

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

KHOA CÔNG NGHỆ THÔNG TIN 1



**BÁO CÁO GIỮA KỲ  
THỰC TẬP CƠ SỞ**

**Đề tài: Xây dựng trang web thi online**

**Giảng viên hướng dẫn: TS. Kim Ngọc Bách**

**Họ và tên: Lê Xuân Thắng**

**Mã sinh viên: B23DCCN753**

# Mục lục

<b>1. GIỚI THIỆU DỰ ÁN .....</b>	2
<b>1.2. Ý nghĩa của đề tài .....</b>	2
<b>1.3. Tính ứng dụng của đề tài .....</b>	2
<b>1.4. Giá trị học thuật.....</b>	2
<b>2. CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ SỬ DỤNG.....</b>	2
<b>2.1. Cơ sở lý thuyết.....</b>	2
2.1.1. Mô hình Client-Server .....	2
2.1.2. RESTful API.....	2
2.1.3. JSON Web Token(JWT).....	3
2.1.4. Chuẩn hóa cơ sở dữ liệu.....	3
<b>2.2. Công nghệ sử dụng.....</b>	3
<b>3. PHÂN TÍCH YÊU CẦU CỦA DỰ ÁN .....</b>	3
<b>3.1. Yêu cầu chức năng .....</b>	3
3.1.1. Người dùng .....	3
3.1.2. Admin.....	4
<b>3.2. Yêu cầu phi chức năng .....</b>	4
<b>3.3. Thiết kế cơ sở dữ liệu .....</b>	4
<b>3.4. Thuật toán tính điểm và leaderboard .....</b>	4
<b>4. KẾ HOẠCH THỰC HIỆN DỰ ÁN.....</b>	4
<b>DANH MỤC TÀI LIỆU THAM KHẢO.....</b>	5

## **1. GIỚI THIỆU DỰ ÁN**

### **1.1. Lý do chọn đề tài**

Trong bối cảnh học tập trực tuyến ngày càng phổ biến, nhu cầu xây dựng các hệ thống kiểm tra và đánh giá năng lực qua môi trường web ngày càng tăng. Các nền tảng quiz hiện nay tuy đa dạng nhưng không tùy biến cao cho mục đích học tập nội bộ. Việc tạo ra một hệ thống thi online giúp loại bỏ các rào cản về địa lý và tiết kiệm chi phí in ấn tài liệu.

### **1.2. Ý nghĩa của đề tài**

Dự án giúp người học tự kiểm tra kiến thức một cách nhanh chóng, đồng thời giúp người quản trị dễ dàng thống kê kết quả mà không cần chấm tay thủ công.

### **1.3. Tính ứng dụng của đề tài**

Hệ thống có thể áp dụng cho các trung tâm ngoại ngữ, các buổi kiểm tra nội bộ tại doanh nghiệp, các bài kiểm tra nhỏ trên lớp học hoặc đơn giản là ứng dụng ôn thi cá nhân.

### **1.4. Giá trị học thuật**

Dự án giúp sinh viên:

- Nắm vững và vận dụng kiến thức về lập trình Web
- Hiểu kiến trúc RESTful API
- Áp dụng chuẩn hóa dữ liệu 3NF
- Vận dụng thuật toán tính toán thống kê cơ bản
- Thực hành triển khai và vận hành hệ thống thực tế
- Làm quen với quy trình phát triển phần mềm hoàn chỉnh

## **2. CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ SỬ DỤNG**

### **2.1. Cơ sở lý thuyết**

#### **2.1.1. Mô hình Client-Server**

Mô hình Client Server là mô hình mạng máy tính trong đó các máy tính con được đóng vai trò như một máy khách, chúng làm nhiệm vụ gửi yêu cầu đến các máy chủ. Để máy chủ xử lý yêu cầu và trả kết quả về cho máy khách đó.[1]

Hệ thống được xây dựng theo kiến trúc Client–Server, trong đó:

- Client (ReactJS) xử lý giao diện và tương tác người dùng
- Server (NodeJS + Express) xử lý logic nghiệp vụ
- Database (PostgreSQL/MySQL) lưu trữ dữ liệu

#### **2.1.2. RESTful API**

RESTful API là một tiêu chuẩn dùng trong việc thiết kế API cho các ứng dụng web (thiết kế Web services) để tiện cho việc quản lý các resource. Nó chú trọng vào tài nguyên hệ thống (tệp văn bản, ảnh, âm thanh, video, hoặc dữ liệu động...),

bao gồm các trạng thái tài nguyên được định dạng và được truyền tải qua HTTP.[2]

Hệ thống API tuân theo nguyên tắc:

- Stateless
- Resource-based
- Sử dụng HTTP methods: GET, POST, PUT, DELETE

#### 2.1.3. JSON Web Token(JWT)

JSON Web Token (hay còn gọi là JWT) là một tiêu chuẩn mã nguồn mở (open industry standard) được sử dụng để chia sẻ thông tin an toàn, khép kín giữa hai thực thể, thường là máy khách (chẳng hạn như frontend của ứng dụng) và máy chủ (backend của ứng dụng).[3]

JWT được sử dụng để:

- Xác thực người dùng
- Bảo vệ các API yêu cầu quyền truy cập

#### 2.1.4. Chuẩn hóa cơ sở dữ liệu

Chuẩn hóa cơ sở dữ liệu để đảm bảo dữ liệu nhất quán, không bị trùng lặp, bảo trì dễ hơn và hỗ trợ khả năng mở rộng hệ thống.

### 2.2. Công nghệ sử dụng

Frontend:

- ReactJS
- Vite
- Chart.js

Backend:

- NodeJS
- ExpressJS

Database:

- PostgreSQL (hoặc MySQL)

Authentication:

- JWT

Deployment:

- Vercel (Frontend)
- Render (Backend)
- Supabase/Neon (Database)

## 3. PHÂN TÍCH YÊU CẦU CỦA DỰ ÁN

### 3.1. Yêu cầu chức năng

#### 3.1.1. Người dùng

- Đăng ký
- Đăng nhập

- Tham gia thi
- Xem kết quả
- Xem lịch sử làm bài

### 3.1.2. Admin

- Tạo câu hỏi
- Thêm/sửa/xóa câu hỏi
- Xem thống kê

### 3.2. *Yêu cầu phi chức năng*

- Thời gian phản hồi <2 giây
- Bảo mật token
- Hỗ trợ trình duyệt Chrome/Edge

### 3.3. *Thiết kế cơ sở dữ liệu*

Các bảng chính:

- users
- exams
- questions
- answers
- results
- attempts

Quan hệ:

- 1 exam có nhiều question
- 1 user có nhiều attempt

### 3.4. *Thuật toán tính điểm và leaderboard*

Xây dựng thuật toán tính điểm dựa trên thời gian trả lời câu hỏi. Trả lời càng nhanh điểm sẽ càng cao.

Leaderboard sắp xếp theo tổng điểm.

## 4. KẾ HOẠCH THỰC HIỆN DỰ ÁN

Giai đoạn 1: Học nền tảng JavaScript và NodeJS(Tuần 1-2)

Giai đoạn 2: Xây dựng Backend(Tuần 3-4)

Giai đoạn 3: Học và xây dựng Frontend(Tuần 5-6)

Giai đoạn 4: Kết nối hệ thống(Tuần 7-8)

Giai đoạn 5: Phân tích nâng cao(Tuần 9-10)

Giai đoạn 6: Kiểm thử và triển khai(Tuần 11-12)

## **DANH MỤC TÀI LIỆU THAM KHẢO**

- [1] Andy Ngo. (31/10/2020). CodeLearn. <https://codelearn.io/sharing/tim-hieu-ve-mo-hinh-client-server>
- [2] Lê Đức Mạnh. (05/09/2020). Viblo. <https://viblo.asia/p/restful-api-la-gi-1Je5EDJ4lnL>
- [3] Uyen Ngo. (29/07/2024). ITViec. <http://itviec.com/blog/json-web-token-la-gi/>