**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC BÌNH DƯƠNG**

**KHOA CÔNG NGHỆ THÔNG TIN – TRÍ TUỆ NHÂN TẠO -ROBOT**

Ảnh có chứa biểu tượng, đồng, phù hiệu

Nội dung do AI tạo ra có thể không chính xác.

**MÔN: PHÁT TRIỂN ỨNG DỤNG MÃ NGUỒN MỞ.**N

**Đề tài: Nền tảng thi trắc nghiệm**

**Ngành: Công Nghệ Thông Tin**

**Giảng viên: Lê Duy Hùng**

Sinh viên thực hiện:

* **Nguyễn Minh Nhật** MSSV: 21050004 Lớp: 24TH01

**TP.HCM, tháng 12 năm 2025**

Contents

[**1. GIỚI THIỆU HỆ THỐNG** 3](#_Toc216683450)

[**2. GIỚI THIỆU CHỨC NĂNG VÀ CÔNG NGHỆ SỬ DỤNG** 3](#_Toc216683451)

[**2.1 Chức năng của hệ thống** 3](#_Toc216683452)

[**2.2 Công nghệ sử dụng** 5](#_Toc216683453)

[**3. KẾ HOẠCH TRIỂN KHAI HỆ THỐNG** 5](#_Toc216683454)

[**4. MÔ HÌNH KIẾN TRÚC VÀ PHÂN TÍCH KIẾN TRÚC HỆ THỐNG** 6](#_Toc216683455)

[**4.1 Mô hình tổng quát** 6](#_Toc216683456)

[**4.2 Phân tích kiến trúc** 6](#_Toc216683457)

[**5. PHÂN TÍCH NGHIỆP VỤ & API PLANNING** 6](#_Toc216683458)

[**5.1 Phân tích nghiệp vụ** 6](#_Toc216683459)

[**5.2 API Planning** 6](#_Toc216683460)

[**6. THIẾT KẾ GIAO DIỆN BAN ĐẦU** 7](#_Toc216683461)

[**7. KẾT QUẢ XÂY DỰNG HỆ THỐNG** 7](#_Toc216683462)

[**7.1 Nộp code và README.md** 7](#_Toc216683463)

[**7.2 Chức năng của chương trình** 7](#_Toc216683464)

[**7.3 Coding Convention** 7](#_Toc216683465)

[**7.4 Triển khai thực tế** 8](#_Toc216683466)

[**8. ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN** 8](#_Toc216683467)

[**8.1 Đánh giá** 8](#_Toc216683468)

[**8.2 Hướng phát triển** 19](#_Toc216683469)

[**9. KẾT LUẬN** 19](#_Toc216683470)

# **1. GIỚI THIỆU HỆ THỐNG**

**Trong bối cảnh chuyển đổi số hiện nay, việc ứng dụng công nghệ thông tin vào giáo dục và kiểm tra đánh giá là xu hướng tất yếu. Các hệ thống kiểm tra trực tuyến giúp tiết kiệm thời gian, chi phí, đồng thời nâng cao tính linh hoạt và khả năng mở rộng so với hình thức thi truyền thống.**

**Xuất phát từ nhu cầu đó, nhóm thực hiện đề tài “Xây dựng hệ thống Quiz Online” với mục tiêu tạo ra một ứng dụng web cho phép người dùng làm bài trắc nghiệm trực tuyến một cách thuận tiện, nhanh chóng và chính xác.**

**Hệ thống cho phép:**

* **Lấy câu hỏi ngẫu nhiên từ cơ sở dữ liệu**
* **Hiển thị từng câu hỏi theo thứ tự**
* **Giới hạn thời gian làm bài**
* **Tự động chấm điểm và hiển thị kết quả**

**Hệ thống được xây dựng với mục tiêu đơn giản, dễ sử dụng nhưng vẫn đảm bảo tính mở rộng trong tương lai.**

# **2. GIỚI THIỆU CHỨC NĂNG VÀ CÔNG NGHỆ SỬ DỤNG**

## **2.1 Chức năng của hệ thống**

**Hệ thống Quiz Online bao gồm các chức năng chính sau:**

* **Bắt đầu bài thi**
* **Hiển thị câu hỏi trắc nghiệm**
* **Chọn đáp án cho từng câu hỏi**
* **Di chuyển giữa các câu hỏi**
* **Đếm ngược thời gian làm bài**
* **Nộp bài và tính điểm tự động**
* **Hiển thị kết quả sau khi hoàn thành**
* **Ảnh có chứa ảnh chụp màn hình, máy tính, phần mềm, văn bản

  Nội dung do AI tạo ra có thể không chính xác.Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, máy tính

  Nội dung do AI tạo ra có thể không chính xác.**

**Ảnh có chứa văn bản, ảnh chụp màn hình, máy tính, phần mềm

Nội dung do AI tạo ra có thể không chính xác.**

## **2.2 Công nghệ sử dụng**

| **Thành phần** | **Công nghệ** |
| --- | --- |
| **Backend** | **Python, FastAPI** |
| **Frontend** | **HTML, CSS, JavaScript** |
| **Database** | **PostgreSQL** |
| **ORM** | **SQLAlchemy** |
| **Web Server** | **Uvicorn (ASGI)** |
| **Công cụ quản lý mã** | **Git, GitHub** |

**FastAPI được lựa chọn vì hiệu năng cao, cú pháp đơn giản, hỗ trợ tốt cho REST API và dễ mở rộng.**

# **3. KẾ HOẠCH TRIỂN KHAI HỆ THỐNG**

**Quá trình triển khai hệ thống được chia thành các giai đoạn:**

1. **Phân tích yêu cầu và nghiệp vụ**
2. **Thiết kế cơ sở dữ liệu**
3. **Xây dựng backend API**
4. **Thiết kế giao diện người dùng**
5. **Kết nối frontend và backend**
6. **Kiểm thử hệ thống**
7. **Hoàn thiện và triển khai**

**Mỗi giai đoạn đều được kiểm tra và điều chỉnh để đảm bảo hệ thống hoạt động đúng yêu cầu đề ra.**

# **4. MÔ HÌNH KIẾN TRÚC VÀ PHÂN TÍCH KIẾN TRÚC HỆ THỐNG**

## **4.1 Mô hình tổng quát**

**Hệ thống được xây dựng theo mô hình Client – Server – Database.**

* **Client: Trình duyệt web của người dùng**
* **Server: Ứng dụng FastAPI xử lý nghiệp vụ**
* **Database: PostgreSQL lưu trữ dữ liệu câu hỏi**

## **4.2 Phân tích kiến trúc**

* **Client gửi request thông qua HTTP**
* **Server xử lý logic, truy vấn database**
* **Database trả kết quả về server**
* **Server trả dữ liệu dạng JSON cho client**
* **Client hiển thị dữ liệu cho người dùng**

**Kiến trúc này giúp hệ thống:**

* **Dễ bảo trì**
* **Dễ mở rộng**
* **Tách biệt rõ ràng giữa giao diện và xử lý logic**

# **5. PHÂN TÍCH NGHIỆP VỤ & API PLANNING**

## **5.1 Phân tích nghiệp vụ**

**Quy trình làm bài của người dùng:**

1. **Người dùng truy cập website**
2. **Nhấn nút Bắt đầu**
3. **Hệ thống tải 20 câu hỏi ngẫu nhiên**
4. **Người dùng trả lời từng câu**
5. **Hệ thống ghi nhận đáp án**
6. **Người dùng nộp bài**
7. **Hệ thống chấm điểm và hiển thị kết quả**

## **5.2 API Planning**

| **Method** | **API** | **Mô tả** |
| --- | --- | --- |
| **GET** | **/quiz/random** | **Lấy 20 câu hỏi ngẫu nhiên** |
| **POST** | **/quiz/submit** | **Chấm điểm (mở rộng)** |

# **6. THIẾT KẾ GIAO DIỆN BAN ĐẦU**

**Giao diện hệ thống được thiết kế theo hướng tối giản, dễ sử dụng:**

* **Trang bắt đầu bài thi**
* **Trang làm bài trắc nghiệm**
* **Khu vực hiển thị câu hỏi**
* **Các lựa chọn đáp án rõ ràng**
* **Thanh hiển thị thời gian**

**Giao diện có thể được thiết kế sơ bộ bằng Figma trước khi triển khai bằng HTML/CSS.**

# **7. KẾT QUẢ XÂY DỰNG HỆ THỐNG**

## **7.1 Nộp code và README.md**

* **Mã nguồn được quản lý bằng Git**
* **Push lên GitHub tài khoản sinh viên**
* **README.md mô tả:**
  + **Giới thiệu**
  + **Công nghệ**
  + **Hướng dẫn cài đặt**
  + **Cách chạy hệ thống**

## **7.2 Chức năng của chương trình**

* **Lấy câu hỏi ngẫu nhiên từ database**
* **Hiển thị đúng nội dung câu hỏi và đáp án**
* **Chọn và lưu đáp án người dùng**
* **Đếm thời gian làm bài**
* **Tính điểm chính xác**

## **7.3 Coding Convention**

* **Đặt tên biến, hàm rõ nghĩa**
* **Tuân thủ PEP8 (Python)**
* **Comment đầy đủ**
* **Cấu trúc thư mục rõ ràng**

## **7.4 Triển khai thực tế**

* **Hệ thống chạy ổn định trên localhost**
* **Có ảnh chụp minh chứng**
* **Demo trực tiếp khi bảo vệ**

# **8. ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN**

## **8.1 Đánh giá**

**Hệ thống hoạt động ổn định, đáp ứng đúng yêu cầu đề tài, giao diện đơn giản, dễ sử dụng. Một số hình ảnh thành phần của ứng dụng:**

**api\_route.py:**

* **Gom các router con lại**
* **Gắn vào app chính (main.py)**

Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, màn hình

Nội dung do AI tạo ra có thể không chính xác.

**crud.py**

**👉 Xử lý truy vấn dữ liệu (Create – Read – Update – Delete)**

**📌 Thường chứa:**

* **Hàm lấy câu hỏi**
* **Hàm thêm câu hỏi**
* **Hàm cập nhật / xóa**

**📌 Lợi ích:**

* **Tách logic DB khỏi router**
* **Dễ bảo trì, dễ test**

**Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Nội dung do AI tạo ra có thể không chính xác.**

**quiz\_route.py**

**👉 ⭐ API chính của chức năng làm bài thi**

**📌 Thường có:**

* **/quiz/random → lấy câu hỏi ngẫu nhiên**
* **/quiz/submit → nộp bài (nếu có)**

**📌 File này:**

* **Giao tiếp trực tiếp với frontend**
* **Trả dữ liệu JSON**

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Nội dung do AI tạo ra có thể không chính xác.

**index.html**

**👉 Giao diện HTML chính**

**📌 Chức năng:**

* **Hiển thị trang làm bài**
* **Load JS để gọi API**
* **Render câu hỏi**

**📌 Dùng cho:**

* **Demo**
* **Triển khai nhanh**
* **Kết hợp FastAPI + Jinja2 (nếu có)**

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, màn hình

Nội dung do AI tạo ra có thể không chính xác.

**database.py**

**👉 ⭐ Kết nối PostgreSQL**

**📌 Chứa:**

* **engine**
* **SessionLocal**
* **Chuỗi kết nối DB**

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, màn hình

Nội dung do AI tạo ra có thể không chính xác.

**main.py**

**👉 ⭐ ENTRY POINT của FastAPI**

**📌 Chức năng:**

* **Tạo app FastAPI**
* **Include router**
* **Mount static / template**
* **Chạy server**

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, phần mềm

Nội dung do AI tạo ra có thể không chính xác.

**models.py**

**👉 Định nghĩa bảng Database (ORM)**

**📌 Ví dụ:**

* **Bảng questions**
* **Các cột: id, text, choices, answer\_key, meta**

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, phần mềm

Nội dung do AI tạo ra có thể không chính xác.

**schemas.py**

**👉 Định nghĩa dữ liệu vào/ra API**

**📌 Dùng Pydantic:**

* **Validate dữ liệu**
* **Đảm bảo frontend – backend thống nhất format**

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, phần mềm

Nội dung do AI tạo ra có thể không chính xác.

**session.py**

**👉 Quản lý phiên làm việc với DB**

**📌 Chứa:**

* **Dependency get\_db()**
* **Tự động đóng session sau request**

**📌 Rất quan trọng với FastAPI**

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, màn hình

Nội dung do AI tạo ra có thể không chính xác.

**frontend\_src\_api.js**

**👉 JS gọi API backend**

**📌 Chức năng:**

* **Fetch câu hỏi**
* **Submit bài**
* **Nhận kết quả**

**📌 Kết nối frontend ↔ backend**

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, màn hình

Nội dung do AI tạo ra có thể không chính xác.

**frontend\_src\_App.jsx**

**👉 Component chính của React**

**📌 Quản lý:**

* **State**
* **Routing**
* **Giao diện tổng**

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, màn hình

Nội dung do AI tạo ra có thể không chính xác.

**frontend\_src\_components\_\***

**👉 Các component nhỏ**

**Ví dụ:**

* **Question**
* **Timer**
* **Result**
* **Button**

**📌 Giúp code:**

* **Dễ đọc**
* **Dễ bảo trì**
* **Chuẩn React**

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, phần mềm

Nội dung do AI tạo ra có thể không chính xác.Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, máy tính

Nội dung do AI tạo ra có thể không chính xác.Ảnh có chứa đồ điện tử, văn bản, ảnh chụp màn hình, màn hình

Nội dung do AI tạo ra có thể không chính xác.

**start.py**

**👉 File chạy nhanh hệ thống**

**📌 Có thể:**

* **Gọi uvicorn**
* **Set config trước khi chạy**

Ảnh có chứa ảnh chụp màn hình, văn bản, phần mềm, màn hình

Nội dung do AI tạo ra có thể không chính xác.

## **8.2 Hướng phát triển**

* **Đăng nhập người dùng**
* **Phân loại độ khó câu hỏi**
* **Lưu lịch sử làm bài**
* **Thống kê kết quả**

# **9. KẾT LUẬN**

**Đề tài đã hoàn thành mục tiêu xây dựng hệ thống Quiz Online cơ bản.  
Kết quả đạt được cho thấy khả năng áp dụng kiến thức lập trình web và cơ sở dữ liệu vào thực tế.**