exams, results |
| AI Integration | Gemini AI: sinh câu hỏi, đề thi |
| OCR Integration | Google Vision API: nhận diện và chấm điểm tự động |
| Infrastructure | Docker containers trên VPS: triển khai, mở rộng dễ dàng |

**4. Kiểm thử**

| **Loại kiểm thử** | **Công cụ & Phương pháp** |
| --- | --- |
| Unit Test Backend | JUnit, Mockito |
| Unit Test Frontend | Jest, React Testing Library |
| Integration Test | Postman/Newman |
| End-to-End Test | Cypress |
| Load Test | Locust |
| Security Test | OWASP ZAP |