**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**



**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**WEBSITE QUẢN LÝ ĐỀ THI KẾT HỢP CHẤM BÀI THI TRẮC NGHIỆM BẰNG MÁY HỌC**

**Sinh viên: Lê Trung Hiếu**

**Mã số: B1606795**

**Khóa: K42**

***Cần Thơ, 01/2020***

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**



**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**WEBSITE QUẢN LÝ ĐỀ THI KẾT HỢP CHẤM BÀI THI TRẮC NGHIỆM BẰNG MÁY HỌC**

**Người hướng dẫn Sinh viên thực hiện**

TS. Nguyễn Thanh HảiLê Trung Hiếu

Mã số: B1606795

Khóa: K42

***Cần Thơ, 01/2020***

* + - * 1. LỜI CẢM ƠN

Trong thời gian học tập tại trường Đại Học Cần Thơ, nhờ sự dìu dắt của quý thầy cô, đặc biệt là quý thầy cô khoa Công Nghệ Thông Tin& TT đã dạy cho em những kiến thức quý giá để bước vào đời.

Việc hoàn thành bài luận văn này cũng là nhờ vào những kiến thức quý báo đó, em xin chân thành cảm ơn sự dạy dỗ quan tâm của các thầy, các cô và đặc biệt em xin được gửi lời cảm ơn sâu sắc đến giáo viên hướng dẫn đề tài luận văn là thầy Nguyễn Thanh Hải đã tận tình hướng dẫn, chỉ bảo và góp ý cho em trong suốt quá trình thực hiện luận văn.

Trong suốt quá trình làm việc, em đã cố gắng, nỗ lực rất nhiều để hoàn thành đề tài này một cách tốt nhất, nhưng sai sót nhỏ là điều khó tránh khỏi. Vì vậy, em rất mong nhận được những ý kiến đóng góp của quí thầy cô và các bạn để đề tài này có thể phát triển hoàn thiện hơn.

Xin chân thành cảm ơn!

* + - * 1. NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………..…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………..………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………....................................................................................................................................................

* + - * 1. NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………..…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………..………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………....................................................................................................................................................

**MỤC LỤC**

[LỜI CẢM ƠN 3](#_Toc60500936)

[NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN 4](#_Toc60500937)

[NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN 5](#_Toc60500938)

[DANH MỤC BẢNG 8](#_Toc60500939)

[DANH MỤC HÌNH 9](#_Toc60500940)

[CÁC KÝ HIỆU VÀ VIẾT TẮT 10](#_Toc60500941)

[TÓM TẮT 11](#_Toc60500942)

[**CHƯƠNG 2: ĐẶC TẢ YÊU CẦU** 25](#_Toc60500943)

[1. Yêu cầu về chức năng 25](#_Toc60500944)

[1.1. Yêu cầu về hệ thống 25](#_Toc60500945)

[1.2. Quản lý đề thi 25](#_Toc60500946)

[2. Yêu cầu phi chức năng 25](#_Toc60500947)

[2.1. Yêu cầu phi chức năng 25](#_Toc60500948)

[2.2. Phần cứng Ứng dụng 26](#_Toc60500949)

[**CHƯƠNG 3: THIẾT KẾ GIẢI PHÁP** 26](#_Toc60500950)

[1. Cơ sở lý thuyết 26](#_Toc60500951)

[1.1. CÔNG NGHỆ WEB 26](#_Toc60500952)

[1.1.1. Giới thiệu về API và RESTful API 26](#_Toc60500953)

[1.2. Hệ quản trị cơ sở dữ liệu MySQL 27](#_Toc60500954)

[1.3. Giới thiệu về Sequelize 28](#_Toc60500955)

[1.4. Giới thiệu về Reactjs 29](#_Toc60500956)

[1.5. Thư viện Axios 30](#_Toc60500957)

[1.6. Giới thiệu về Nodejs 30](#_Toc60500958)

[1.7. Giới thiệu về Express 31](#_Toc60500959)

[1.8. Flask REST API 31](#_Toc60500960)

[2. Công nghệ nhận diện 32](#_Toc60500961)

[2.1. Giới thiệu OpenCv 32](#_Toc60500962)

[3. Thiết kế hệ thống 33](#_Toc60500963)

[3.1. RESTful API 33](#_Toc60500964)

[3.1.1. Định dạng URI 33](#_Toc60500965)

[3.1.2. Định dạng status code từ response 33](#_Toc60500966)

[3.1.3. Ứng dụng web quản lý 33](#_Toc60500967)

[**CHƯƠNG 4: CÀI ĐẶT GIÃI PHÁP** 36](#_Toc60500968)

[1. REST API server 36](#_Toc60500969)

[2. Ứng dụng web quản lý 36](#_Toc60500970)

[3. Quá trình xử lý ảnh 36](#_Toc60500971)

[**CHƯƠNG 5: ĐÁNH GIÁ KIỂM THỬ** 36](#_Toc60500972)

[1. Kết quả đạt được 37](#_Toc60500973)

[1.1. Kết quả 37](#_Toc60500974)

[1.1.1. Về mặt kiến thức 37](#_Toc60500975)

[1.1.2. Kinh nghiệm thực tiễn 37](#_Toc60500976)

[1.1.3. Mục tiêu đề ra 37](#_Toc60500977)

[1.2. Hạn chế 38](#_Toc60500978)

[2. Hướng phát triển 38](#_Toc60500979)

[TÀI LIỆU THAM KHẢO 39](#_Toc60500980)

* + - * 1. DANH MỤC BẢNG

[Bảng 1 Đối tượng nghiên cứu của đề tài 22](#_Toc60483413)

* + - * 1. DANH MỤC HÌNH

[Hình 1. 2 Công nghệ nhận dạng đánh dấu OMR 17](#_Toc60483323)

[Hình 1. 3 Mô hình hoạt động cơ bản của REST 28](#_Toc60483324)

[Hình 1. 4 Logo của MySQL 29](#_Toc60483325)

[Hình 1. 5 Logo của thư viện Sequelize 29](#_Toc60483326)

[Hình 1. 6 Logo thư viện Reactjs 30](#_Toc60483327)

[Hình 1. 7 Logo Nodejs Express 32](#_Toc60483328)

[Hình 1. 8 Logo Flask framework 32](#_Toc60483329)

* + - * 1. CÁC KÝ HIỆU VÀ VIẾT TẮT
        2. TÓM TẮT

Trong những năm gần đây với sự phát triển của công nghệ và internet. Xu hướng trắc nghiệm hóa đang ngày phát triển.Với việc chuyển đổi hình thức thi và kiểm tra, từ hình thức thi tự luận sang trắc nghiệm đang rất phổ biến.Vì thế để dễ dàng cho việc quản lý các đề thi cũng như chấm các bài thi trắc nghiệm và cả hình thức thi trắc online như hiện nay, một cách dễ dàng nhanh chóng và ít tốn thời gian nhất. Sự cần thiết của một website chấm bài kiểm tra trắc nghiệm đóng vai trò quan trọng trong việc giảng dạy để tiết kiệm thời gian lẫn công sức.

Luận văn tốt nghiệp là một đề tài nghiên cứu xây dựng một website quản lý cho phép người dùng có thể dễ dàng quản lý. Website được thiết kế với các công nghệ mới hiện nay như Reactjs, Nodejs, Sequelize, Flask, ORM và OpenCV. Kết quả của đề tài là một website quản lý đáp ứng được các yêu cầu của người dùng, với giao diện thân thiện và dễ sử dụng.

Từ khóa: Website, Nodejs, Sequelize, Flask, ORM, OpenCV

# ABSTRACT

In recent years, with the development of technology and the internet. The trend of multiple-choice exams is growing day by day. With the change in exam format, from official essay exams to multiple choice exams is very popular. So for easy management, from multiple choice test management and multiple choice scores to easily, quickly and in the shortest time. A bubble sheet grading website necessary, it plays an important role in teaching to save both time and effort.

Graduation thesis is a research topic aimed at building a management website that allows users to easily manage and use. The site is designed with existing new technologies like Reactjs, Nodejs, Sequelize, Flask and using OMR and OpenCV. The result of the topic is a managed website that responds to user requirements, with a friendly and easy-to-use interface.

Keyword: Website, Nodejs, Sequelize, Flask, ORM

**Phần 1. Giới thiệu**

1. **Đặt vấn đề**
   * **Xu hướng chung**

Những năm gần đây công nghệ thông tin ngày càng phát triển đang dạng, công nghệ thông tin đang được áp dụng ngày càng nhiều vào trong thực tiễn đời sống của con người trong các lĩnh vực như kinh tế, công nghiệp, y khoa,… Nhờ vào công nghệ thông tin mà các công việc trở nên nhanh hơn, tiết kiệm được chi phí thời gian, công sức hơn. Trong đó đặc biệt là lĩnh vực giáo dục, các phần mềm công nghệ thông tin các website giống chúng ta để dàng quản lý, lưu trữ đồng thời công nghệ giúp cho việc giảng dạy ngày được nâng cao. Trong thi cử vẫn thế với xu hướng hiện nay việc trắc nghiệm hóa đang ngày được áp dụng phổ biến vào thi cử thông qua các kỳ thi. Với ưu điểm khách quan, chính xác và thuận tiện cho cả người ra đề và thí sinh đi thi, hình thức thi này đã được áp dụng ở hầu khắp các nước qua các kỳ thi tuyển sinh trung học, trung học phổ thông quốc gia.

Cùng với việc đổi mới phương pháp dạy và học, cùng với việc đổi mới hình thức thi cử cũng trở thành một việc làm cấp thiết xây dựng hệ thống giáo dục tiên tiến, đào tạo ra thế hệ trẻ với tầm kiến thức sâu rộng. Chính sự kiểm tra đầu ra của quá trình đào tạo này đã khiến cho hình thức kiểm tra trắc nghiệm được trãi rộng không co cụm, không trọng tâm. Người làm bài phải đảm bảo tính chính xác, không nhầm lẫn, tính chất của mỗi lựa chọn của câu trắc nghiệm chính là thử thách của người làm bài. Trắc nghiệm là hình thức sử dụng rộng rãi trong các kỳ thi. Với số lượng bài thi lớn, dụng cụ thô sơ (giấy đục lỗ hoặc quan sát), người chấm không tránh khỏi những nhầm lẫn xảy ra. Thời gian chấm bài cũng là vấn đề của người chấm và là yếu tố khách quan ảnh hưởng đến kết quả bài thi. Vận dụng những kiến thức đã học vào thực tiễn, tạo một công cụ giúp giải phóng sức lao động cho người chấm, những người đã vất vả trong lĩnh vực lao động trí óc. Và được sự cho phép của Khoa Công Nghệ Thông Tin & Truyền thông của trường Đại học Cần Thơ, em đã tiến hành thực hiện đề tài” Xây dựng website quản lý đề thi và chấm bài thi trắc nghiệm bằng máy học”.

1. **Những nghiên cứu liên quan**

Từ hệ thống quản lý học tập LMS-Moodle [hình 1]. [1]Moodle là một hệ thống quản lý học tập (Learning Management System - LMS hoặc người ta còn gọi là Course Management System hoặc VLE - Virtual Learning Environment) mã nguồn mở (do đó miễn phí và có thể chỉnh sửa được mã nguồn), cho phép tạo các khóa học trên mạng Internet hay các website học tập trực tuyến.

Moodle (viết tắt của Modular Object-Oriented Dynamic Learning Environment) được sáng lập năm 1999 bởi Martin Dougiamas, người tiếp tục điều hành và phát triển chính của dự án. Do không hài lòng với hệ thống LMS/LCMS thương mại WebCT trong trường học Curtin của Úc, Martin đã quyết tâm xây dựng một hệ thống LMS mã nguồn mở hướng tới giáo dục và người dùng hơn. Từ đó đến nay Moodle có sự phát triển vượt bậc và thu thu hút được sự quan tâm của hầu hết các quốc gia trên thế giới và ngay cả những công ty bán LMS/LCMS thương mại lớn nhất như BlackCT (BlackBoard + WebCT) cũng có các chiến lược riêng để cạnh tranh với Moodle.



Hình 1. Hệ thống quản lý học tập LMS-Moodle cục bộ trên Windows

Công nghệ nhận dạng đánh dấu OMR [hình 1.1] (OMR - Optical Mark Recognition): Là công nghệ nhận biết dấu quang học trên trang giấy với một định dạng nhất định. Công nghệ này có thể xác định các dấu hiệu đặc biệt đã được đánh dấu trên trang giấy tại các vị trí định trước bằng quang học. Nó cho phép điều khiển một số loại máy quét ảnh, nhập tự động và số hóa các dữ liệu ảnh với biểu mẫu được thiết kế phù hợp, kiểm tra, điều chỉnh điểm chọn trực quan trên ảnh quét và kết xuất ra báo cáo dạng text dễ truy nhập vào các phần mềm xử lý dữ liệu khác. Công nghệ này thường được áp dụng để xử lý các dữ liệu từ phiếu điều tra hay các bài thi trắc nghiệm.

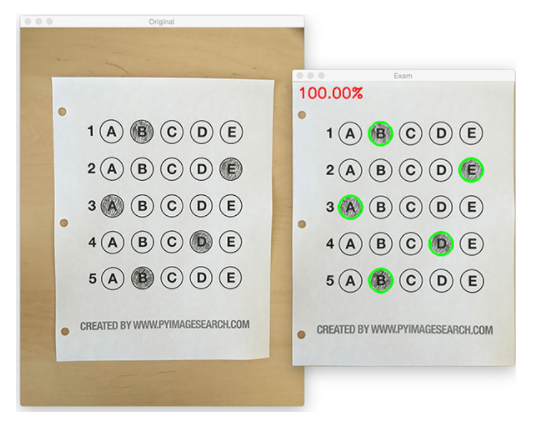
Tính năng ưu việt của công nghệ:

- Công nghệ có thể nhận biết các kích cỡ đánh dấu khác nhau với độ chính xác và linh hoạt cao.

- Điều chỉnh hình ảnh được scan để bù lại chất lượng thấp của máy scan - Sử dụng được với nhiều loại công cụ viết (bút chì, bút bi, bút nhớ…)

- Dễ dàng kiểm tra lại câu trả lời thông qua việc dùng xóa hoặc đánh dấu với kích cỡ lớn hơn

- Để tránh việc đọc sai hình ảnh, công nghệ OMR đọc dựa trên đánh dấu thời gian nên bù đắp được các lỗi kỹ thuật của máy scan

- Giúp tiết kiệm thời gian và chi phí

Hình 1. 1 Công nghệ nhận dạng đánh dấu OMR

1. **Mục tiêu đề tài**

Vì vậy, bài toán đặt ra là cần hệ thống quản lý một cách tối ưu với vấn đề thi trắc nghiệm và chấm bài thi trắc nghiệm.

Ở đây, vấn đề được giải quyết trong khuôn khổ nhà trường với đối tượng hướng đến là học sinh, sinh viên. Việc triển khai việc học và thi với hình thức trắc nghiệm từ cấp bậc trung học phổ thông là bước chạy đà để làm quen với chương trình trắc nghiệm ở bậc cao đẳng, đại học (sinh viên). Do đó quyết định chọn vấn đề xử lý việc quản lý một hệ thống phục vụ quản lý đề thi và chấm bài thi trắc nghiệm là lý do của đề tài “Hệ thống quản lý đề thi và chấm bài thi trắc nghiệm”.

Mong muốn góp một phần vào việc giúp đỡ học sinh, sinh viên và các giáo viên trong hình thức thi và chấm điểm này, tôi quyết định tìm hiểu và nghiên cứu một số vấn đề quan trọng sau:

* Tìm hiểu các loại câu hỏi trắc nghiệm về bản chất.
* Xây dựng một trang web quản lý đề thi, thi trắc nghiệm trực tuyến hoặc thi trên giấy và chấm bài thi trên giấy hỗ trợ cho việc rèn luyện , ôn tập kiến thức một cách cơ bản nhất với những chức năng sau :
* Quản lý các câu hỏi.
  + Chép phép người dùng thêm sử xóa câu hỏi.
* Quản lý các đề thi.
  + Cho phép người dùng tạo đề thi từ ngân hàng câu hỏi đã cho trước hoặc có thể import đề thi vào.
  + Tạo ra số lượng đề thi tướng ướng vế từng học sinh từ một đề thi góc.
  + Export đề thi và đáp án để giáo viên triển khai thi trắc nghiệm trên giấy
* Cho phép thi trắc nghiệm online.
* Cho phép chấm bài thi trên giấy theo mẫu có sẵn.
* Quản lý kết quả thi của từng học sinh.
* Quản lý được chất lượng của từng đề thi.

1. **Đối tượng và phạm vi nghiên cứu**
   * **Đối tượng nghiên cứu**

Đối tượng nghiên cứu ở đề tài chủ yếu là học sinh và giáo viên với việc thi trắc nghiệm phục vụ các môn học và việc ôn tập các kỳ kiểm tra và chấm các bài kiểm tra.

Ngoài ra, cũng sẽ nghiên cứu thêm về việc xây dựng một kiến trúc hệ thống ứng dụng sử dụng máy chủ dữ liệu (API server) và công cụ quản lý (Web client). Đối tượng nghiên cứu đầy đủ sẽ gồm các thành phần:

− Rest API server: xây dựng một server cung cấp các thao tác tạo, thêm, sửa,xóa (CRUD) dữ liệu của hệ thống thông qua HTTP request.

−Flask Framework: − Website: xây dựng một ứng dụng web trên desktop giao tiếp với server thông qua REST API, Flask nhằm mục đích quản lý dữ liệu cốt lõi của hệ thống.

−ORM và thư viện OpenCV để nhận diện bài kiểm tra trong hình ảnh, áp dụng chuyển đổi phối cảnh để được góc nhìn chính diện, trích xuất khu vực trả lời, phát hiện và sắp xếp các trả lời, xác định câu trả lời được đánh dấu và so với câu trả lời đúng của giáo viên nhập vào.

* + **Phạm vi nghiên cứu**

Phạm vi áp dụng của đề tài nghiên cứu là nội bộ trực thuộc các trường trung học phổ thông có hình thức kiểm tra trắc nghiệm.

Ngoài ra, sẽ sử dụng một vài thư viện Nodejs và Flask để xây dựng nhanh một hệ thống RESTful API và một số thư viện frontend như Reactjs, Axios để giao tiếp với API server để tạo giao diện người dùng web.

1. **Phương pháp nghiên cứu**

Phương pháp nghiên cứu chủ yếu là tìm hiểu thông qua tài liệu trên trang chủ của các thư viện, thông qua các video trên YouTube và làm các ví dụ trực tiếp. Từ đó, sẽ vận dụng các kiến thức đã tiếp thu được vào thực hiện đề tài. Tương ứng với mỗi thành phần của hệ thống sẽ có những phương pháp nghiên cứu khác nhau:

− API server: nghiên cứu cơ bản về các thư viện dùng để xây dựng nhanh một hệ thống RESTful API gồm Nodejs và Flask Framework và hệ quản trị cơ sở dữ liệu MySQL.

− Web client: nghiên cứu cơ bản về ứng dụng web sử dụng các thư viện react, axios.

* ORM, OpenCv: nghiên cứu cơ bản về các tính năng của các hàm xử lý.

1. **Nội dung nghiên cứu**

* Phân tích yêu cầu:

Phân chia thời gian, công việc hợp lý cho quá trình làm luận văn. Lên kế hoạch và báo cáo tiến độ làm việc, từ đó xem xét và điều chỉnh tiến độ cho phù hợp. Trong quá trình làm việc tranh thủ sự quan tâm giúp đỡ của giáo viên hướng dẫn cũng như các giáo viên hướng dẫn khác, trong quá trình thực hiện cần phải:

* Website quản lý đề thi
  + - Thu thập tài liệu nghiên cứu, biểu mẫu, dữ liệu có sẵn.
    - Tìm hiểu các loại câu hỏi có trong đề thi và chọn ra biểu mẫu thích hợp.
    - Xác định phạm vi và kết quả cần đạt được cho đề tài.
    - Viết đặc tả yêu cầu, chi tiết cho từng chức năng.
* Phân tích hệ thống
  + - Mô tả các thực thể trong hệ thống.
    - Xây dựng mô hình dữ liệu quan hệ mức quan niệm.
* Thiết kế cơ sở dữ liệu
* Thiết kế mô hình luận lý.
* Thiết kế mô hình vật lý.
* Thiết kế từ điển dữ liệu
* Tạo các ràng buộc trên cơ sở dữ liệu (Ràng buộc thực thể, tham chiếu, luận lý...)
* Thiết kế các chức năng
* Cài đặt và kiểm thử
  + Cài đặt website (trước tiên là chạy thử trên Localhost)
  + Nhập liệu cho hệ thống.
  + Kiểm thử tất cả các chức năng hệ thống
* Chỉnh sửa lỗi Viết báo cáo kết quả thực hiện.
* Nội dung và đối tượng nghiên cứu chi tiết qua bảng[bảng 1] dưới đây như sau:

|  |  |  |
| --- | --- | --- |
| STT | Đối tượng | Nội dung |
| 1 | API và RESTful API | − Tìm hiểu về API và RESTful API |
| 2 | MySQL | − Tìm hiểu hệ quản trị cơ sở dữ liệu MySQL  − Cấu hình MySQL trong sequelize |
| 3 | Sequelize | −Tìm hiểu cách sử dụng sequelize  −Cách kết nối với cơ sở dữ liệu |
| 4 | Reactjs | − Tìm hiểu về reactjs: components, state và props  − Routing bằng reactjs và chuyển hướng đăng nhập |
| 5 | Axios | − Tìm hiểu về axios –  − Trao đổi với API server thông qua axios |
| 6 | Nodejs | − Tìm hiểu về nodejs cách thức vận hành.  − Cách truyền dữ liệu  − Xây dựng RESTful API server |
| 7 | Flask | −Tìm hiểu về Flask Framework  −Route cách thức hoạt động  −Xây dựng API kết nối |
| 8 | Opencv | −Các thư viện hỗ trợ |
|  | Express |  |

Bảng 1 Đối tượng nghiên cứu của đề tài

1. **Bố cục quyển luận văn**

Bố cục luận văn gồm các phần và các chương sau:

Phần 1 – Giới thiệu, gồm:

* Đặt vấn đề
* Những nghiên cứu liên quan
* Mục tiêu đề tài
* Đối tượng và phạm vi nghiên cứu
* Phương pháp nghiên cứu
* Nội dung nghiên cứu
* Bố cục quyển luận văn

Phần 2 – Nội dung, gồm:

Chương 1. Đặc tả yêu cầu

* Yêu cầu chức năng
* Yêu cầu phi chức năng

Chương 2. Thiết kế giải pháp

* Cơ sở lý thuyết
* Thiết kế hệ thống

Chương 3: Cài đặt giải pháp

* REST API server
* Ứng dụng web quản lý

Chương 4. Đánh giá kiểm thử

* Mục tiêu kiểm thử
* Kịch bản kiểm thử
* Đánh giá kiểm thử

Phần 3 – Kết luận, gồm:

* Kết luận
* Hướng phát triển

**Phần 2: Nội dung**

1. **ĐẶC TẢ YÊU CẦU**
   1. Yêu cầu về chức năng
      1. Yêu cầu về hệ thống

* Quản lý đăng nhập/đăng xuất user bằng localStorage.
  + Phân quyền hệ thống cơ bản với người dùng với từng chức năng.
* Cho phép thi trực tuyến.
  + Với từng học sinh sẽ có mã đề riêng biệt và thứ tự các câu hỏi và đáp án đề khác nhau, đảm bảo được chất lượng của đề thi
    1. Quản lý đề thi
* Quản lý câu hỏi:
  + Nội dung câu hỏi, đáp án câu hỏi (cơ bản với 4 đáp án trắc nghiệm a, b, c, d), câu hỏi lý thuyết,thuộc chủ đề, độ khó của câu hỏi
* Quản lý tạo đề thi:
  + Chọn môn, chọn chủ đề trong môn, chọn mức độ khó câu hỏi (dễ, trung bình, khó), chọn số lượng câu hỏi cần thêm vào đề thi
  + Xáo trộn thứ tự câu hỏi trong bộ đề thi sinh mã đề mới, xáo trộn thứ tự đáp án trong câu hỏi
  + Cho phép xuất đề thi để thi ra dạng PDF và xuất đáp án của đề để thi trên giấy.
* Quản lý bài thi:
  + Tạo mật khẩu đề thi, quản lý tình trạng đề thi.
  + Xem lại kết quả của bài thi.
  + Thống kê điểm, chất lượng câu hỏi.
  1. Yêu cầu phi chức năng
     1. Yêu cầu phi chức năng
     + Giao diện Giao diện đơn giản, dễ sử dụng.
     + Đối với ứng dụng web quản lý thì giao diện sẽ thống nhất với một mẫu chung để hướng tới tính quen thuộc, dễ sử dụng.
     1. Phần cứng Ứng dụng
* Ứng dụng web chỉ hỗ trợ trên trình duyệt desktop

1. **THIẾT KẾ GIẢI PHÁP**
   1. Cơ sở lý thuyết
      1. CÔNG NGHỆ WEB
         1. Giới thiệu về API và RESTful API

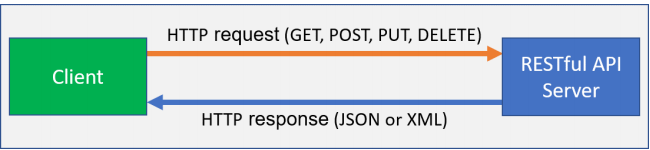
API (Application programming interface) là những quy tắc tương tác để các ứng dụng hoặc thành phần của ứng dụng có thể giao tiếp được với nhau. Mục đích của API là để đơn giản hóa việc lập trình bằng cách trừu tượng hóa đi những cài đặt phức tạp và chỉ phơi bày ra các cách thức để thực hiện hành động mà lập trình viên cần. Trong ngữ cảnh lập trình web, API là cách thức để ứng dụng trao đổi dữ liệu với một dịch vụ trực tuyến. API sẽ cung cấp các chức năng dịch vụ thông qua đường dẫn URL (Uniform Resource Locator) với định nghĩa thông số kỹ thuật như gửi request bằng HTTP (Hypertext Transfer Protocol) và trả về response message theo định dạng XML (Extensible Markup Language) hoặc JSON (JavaScript Object Notation).

REST (Representation State Transfer) là một kiến trúc phần mềm bao gồm các quy tắc để tạo ra dịch vụ web (webservice) [Hình 1. 2]. Một webservice tuân thủ theo kiến trúc REST thì được gọi là RESTful webservice. Webservice này sử dụng giao thức HTTP để triển khai kiến trúc web. Như vậy, RESTful API chính là kiến trúc thiết kế API tuân thủ theo kiến trúc REST thông qua các phương thức của HTTP (GET, POST, PUT, DELETE,..). Tương ứng với mỗi phương thức HTTP sẽ thực hiện những tác vụ tương ứng[bảng1. 1]:

|  |  |
| --- | --- |
| Phương thức HTTP | Tác vụ |
| GET | Lấy dữ liệu |
| POST | Tạo dữ liệu mới |
| PUT | Cập nhật dữ liệu |
| DELETE | Xóa dữ liệu |

Bảng 1. 1 Các tác vụ cơ bản của REST dựa trên phương thức HTTP

Các tác vụ đọc, tạo, cập nhật, xóa được gọi là CRUD service (Create, Read, Update, Delete). Mỗi tác vụ trên phải được gọi thông qua địa chỉ URI (Uniform Resource Identifier) kèm theo phương thức và payload (có thể có hoặc không, thường là định dạng XML hoặc JSON).



Hình 1. 2 Mô hình hoạt động cơ bản của REST

RESTful API sử dụng giao thức stateless (là một giao thức truyền thông không sử dụng session) và theo tiêu chuẩn nên hệ thống sẽ nhanh, đáng tin cậy và có thể mở rộng dễ dàng. Thông thường, RESTful API sẽ xác thực người dùng khi gửi yêu cầu đối với những tác vụ nguy hiểm như cập nhật hoặc xóa dữ liệu hoặc chỉ cho phép đối với người quản trị.

* + 1. Hệ quản trị cơ sở dữ liệu MySQL



Hình 1. 3 Logo của MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở miễn phí, được những doanh nghiệp lớn nhất thế giới như Facebook, Google, Adobe sử dụng để tiết kiệm thời gian và chi phí cho việc quản lý dữ liệu hiệu quả. MySQL được sáng lập bởi Michael "Monty" Widenius và David Axmark năm 1995. Hiện nay được phát triển bởi tập đoàn Oracle. MySQL hỗ trợ trên nhiều hệ điều hành: Windows, Linux, MacOS,... MySQL cũng được công nhận bởi DB-Engines là hệ quản trị cơ sở dữ liệu của năm 2019.

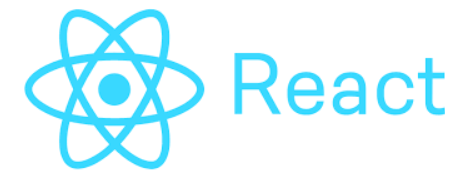
* + 1. Giới thiệu về Sequelize



Hình 1. 4 Logo của thư viện Sequelize

**Sequelize**: là một ORM dành cho NodeJS, hỗ trợ truy cập một cách dễ dàng đến các hệ quản trị cơ sở dữ liệu như PostgreSQL, MySQL, MariaDB, SQLite, MSSQL, … và cung cấp nhiều tính năng như transaction support, relations, eager và lazy loading, read replication và hơn nữa.

* + 1. Giới thiệu về Reactjs



Hình 1. 5 Logo thư viện Reactjs

**ReactJS (**hay React, React.js) là một thư viện mã nguồn mở (MIT License) dùng để xây dựng giao diện người dùng được viết bằng JavaScript. React được phát hành.lần đầu tiên vào năm 2013 bởi Facebook. React hiện nay được duy trì bởi Facebook và cộng đồng lập trình viên. React được sử dụng bởi những doanh nghiệp công nghệ hàng đầu như Facebook, Twitter, Instagram.

Sức mạnh của React nằm ở việc tự động cập nhật lại UI (user interfaces) khi dữ liệu thay đổi với chi phí ít nhất mà không cần phải tải lại toàn bộ trang web. Điều này làm cho ứng dụng phản hồi lại nhanh hơn, tiết kiệm băng thông, tăng tốc ứng dụng và tăng trải nghiệm người dùng. Trong React sẽ có một số khái niệm cơ bản về JSX, components, props, state và hook.

**SX** (JavaScript XML) là một cú pháp mở rộng của JavaScript cho phép định nghĩa các thành phần HTML trong React. Nói một cách đơn giản, cú pháp này cho phép ta viết các HTML tag trực tiếp trong JavaScript. Components là những thành phần UI được chia nhỏ ra, độc lập và có thể tái sử dụng. Component có thể là những function (stateless) hoặc class (stateful) trong JS.

**Component s**ẽ có các thuộc tính props (properties) và state (nếu được định nghĩa bằng class). Để phân biệt giữa React component và HTML tag, tất cả các React components phải được viết kiểu CamelCase (các cụm từ được viết liền nhau và bắt đầu mỗi từ bằng chữ in hoa, không có khoảng cách hoặc dấu câu xen kẽ) và phải bắt đầu bằng ký tự in hoa.

**Props** là một object được truyền vào trong một components, mỗi components sẽ nhận vào props và trả về react element. Props cho phép chúng ta giao tiếp giữa các components với nhau bằng cách truyền tham số qua lại giữa các components. Khi một components cha truyền cho component con một props thì components con chỉ có thể đọc và không có quyền chỉnh sửa nó bên phía components cha.

**State** là trạng thái thuộc về chính component đó, được quản lý bởi chính nó và không được truy xuất từ bên ngoài. Chỉ có thể sử dụng state khi dùng stateful component.

**Life cycle** là một vòng đời của một React component từ lúc được render lần đầu tiên và mỗi lần render lại (mounting) và khi gỡ bỏ component (unmounting). Có 2 phương thức được tự động gọi khi sự kiện mounting (component được render lần đầu tiên hoặc được render lại) và unmounting (component bị gỡ bỏ) xảy ra lần lượt là componentDidMount và componentWillUnmount. Có thể ghi đè 2 phương thức này khi sử dụng stateful component (class).

**Hook** là tính năng được thêm vào React ở phiên bản 16.8. Cho phép sử dụng một số tính năng chỉ có ở stateful component (class) khi dùng stateless component (function) như state (useState), life cycle (useEffect),..

* + 1. Thư viện Axios

Axios là một thư viện HTTP client dựa trên Promise dùng để gửi các request HTTP bất đồng bộ đến REST endpoint để sử dụng các dịch vụ CRUD.

Promise là một cải tiến để loại bỏ try catch, callback rườm rà trong xử lý bất đồng bộ, thay vào đó là then – catch. Hiểu đơn giản là khi thực hiện một tác vụ bất đồng bộ, sau khi thực hiện thành công thì “then” sẽ được gọi, ngược lại khi có lỗi phát sinh thì “catch” sẽ được gọi. Tương tự như promise, axios cũng có thể sử dụng theo kiểu gửi request lồng nhau.

Ngoài ra, axios còn hỗ trợ interceptor dùng để thực hiện một số công việc khác trước khi gửi request hoặc ngay khi nhận được response. Ví dụ như tạo mới access token khi nhận được response thông báo lỗi “401 – Unauthorized”.

* + 1. Giới thiệu về Nodejs



Hình 1. 6 Logo Nodejs Express

NodeJS: là một mã nguồn mở được xây dựng trên nền tảng Javascript V8 Engine. Nodejs sử dụng rộng bởi hàng ngàn lập trình viên trên toàn thế giới. Nó có thể chạy trên nhiều nền tảng hệ điều hành như Windows, Linux, MacOS. NodeJS cung cấp các thư viện phong phú ở dạng Javascript Module khác nhau giúp đơn giản hóa việc lập trình và giảm thời gian ở mức thấp nhất. Đặc tính nổi trội của Nodejs là tính bất đồng bộ. Điều này giúp các request được xử lý ngay lập tức.

* + 1. Giới thiệu về Express

Express: là một framework nhỏ nhưng linh hoạt, được xây dựng trên nền tảng NodeJS. Nó cung cấp nhiều tính năng mạnh mẽ để phát triển web và ứng dụng di động. Express có nhiều package hỗ trợ lập trình. Một số chức năng chính của express là thiết lập các lớp trung gian để trả về các request, định nghĩa router cho phép sử dụng với cá hành động khác nhau dựa trên phương thức HTTP và URL, cho phép trả về các trang HTML dựa vào các tham số.

* + 1. Flask REST API



Hình 1. 7 Logo Flask framework

Các dịch vụ REST API cho phép bạn tương tác với cơ sở dữ liệu bằng cách thực hiện các yêu cầu HTTP. Trong bài viết này, bạn học cách viết một máy chủ REST bằng Flask.

Đây thường là cách phần phụ trợ của ứng dụng web được tạo. Dữ liệu trả về ở định dạng JSON và các yêu cầu chúng tôi đang sử dụng là PUT, DELETE, POST và GET

* 1. Công nghệ nhận diện
     1. Giới thiệu OpenCv

OpenCv (Open Source Computer Vision) là một thư viện mã nguồn mở về thị giác máy với hơn 500 hàm và hơn 2500 các thuật toán đã được tối ưu về xử lý ảnh, và các vấn đề liên quan tới thị giác máy. OpenCv được thiết kế một cách tối ưu, sử dụng tối đa mạnh của các dòng chip đa lõi... để thực hiện các phép tính toán trong thời gian thực, nghĩa là tốc độ đáp ứng của nó thể đủ nhanh cho các ứng dụng thông thường.

OpenCv là thư viện được thiết kế để chạy trên nhiều nền tảng khác nhau (cross-platform), nghĩa là nó có thể chạy trên hệ điều hành Window, Linux, Mac, iOS... Việc sử dụng thư viện OpenCv tuân theo các quy định về sử dụng phần mềm mã nguồn mở BSD do đó bạn có thể sử dụng thư viện này một cách miễn phí cho các mục đích phi thương mại lẫn thương mại.

Thư viện có hơn 2500 thuật toán được tối ưu hóa, bao gồm một bộ toàn diện về cả thuật toán máy tính cổ điển và hiện đại và thuật toán học máy. Các thuật toán này có thể được sử dụng để phát hiện và nhận diện khuôn mặt, nhận dạng đối tượng, phân loại hành động của con người trong video, theo dõi chuyển động của camera, theo dõi các vật thể chuyển động, trích xuất mô hình 3D của vật thể, tạo ra các đám mây điểm 3D từ camera stereo, ghép các hình ảnh lại với nhau để tạo ra độ phân giải cao hình ảnh của toàn bộ cảnh, tìm hình ảnh tương tự từ cơ sở dữ liệu hình ảnh, loại bỏ mắt đỏ khỏi hình ảnh được chụp bằng đèn flash, theo dõi chuyển động của mắt, nhận ra cảnh quan và thiết lập các điểm đánh dấu để che phủ nó với thực tế tăng cường, v.v. OpenCV có hơn 47 nghìn người dùng cộng đồng và số lượt tải xuống ước tính hơn  18 triệu. Thư viện được sử dụng rộng rãi trong các công ty, nhóm nghiên cứu và bởi các cơ quan chính phủ.

Nhờ một hệ thống các giải thuật chuyên biệt, tối ưu cho việc xử lý ảnh, thị giác máy tính, các ứng dụng của OpenCv rất nhiều, có thể kể đến như:

* Nhận dạng ảnh: nhận dạng khuôn mặt, nhận dạng các vật thể, …
* Xử lý hình ảnh: khử nhiễu, điều chỉnh độ sáng, độ tương phản của ảnh, …
* Tìm kiếm và phục hồi hình ảnh/video
* Thực tế ảo
* Xử lý các đối tượng 3D
* Các thuật toán Machine learning & clustering
* Và còn rất nhiều ứng dụng khác
  1. Thiết kế hệ thống
     1. RESTful API
        1. Định dạng URI

# Các endpoint của REST API chính là URI. Định dạng URI chung cho các tác vụ CRUD được thiết kế theo quy tắc /api/<version>/<model>/<id>/. Trong đó:

# − Version: là phiên bản mà API cung cấp. Ví dụ v1, v2, ...

# − Model: là danh từ số nhiều chỉ model sẽ được thực hiện tác vụ CRUD. Ví dụ như thao tác CRUD với đồ nội thất thì model lúc này là “furnitures”.

# − Id: là số định danh của đối tượng, dùng để chỉ chính xác đối tượng của một model Dưới đây là danh sách mô tả các REST endpoint v1:

# −Tạo các API với các url theo từng phương thức để triển khai.

* + - 1. Định dạng status code từ response

Dưới đây là danh sách một số status code thông dụng được trả về từ response:

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Code** | **TextCode** | **Mô tả** |
| 1 | 200 | Ok | Thành công |
| 2 | 201 | Created | Tạo thành công |
| 3 | 401 | Unauthorized | Lỗi yêu cầu không được xác thực hợp lệ |
| 4 | 404 | Not Found | Lỗi server từ chối tiếp nhận loại phương thức HTTP cụ thể |
| 5 | 500 | Server Internal Error | Error Lỗi không mong muốn trên server nhằm từ chối việc thực hiện yêu cầu |

Hình 1. 8 Danh sách status code từ response

* + - 1. Ứng dụng web quản lý

1. **CÀI ĐẶT GIÃI PHÁP**
   1. REST API server
   2. Ứng dụng web quản lý
   3. Quá trình xử lý ảnh
2. **ĐÁNH GIÁ KIỂM THỬ**

**PHẦN 3: KẾT LUẬN**

* 1. Kết quả đạt được
     1. Kết quả
        1. Về mặt kiến thức
* Hiểu được quá trình xây dựng website nói chung và website thi trắc nghiệm trực tuyến nói riêng. Đồng thời cũng biết thêm kiến thức về lập trình web.
* Nâng cao khả năng phân tích và thiết kế hệ thống thông tin, khắc phục điểm yếu và phát huy điểm mạnh của bản thân.
* Bổ sung, củng cố và nâng cao kiến thức về cơ sở dữ liệu
* Lập trình thành thạo hơn về các ngôn ngữ và các thư viện như:
  + Nodejs
  + Reactjs
  + Sequelize
  + Flask framework
  + OpenCv
    - 1. Kinh nghiệm thực tiễn
* Nâng cao tính tự giác, kỹ năng tự học và nghiên cứu các vấn đề thắc mắc qua Internet.
* Củng cố kỹ năng tìm và khắc phục các vấn đề phát sinh.
* Phát huy tinh thần sáng tạo, có thêm kinh nghiệm lập trình thông qua việc giải quyết các vấn đề thực tiễn.
  + - 1. Mục tiêu đề ra

Xây dựng hoàn thiện hệ thống gồm 2 phần RESTful API server, web đạt được mục tiêu ban đầu đề ra.

* Quản lý đề thi
  + Dánh sach đề thi
  + Tạo đề thi từ ngân hàng câu hỏi
  + Tạo đề thi từ import file excel
  + Xuất được đề thi và đáp án để thi trên giấy
* Quản lý câu hỏi
  + Danh sách câu hỏi tho từng chương
  + Danh sách câu hỏi theo độ khó
* Quản lý thi
  + Tổ chức được thi online
  + Xuất được đề thi, thi trên giấy
* Quản lý điểm
  + Tổng hợp được kết quả.
  + Danh sách bài thi được được thực hiện từ đó quản lý được chất lượng đề thi củng như chất lượng câu hỏi tứ đó chọn câu hỏi phù hợp cho sinh viên học sinh.
    1. Hạn chế
* Website đang được phát triển ngày một hoàn thiện hơn nên hiện tại chưa hỗ trợ tính giao diện trên mobile.
* Ngoài ra, vẫn còn tồn tại một số lỗi nhỏ và đang tìm cách khắc phục ở phía server và client.
* Giao diện còn thô sơ khá đơn giãn chức năng còn nhiều thiếu soát.
* Chưa tối ưu hoá khả năng truy cập cũng như bảo mật hệ thống.
* Nhiều chức năng nâng cao như hệ thống gợi ý, so sánh, đánh giá, trao đổi trực tuyến… chưa được áp dụng.
  1. Hướng phát triển
* Tiến hành cải tiến các giải thuật cũng như áp dụng JWT Authentication để xét access truy cập
* Cần chỉnh sửa lại bố cục điều hướng để tạo nên sự thân thiện cho người dùng cũng như tăng cao độ bảo mật.
* Tích hợp thêm các chức năng nhập đề thi câu hỏi thông qua các biểu mẫu dữ liệu định dạng sẵn như file word, text.
* Tích hợp thêm các chức năng như đa ngôn ngữ, đa nền tảng, các giải thuật khai khoáng dữ liệu để hệ thống được hoàn chỉnh hơn.
* Điều chỉnh lại hệ thống với loại câu hỏi với số điểm riêng biệt để áp dụng cho nhiều loại hình thi trắc nghiệm hơn hay loại hình thi trắc nghiệm với nhiều lựa chọn đúng.
  + - * 1. TÀI LIỆU THAM KHẢO

[1] Moodle-va-e-learning. [Online]. Có sẵn tại: <https://oes.vn/moodle-va-e-learning-p1-khai-niem-tinh-nang-cach-cai-dat/> Ngày đầu tiên truy cập: 15/10/2020.

[2] Bubble sheet multiple choice scanner and test grader using OMR, Python and OpenCV. [Online]. Có sẵn tại: <https://www.pyimagesearch.com/2016/10/03/bubble-sheet-multiple-choice-scanner-and-test-grader-using-omr-python-and-opencv>/. Ngày đầu tiên truy cập: 15/10/2020.

[3] RESTful API. [Online]. Có sẵn tại: <https://topdev.vn/blog/restful-api-la-gi/> Ngày đầu tiên truy cập: 15/10/2020.