**TRƯỜNG ĐẠI HỌC VINH**

**VIỆN KỸ THUẬT VÀ CÔNG NGHỆ**

****

**ĐỒ ÁN HỌC PHẦN  
CÔNG NGHỆ PHẦN MỀM**

**XÂY DỰNG WEBSITE**

**VẬN HÀNH VÀ QUẢN LÝ QUY TRÌNH**

**THI TRẮC NGHIỆM KHÁCH QUAN**

**NHÓM: 05**

|  |  |
| --- | --- |
| GVHD: | TS. Cao Thanh Sơn |
| SVTH: | Đặng Ngọc Anh, 215748020110333 (NT) |
|  | Thái Văn Tuấn, 215748020110086 |
|  |
|  |

**Nghệ An, 12/202****4**

MỞ ĐẦU

Với sự phát triển không ngừng của ngành công nghệ thông tin, trí tuệ nhân tạo hay cụ thể hơn là học máy thì các hệ thống quản lý thông minh đã và đang góp phần tối ưu hóa quy trình làm việc, nâng cao hiệu quả và đảm bảo tính minh bạch trong mọi hoạt động. Trong đó việc áp dụng các giải pháp công nghệ phần mềm vào quản lý và vận hành quy trình của giáo dục ngày càng trở nên cần thiết. Hiểu rõ tầm quan trọng của việc ứng dụng kiến thức vào thực tiễn, chúng tôi đã lựa chọn triển khai thực hiện đồ án với đề tài "*Xây dựng Website vận hành và quản lý quy trình thi trắc nghiệm khách quan*" nhằm đưa ra một giải pháp hỗ trợ tốt hơn cho quá trình tổ chức, vận hành, quản lý và giám sát các kỳ thi trắc nghiệm trong môi trường giáo dục bậc đại học.

Qua đồ án này, chúng tôi tập trung xây dựng một hệ thống Website hướng đến việc tự động hoá toàn bộ quy trình tổ chức thi với các chức năng nổi bật như quản lý thông tin sinh viên, tổ chức thi trắc nghiệm, giám sát, và lưu trữ dữ liệu kết quả một cách tự động và hiệu quả. Đặc biệt, hệ thống sử dụng công nghệ thị giác máy tính để xác minh danh tính thí sinh, giám sát quá trình thi, nâng cao tính khách quan, tính chính xác và tính bảo mật thông tin.

Đồ án sử dụng ngôn ngữ lập trình Java với framework Spring Boot cho phần xử lý backend, framework FastAPI của Python để xử lý hình ảnh, hệ quản trị cơ sở dữ liệu MongoDB và cuối cùng là giao diện người dùng bằng HTML, CSS, và JavaScript. Việc tiếp cận với các framework và công nghệ đang được ứng dụng rộng rãi trong thực tế là bài học rất lớn không chỉ giúp ôn lại kiến thức đã học mà còn là cái nhìn tổng quát hơn cho định hướng lựa chọn công việc sau này.

Tuy nhiên chúng tôi nhận thức rõ rằng để xây dựng một Website hoàn chỉnh đòi hỏi lập trình viên có nhiều kinh nghiệm, sự nỗ lực, tính kỉ luật cao và khả năng giải quyết các thách thức phát sinh. Do đó sản phẩm trong phạm vi đồ án này không thể tránh khỏi nhiều thiếu sót, vì vậy nhóm mong nhận được sự đóng góp ý kiến của các thầy cô để có thể nhận ra, khắc phục và cải thiện hơn cho sản phẩm.

LỜI CẢM ƠN

Trước tiên, chúng tôi xin gửi lời cảm ơn sâu sắc và chân thành nhất đến TS. Cao Thanh Sơn, người đã tận tình hướng dẫn, đồng hành và hỗ trợ chúng tôi xuyên suốt quá trình thực hiện đồ án học phần Công nghệ phần mềm. Trong suốt thời gian qua, thầy không chỉ truyền đạt những kiến thức nền tảng vững chắc mà còn chia sẻ nhiều chỉ dẫn thực tế quý báu, giúp chúng tôi có thể áp dụng lý thuyết vào thực tiễn một cách hiệu quả. Sự tận tâm và những góp ý chân thành của thầy đã giúp chúng tôi hoàn thành đồ án này, đồng thời mang lại cho chúng tôi nhiều bài học bổ ích và kinh nghiệm quý giá cho chặng đường nghề nghiệp sau này.

Chúng tôi cũng xin bày tỏ lòng biết ơn sâu sắc đến toàn thể các thầy cô trong Viện Kỹ thuật Công nghệ, những người đã luôn tận tình giảng dạy, trang bị cho chúng tôi một nền tảng kiến thức vững chắc trong lĩnh vực công nghệ thông tin. Sự nhiệt huyết trong giảng dạy và tinh thần hỗ trợ của các thầy cô không chỉ giúp chúng tôi hiểu rõ các vấn đề học thuật mà còn khuyến khích chúng tôi sáng tạo và phát triển tư duy phản biện, tạo động lực mạnh mẽ để chúng tôi vững bước trên hành trình học tập và nghiên cứu.

Chúng tôi cũng không thể không nhắc đến sự đồng hành và nỗ lực của các thành viên trong nhóm, những người đã cùng nhau vượt qua những khó khăn và thử thách trong suốt quá trình làm việc. Đặc biệt, chúng tôi xin gửi lời cảm ơn sâu sắc đến bạn Thái Văn Tuấn, người đã luôn sát cánh, đóng góp ý kiến và hỗ trợ nhóm với tinh thần hợp tác, trách nhiệm và đoàn kết. Sự hỗ trợ lẫn nhau, sự kiên trì và nhiệt huyết của tất cả thành viên đã giúp chúng tôi hoàn thành đồ án này một cách tốt nhất.

Cuối cùng, chúng tôi chân thành mong muốn nhận được những góp ý, nhận xét từ quý thầy cô và các bạn để đồ án của chúng tôi ngày càng hoàn thiện và có giá trị thực tiễn hơn. Những ý kiến đóng góp ấy sẽ là nguồn động lực to lớn để chúng tôi tiếp tục phấn đấu và học hỏi trong tương lai.

Trân trọng cảm ơn,

*Đặng Ngọc Anh*,

*Thái Văn Tuấn.*

DANH MỤC CÁC HÌNH

[Hình 1.1: Ảnh mẫu thẻ sinh viên 5](#_Toc185332270)

[Hình 1.2: Hình ảnh minh hoạ danh sách 1 ca thi trắc nghiệm khách quan 6](#_Toc185332271)

[Hình 1.x: Giao diện hệ thống kiểm tra thi trực tuyến 10](#_Toc185332272)

[Hình 1.x: Ứng dụng xác minh khuôn mặt tại Vinpearl 11](#_Toc185332273)

[Hình 1.x: Ứng dụng xác minh khuôn mặt tại Vinpearl 11](#_Toc185332274)

[Hình 1.x: Ứng dụng thi lý thuyết trong sát hạch bằng lái xe ô tô 12](#_Toc185332275)

[Hình 2.x: Biểu đồ ca sử dụng quản lý quy trình thi của giám thị 18](#_Toc185332276)

[Hình 2.x: Biểu đồ tuần tự chức năng xử lý đầu vào và trích xuất thông tin trên thẻ 22](#_Toc185332277)

[Hình 2.x: Thiết kế giao diện của trích xuất thông tin sinh viên 23](#_Toc185332278)

[Hình 2.x: Biểu đồ ca sử dụng của giám thị trong chức năng so khớp thông tin 24](#_Toc185332279)

[Hình 2.x: Biểu đồ tuần tự chức năng đối chiếu thông tin với danh sách trong phòng thi 27](#_Toc185332280)

[Hình 2.x: Thiết kế giao diện của chức năng đối chiếu với các sinh viên trong phòng thi 28](#_Toc185332281)

[Hình 2.x: Biểu đồ ca sử dụng đối chiếu khuôn mặt của giám thị 29](#_Toc185332282)

[Hình 2.x: Biểu đồ ca sử dụng đối chiếu khuôn mặt của sinh viên 29](#_Toc185332283)

[Hình 2.x: Biểu đồ tuần tự chức năng đối chiếu khuôn mặt thực tế với ảnh thẻ sinh viên 33](#_Toc185332284)

[Hình 2.x: Giao diện của chức năng đối chiếu khuôn mặt 34](#_Toc185332285)

[Hình 2.x: Biểu đồ ca sử dụng chức năng quản lý sinh viên 37](#_Toc185332286)

[Hình 2.x: Biểu đồ ca sử dụng chức năng quản lý giám thị 40](#_Toc185332287)

[Hình 2.x. Biểu đồ ca sử dụng chức năng quản lý phòng thi 42](#_Toc185332288)

[Hình 3.x: Công cụ quản lý dự án Trello 46](#_Toc185332289)

[Hình 3.x: Phần lập trình Mô-đun tiền xử lý ảnh đầu vào 47](#_Toc185332290)

[Hình 3.x: Phần lập trình Mô-đun trích xuất văn bản 48](#_Toc185332291)

[Hình 3.x: Giao diện chức năng trích xuất thông tin trên thẻ sinh viên 50](#_Toc185332292)

[Hình 3.x: Mã nguồn Mô-đun đọc danh sách từ file Excel 55](#_Toc185332293)

[Hình 3.x: Mã nguồn Mô-đun so khớp thông tin sinh viên 56](#_Toc185332294)

[Hình 3.x: Giao diện chức năng so khớp thông tin sinh viên 57](#_Toc185332295)

[Hình 3.x: Mã nguồn Mô-đun nhận diện và cắt khuôn mặt có trên thẻ sinh viên 63](#_Toc185332296)

[Hình 3.x: Mã nguồn Mô-đun đối chiếu khuôn mặt đã cắt với khuôn mặt thực tế 65](#_Toc185332297)

[Hình 3.x: Mã nguồn Mô-đun in phiếu thi cho sinh viên 72](#_Toc185332298)

DANH MỤC CÁC BẢNG

[Bảng 1.1: Bảng mẫu ảnh thẻ và thông tin tương ứng 5](#_Toc185332305)

[Bảng 1.2: Bản mẫu dữ liệu phiếu thi trắc nghiệm khách quan hiện nay 8](#_Toc185332306)

[Bảng 1.3: Bảng mô tả yêu cầu chức năng 13](#_Toc185332307)

[Bảng 1.4: Bảng mô tả yêu cầu phi chức năng 14](#_Toc185332308)

[Bảng 2.1: Bảng mô tả yêu cầu phi chức năng 16](#_Toc185332309)

[Bảng 2.1: Dữ liệu lưu trữ thông tin sinh viên 20](#_Toc185332310)

[Bảng 2.x: Bảng chức năng xử lý đầu vào và trích xuất thông tin trên thẻ sinh viên 22](#_Toc185332311)

[Bảng 2.x: Dữ liệu lưu trữ thông tin phòng thi 25](#_Toc185332312)

[Bảng 2.x: Bảng chức năng đối chiếu thông tin sinh viên với danh sách phòng thi 26](#_Toc185332313)

[Bảng 2.x: Dữ liệu lưu trữ thông tin kết quả kiểm tra trước khi thi 31](#_Toc185332314)

[Bảng 2.x: Bảng chức năng đối chiếu khuôn mặt thực tế với ảnh thẻ sinh viên. 32](#_Toc185332315)

[Bảng 2.x: Bảng chức năng xuất phiếu thi cá nhân cho sinh viên 36](#_Toc185332316)

[Bảng 2.x: Quy trình quản lý sinh viên của quản trị viên 39](#_Toc185332317)

[Bảng 2.x: Quy trình quản lý giám thị của quản trị viên 41](#_Toc185332318)

[Bảng 2.x: Quy trình quản lý giám thị của quản trị viên 42](#_Toc185332319)

[Bảng 3.x: Bảng mô tả Mô-đun tiền xử lý ảnh đầu vào 47](#_Toc185332320)

[Bảng 3.x: Bảng mô tả Mô-đun trích xuất văn bản 49](#_Toc185332321)

[Bảng 3.x: Kiểm thử chức năng xử lý ảnh đầu vào và trích xuất thông tin 52](#_Toc185332322)

[Bảng 3.x: Bảng mô tả Mô-đun đọc danh sách từ file Excel 54](#_Toc185332323)

[Bảng 3.x: Bảng mô tả Mô-đun so khớp thông tin sinh viên 56](#_Toc185332324)

[Bảng 3.x: Kiểm thử chức năng so khớp thông tin sinh viên 61](#_Toc185332325)

[Bảng 3.x: Bảng mô tả Mô-đun nhận diện và cắt khuôn mặt có trên thẻ sinh viên 63](#_Toc185332326)

[Bảng 3.x: Bảng mô tả Mô-đun đối chiếu khuôn mặt đã cắt với khuôn mặt thực tế 65](#_Toc185332327)

[Bảng 3.x: Kiểm thử chức năng đối chiếu khuôn mặt trên thẻ sinh viên với ảnh thực tế 69](#_Toc185332328)

[Bảng 3.x: Bảng mô tả Mô-đun in phiếu thi cho sinh viên 70](#_Toc185332329)

MỤC LỤC

[MỞ ĐẦU ii](#_Toc185286614)

[LỜI CẢM ƠN iii](#_Toc185286615)

[DANH MỤC CÁC HÌNH iv](#_Toc185286616)

[DANH MỤC CÁC BẢNG v](#_Toc185286617)

[MỤC LỤC 1](#_Toc185286618)

[CHƯƠNG 1. MÔ TẢ BÀI TOÁN VẬN HÀNH VÀ QUẢN LÝ QUY TRÌNH THI TRẮC NGHIỆM KHÁCH QUAN 3](#_Toc185286619)

[1.1. Mô tả tổng quan bài toán 3](#_Toc185286620)

[1.2. Tìm hiểu bài toán vận hành và quản lý thi trắc nghiệm khách quan 4](#_Toc185286621)

[1.3. Một số ứng dụng liên quan đến đề tài 9](#_Toc185286622)

[1.4. Các yêu cầu cho hệ thống mới 12](#_Toc185286623)

[CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG VẬN HÀNH VÀ QUẢN LÝ QUY TRÌNH THI TRẮC NGHIỆM KHÁCH QUAN 15](#_Toc185286624)

[2.1. Phân tích yêu cầu và xác định tác nhân 15](#_Toc185286625)

[2.2. Sprint 1: Xử lý ảnh đầu vào và trích xuất thông tin trên thẻ sinh viên 18](#_Toc185286626)

[2.3. Sprint 2: Chức năng so khớp thông tin sinh viên với danh sách trong phòng thi 23](#_Toc185286627)

[2.4. Sprint 3: Đối chiếu khuôn mặt trên thẻ với ảnh thực tế 28](#_Toc185286628)

[2.5. Sprint 4: Xuất phiếu thi và giám sát quy trình thi 34](#_Toc185286629)

[2.6. Sprint 5: Các chức năng quản lý của quản trị viên 36](#_Toc185286630)

[2.7. Biều đồ ca sử dụng 43](#_Toc185286631)

[CHƯƠNG 3. XÂY DỰNG HỆ THỐNG VẬN HÀNH VÀ QUẢN LÝ QUY TRÌNH THI TRẮC NGHIỆM KHÁCH QUAN 44](#_Toc185286632)

[3.1. Môi trường phát triển 44](#_Toc185286633)

[3.2. Chức năng xử lý ảnh đầu vào và trích xuất thông tin trên thẻ sinh viên 46](#_Toc185286634)

[3.3. Chức năng so khớp thông tin sinh viên với danh sách phòng thi 53](#_Toc185286635)

[3.4. Chức năng đối chiếu khuôn mặt trên thẻ sinh viên với ảnh thực tế 61](#_Toc185286636)

[3.5. Chức năng xuất phiếu và giám sát quy trình thi 69](#_Toc185286637)

[3.6. Các chức năng quản lý của quản trị viên 73](#_Toc185286638)

[KẾT LUẬN 74](#_Toc185286639)

[Kết quả đạt được 74](#_Toc185286640)

[Những tồn tại và hạn chế 74](#_Toc185286641)

[TÀI LIỆU THAM KHẢO 75](#_Toc185286642)

1. MÔ TẢ BÀI TOÁN VẬN HÀNH VÀ QUẢN LÝ QUY TRÌNH THI TRẮC NGHIỆM KHÁCH QUAN

Việc sử dụng hình thức thi trắc nghiệm khách quan để kiểm tra và đánh giá kết quả học tập của sinh viên đang được các cơ sở giáo dục bậc đại học lựa chọn, dù vậy vẫn còn tồn tại nhiều khó khăn và thách thức. Trong chương đầu tiên này, chúng tôi sẽ trình bày bối cảnh thực tế dẫn đến việc chọn lựa đề tài, nhấn mạnh những yêu cầu cần thiết để xây dựng một hệ thống quản lý thi hiệu quả, và phân tích những thách thức đặt ra khi triển khai. Qua đó, chương cũng sẽ nêu ra và giới thiệu sơ bộ các công nghệ được sử dụng để cải thiện và khắc phục các nhược điểm hiện tại.

* 1. Mô tả tổng quan bài toán

Thực tế, tổ chức thi trắc nghiệm khách quan cho số lượng lớn sinh viên đã mang lại nhiều cải tiến so với hình thức thi truyền thống, như tăng tốc độ xử lý kết quả, chuẩn hóa cách đánh giá và giảm bớt áp lực chấm thi. Tuy nhiên, phương thức này vẫn còn nhiều điểm hạn chế cần khắc phục. Một trong những bất cập lớn nhất hiện nay là **quá trình xác minh danh tính sinh viên** trước mỗi ca thi vẫn chủ yếu được thực hiện thủ công. Phương pháp này không chỉ tiêu tốn thời gian mà còn dễ dẫn đến sai sót, nhất là với các kỳ thi có số lượng thí sinh tham gia lớn. Thêm vào đó, việc **phát phiếu thi chứa thông tin quan trọng** như tài khoản và mật khẩu truy cập thường diễn ra trước ca thi và trước cả bước xác minh danh tính. Đây là một lỗ hổng lớn trong khâu quản lý, tiềm ẩn nguy cơ gian lận, khi thông tin này có thể bị lạm dụng hoặc trao đổi trước giờ thi. Ở khâu **giám sát thi cử**, hầu hết các cơ sở giáo dục vẫn sử dụng phương pháp truyền thống, bố trí từ 1-2 giám thị cho mỗi phòng thi để quan sát trực tiếp sinh viên làm bài. Mặc dù phương pháp này đã được áp dụng từ lâu, nhưng nó đang bộc lộ nhiều hạn chế trước sự phát triển ngày càng tinh vi của các hành vi gian lận. Những bất cập trong quá trình vận hành này không chỉ làm giảm hiệu quả mà còn gây lãng phí thời gian, nguồn lực và thậm chí có thể ảnh hưởng đến uy tín của tổ chức giáo dục. Từ thực trạng trên việc ứng dụng các **giải pháp công nghệ hiện đại** để tự động hóa quy trình thi là một đòi hỏi cấp thiết. Các công nghệ như nhận diện khuôn mặt để xác minh danh tính, hệ thống giám sát thông minh sử dụng trí tuệ nhân tạo, hoặc phần mềm bảo mật để quản lý dữ liệu thi đều có thể giúp khắc phục những hạn chế hiện tại. Những giải pháp này không chỉ giúp giảm thiểu tối đa sai sót và gian lận mà còn tối ưu hóa thời gian, nhân lực trong tổ chức thi. Quan trọng hơn, chúng tạo ra một môi trường thi cử minh bạch, công bằng và hiệu quả hơn, góp phần nâng cao chất lượng giáo dục cũng như sự tin cậy của xã hội đối với hệ thống giáo dục hiện đại.

* 1. Tìm hiểu bài toán vận hành và quản lý thi trắc nghiệm khách quan

Hiện tại, trong công cuộc tổ chức thi trắc nghiệm khách quan yêu cầu giám thị thực hiện tuần tự nhiều bước nhằm đảm bảo tính chính xác và minh bạch. Đầu tiên, giám thị nhận danh sách sinh viên thông qua văn bản giấy được cấp phát ngay trước giờ thi, sau đó gọi tên từng sinh viên để xác nhận sự có mặt. Tiếp theo, giám thị đối chiếu thông tin cá nhân trên thẻ sinh viên, bao gồm họ tên và mã số, với danh sách để đảm bảo khớp đúng. Đồng thời, giám thị kiểm tra khuôn mặt thực tế của sinh viên với ảnh trên thẻ nhằm xác minh danh tính. Chỉ khi hoàn thành đầy đủ các bước xác minh này, sinh viên mới nhận được phiếu thi để tham gia thực hiện bài thi, nhằm đảm bảo quá trình diễn ra chặt chẽ, hạn chế gian lận và sai sót.

Qua thực tiễn từ bài toán này, để hướng đến tự động hóa và tối ưu quy trình, nhóm chúng tôi đề xuất các bước thực hiện theo thứ tự ưu tiên dựa trên mức độ quan trọng, tạo cơ sở cho các bước tiếp theo:

* + 1. Mô tả bước thực hiện đọc thông tin trên thẻ sinh viên

Trong thực tế giám thị phải trực tiếp kiểm tra thông tin trên thẻ sinh viên, đối chiếu mã số và họ tên với danh sách. Quy trình này dễ gặp phải những nhược điểm như mất thời gian khi số lượng sinh viên lớn, dễ xảy ra sai sót khi giám thị đọc nhầm thông tin hoặc không phát hiện kịp thời các gian lận như sử dụng thẻ giả hoặc thẻ không hợp lệ.

Việc tự động hóa bước đối chiếu thông tin từ thẻ sinh viên với danh sách không chỉ giúp giảm tải áp lực cho giám thị mà còn tăng cường độ chính xác và tốc độ xử lý. Thay vì giám thị kiểm tra lần lượt từng thẻ một, hệ thống sẽ tự động trích xuất các thông tin văn bản từ thẻ sinh viên. Điều này đảm bảo khâu quan trọng nhất trọng vận hành 1 ca thi trở nên hiệu quả, minh bạch và giảm thiểu tối đa sai sót thủ công.

Trong đó để xác minh danh tính của một sinh viên ta cần: ***mã số sinh viên*** và ***họ tên sinh viên***. Lý do lựa chọn hai trường này là bởi chúng là các yếu tố cơ bản, quan trọng và dễ dàng xác minh trong quá trình đối chiếu. *Mã số sinh viên* là một thông tin duy nhất, không trùng lặp, giúp nhận diện chính xác từng cá nhân, trong khi *họ tên sinh viên* giúp xác nhận thông tin một cách bổ sung. Các trường khác dù có trên thẻ sinh viên như *ngành học*, *khóa học*, *đơn vị* nhưng để tối ưu tốc độ xử lý, hệ thống chỉ tập trung vào hai trường thông tin cốt lõi này. Dưới đây là ảnh minh hoạ về các trường thông tin:

|  |  |
| --- | --- |
| **Ảnh mẫu thẻ sinh viên** | **Thông tin trên thẻ** |
| Hình 1.1: Ảnh mẫu thẻ sinh viên | Mã số sinh viên: 215748020110333  Họ và tên: Đặng Ngọc Anh  Ngành: Công nghệ thông tin  Đơn vị: Viện kỹ thuật công nghệ  Khoá học: 62 (2021-2026) |

Bảng 1.1: Bảng mẫu ảnh thẻ và thông tin tương ứng

Tiếp theo, hệ thống sẽ thực hiện bước so khớp sinh viên với danh sách phòng thi, đảm bảo rằng mỗi sinh viên được phân vào đúng phòng thi và không có sai sót nào trong việc phân bổ phòng.

* + 1. Mô tả bước thực hiện so khớp sinh viên với danh sách phòng thi

Chức năng so khớp thông tin sinh viên với danh sách phòng thi đóng vai trò quan trọng trong việc xác minh sinh viên có đến đúng phòng thi đã được phân công hay không. Với phương pháp hiện nay, giám thị sẽ nhận danh sách sinh viên bằng bản giấy ngay trước ca thi, sau đó thực hiện việc gọi tên từng sinh viên theo danh sách và đối chiếu với thông tin trên thẻ sinh viên. Tuy nhiên, quy trình này khá rườm rà và không tối ưu. Khi giám thị gọi tên mà sinh viên chưa có mặt, họ sẽ phải đánh dấu và tạm bỏ qua để xác minh sinh viên khác, tạo ra khoảng thời gian trống không cần thiết, làm chậm tiến độ xác minh dẫn đến thâm hụt thời gian cho cả giám thị và sinh viên đã có mặt.

A document with numbers and letters

Description automatically generated

Hình 1.2: Hình ảnh minh hoạ danh sách 1 ca thi trắc nghiệm khách quan

Hướng đến việc tự động hóa quy trình này, dựa trên thông tin đã được trích xuất từ thẻ sinh viên, hệ thống có thể tự động so sánh thông tin với danh sách được lấy ra từ cơ sở dữ liệu, từ đó xác nhận đúng sinh viên và phòng thi một cách nhanh chóng. Điều này giúp tối ưu hóa thời gian và tiện lợi hơn rất nhiều so với phương pháp thủ công. Được biết mỗi phòng thi sẽ bao gồm các trường thông tin như: *phòng thi, mã đề thi, mã số giám thị, số lượng tối đa sinh viên, và danh sách mã số sinh viên*. Qua các trường thông tin này ta chỉ cần lấy *mã số sinh viên* trong *danh sách mã số sinh viên* để đối chiếu với phần đã đọc được trên thẻ từ trước.

* + 1. Mô tả bước thực hiện xác minh khuôn mặt sinh viên

Việc đối chiếu khuôn mặt trên thẻ sinh viên với khuôn mặt thực tế của sinh viên là một công đoạn quan trọng để đảm bảo tính xác thực của danh tính. Hiện tại, giám thị sẽ phải dựa vào sự quan sát và kinh nghiệm của mình để so sánh hai khuôn mặt, một trên thẻ và một ngoài đời. Tuy nhiên, phương pháp này có nhược điểm lớn như việc dễ bị ảnh hưởng bởi yếu tố chủ quan, sự mệt mỏi của giám thị hoặc điều kiện ánh sáng không phù hợp, dẫn đến khả năng nhầm lẫn hoặc sai sót trong quá trình đối chiếu. Nếu sinh viên có thay đổi nhỏ về ngoại hình hoặc khuôn mặt bị mờ do góc chụp, quá trình xác minh cũng có thể gặp khó khăn.

Để tối ưu hóa và tự động hóa quá trình này, hệ thống sẽ sử dụng công nghệ nhận diện khuôn mặt để so sánh khuôn mặt trên thẻ sinh viên và khuôn mặt thực tế của sinh viên. Sau khi nhận được hình ảnh của thẻ sinh viên, hệ thống sẽ tiến hành nhận diện phân vùng chứa ảnh của sinh viên rồi tiến hành tách nó đồng thời sẽ so khớp với hình ảnh thực tế được chụp qua camera ngay lúc đó. Quá trình này không chỉ giúp tăng cường tính chính xác trong việc xác minh danh tính mà còn giảm thiểu tối đa sai sót do yếu tố con người. Việc sử dụng công nghệ nhận diện khuôn mặt cũng giúp tăng tốc độ kiểm tra và đảm bảo an toàn hơn so với phương pháp thủ công truyền thống, mang lại một giải pháp hiệu quả và dễ dàng hơn trong việc quản lý và giám sát sinh viên.

* + 1. Mô tả bước thực hiện phát phiếu dự thi cho sinh viên

Sau khi xác minh danh tính sinh viên, giám thị sẽ phát phiếu thi chứa các thông tin như tài khoản, mật khẩu bài thi cho sinh viên tương ứng. Hiện tại, việc phát phiếu thi thủ công gặp nhiều vấn đề: phiếu thi được sắp xếp theo thứ tự nhưng sinh viên đến xác thực có thể không theo thứ tự hoặc có sinh viên vắng mặt, dẫn đến phiếu thi bị bỏ qua hoặc gửi nhầm cho sinh viên khác. Việc in phiếu sẵn cho tất cả sinh viên tham gia thi cũng gây lãng phí khi có sinh viên vắng thi trong ca thi đó.

|  |  |
| --- | --- |
| **Môn thi** | Trí tuệ nhân tạo |
| **Ca thi** | 1(7:30-8:30) |
| **Tên đăng nhập** | Akgo2817 |
| **Mật khẩu** | Jkh63g2 |

Bảng 1.2: Bản mẫu dữ liệu phiếu thi trắc nghiệm khách quan hiện nay

Để giải quyết các bất cập này, chúng tôi đề xuất chức năng xuất phiếu thi ngay sau khi xác thực thành công. Cụ thể, phiếu thi sẽ được in ngay lập tức sau khi sinh viên xác thực thành công, với các thông tin như: *mã số sinh viên, tên môn thi, tên đăng nhập, mật khẩu, thời gian thi, và vị trí ngồi*. Việc này giúp sinh viên dễ dàng tìm được vị trí của mình và tránh các tình huống phát sinh như trong quy trình truyền thống.

Về phần giám sát quy trình thi, mặc dù phòng thi đã được trang bị camera, nhưng các camera này chưa được ứng dụng vào việc phát hiện gian lận thời gian thực, mà vẫn cần đến sự giám sát trực tiếp của giám thị. Với số lượng sinh viên trong phòng thi và các hành vi gian lận ngày càng tinh vi (như trao đổi bài, thay đổi vị trí ngồi, sử dụng tài liệu hoặc thiết bị công nghệ cao), chúng tôi đề xuất ứng dụng công nghệ truy vết đối tượng và giám sát hành vi để nhanh chóng và chính xác phát hiện các hành vi gian lận trong quá trình thi.

* + 1. Mô tả các chức năng của người quản lý

Hiện nay, việc quản lý trong các kỳ thi tại các cơ sở giáo dục phần lớn đã triển khai thông qua các hệ thống được xây dựng riêng phù hợp với từng hoàn cảnh và yêu cầu thực tế của cơ sở đó. Trong đồ án này chúng tôi chỉ nhắm đến việc xây dựng lại một vài chức năng liên quan đến đề tài. Các chức năng quản lý bao gồm quản lý sinh viên, quản lý giảng viên và quản lý phòng thi. Từng chức năng cụ thể được mô tả như sau:

* + - 1. Chức năng quản lý sinh viên

Người quản lý chịu trách nhiệm lưu trữ, cập nhật và theo dõi thông tin cá nhân của sinh viên, bao gồm *mã số sinh viên*, *họ tên*, *ngày sinh*, *lớp học*, *đơn vị* và *khóa học*. Ngoài ra, *kết quả thi* của từng sinh viên cũng cần được cập nhật, lưu trữ để thuận tiện trong việc đối chiếu thông tin và đánh giá năng lực trong suốt quá trình học tập. Bên cạnh đó, người quản lý còn theo dõi *trạng thái thi cử*, cập nhật danh sách sinh viên đủ điều kiện dự thi, vắng thi hoặc vi phạm quy chế để đưa ra các biện pháp xử lý kịp thời.

* + - 1. Chức năng quản lý giám thị

Trong kỳ thi, người quản lý phân công nhiệm vụ cho giám thị, đảm bảo phân bổ hợp lý các phòng thi, ca thi. Đồng thời, họ cũng theo dõi việc thực hiện nhiệm vụ của giám thị để đảm bảo tuân thủ quy chế thi cử và giám sát nghiêm túc quá trình thi. Để quản lý thông tin của giám thị ta vẫn sử dụng các trường như: *họ tên*, *mã số giám thị*, *số điện thoại* và *email.*

* + - 1. Chức năng quản lý phòng thi

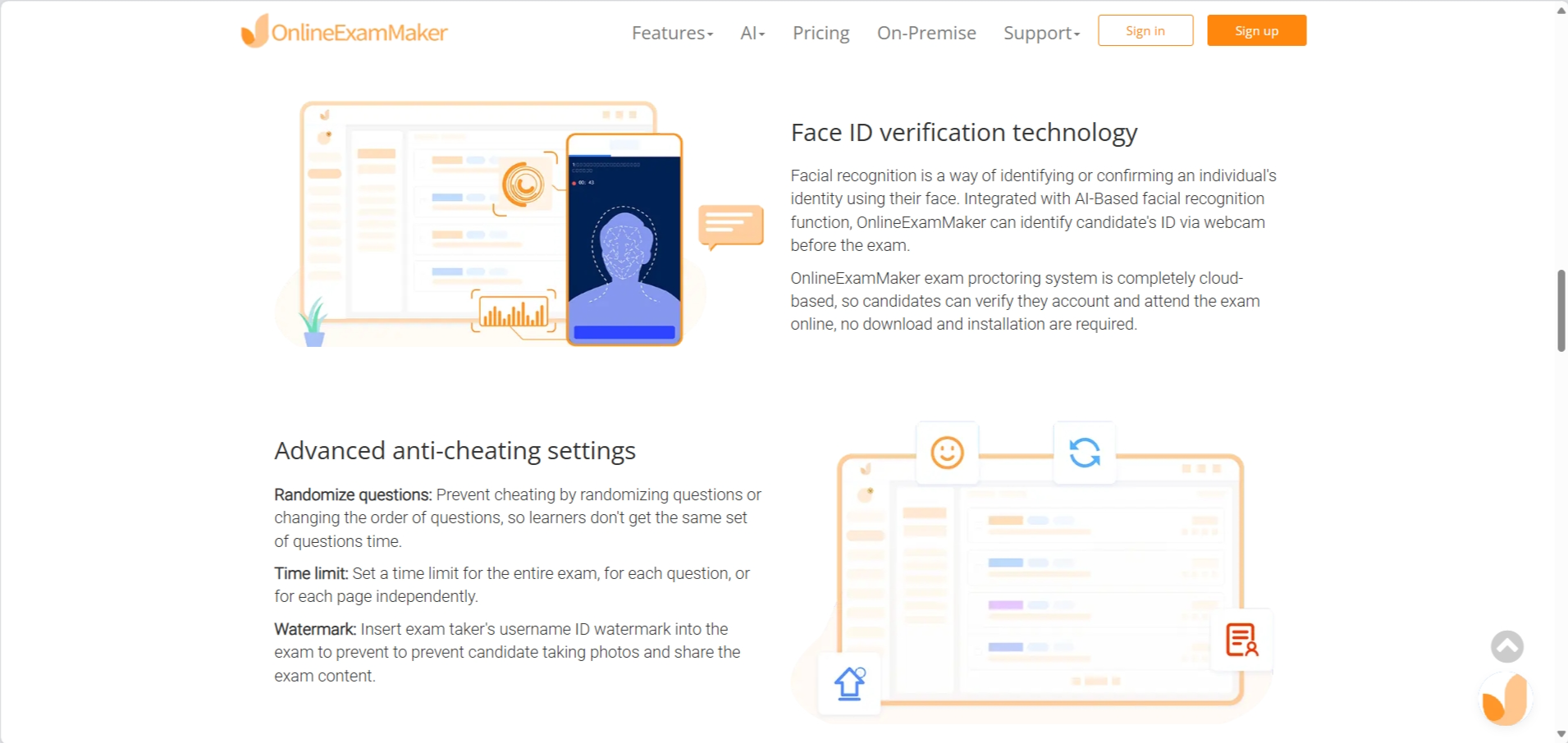
Người quản lý chịu trách nhiệm phân bổ phòng thi dựa trên số lượng sinh viên tham gia, đảm bảo không xảy ra tình trạng quá tải hoặc lãng phí không gian. Đồng thời quản lý cũng là người trực tiếp tạo phòng thi có ca thi cụ thể cho từ môn học tương ứng và chỉ định giám thị cũng như danh sách sinh viên sẽ xuất hiện trong phòng thi đó. Các thông tin để xác định một phòng thi bao gồm: *phòng thi*, *môn thi*, *giám thị*, *danh sách sinh viên* cũng như là trang thiết bị có trong phòng như: *máy tính* và *camera.*

* + 1. Giới thiệu thực tế quy trình triển khai, kiểm thử và bảo trì hệ thống
  1. Một số ứng dụng liên quan đến đề tài

Hệ thống nhận diện và xác minh danh tính đang ngày càng được ứng dụng rộng rãi trong nhiều lĩnh vực nhằm cải thiện hiệu quả quản lý, tăng cường an ninh và đảm bảo tính minh bạch. Đặc biệt, trong bối cảnh chuyển đổi số hiện nay, việc ứng dụng công nghệ nhận diện khuôn mặt không chỉ mang lại sự tiện lợi mà còn giúp tối ưu hoá quy trình làm việc. Dưới đây là một số ứng dụng liên quan đến đề tài:

1) Hệ thống kiểm tra thi trực tuyến

Một trong những ứng dụng nổi bật của công nghệ nhận diện khuôn mặt là trong các hệ thống tổ chức thi trực tuyến, giúp đảm bảo tính minh bạch và trung thực trong quá trình thi cử. Ví dụ, trang web <https://onlineexammaker.com/> đã áp dụng nhận diện khuôn mặt để xác minh danh tính thí sinh ngay trước mỗi ca thi. Hệ thống này không chỉ giúp loại bỏ rủi ro gian lận danh tính mà còn nâng cao hiệu quả trong quản lý số lượng lớn thí sinh từ xa. Thông qua tính năng này, các trung tâm thi có thể tự động kiểm tra danh tính thí sinh, đối chiếu với cơ sở dữ liệu, và ghi nhận thời gian tham gia của từng cá nhân. Ngoài ra, việc kết hợp với các tính năng khác như giám sát bằng webcam hay kiểm soát hành vi cũng giúp tăng cường độ tin cậy và chính xác cho hệ thống thi trực tuyến.



Hình 1.x: Giao diện hệ thống kiểm tra thi trực tuyến

2) Hệ thống xác minh danh tính bằng công nghệ nhận diện khuôn mặt tại Vinpearl

Công nghệ nhận diện khuôn mặt cũng đã được ứng dụng thực tế trong lĩnh vực du lịch và dịch vụ khách sạn, điển hình là hệ thống xác minh danh tính tại Tổ hợp Vinpearl Nha Trang. Đây là một trong những đơn vị đầu tiên tại Việt Nam triển khai công nghệ này nhằm tối ưu hóa trải nghiệm của khách hàng. Hệ thống sử dụng các thiết bị nhận diện khuôn mặt hiện đại để xác thực danh tính của du khách khi ra vào các khu vực như khách sạn, khu nghỉ dưỡng, khu vui chơi giải trí VinWonders, và nhà hàng. Thay vì các phương pháp kiểm soát truyền thống như xuất trình giấy tờ tùy thân hay thẻ ra vào, du khách chỉ cần quét khuôn mặt tại các cổng kiểm soát. Hệ thống tự động đối chiếu thông tin nhận diện với dữ liệu đã được lưu trữ trước đó để xác minh danh tính một cách nhanh chóng và chính xác.



Hình 1.x: Ứng dụng xác minh khuôn mặt tại Vinpearl



Hình 1.x: Ứng dụng xác minh khuôn mặt tại Vinpearl

3) Hệ thống thi lí thuyết trong sát hạch bằng lái xe ô tô

Trong lĩnh vực sát hạch bằng lái xe ô tô, việc ứng dụng công nghệ nhận diện khuôn mặt trong quy trình thi lý thuyết đã góp phần tăng cường tính minh bạch và chính xác. Trước khi bước vào ca thi, học viên được yêu cầu xuất trình căn cước công dân và đứng trước camera để hệ thống tiến hành xác minh danh tính.

A person holding up a card

Description automatically generated

Hình 1.x: Ứng dụng thi lý thuyết trong sát hạch bằng lái xe ô tô

* 1. Các yêu cầu cho hệ thống mới

Hệ thống nhận diện và xác minh danh tính sinh viên thông qua thẻ sinh viên và khuôn mặt cần phải đáp ứng một số yêu cầu chức năng và phi chức năng để đảm bảo hoạt động hiệu quả và ổn định. Dưới đây là các yêu cầu chi tiết cho hệ thống.

* + 1. Yêu cầu chức năng

|  |  |  |
| --- | --- | --- |
| **STT** | **Nội dung yêu cầu** | **Mô tả** |
| 1 | Hệ thống phải nhận diện các trường thông tin trên thẻ sinh viên. | Các trường cần nhận diện gồm mã số sinh viên, họ tên, ngày sinh, ngành học, khóa học, đơn vị, địa chỉ email, số điện thoại. Thông tin trích xuất được lưu trữ chính xác vào cơ sở dữ liệu để phục vụ các quy trình sau. |
| 2 | Hệ thống phải nhận diện khuôn mặt sinh viên qua camera. | So khớp khuôn mặt sinh viên với dữ liệu đã lưu trong cơ sở dữ liệu. Đảm bảo nhận diện chính xác trong các điều kiện khác nhau (ánh sáng yếu, góc nhìn khác). |
| 3 | Hệ thống phải đối chiếu thông tin từ thẻ và khuôn mặt để xác minh danh tính sinh viên. | Đối chiếu các thông tin với cơ sở dữ liệu hiện có. Quá trình xác minh phải nhanh, chính xác, không xảy ra sai sót. |
| 4 | Hệ thống phải lưu trữ thông tin sinh viên vào cơ sở dữ liệu. | Lưu trữ cả ảnh thẻ sinh viên và ảnh khuôn mặt. Thông tin phải được bảo mật và dễ dàng truy xuất. |
| 5 | Hệ thống phải quản lý thông tin phòng thi và danh sách sinh viên. | Quản lý danh sách sinh viên, giờ thi, môn thi. Xác nhận sinh viên được tham gia thi khi thông tin khớp với dữ liệu trong danh sách phòng thi. |
| 6 | Hệ thống cần kiểm tra và thông báo kết quả xác minh ngay lập tức. | Cung cấp thông báo “thành công” hoặc “thất bại” dựa trên độ khớp của thông tin. |
| 7 | Hệ thống cần có giao diện người dùng dễ sử dụng. | Giao diện được thiết kế cho sinh viên, giám thị và quản trị viên. Đơn giản, dễ hiểu, không yêu cầu kiến thức kỹ thuật đặc biệt. |

Bảng 1.3: Bảng mô tả yêu cầu chức năng

* + 1. Yêu cầu phi chức năng

|  |  |  |
| --- | --- | --- |
| **STT** | **Nội dung yêu cầu** | **Mô tả** |
| 1 | Hệ thống phải xử lý nhanh chóng và hiệu quả. | Đảm bảo trích xuất và xác minh thông tin nhanh chóng, Tránh gây trì hoãn trong các quy trình như tham gia thi. |
| 2 | Hệ thống phải có khả năng mở rộng. | Đáp ứng nhu cầu trong tương lai như số lượng sinh viên tăng hoặc tích hợp tính năng mới. |
| 3 | Hệ thống phải đảm bảo tính bảo mật cao. | Mã hóa các thông tin nhạy cảm như mã số sinh viên, thông tin cá nhân. Quy trình truy cập được bảo mật, chỉ người có quyền mới truy cập, chỉnh sửa dữ liệu. |
| 4 | Hệ thống phải đảm bảo tính khả dụng và độ tin cậy. | Luôn hoạt động ổn định trong quá trình triển khai. Có khả năng tự động khôi phục sau sự cố và tránh mất dữ liệu quan trọng. |
| 5 | Hệ thống phải tương thích với các thiết bị đầu vào và nền tảng khác nhau. | Hỗ trợ máy ảnh, webcam và các thiết bị nhận diện khuôn mặt. Hỗ trợ trên nhiều nền tảng như máy tính và thiết bị di động. |
| 6 | Hệ thống phải dễ bảo trì và cập nhật. | Dễ dàng nâng cấp phần mềm, triển khai tính năng mới hoặc sửa lỗi mà không làm gián đoạn hoạt động. |
| 7 | Hệ thống phải tương tác tốt với các hệ thống khác. | Tích hợp với hệ thống quản lý sinh viên, hệ thống thi cử. API phải được thiết kế để tương tác được với hệ thống bên ngoài. |

Bảng 1.4: Bảng mô tả yêu cầu phi chức năng

1. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG VẬN HÀNH VÀ QUẢN LÝ QUY TRÌNH THI TRẮC NGHIỆM KHÁCH QUAN

Chương 2 sẽ tập trung vào việc phân tích và thiết kế hệ thống vận hành và quản lý quy trình thi trắc nghiệm khách quan, nhằm hướng đến việc xây dựng một giải pháp hiện đại, hiệu quả, giảm thiểu các sai sót và tối ưu hóa quy trình kiểm tra, giám sát, cũng như đánh giá trong giáo dục. Nội dung chương được trình bày chi tiết qua ba phần chính: phân tích yêu cầu, thiết kế hệ thống, và xác định các yếu tố cần thiết để phát triển một hệ thống vận hành và quản lý quy trình thi trắc nghiệm khách quan hoàn chỉnh. Quá trình phát triển hệ thống sẽ áp dụng theo phương pháp mô hình Scrum, trong đó công việc được chia thành các nhiệm vụ nhỏ, triển khai theo từng giai đoạn (sprint), đảm bảo linh hoạt trong quá trình xây dựng và thích ứng kịp thời với những yêu cầu phát sinh. Thông qua đó chương sẽ cung cấp cái nhìn tổng quan về cách thức triển khai hoàn thiện hệ thống để xử lý các yêu cầu đã đặt ra.

* 1. Phân tích yêu cầu và xác định tác nhân
     1. Phân tích yêu cầu hệ thống

Mục tiêu của phân tích yêu cầu hệ thống là xác định các tính năng và yêu cầu mà hệ thống phải đáp ứng trong quá trình phát triển. Để đảm bảo tính hoàn chỉnh, hiệu quả và khả năng mở rộng của hệ thống, các yêu cầu được phân chia thành các Sprint cụ thể, mỗi Sprint có mục đích và yêu cầu rõ ràng. Dưới đây là mô tả tổng thể các yêu cầu hệ thống được chia theo từng Sprint:

|  |  |  |
| --- | --- | --- |
| **Sprint** | **Nội dung** | **Mục đích hướng đến** |
| 1 | Xử lý đầu vào và trích xuất thông tin trên thẻ sinh viên | Phát triển khả năng xử lý và trích xuất thông tin từ ảnh thẻ sinh viên. Sprint này tập trung vào việc phát triển tính năng tải ảnh, xử lý ảnh (chỉnh sửa độ sáng, độ tương phản, xoay ảnh) và trích xuất các thông tin như mã số sinh viên, tên, lớp học, khoa/ngành học từ thẻ. |
| 2 | So khớp thông tin sinh viên với danh sách phòng thi | Phát triển tính năng kiểm tra tính hợp lệ của thông tin trích xuất từ ảnh thẻ sinh viên và đối chiếu với cơ sở dữ liệu để xác nhận danh tính. |
| 3 | Đối chiếu khuôn mặt trên thẻ với ảnh thực tế | Tích hợp khả năng nhận diện khuôn mặt từ ảnh thẻ sinh viên và khuôn mặt thực tế trong môi trường thi để xác nhận đúng danh tính sinh viên. |
| 4 | Xuất phiêu và giám sát quy trình thi | Xây dựng khả năng quản lý, lưu trữ và truy xuất thông tin sinh viên, bao gồm việc lưu trữ dữ liệu ảnh thẻ, thông tin sinh viên và kết quả xác nhận. |
| 5 | Các chức năng quản lý của quản trị viên | Đảm bảo hệ thống hoạt động ổn định, hiệu quả và đạt được hiệu suất cao khi xử lý lượng lớn dữ liệu. |
| 6 | … | … |

Bảng 2.1: Bảng mô tả yêu cầu phi chức năng

Thông qua các Sprint này, hệ thống sẽ dần được xây dựng từ các tính năng cơ bản đến các tính năng phức tạp hơn, đồng thời đảm bảo chất lượng và hiệu quả hoạt động của hệ thống trong môi trường thực tế.

* + 1. Xác định các tác nhân tác động lên hệ thống

Hệ thống vận hành và quản lý quy trình thi trắc nghiệm khách quan xác định ba tác nhân chính là *Sinh viên*, *Giám thị* và *Quản trị viên*. Mỗi tác nhân có vai trò và trách nhiệm cụ thể trong việc đảm bảo hệ thống vận hành trơn tru, hiệu quả và minh bạch.

* + - 1. Sinh viên

Sinh viên là đối tượng chính tham gia vào các kỳ thi và thực hiện các nhiệm vụ liên quan trong hệ thống. Trách nhiệm của sinh viên bao gồm việc cung cấp đầy đủ và chính xác thông tin cá nhân, tuân thủ quy trình xác minh danh tính trước mỗi ca thi và tham gia thi theo các yêu cầu đặt ra.

* + - 1. Giám thị

Giám thị là tác nhân hỗ trợ và giám sát quá trình thi, đóng vai trò quan trọng trong việc đảm bảo sự trung thực và minh bạch trong suốt kỳ thi. Giám thị đăng nhập vào hệ thống qua tài khoản được cấp và thực hiện nhiều chức năng như kiểm tra danh sách sinh viên tham gia ca thi, giám sát quy trình xác minh danh tính tự động và hỗ trợ theo dõi nhằm phát hiện các hành vi gian lận trong phòng thi. Ngoài ra, giám thị còn chịu trách nhiệm ghi nhận sự cố hoặc vi phạm xảy ra trong kỳ thi và báo cáo kết quả sau khi ca thi kết thúc. Vai trò của giám thị giúp đảm bảo các quy trình thi cử được diễn ra một cách chặt chẽ và nghiêm túc.

* + - 1. Quản trị viên

Quản trị viên là tác nhân thiết yếu phải có trong hệ thống, chịu trách nhiệm quản lý và điều phối toàn bộ các hoạt động nhằm đảm bảo hệ thống vận hành ổn định, dữ liệu được bảo mật và các yêu cầu người dùng được đáp ứng. Với quyền hạn cao nhất, quản trị viên thực hiện nhiều chức năng quan trọng như quản lý người dùng, bao gồm tạo, chỉnh sửa hoặc xóa tài khoản của giám thị và sinh viên, đảm bảo quyền truy cập hợp lệ và hợp lý trong hệ thống. Bên cạnh đó, quản trị viên có thể tùy chỉnh các tham số hệ thống như quy trình thi, thông tin kỳ thi và cài đặt bảo mật để đáp ứng nhu cầu sử dụng và đảm bảo tính linh hoạt của hệ thống. Ngoài ra, họ còn chịu trách nhiệm theo dõi toàn bộ quá trình thi, bao gồm thời gian tham gia của sinh viên, trạng thái nhận diện khuôn mặt và hoạt động của giám thị, nhằm phát hiện và xử lý kịp thời các bất thường. Để duy trì tính toàn vẹn của thông tin, quản trị viên quản lý cơ sở dữ liệu thông qua các hoạt động như sao lưu, phục hồi và bảo mật dữ liệu, ngăn ngừa mất mát hoặc rò rỉ thông tin. Trong trường hợp phát sinh sự cố, quản trị viên tiếp nhận các báo lỗi hoặc khiếu nại từ giám thị và sinh viên, xử lý nhanh chóng để đảm bảo hệ thống hoạt động liên tục và ổn định. Với vai trò điều hành và giám sát toàn diện, quản trị viên không chỉ đảm bảo rằng mọi quy trình trong hệ thống diễn ra hiệu quả và minh bạch, mà còn đóng vai trò quyết định trong việc nâng cao chất lượng vận hành và hiệu quả tổng thể của hệ thống.

* 1. Sprint 1: Xử lý ảnh đầu vào và trích xuất thông tin trên thẻ sinh viên
     1. Giới thiệu Sprint 1

Sprint 1 của dự án tập trung vào việc triển khai quy trình xác minh danh tính sinh viên qua ảnh thẻ sinh viên mà người dùng tải lên. Trong bước đầu, giám thị sẽ yêu cầu sinh viên đưa thẻ để chụp ảnh, sau đó hệ thống tiếp nhận ảnh và thực hiện tiền xử lý để chuẩn hóa. Quá trình này bao gồm các thao tác như cắt thẻ, điều chỉnh độ sáng, độ tương phản và làm thẳng ảnh nếu cần thiết, giúp ảnh trở nên rõ ràng và dễ xử lý. Sau khi hoàn tất tiền xử lý, hệ thống sử dụng công nghệ nhận dạng ký tự quang học (OCR) để trích xuất các thông tin quan trọng từ thẻ, bao gồm mã số sinh viên, tên, ngày tháng năm sinh và khoa/ngành học. Các thông tin này sẽ được ghi nhận và chuẩn bị cho các bước xác thực danh tính tiếp theo. Sprint 1 dự kiến kéo dài trong 11 ngày, từ 30/9/2024 đến 13/10/2024, bao gồm các công việc như thu thập và chuẩn bị dữ liệu mẫu, xây dựng chức năng trích xuất thông tin từ thẻ sinh viên, huấn luyện Mô hình Yolo v11 với tập dữ liệu THE SINH VIEN, thiết kế giao diện và tích hợp chức năng đối chiếu với hệ thống backend và cơ sở dữ liệu lưu trữ thông tin sinh viên, cuối cùng là kiểm thử mô hình và API trích xuất thông tin.

A diagram with black text and white text

Description automatically generated

Hình 2.x: Biểu đồ ca sử dụng quản lý quy trình thi của giám thị

* + 1. Thiết kế dữ liệu sinh viên

Để lưu trữ thông tin sinh viên như đã đề cập trong Chương 1, phần này chỉ trình bày chi tiết về thiết kế bảng dữ liệu nhằm đáp ứng yêu cầu lưu trữ và quản lý hiệu quả. Bảng dữ liệu được xây dựng dựa trên việc phân tích các thông tin của sinh viên, bổ sung thêm một số trường thông tin cần thiết cho quá trình xây dựng hệ thống sau này. Các thông tin cụ thể về cấu trúc và nội dung của bảng dữ liệu lưu trữ thông tin sinh viên được mô tả chi tiết trong **Bảng 2.1.**

*Mục đích*: Lưu trữ các thông tin của sinh viên.

*Tên bảng*: Collection Students

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Diễn giải** | **Tên trường** | **Kiểu** | **Mặc định** | **Ràng buộc** | **Ghi chú** |
| 1 | MongoDB tự tạo | \_id | ObjectId |  |  |  |
| 2 | Mã sinh viên | std\_id | String |  | PK |  |
| 3 | Họ tên sinh viên | std\_name | String |  |  |  |
| 4 | Ngày sinh | std\_dob | Date |  |  |  |
| 5 | Lớp | std\_class | String |  |  |  |
| 6 | Ngành | std\_faculty | String |  |  |  |
| 7 | Đơn vị | std\_course | String |  |  |  |
| 8 | Email | std\_email | String |  |  |  |
| 9 | Số điện thoại | std\_phone | String |  |  |  |
| 10 | Kết quả thi | exam\_result | Object |  | FK |  |
| 11 | Trạng thái | status | Boolean | 1 |  | Mô tả tình trạng của sinh viên(đang học/đã nghỉ) |

Bảng 2.1: Dữ liệu lưu trữ thông tin sinh viên

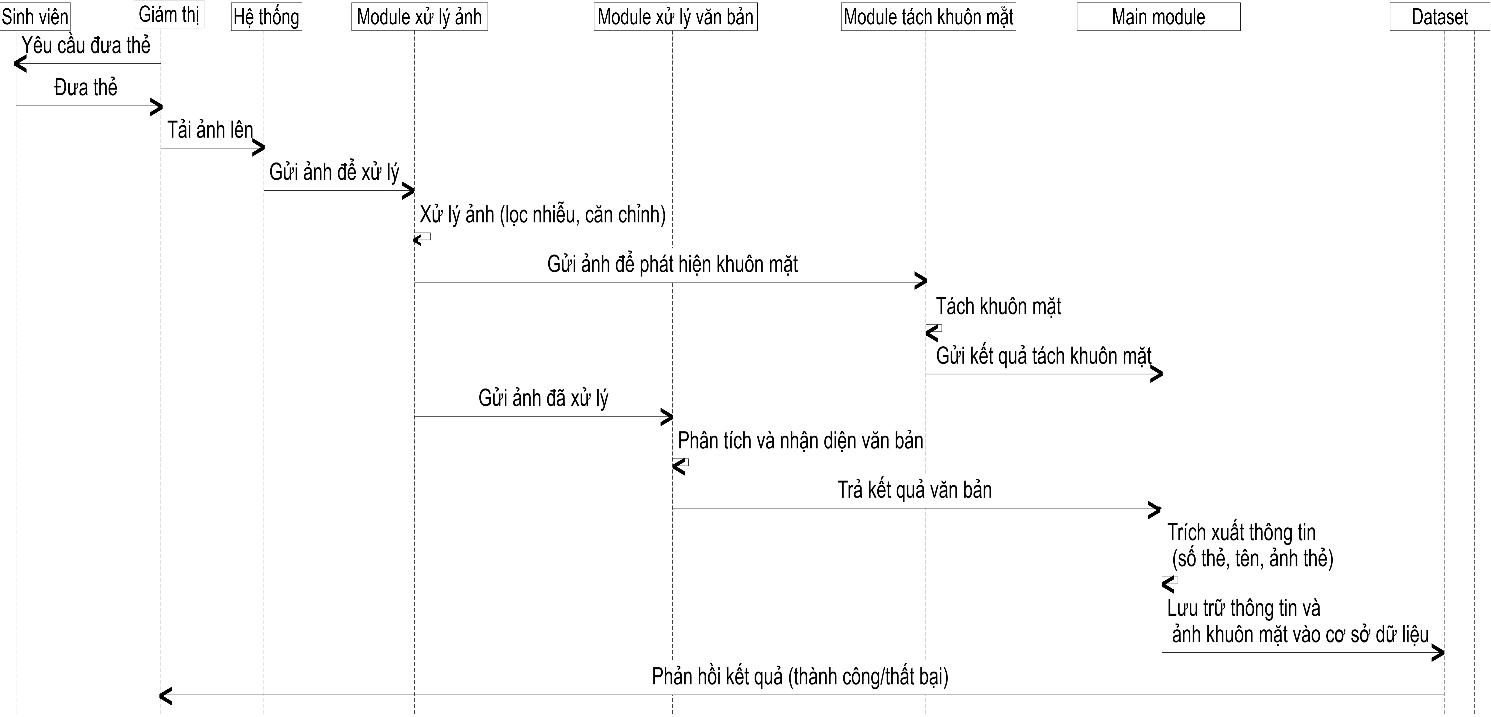
* + 1. Quy trình xử lý ảnh đầu vào và trích xuất thông tin

Quy trình xử lý ảnh đầu vào và trích xuất thông tin trên thẻ sinh viên được thiết kế để vận hành tự động và tuần tự nhằm đảm bảo độ chính xác và tối ưu thời gian thực hiện. Với kết quả mong muốn là đọc và lưu trữ được các thông tin trên thẻ thì sau khi nhận được ảnh thẻ sinh viên, hệ thống sẽ tiến hành các bước với chi tiết như sau:

|  |  |
| --- | --- |
| **Mục đích** | Đảm bảo trích xuất chính xác thông tin sinh viên từ ảnh thẻ sinh viên để phục vụ cho quá trình xác nhận danh tính. |
| **Các bước thực hiện** | **Bước 1:** Tiếp nhận ảnh thẻ sinh viên từ người dùng thông qua giao diện hoặc nguồn dữ liệu.  **Bước 2:** Áp dụng các thuật toán phát hiện, nhận diện để xác định vùng chứa thẻ sinh viên trong ảnh.  **Bước 3:** Cắt và làm thẳng ảnh thẻ, chuyển thẻ về cùng 1 dạng ma trận màu, đảm bảo thẻ hiển thị rõ ràng và đúng góc độ.  **Bước 4:** Sử dụng OCR (như PaddleOCR) để trích xuất thông tin chữ trên thẻ sinh viên, như mã số sinh viên, tên, ngày sinh, lớp.  **Bước 5:** Xử lý và chuẩn hóa dữ liệu trích xuất để đưa vào hệ thống quản lý. |
| **Các bước**  **bổ sung** | * Áp dụng các phương pháp tiền xử lý ảnh, như làm mịn, điều chỉnh độ tương phản, để cải thiện độ chính xác của OCR. * Sử dụng thuật toán sửa lỗi ký tự để tăng tính chính xác cho dữ liệu trích xuất từ thẻ. * Sử dụng kỹ thuật phát hiện và cắt vùng chứa văn bản (text region để cô lập vùng cần xử lý. * Tối ưu hóa góc nghiêng và cân chỉnh hình ảnh bằng các phương pháp xoay và biến đổi hình học (geometric transformations). * Sử dụng thuật toán sửa lỗi ký tự như các kỹ thuật phát hiện và thay thế lỗi chính tả (spell checking) để tăng tính chính xác cho dữ liệu trích xuất từ thẻ. |
| **Các ngoại lệ** | * Ảnh đầu vào mờ hoặc bị nhòe, dẫn đến việc không thể trích xuất chính xác thông tin. * Thẻ sinh viên bị hư hỏng hoặc thông tin không rõ ràng. * OCR không nhận diện được các ký tự đặc biệt, cần có cơ chế sửa thủ công hoặc thông báo lỗi. * Kích thước ảnh quá lớn hoặc quá nhỏ, gây khó khăn trong quá trình trích xuất, cần bổ sung bước tự động điều chỉnh kích thước ảnh. |
| **Tài liệu liên quan** | * PaddleOCR: Công cụ nhận dạng ký tự quang học và trích xuất văn bản từ hình ảnh. * OpenCV: Thư viện xử lý hình ảnh và video hỗ trợ các bước tiền xử lý như làm mờ, phân ngưỡng và cân chỉnh ảnh. * Các phương pháp tiền xử lý ảnh: Các kỹ thuật làm sạch và tối ưu ảnh trước khi đưa vào OCR. * Tài liệu phân tích và xử lý ký tự: Phương pháp sửa lỗi chính tả, đối chiếu ký tự, và tối ưu hóa thông tin trích xuất. * Các thuật toán phát hiện cạnh và vùng chứa văn bản như Canny Edge Detection và Contour Detection. |
| **Tham chiếu** | Mục 1.6.1.1: Xử lý đầu vào và trích xuất thông tin trên thẻ sinh viên  Mục 3.3.1.1: Mô-đun xử lý đầu vào và trích xuất thông tin trên thẻ sinh viên. |

Bảng 2.x: Bảng chức năng xử lý đầu vào và trích xuất thông tin trên thẻ sinh viên

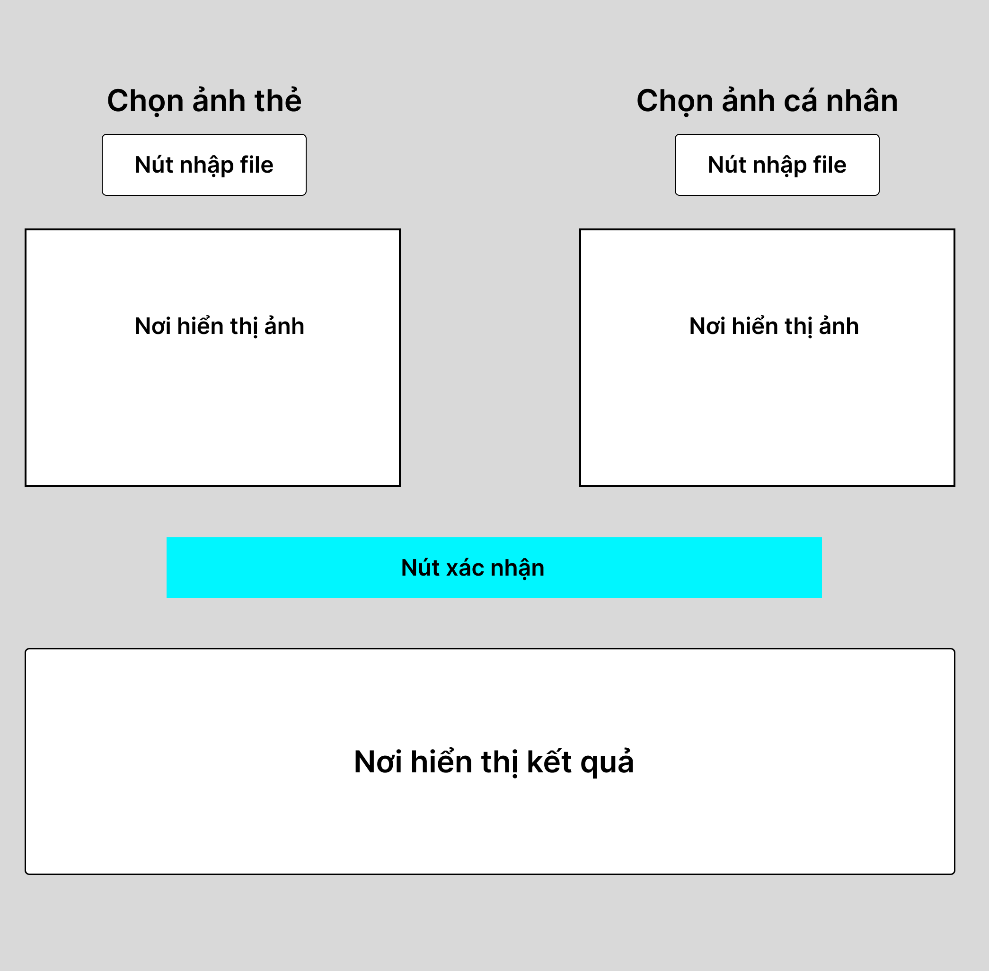
* + 1. Thiết kế quy trình nghiệp vụ

Chức năng trích xuất thông tin trên thẻ sinh viên được bắt đầu bằng việc hệ thống nhận được ảnh thẻ sinh viên thông qua camera. Sau khi đã có ảnh, hệ thống gọi đến các Mô-đun tương ứng để xử lý tuần tự như: Mô-đun xử lý ảnh, Mô-đun trích xuất văn bản. Sau khi quá trình kết thúc, hệ thống phản hồi kết quả và lưu trữ dữ liệu để phục vụ cho các bước tiếp theo.

Hình 2.x: Biểu đồ tuần tự chức năng xử lý đầu vào và trích xuất thông tin trên thẻ

* + 1. Thiết kế giao diện

**Hình 2.x** mô tả giao diện mẫu của chức năng đọc danh sách sinh viên trong hệ thống. Giao diện bao gồm hai mục chính: nơi tiếp nhận ảnh thẻ sinh viên và ảnh thẻ thực tế. Ảnh thẻ sinh viên được tải lên sẽ trải qua quá trình xử lý. Kết quả sau khi xử lý sẽ được hiển thị tại khu vực "Nơi hiển thị kết quả", giúp giám thị dễ dàng theo dõi và xác minh thông tin của sinh viên.



Hình 2.x: Thiết kế giao diện của trích xuất thông tin sinh viên

* 1. Sprint 2: Chức năng so khớp thông tin sinh viên với danh sách trong phòng thi
     1. Giới thiệu Sprint 2

Sau khi hoàn tất Mô-đun trích xuất thông tin trên thẻ sinh viên, bước tiếp theo là xây dựng chức năng hiển thị danh sách sinh viên trong phòng thi kết hợp với Mô-đun xác nhận danh tính sinh viên để hỗ trợ giám thị trong việc kiểm tra thông tin. Giám thị sẽ đăng nhập vào hệ thống, tìm kiếm phòng thi được phân công và hệ thống sẽ hiển thị danh sách sinh viên tương ứng. Khi thông tin đã được xác nhận chính xác, giám thị có thể trích xuất danh sách sinh viên thành file Excel và nhập vào giao diện kiểm tra khuôn mặt kết hợp với trích xuất thông tin thẻ sinh viên để kiểm tra từng sinh viên một cách cụ thể. Chức năng này được triển khai trong vòng 2 tuần từ 13/10/2024 đến 27/10/2024, bao gồm các công việc như chuẩn bị danh sách thí sinh, xây dựng chức năng đối chiếu danh sách sinh viên trong phòng thi, thêm, sửa, xóa thông tin sinh viên, xem danh sách sinh viên và xuất kết quả ra file Excel, cùng với việc kiểm thử chức năng đối chiếu và CRUD dữ liệu vào cơ sở dữ liệu. Việc hoàn thành chức năng này giúp quy trình kiểm tra và xác minh thông tin sinh viên trong phòng thi diễn ra thuận tiện, chính xác và hiệu quả hơn.

A diagram of a diagram

Description automatically generated

Hình 2.x: Biểu đồ ca sử dụng của giám thị trong chức năng so khớp thông tin

* + 1. Thiết kế dữ liệu phòng thi

Để hỗ trợ việc lưu trữ và quản lý thông tin sinh viên trong quá trình xác minh danh tính, phần này sẽ trình bày chi tiết về thiết kế bảng dữ liệu phục vụ cho việc quản lý phòng thi. Bảng dữ liệu được thiết kế dựa trên việc phân tích các thông tin liên quan đến phòng thi và sinh viên, đồng thời bổ sung một số trường thông tin quan trọng nhằm đảm bảo hệ thống vận hành hiệu quả và đáp ứng các yêu cầu mở rộng trong tương lai. Cấu trúc và nội dung cụ thể của bảng dữ liệu lưu trữ thông tin phòng thi sẽ được trình bày chi tiết trong **Bảng 2.x**.

*Mục đích*: Lưu trữ các thông tin của phòng thi

*Tên bảng*: Collection ExamRooms

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Diễn giải** | **Tên trường** | **Kiểu** | **Mặc định** | **Ràng buộc** | **Ghi chú** |
| 1 | MongoDB tự tạo | \_id | ObjectId |  |  | Đối tượng MongoDb tự sinh |
| 2 | Mã phòng thi | Room\_id | String |  | PK | Khoá chính |
| 3 | Số lượng sinh viên tối đa | Capacity | String |  |  |  |
| 4 | Mã bài thi | Exam\_id | Date |  | FK |  |
| 5 | Mã giám thị | Invigilator\_id | String |  | FK |  |
| 6 | Danh sách mã sinh viên | Students | String |  |  | Chứa mã sinh viên có trong phòng thi |
| 7 | Danh sách camera | Cameras | String |  |  |  |
| 8 | Danh sách máy tính | Computers | String |  |  |  |

Bảng 2.x: Dữ liệu lưu trữ thông tin phòng thi

* + 1. Quy trình đối chiếu thông tin sinh viên với danh sách phòng thi

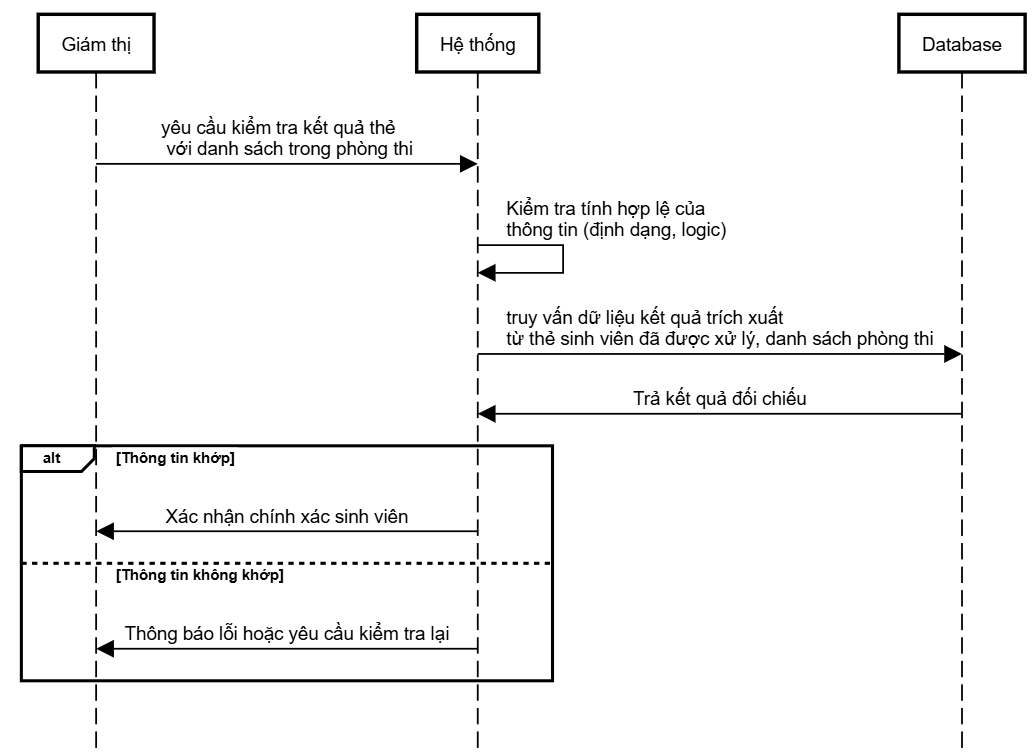
Quy trình đối chiếu thông tin sinh viên với danh sách phòng thi được thiết kế để vận hành tự động, tuần tự nhằm đảm bảo độ chính xác và tối ưu thời gian thực hiện. Sau khi đã có thông tin trích xuất từ thẻ, hệ thống sẽ tiến hành so khớp với danh sách sinh viên được giám thị nhập vào. Với kết quả mong muốn là xác nhận sự có mặt của sinh viên trong phòng thi, hệ thống sẽ tiến hành các bước với chi tiết như sau:

|  |  |
| --- | --- |
| **Mục đích** | Xác nhận sinh viên có tên trong danh sách phòng thi |
| **Các bước thực hiện** | **Bước 1:** Tạo trang hiển thị phòng thi và danh sách sinh viên trong phòng thi đó.  **Bước 2:** Trích xuất danh sách sinh viên thành file Excel.  **Bước 3:** Nhập file Excel và hiển thị nó vào giao diện của “Kiểm tra khuôn mặt và trích xuất thông tin thẻ sinh viên”. |
| **Các bước**  **bổ sung** | * Áp dụng các công nghệ của Java Spring Boot để có thể dễ dàng quản lý dữ liệu, trích xuất danh sách sinh viên thành file Excel |
| **Các ngoại lệ** | * Thông tin danh sách sinh viên bị sai. * Không thể trích xuất được danh sách sinh viên thành file Excel. |
| **Tài liệu liên quan** | Java Spring Boot, thư viện XSSWorkbook |
| **Tham chiếu** | Mục 1.6.1.2: Chức năng đối chiếu sinh viên với danh sách phòng thi  Mục 3.3.1.2: Mô-đun đối chiếu sinh viên với danh sách phòng thi. |

Bảng 2.x: Bảng chức năng đối chiếu thông tin sinh viên với danh sách phòng thi

* + 1. Thiết kế quy trình nghiệp vụ

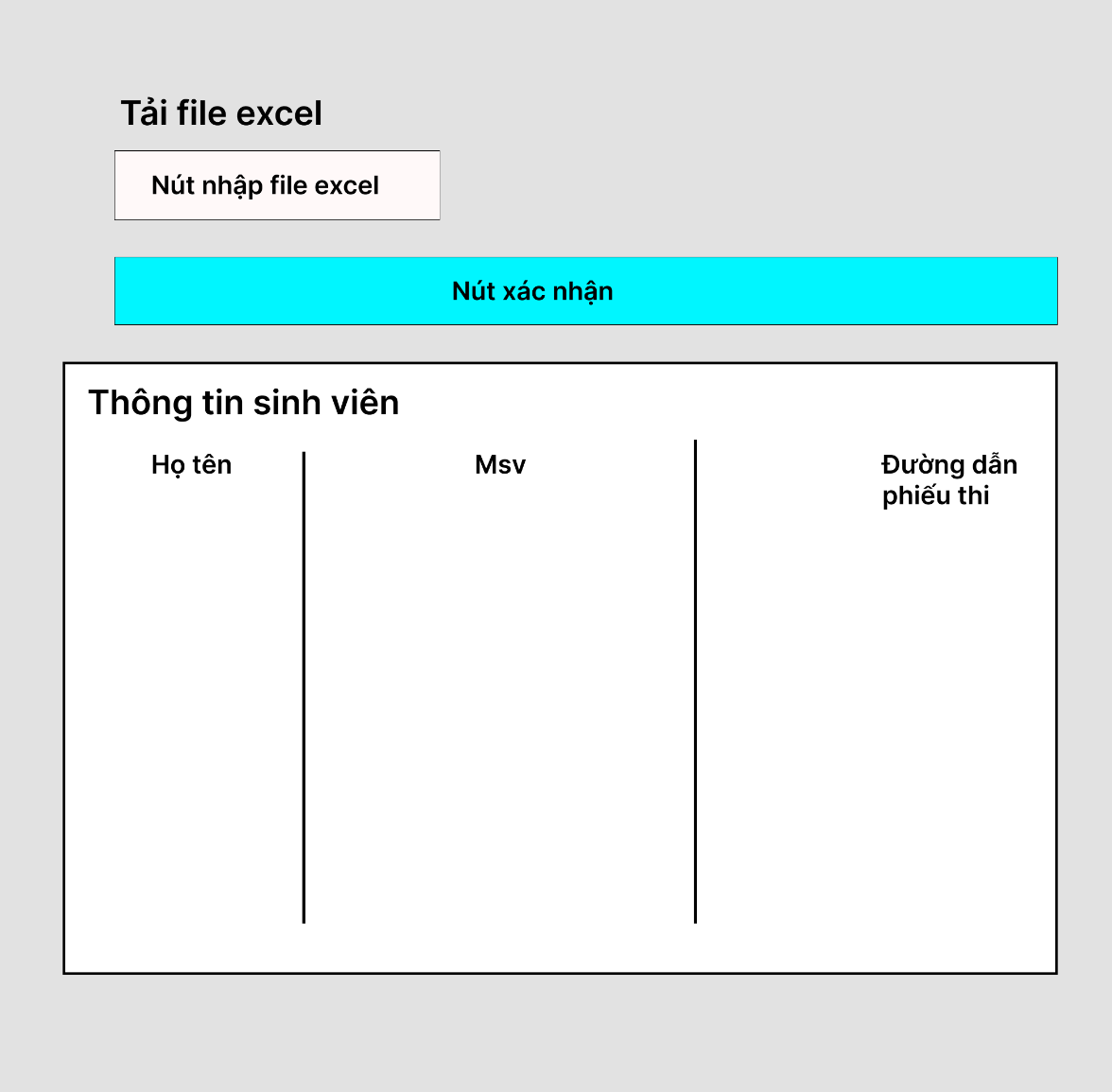
Quy trình đối chiếu thông tin sinh viên với danh sách trong phòng thi bắt đầu bằng việc giám thị truy cập vào hệ thống để lấy danh sách sinh viên của phòng thi mà mình được phân công. Sau khi đã hiển thị danh sách, giám thị tiến hành tải về file excel để sau nhập vào tính năng xác minh, hệ thống sẽ chỉ lưu trữ tạm thông tin cần thiết để xác minh danh tính như *mã số sinh viên*, *họ tên sinh viên*. Kết hợp với thông tin đã đọc được từ trước quy trình đối chiếu được triển khai như sau:



Hình 2.x: Biểu đồ tuần tự chức năng đối chiếu thông tin với danh sách trong phòng thi

* + 1. Thiết kế giao diện

**Hình 2.x** mô tả giao diện mẫu của chức năng đối chiếu thông tin sinh viên với danh sách trong phòng thi. Giao diện bao gồm nút tải lên file excel và nơi để hiển thị kết quả đối chiếu.



Hình 2.x: Thiết kế giao diện của chức năng đối chiếu với các sinh viên trong phòng thi

* 1. Sprint 3: Đối chiếu khuôn mặt trên thẻ với ảnh thực tế
     1. Giới thiệu Sprint 3

Sprint 3 của đề tài tập trung vào việc phát triển chức năng đối chiếu khuôn mặt trên thẻ sinh viên với khuôn mặt thực tế trong quá trình xác minh danh tính. Đây là bước quan trọng nhằm đảm bảo tính chính xác và độ tin cậy của hệ thống trong việc nhận diện và xác thực sinh viên tham gia kỳ thi. Trong thời gian 12 ngày, từ 27/10/2024 đến 8/11/2024, các công việc chính được thực hiện bao gồm: thu thập và chuẩn bị dữ liệu mẫu, xây dựng chức năng các Mô-đun chính, tạo giao diện và tích hợp chức năng đối chiếu qua backend, truy vấn cơ sở dữ liệu, và kiểm thử chức năng đối chiếu khuôn mặt. Sprint này không chỉ hoàn thiện chức năng đối chiếu mà còn tích hợp giao diện người dùng, đảm bảo tính liền mạch và hiệu quả trong quá trình sử dụng thực tế.

A diagram with black text and black circles

Description automatically generated

Hình 2.x: Biểu đồ ca sử dụng đối chiếu khuôn mặt của giám thị

A diagram of a person

Description automatically generated

Hình 2.x: Biểu đồ ca sử dụng đối chiếu khuôn mặt của sinh viên

* + 1. Thiết kế dữ liệu

Để hỗ trợ chức năng đối chiếu khuôn mặt trên thẻ sinh viên với khuôn mặt thực tế, phần này sẽ trình bày chi tiết về thiết kế bảng dữ liệu. Bảng dữ liệu được thiết kế dựa trên việc phân tích các thông tin liên quan đến phòng thi và sinh viên, đồng thời bổ sung một số trường thông tin quan trọng nhằm đảm bảo hệ thống vận hành hiệu quả và đáp ứng các yêu cầu mở rộng trong tương lai. Cấu trúc và nội dung cụ thể của bảng dữ liệu lưu trữ thông tin phòng thi sẽ được trình bày chi tiết trong **Bảng 2.x**.

*Mục đích*: Lưu trữ các thông tin của phòng thi

*Tên bảng*: Collection ExamParticipants

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Diễn giải** | **Tên trường** | **Kiểu** | **Mặc định** | **Ràng buộc** | **Ghi chú** |
| 1 | MongoDB tự tạo | \_id | ObjectId |  |  | Đối tượng MongoDb tự sinh |
| 2 | Mã sinh viên | Std\_id | String |  | PK | Khoá chính |
| 3 | Mã phòng thi | Room\_id | String |  | FK |  |
| 4 | Đường dẫn ảnh thẻ | Card\_image | String |  |  |  |
| 5 | Đường dẫn ảnh khuôn mặt đã cắt | Face\_image | String |  |  |  |
| 6 | Đường dẫn ảnh qua camera | Face\_camera | String |  |  |  |
| 7 | Kết quả so sánh khuôn mặt | Face\_result | Boolean |  |  |  |
| 8 | Thời gian kiểm tra sinh viên | Verification\_time | date |  |  |  |
| 9 | Phiếu thi của sinh viên | Exam\_ticket | Object |  |  |  |

Bảng 2.x: Dữ liệu lưu trữ thông tin kết quả kiểm tra trước khi thi

* + 1. Quy trình đối chiếu khuôn mặt

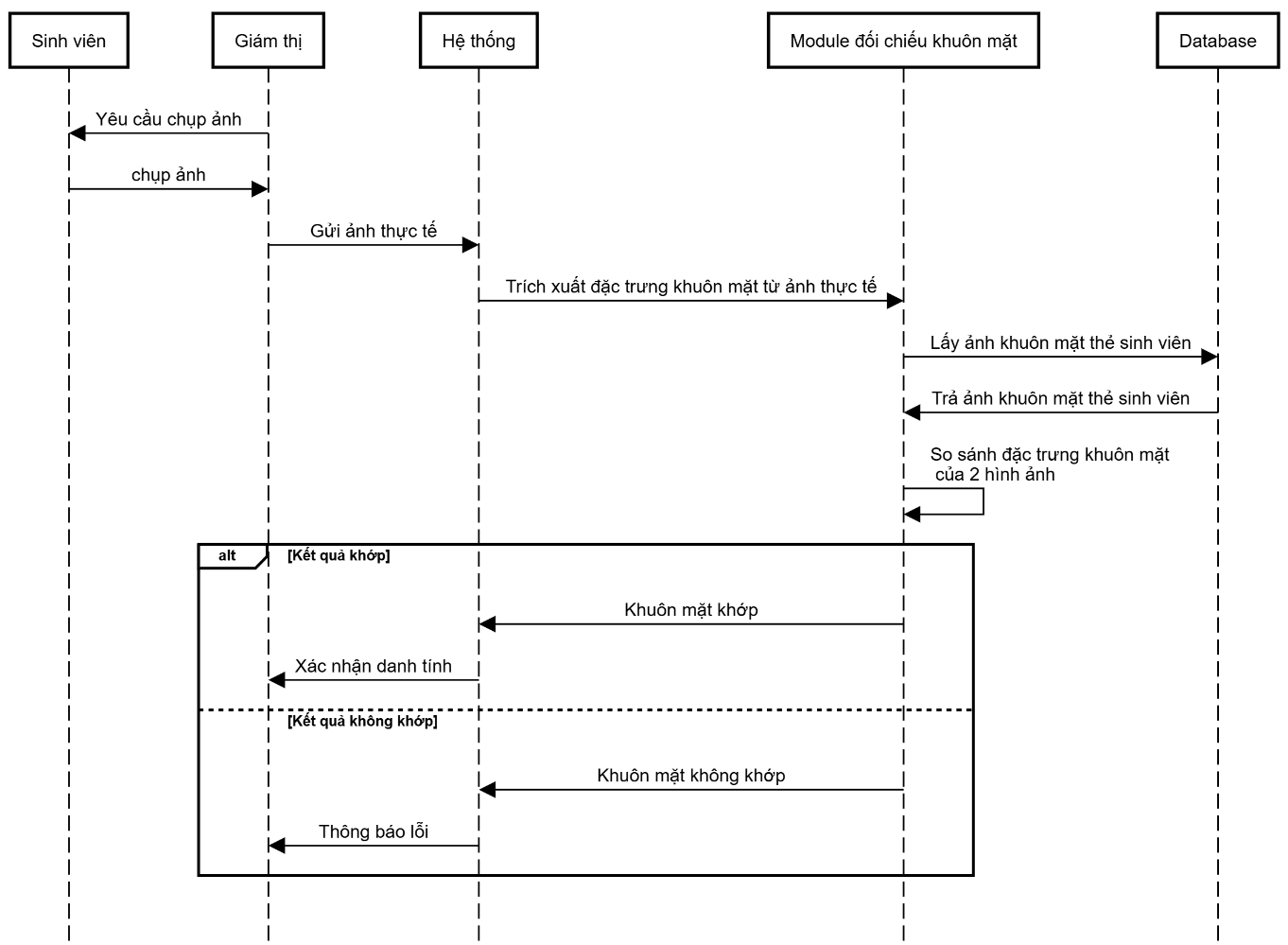
Quy trình thực hiện chức năng đối chiếu khuôn mặt bao gồm ba bước chính. Bước đầu tiên là chuẩn bị dữ liệu đầu vào, trong đó đảm bảo rằng hai hình ảnh quan trọng, bao gồm ảnh sinh viên trên thẻ và ảnh sinh viên thực tế, phải được cắt đúng khuôn mặt và chính xác để sẵn sàng cho các bước tiếp theo. Bước thứ hai là xây dựng chương trình đối chiếu khuôn mặt. Với sự hỗ trợ của ngôn ngữ Python và các thư viện liên quan, thuật toán khoảng cách Euclid được sử dụng để tiền xử lý và so sánh các đặc điểm khuôn mặt, từ đó đưa ra kết quả đối chiếu. Cuối cùng, bước ba là đẩy kết quả lên giao diện, nơi các giám thị có thể xác nhận và thông báo kết quả đối chiếu, đảm bảo tính chính xác và hiệu quả trong quá trình xác minh danh tính sinh viên.

|  |  |
| --- | --- |
| **Mục đích** | Đối chiếu khuôn mặt sinh viên với thẻ sinh viên thực tế để xác minh danh tính. |
| **Các bước thực hiện** | **Bước 1:** Chuẩn bị dữ liệu đầu vào, đảm bảo ảnh thẻ sinh viên và ảnh thực tế của sinh viên được cắt đúng khuôn mặt và chính xác.  **Bước 2:** Xây dựng chương trình đối chiếu khuôn mặt sử dụng thuật toán khoảng cách Euclid trong Python để so sánh và đưa ra kết quả.  **Bước 3:** Đẩy kết quả đối chiếu lên giao diện để giám thị có thể xác nhận và thông báo kết quả. |
| **Các bước**  **bổ sung** | * Sử dụng thư viện hỗ trợ Python như OpenCV và dlib để thực hiện các thao tác tiền xử lý ảnh trước khi đối chiếu khuôn mặt. * Tăng cường độ chính xác của thuật toán qua các bài huấn luyện và cải tiến mô hình. * Tăng cường độ chính xác của thuật toán qua các bài huấn luyện và cải tiến mô hình. |
| **Các ngoại lệ** | * Ảnh đầu vào bị mờ hoặc không rõ nét, ảnh thực tế không trùng khớp với ảnh trên thẻ. * Khuôn mặt trong ảnh thực tế không đủ rõ ràng hoặc bị che khuất. |
| **Tài liệu liên quan** | Python, OpenCV, dlib, thuật toán khoảng cách Euclid. |
| **Tham chiếu** | Mục 1.6.1.3: Chức năng đối chiếu khuôn mặt với thẻ sinh viên.  Mục 3.3.1.3: Chức năng đối chiếu khuôn mặt với thẻ sinh viên.. |

Bảng 2.x: Bảng chức năng đối chiếu khuôn mặt thực tế với ảnh thẻ sinh viên.

* + 1. Thiết kế quy trình nghiệp vụ

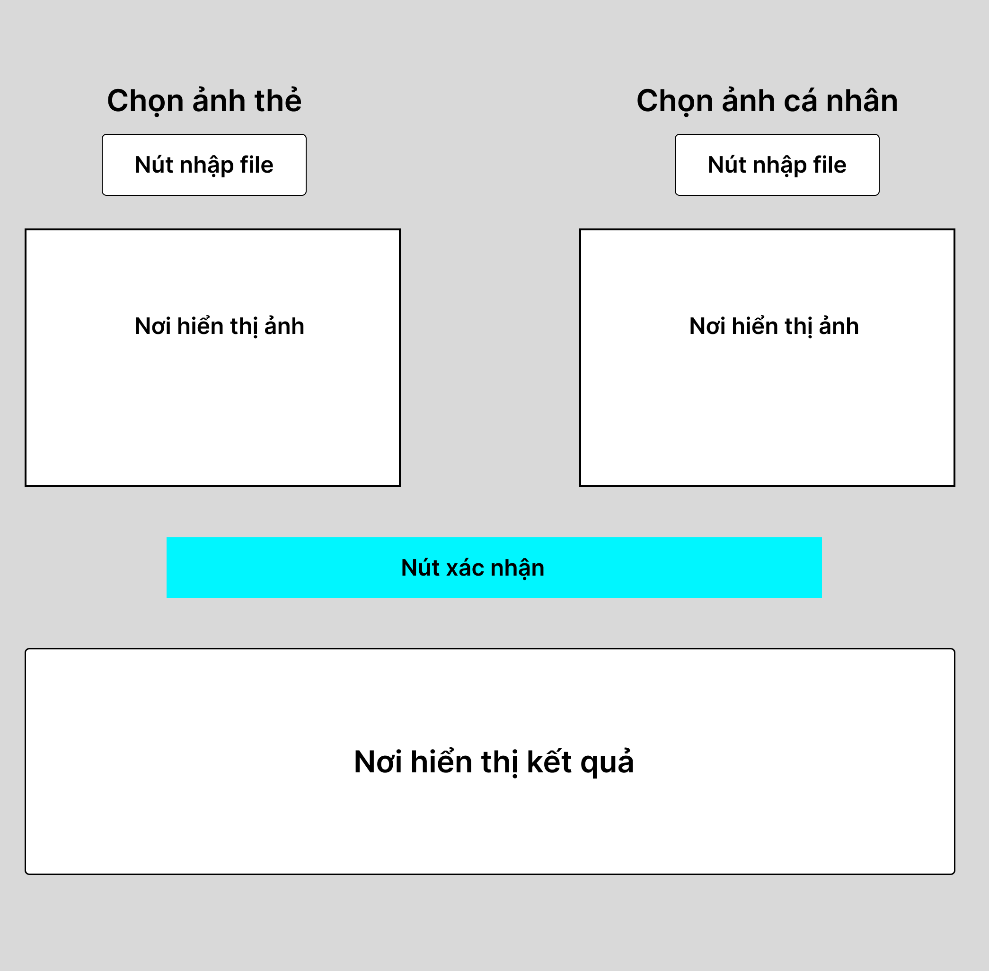
Quy trình đối chiếu khuôn mặt trong Sprint 3 bao gồm ba bước chính để đảm bảo tính chính xác và hiệu quả trong việc xác minh danh tính sinh viên. Đầu tiên, hệ thống sẽ trích xuất khuôn mặt từ cả ảnh thẻ sinh viên và ảnh thực tế, sau đó sử dụng các phương pháp như Face Recognition hoặc Deep Learning để so sánh độ tương đồng giữa hai khuôn mặt. Nếu độ tương đồng đạt mức chấp nhận được, hệ thống sẽ xác nhận danh tính của sinh viên. Nếu không khớp, hệ thống sẽ yêu cầu kiểm tra lại. Tiếp theo, nếu khuôn mặt khớp, kết quả sẽ được lưu vào cơ sở dữ liệu và cập nhật để xác nhận danh tính. Nếu không khớp, hệ thống sẽ ghi lại kết quả và thông báo cho giám thị để kiểm tra lại thủ công. Cuối cùng, sau khi hoàn tất quá trình đối chiếu, hệ thống sẽ thông báo kết quả cho sinh viên và giám thị qua giao diện, giúp xác nhận thành công hoặc thất bại trong việc đối chiếu khuôn mặt.



Hình 2.x: Biểu đồ tuần tự chức năng đối chiếu khuôn mặt thực tế với ảnh thẻ sinh viên

* + 1. Thiết kế giao diện

**Hình 2.x** mô tả giao diện mẫu của chức năng đọc danh sách sinh viên trong hệ thống. Giao diện bao gồm hai mục chính: nơi tiếp nhận ảnh thẻ sinh viên và ảnh thẻ thực tế. Ảnh thẻ sinh viên được tải lên sẽ trải qua quá trình xử lý. Kết quả sau khi xử lý sẽ được hiển thị tại khu vực "Nơi hiển thị kết quả", giúp giám thị dễ dàng theo dõi và xác minh thông tin của sinh viên.



Hình 2.x: Giao diện của chức năng đối chiếu khuôn mặt

* 1. Sprint 4: Xuất phiếu thi và giám sát quy trình thi
     1. Giới thiệu Sprint 4

Sprint 4 tập trung vào hai nhiệm vụ chính. Đầu tiên, hệ thống sẽ thực hiện việc xuất phiếu thi cá nhân cho sinh viên, sử dụng thông tin đã lưu trong cơ sở dữ liệu để tạo phiếu thi với đầy đủ thông tin cá nhân và lịch thi. Phiếu thi sẽ được xuất dưới dạng PDF. Nhiệm vụ thứ hai là phát triển Mô-đun giám sát quy trình thi, nhằm phát hiện các hành vi gian lận, xác minh danh tính sinh viên và theo dõi vị trí thi theo thời gian thực. Mô-đun này sẽ kết hợp công nghệ nhận diện khuôn mặt và giám sát từ camera, đảm bảo tính minh bạch và công bằng trong suốt kỳ thi.

* + 1. Thiết kế dữ liệu

Hiện tại, chúng tôi mới chỉ triển khai được chức năng in phiếu thi. Vì phiếu thi chỉ là công cụ tạm thời, không mang tính chất lưu trữ lâu dài và sẽ bị huỷ ngay sau khi ca thi hoàn thành, nên việc lưu trữ phiếu thi là không cần thiết. Do đó, phiếu thi sẽ gồm các trường thông tin cơ bản như: *tên sinh viên*, *mã sinh viên*, *môn thi*, *thời gian thi*, và *tài khoản*, *mật khẩu đăng nhập*. Các trường này sẽ được lấy trực tiếp hoặc tự sinh từ cơ sở dữ liệu của hệ thống. Phiếu thi được triển khai dưới dạng PDF, giúp dễ dàng in ra hoặc gửi qua email cho sinh viên trước khi kỳ thi diễn ra.

* + 1. Quy trình xuất phiếu thi

Quy trình xuất phiếu thi được thiết kế nhằm tự động hóa việc tạo phiếu thi cho sinh viên, từ việc truy xuất thông tin sinh viên trong cơ sở dữ liệu đến việc tạo và gửi phiếu thi cho sinh viên. Hệ thống sẽ sử dụng dữ liệu đã lưu trữ, bao gồm thông tin cá nhân và lịch thi, để tạo ra phiếu thi chuẩn xác. Phiếu thi sẽ được xuất dưới dạng PDF chứa thông tin quan trọng giúp giám thị và sinh viên xác minh dễ dàng tại phòng thi. Toàn bộ quy trình không chỉ giúp tiết kiệm thời gian mà còn đảm bảo tính chính xác và thuận tiện trong việc cấp phát phiếu thi cho sinh viên.

|  |  |
| --- | --- |
| **Mục đích** | Tạo phiếu thi cá nhân cho sinh viên |
| **Các bước thực hiện** | **Bước 1:** Truy xuất dữ liệu sinh viên từ cơ sở dữ liệu để lấy thông tin cá nhân và lịch thi.  **Bước 2:** Tạo phiếu thi bằng cách định dạng dữ liệu vào mẫu phiếu thi chuẩn, bao gồm các thông tin như mã sinh viên, tên, lớp, ngày thi, giờ thi và địa điểm thi, tự sinh dữ liệu tài khoản và mật khẩu ca thi.  **Bước 3:** Xuất phiếu thi dưới dạng file PDF phục vụ việc in phiếu để dễ dàng tiến hành tại phòng thi.  **Bước 4:** Sinh viên kiểm tra phiếu thi và xác nhận thông tin, yêu cầu chỉnh sửa nếu có sai sót. |
| **Các bước**  **bổ sung** | * Sử dụng thư viện iText PDF và Java Spring Boot để tạo và xuất file PDF. |
| **Các ngoại lệ** | * Thiếu thông tin sinh viên trong cơ sở dữ liệu hoặc lỗi khi truy xuất dữ liệu. * Không thể tạo hoặc xuất file PDF |
| **Tài liệu liên quan** | Java Spring Boot, iText PDF Library, MongoDB. |
| **Tham chiếu** | Mục 1.7.1.3: Chức năng xuất phiếu thi cá nhân cho sinh viên.  Mục 3.3.1.3: Chức năng xuất phiếu thi cá nhân cho sinh viên. |

Bảng 2.x: Bảng chức năng xuất phiếu thi cá nhân cho sinh viên

* + 1. Thiết kế quy trình nghiệp vụ

Text+img

* + 1. Thiết kế giao diện

Text+img

* 1. Sprint 5: Các chức năng quản lý của quản trị viên

Sprint 5 tập trung vào việc triển khai các chức năng quản lý dành cho quản trị viên, giúp quản trị viên dễ dàng theo dõi, quản lý và điều phối các hoạt động trong hệ thống. Hiện nay, hầu hết các cơ sở giáo dục đã áp dụng các hệ thống quản lý thông minh để tự động hóa và tối ưu hóa quy trình thi cử. Do đó, trong phạm vi đồ án này, chúng tôi chỉ xây dựng lại một cách sơ bộ các tính năng liên quan trực tiếp đến vận hành kỳ thi mà vẫn đảm bảo hệ thống hoạt động hiệu quả, nhằm tiết kiệm thời gian và kịp tiến độ.

Các chức năng chính sẽ bao gồm việc quản lý danh sách sinh viên, giám thị, phòng thi, và các kỳ thi. Hệ thống sẽ cung cấp giao diện quản trị để quản trị viên có thể dễ dàng thêm, sửa, xóa thông tin sinh viên, giám thị và các thông tin liên quan đến kỳ thi. Quản trị viên cũng có thể truy cập vào các báo cáo thống kê liên quan đến việc tham gia thi của sinh viên, kết quả thi và các hoạt động giám sát trong suốt kỳ thi. Việc triển khai các chức năng này nhằm tăng cường khả năng giám sát và quản lý hệ thống một cách hiệu quả, đồng thời nâng cao tính bảo mật và minh bạch trong quy trình thi.

* + 1. Chức năng quản lý sinh viên
       1. Giới thiệu chức năng quản lý sinh viên của quản trị viên

Chức năng quản lý sinh viên của quản trị viên cho phép quản lý thông tin sinh viên như thêm mới, chỉnh sửa, xóa và tìm kiếm theo các tiêu chí như mã sinh viên, tên, lớp và khoa. Quản trị viên có thể cập nhật kết quả thi và xuất báo cáo về số lượng sinh viên, tình trạng tham gia thi và kết quả thi, giúp giám sát và quản lý hiệu quả trong quá trình tổ chức kỳ thi.

A diagram of a person

Description automatically generated

Hình 2.x: Biểu đồ ca sử dụng chức năng quản lý sinh viên

* + - 1. Thiết kế các bảng dữ liệu liên quan

Các bảng dữ liệu liên quan đến chức năng quản lý sinh viên chủ yếu kế thừa từ các bảng đã được xây dựng trước đó. Chức năng này tập trung vào việc thực hiện các câu lệnh truy vấn đơn giản để quản lý thông tin sinh viên, bao gồm việc tìm kiếm, cập nhật, và xóa dữ liệu. Các bảng dữ liệu sẽ bao gồm thông tin sinh viên, kết quả thi và lịch thi, giúp quản trị viên dễ dàng theo dõi và quản lý các hoạt động liên quan đến sinh viên trong quá trình thi. Bổ sung bảng kết quả thi của sinh viên có tên ExamResults.

* + - 1. Quy trình quản lý sinh viên

Trong quá trình quản lý sinh viên, các chức năng CRUD (Create, Read, Update, Delete) đóng vai trò quan trọng trong việc đảm bảo tính linh hoạt và chính xác của hệ thống. Cụ thể, chức năng thêm sinh viên (Create) cho phép quản trị viên nhập thông tin mới vào hệ thống; chức năng cập nhật (Update) giúp chỉnh sửa thông tin sinh viên khi có thay đổi; chức năng xóa (Delete) hỗ trợ việc loại bỏ dữ liệu không còn cần thiết; và chức năng tìm kiếm (Read) cho phép quản trị viên nhanh chóng tra cứu thông tin sinh viên. Các bước thực hiện cho mỗi chức năng như sau:

|  |  |
| --- | --- |
| **Mục đích** | **Các bước thực hiện** |
| Thêm mới sinh viên | 1. Quản trị viên nhập thông tin sinh viên vào giao diện quản lý.  2. Hệ thống thực hiện câu lệnh thêm vào cơ sở dữ liệu.  3. Hệ thống xác nhận việc thêm mới thành công và cập nhật lại danh sách sinh viên. |
| Cập nhật thông tin sinh viên | 1. Quản trị viên tìm kiếm sinh viên cần cập nhật thông tin.  2. Hệ thống truy vấn và hiển thị thông tin sinh viên đã chọn.  3. Quản trị viên chỉnh sửa thông tin cần thay đổi và gửi yêu cầu cập nhật.  4. Hệ thống thực hiện câu lệnh cập nhật vào cơ sở dữ liệu. |
| Xóa sinh viên | 1. Quản trị viên tìm kiếm và chọn sinh viên cần xóa.  2. Hệ thống xác nhận yêu cầu xóa sinh viên.  3. Quản trị viên xác nhận và hệ thống thực hiện câu lệnh xóa trong cơ sở dữ liệu. |
| Tìm kiếm sinh viên | 1. Quản trị viên nhập thông tin cần tìm kiếm (mã sinh viên, tên, lớp học,...) vào ô tìm kiếm.  2. Hệ thống thực hiện câu lệnh truy vấn để tìm kiếm thông tin sinh viên trong cơ sở dữ liệu.  3. Hệ thống hiển thị kết quả tìm kiếm cho quản trị viên. |

Bảng 2.x: Quy trình quản lý sinh viên của quản trị viên

* + - 1. Thiết kế giao diện quản lý sinh viên

Text+img

* + 1. Chức năng quản lý giám thị
       1. Giới thiệu chức năng quản lý giám thị của quản trị viên

Chức năng quản lý giám thị của quản trị viên giúp quản lý thông tin liên quan đến giám thị trong kỳ thi, bao gồm các chức năng như thêm mới, chỉnh sửa, xóa và tìm kiếm giám thị theo các tiêu chí như mã giám thị, tên và phòng thi. Quản trị viên có thể phân công giám thị cho các phòng thi, theo dõi lịch trình làm việc của giám thị và cập nhật thông tin về kết quả giám sát trong suốt kỳ thi. Điều này giúp đảm bảo tính minh bạch và công bằng trong việc giám sát kỳ thi, cũng như hỗ trợ quản lý hiệu quả việc phân công công việc cho giám thị.

A diagram of a person's process

Description automatically generated

Hình 2.x: Biểu đồ ca sử dụng chức năng quản lý giám thị

* + - 1. Thiết kế các bảng dữ liệu liên quan

Các bảng dữ liệu liên quan đến chức năng quản lý giám thị chủ yếu kế thừa từ các bảng đã được xây dựng trước đó. Chức năng này tập trung vào việc thực hiện các câu lệnh truy vấn đơn giản để quản lý thông tin sinh viên, bao gồm việc tìm kiếm, cập nhật, và xóa dữ liệu. Các bảng dữ liệu sẽ bao gồm thông tin sinh viên, kết quả thi và lịch thi, giúp quản trị viên dễ dàng theo dõi và quản lý các hoạt động liên quan đến sinh viên trong quá trình thi.

* + - 1. Quy trình quản lý giám thị

Trong quá trình quản lý giám thị, các chức năng CRUD (Create, Read, Update, Delete) đóng vai trò quan trọng trong việc đảm bảo tính linh hoạt và chính xác của hệ thống. Cụ thể, chức năng thêm sinh viên (Create) cho phép quản trị viên nhập thông tin mới vào hệ thống; chức năng cập nhật (Update) giúp chỉnh sửa thông tin sinh viên khi có thay đổi; chức năng xóa (Delete) hỗ trợ việc loại bỏ dữ liệu không còn cần thiết; và chức năng tìm kiếm (Read) cho phép quản trị viên nhanh chóng tra cứu thông tin sinh viên. Các bước thực hiện cho mỗi chức năng như sau:

|  |  |
| --- | --- |
| **Mục đích** | **Các bước thực hiện** |
| Thêm mới giám thị | 1. Quản trị viên nhập thông tin vào giao diện quản lý.  2. Hệ thống thực hiện câu lệnh thêm vào cơ sở dữ liệu.  3. Hệ thống xác nhận việc thêm mới thành công và cập nhật lại danh sách sinh viên. |
| Cập nhật thông tin giám thị | 1. Quản trị viên tìm kiếm giám thị cần cập nhật thông tin.  2. Hệ thống truy vấn và hiển thị thông tin đã chọn.  3. Quản trị viên chỉnh sửa thông tin cần thay đổi và gửi yêu cầu cập nhật.  4. Hệ thống thực hiện câu lệnh cập nhật vào cơ sở dữ liệu. |
| Xóa giám thị | 1. Quản trị viên tìm kiếm và chọn giám thị cần xóa.  2. Hệ thống xác nhận yêu cầu xóa giám thị.  3. Quản trị viên xác nhận và hệ thống thực hiện câu lệnh xóa trong cơ sở dữ liệu. |
| Tìm kiếm giám thị | 1. Quản trị viên nhập thông tin cần tìm kiếm (mã giám thị, họ tên, đơn vị,…) vào ô tìm kiếm.  2. Hệ thống thực hiện câu lệnh truy vấn để tìm kiếm thông tin giám thị trong cơ sở dữ liệu.  3. Hệ thống hiển thị kết quả tìm kiếm cho quản trị viên. |

Bảng 2.x: Quy trình quản lý giám thị của quản trị viên

* + - 1. Thiết kế giao diện quản lý giám thị

Text+img

* + 1. Chức năng quản lý phòng thi
       1. Giới thiệu chức năng quản lý phòng thi của quản trị viên

Hình 2.x. Biểu đồ ca sử dụng chức năng quản lý phòng thi

* + - 1. Thiết kế các bảng dữ liệu liên quan
      2. Quy trình quản lý giám thị

|  |  |
| --- | --- |
| **Mục đích** | **Các bước thực hiện** |
| Thêm mới giám thị | 1. Quản trị viên nhập thông tin vào giao diện quản lý.  2. Hệ thống thực hiện câu lệnh thêm vào cơ sở dữ liệu.  3. Hệ thống xác nhận việc thêm mới thành công và cập nhật lại danh sách sinh viên. |
| Cập nhật thông tin giám thị | 1. Quản trị viên tìm kiếm giám thị cần cập nhật thông tin.  2. Hệ thống truy vấn và hiển thị thông tin đã chọn.  3. Quản trị viên chỉnh sửa thông tin cần thay đổi và gửi yêu cầu cập nhật.  4. Hệ thống thực hiện câu lệnh cập nhật vào cơ sở dữ liệu. |
| Xóa giám thị | 1. Quản trị viên tìm kiếm và chọn giám thị cần xóa.  2. Hệ thống xác nhận yêu cầu xóa giám thị.  3. Quản trị viên xác nhận và hệ thống thực hiện câu lệnh xóa trong cơ sở dữ liệu. |

Bảng 2.x: Quy trình quản lý giám thị của quản trị viên

* + - 1. Thiết kế giao diện quản lý giám thị

Text+img

* 1. Biểu đồ ca sử dụng

Biểu đồ ca sử dụng trong hệ thống quản lý kỳ thi giúp mô tả các chức năng mà hệ thống cung cấp và cách thức các người dùng tương tác với hệ thống. Biểu đồ này cung cấp cái nhìn tổng quan về các yêu cầu chức năng của hệ thống và mô tả các tác vụ mà mỗi đối tượng (người dùng là quản trị viên, giám thị, sinh viên) có thể thực hiện.

A diagram of a diagram

Description automatically generatedHình 2.x: Biểu đồ ca sử dụng

1. XÂY DỰNG HỆ THỐNG VẬN HÀNH VÀ QUẢN LÝ QUY TRÌNH THI TRẮC NGHIỆM KHÁCH QUAN
   1. Môi trường phát triển
      1. Giới thiệu công nghệ và công cụ sử dụng
         1. Python FastAPI

Python FastAPI là một framework hiện đại, nhanh và mạnh mẽ dùng để xây dựng các API với Python. FastAPI được đánh giá cao nhờ hiệu suất cao, dễ sử dụng, và hỗ trợ tạo tài liệu tự động cho các API. Framework này được thiết kế để tận dụng khả năng đồng bộ và bất đồng bộ của Python, giúp tối ưu hóa tốc độ xử lý và cải thiện hiệu quả của ứng dụng. Với cú pháp rõ ràng và tài liệu dễ hiểu, FastAPI rất phù hợp cho các ứng dụng yêu cầu tốc độ phản hồi nhanh và khả năng mở rộng linh hoạt, chẳng hạn như hệ thống quản lý và vận hành quy trình thi trắc nghiệm khách quan.

* + - 1. Java Spring Boot

Java Spring Boot là một framework mạnh mẽ và phổ biến để phát triển các ứng dụng web và API backend. Được xây dựng trên nền tảng Spring Framework, Spring Boot giúp đơn giản hóa quá trình thiết lập và cấu hình đề tài Java, giúp cho việc phát triển ứng dụng trở nên nhanh chóng và hiệu quả. Spring Boot cung cấp các công cụ tự động hóa, tạo cấu trúc đề tài sẵn sàng sử dụng, và hỗ trợ xử lý dữ liệu linh hoạt. Với khả năng tích hợp tốt với cơ sở dữ liệu như MongoDB và các công nghệ bảo mật mạnh mẽ, Spring Boot là lựa chọn lý tưởng để xây dựng backend của hệ thống quản lý quy trình thi trắc nghiệm, đảm bảo tốc độ, tính mở rộng, và bảo mật cho hệ thống.

* + - 1. Hệ quản trị cơ sở dữ liệu MongoDB Atlas

MongoDB Atlas là một dịch vụ cơ sở dữ liệu đám mây được quản lý hoàn toàn, được thiết kế để làm việc với các ứng dụng hiện đại đòi hỏi khả năng mở rộng và linh hoạt. Atlas cung cấp một môi trường đáng tin cậy và an toàn để lưu trữ, quản lý và truy vấn dữ liệu NoSQL, đồng thời hỗ trợ tích hợp dễ dàng với các công cụ khác trong hệ sinh thái phát triển.

Việc sử dụng MongoDB Atlas cho đề tài giúp quản lý dữ liệu sinh viên và kết quả thi một cách hiệu quả, đảm bảo hệ thống có thể mở rộng quy mô và xử lý các yêu cầu cao về tốc độ cũng như tính sẵn sàng. Atlas cung cấp các tính năng như sao lưu tự động, quản lý bảo mật nâng cao, và khả năng giám sát hệ thống theo thời gian thực, giúp tối ưu hóa hiệu suất và bảo vệ dữ liệu quan trọng.

* + 1. Quản lý mã nguồn bằng Github

GitHub là một nền tảng quản lý mã nguồn và hợp tác phát triển dựa trên hệ thống kiểm soát phiên bản Git. Việc sử dụng GitHub trong đề tài giúp nhóm phát triển theo dõi sự thay đổi của mã nguồn, quản lý các phiên bản và hợp tác hiệu quả hơn khi làm việc nhóm.

GitHub hỗ trợ việc phân nhánh (branching) và hợp nhất (merging) dễ dàng, giúp phát triển và thử nghiệm các tính năng mới mà không làm gián đoạn phiên bản chính của hệ thống. Ngoài ra, GitHub cung cấp công cụ để quản lý vấn đề, theo dõi tiến độ, và tích hợp CI/CD để tự động hoá việc kiểm thử và triển khai ứng dụng, đảm bảo quy trình phát triển liên tục và linh hoạt.

* + 1. Phân chia công việc, quản lý tiến độ đề tài bằng Trello

Trello là một công cụ quản lý đề tài trực tuyến dựa trên phương pháp Kanban, giúp theo dõi tiến độ công việc một cách trực quan và hiệu quả. Trong đề tài này, Trello được sử dụng để phân chia công việc giữa các thành viên trong nhóm, từ đó đảm bảo mọi người có thể theo dõi tiến độ công việc và điều chỉnh kế hoạch khi cần thiết.

Mỗi công việc sẽ được đại diện dưới dạng các thẻ trong các cột như "To Do", "In Progress" và "Completed", giúp dễ dàng xác định tình trạng hiện tại của từng nhiệm vụ. Các nhiệm vụ được phân chia theo các giai đoạn của đề tài, ví dụ như phân tích yêu cầu, phát triển tính năng, kiểm thử và triển khai.

Trello cũng cho phép thêm các deadline, bình luận, đính kèm tài liệu và phân công nhiệm vụ cho các thành viên cụ thể, tạo nên một không gian làm việc hợp tác và có tổ chức. Bằng cách này, Trello giúp nhóm theo dõi tiến độ đề tài, đảm bảo đúng thời hạn và tối ưu hóa hiệu quả công việc.

A screenshot of a computer

Description automatically generated

Hình 3.x: Công cụ quản lý dự án Trello

* 1. Chức năng xử lý ảnh đầu vào và trích xuất thông tin trên thẻ sinh viên

Trong quá trình phát triển hệ thống, một trong những chức năng quan trọng nhất là xử lý ảnh đầu vào và trích xuất thông tin từ thẻ sinh viên. Để đảm bảo quá trình nhận diện và trích xuất thông tin chính xác, hệ thống sẽ được chia thành các Mô-đun chuyên biệt. Dưới đây là quy trình hoàn thiện chức năng

* + 1. Xây dựng các thành phần liên quan
       1. Mô-đun xử lý ảnh đầu vào

Mô-đun xử lý ảnh đầu vào sẽ chịu trách nhiệm tiếp nhận ảnh từ nhiều nguồn khác nhau (chẳng hạn như ảnh tải lên từ người dùng hoặc ảnh chụp trực tiếp qua camera) và thực hiện các bước tiền xử lý để chuẩn bị ảnh cho quá trình nhận diện.

Hình 3.x: Phần lập trình Mô-đun tiền xử lý ảnh đầu vào

def preprocess\_image(*image\_path*: str) -> str:

    image = cv2.imread(image\_path)

    gray = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)

    \_, thresh = cv2.threshold(gray, 150, 255, cv2.THRESH\_BINARY)

    processed\_image\_path = os.path.join(PROCESSED\_FOLDER, f"processed\_{os.path.basename(image\_path)}")

    cv2.imwrite(processed\_image\_path, thresh)

    return processed\_image\_path

|  |  |
| --- | --- |
| **Chức năng chính** | **Mô tả chi tiết** |
| Tiếp nhận ảnh đầu vào | - Nhận ảnh từ nhiều nguồn: ảnh tải lên hoặc chụp trực tiếp qua camera. - Xác định định dạng và kích thước hợp lệ. |
| Tiền xử lý ảnh | Chuyển đổi màu sắc: Chuyển ảnh về ảnh xám để giảm độ phức tạp xử lý. Làm sạch ảnh: Loại nhiễu nền, cải thiện độ sáng tối, làm sắc nét ảnh. Cắt ảnh thẻ: Tách phần chứa thẻ sinh viên khỏi ảnh lớn. Chuyển hướng ảnh: Quay ảnh về vị trí ngang nếu bị nghiêng. |
| Giảm nhiễu và tối ưu hóa | Áp dụng kỹ thuật lọc Gaussian, làm mịn ảnh để giảm nhiễu nền và cải thiện độ chính xác nhận diện. |

Bảng 3.x: Bảng mô tả Mô-đun tiền xử lý ảnh đầu vào

* + - 1. Mô-đun trích xuất thông tin trên thẻ sinh viên

Sau khi ảnh đã được tiền xử lý, Mô-đun trích xuất thông tin sẽ chịu trách nhiệm nhận diện và trích xuất các dữ liệu quan trọng từ thẻ sinh viên. Mục tiêu của Mô-đun này là thu thập các thông tin như mã số sinh viên, họ tên, ngày sinh, ngành học, và các thông tin khác từ ảnh thẻ sinh viên.

def perform\_ocr(*image\_path*):

    try:

        result = ocr.ocr(image\_path, *cls*=True)

        if result and len(result) > 0:

            sorted\_result = sorted(result[0], *key*=lambda *x*: x[0][0][1])

            logging.info(f"OCR thành công với {len(sorted\_result)} dòng.")

            return sorted\_result

        else:

            logging.warning("Không tìm thấy văn bản nào.")

            return None

    except Exception as e:

        logging.error(f"Lỗi trong quá trình OCR: {e}")

        return None

Hình 3.x: Phần lập trình Mô-đun trích xuất văn bản

|  |  |
| --- | --- |
| **Chức năng chính** | **Mô tả chi tiết** |
| Nhận diện và trích xuất thông tin | Sử dụng OCR để nhận diện các trường thông tin trên thẻ sinh viên như mã số sinh viên, họ tên, ngành học, khóa học, đơn vị. |
| Sử dụng công cụ OCR | Tích hợp các thư viện OCR như Tesseract hoặc PaddleOCR để chuyển đổi văn bản từ ảnh thành văn bản số. |
| Tiền xử lý văn bản | Loại bỏ ký tự không cần thiết, chuẩn hóa văn bản, làm sạch lỗi chính tả, và định dạng ngày tháng. |
| Trích xuất trường thông tin | Xác định và tách các trường quan trọng như mã số sinh viên, họ tên, ngành học, khóa học, email, và số điện thoại. |
| Kiểm tra tính hợp lệ | Xác minh các trường thông tin qua các tiêu chí như định dạng email, số điện thoại, và mã số sinh viên hợp lệ. |
| Công cụ/Thư viện sử dụng | Sử dụng các thư viện như Tesseract OCR, PaddleOCR, Regex, Pandas để hỗ trợ xử lý và lưu trữ dữ liệu. |

Bảng 3.x: Bảng mô tả Mô-đun trích xuất văn bản

* + 1. Giao diện chức năng xử lý ảnh đầu vào và trích xuất thông tin trên thẻ

**Hình 3.x** mô tả giao diện chức năng trích xuất thông tin trên thẻ sinh viên, bao gồm ba khu vực chính. Đầu tiên là *khu vực tải ảnh*, nơi người dùng có thể dễ dàng tải ảnh lên thông qua nút *Tải ảnh*. Sau khi ảnh được tải lên, hệ thống sẽ hiển thị ảnh đầu vào trực tiếp trên giao diện, giúp người dùng kiểm tra và xác nhận ảnh cần xử lý. Tiếp theo là *khu vực kết quả nhận diện*, nơi hiển thị các thông tin đã trích xuất từ ảnh như tên sinh viên, mã số thẻ, ngày sinh, và các trường thông tin liên quan. Đồng thời, khu vực này còn có mục *Thông báo trạng thái* để cung cấp phản hồi về quá trình nhận diện, chẳng hạn như "*Nhận diện thành công*" hoặc "*Không nhận diện được thông tin*". Cuối cùng, các nút chức năng được bố trí để hỗ trợ người dùng thao tác. Sau khi tải ảnh, người dùng có thể nhấn nút *Xử lý ảnh* để hệ thống thực hiện quá trình nhận diện và trích xuất thông tin từ ảnh một cách tự động. Giao diện được thiết kế đơn giản và trực quan, đảm bảo sự tiện lợi và hiệu quả cho người sử dụng.

A screenshot of a computer

Description automatically generated

Hình 3.x: Giao diện chức năng trích xuất thông tin trên thẻ sinh viên

Khi người dùng mở và tương tác với giao diện, họ sẽ thực hiện các bước sau:

Bước 1: Tải ảnh lên

* Nhấn vào nút "Chọn ảnh" để chọn một ảnh từ máy tính.
* Sau khi ảnh được tải lên, ảnh sẽ được hiển thị ngay lập tức trên giao diện.
* Cuối cùng nhấn nút “Gửi ảnh” để tải lên cho hệ thống xử lý

Bước 2: Xử lý ảnh người dùng đã tải lên

* Hệ thống sẽ bắt đầu phân tích và trích xuất các thông tin từ thẻ trong ảnh.
* Quá trình này có thể mất vài giây tùy vào độ phức tạp của ảnh.

Bước 3: Xem kết quả hệ thống trả về

* Sau khi xử lý xong, kết quả trích xuất (tên sinh viên, mã số sinh viên, ngày sinh, v.v.) sẽ được hiển thị trong khu vực kết quả.
* Người dùng có thể kiểm tra thông tin.

Bước 4: Thực hiện lại hoặc thoát

* Người dùng thực hiện lại bước 1 để tiến hành xử lý ảnh mới.
* Nếu muốn thoát khỏi chương trình, người dùng có thể nhấn nút "Thoát".
  + 1. Kiểm thử chức năng

Dưới đây là bảng trình bày các thử nghiệm chi tiết cho chức năng xử lý ảnh đầu vào và trích xuất thông tin trên thẻ sinh viên. Các ca thử nghiệm này bao gồm mục tiêu kiểm thử, kịch bản kiểm thử, kết quả mong đợi và kết quả thực tế để đảm bảo hệ thống hoạt động đúng như mong đợi trong các tình huống khác nhau.

Bảng dưới đây sẽ giúp bạn kiểm tra từng chức năng của hệ thống một cách cụ thể và có cơ sở để so sánh kết quả thực tế với kết quả mong đợi.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Mục tiêu kiểm thử** | **Kịch bản kiểm thử** | **Kết quả mong đợi** | **Kết quả thực tế** |
| 1 | Kiểm thử tải ảnh lên | Nhấn nút “Tải ảnh” và chọn ảnh hợp lệ từ máy | Ảnh được tải lên thành công và hiển thị đúng trên giao diện | Như kết quả mong đợi |
| 2 | Kiểm thử tải ảnh với định dạng không hợp lệ | Chọn file có định dạng không phải là ảnh (ví dụ: .txt, .docx) để tải lên | Hệ thống thông báo lỗi về định dạng | Thông báo định dạng không hợp lệ |
| 3 | Kiểm thử xử lý ảnh với chất lượng cao | Sau khi tải ảnh lên. Kiểm tra kết quả trích xuất được từ ảnh rõ nét | Thông tin (họ tên sinh viên, mã số sinh viên, khoá học, ngành học,…) được trích xuất chính xác | Các thông tin được trích xuất đúng với thực tế |
| 4 | Kiểm thử xử lý với ảnh chất lượng kém | Tải ảnh bị mờ, nhiễu lên. Kiểm tra kết quả trích xuất | Hệ thống thông báo lỗi không trích xuất được thông tin | Trả về kết quả sai hoặc chưa đầy đủ. Không thông báo lỗi |
| 5 | Kiểm thử thông báo lỗi khi không thể nhận diện ảnh | Sử dụng ảnh không phải là thẻ sinh viên (ví dụ: ảnh phong cảnh) và tải ảnh lên | Hệ thống hiển thị thông báo lỗi “Không nhận diện được thông tin” | Thông báo lỗi như mong đợi |
| 6 | Kiểm thử với ảnh thẻ có định dạng khác | Sử dụng ảnh của khoá 61 (định dạng thẻ dọc), tải lên và xử lý | Hệ thống thông báo lỗi không đúng định dạng thẻ | Chưa thông báo lỗi |
| 7 | Kiểm thử giao diện người dùng (UI) | Kiểm tra tính năng các nút “Tải ảnh”, “Xử lý” trên giao diện người dùng | Giao diện dễ sử dụng, các nút chức năng hoạt động đúng và không gặp lỗi. | Giao diện dễ sử dụng, không gặp lỗi |
| 8 | Kiểm thử hệ thống phản hồi khi không có ảnh tải lên | Nhấn "Xử lý ảnh" khi chưa tải ảnh nào lên. | Hệ thống hiển thị thông báo lỗi "Chưa tải ảnh lên". | Thông báo lỗi "Chưa tải ảnh lên". |

Bảng 3.x: Kiểm thử chức năng xử lý ảnh đầu vào và trích xuất thông tin

* 1. Chức năng so khớp thông tin sinh viên với danh sách phòng thi

Chức năng này nhằm mục đích so khớp thông tin sinh viên đã đọc được ở bước xử lý ảnh đầu vào và trích xuất thông tin với danh sách thực tế có trong phòng thi, đảm bảo rằng mỗi sinh viên có thể được phân bổ và có mặt đúng phòng thi giúp việc quản lý và giám sát dễ dàng trong quá trình thi. Dưới đây là các Mô-đun cần thiết để thực hiện chức năng này.

* + 1. Xây dựng các Mô-đun liên quan
       1. Mô-đun đọc danh sách sinh viên từ file excel

**Mục đích**: Hàm này đọc dữ liệu từ file Excel chứa danh sách sinh viên, xác định các cột "Họ tên" và "Mã sinh viên", sau đó trả về danh sách sinh viên với định dạng "HỌ TÊN - MÃ SINH VIÊN".

|  |  |
| --- | --- |
| **Thành phần** | **Mô tả chi tiết** |
| Tham số đầu vào | |  | | --- | |  |  |  | | --- | | excel\_path (str): Đường dẫn đến file Excel cần đọc. | |
| Thư viện sử dụng | pandas: Đọc và xử lý dữ liệu từ file Excel.  unidecode: Loại bỏ dấu tiếng Việt và chuẩn hóa chuỗi. |
| Quy trình hoạt động | 1. Đọc file Excel vào DataFrame bằng thư viện pandas.  2. Xác định cột "Họ tên" và "Mã sinh viên" dựa trên tiêu đề cột.  3. Kiểm tra giá trị hợp lệ, loại bỏ các ô dữ liệu trống hoặc không xác định.  4. Chuyển họ tên thành chữ in hoa, loại bỏ dấu và định dạng đầu ra theo yêu cầu.  5. Trả về danh sách chứa thông tin sinh viên đã được xử lý hoặc danh sách rỗng nếu gặp lỗi. |
| Kết quả trả về | list: Danh sách sinh viên dưới dạng ["HỌ TÊN - MÃ SINH VIÊN"]. |

Bảng 3.x: Bảng mô tả Mô-đun đọc danh sách từ file Excel

def read\_from\_excel(excel\_path): #hàm đọc tt từ file Excel

    try:

        df = pd.read\_excel(excel\_path)

        name\_col = None

        msv\_col = None

        for col in df.columns:

            if "họ tên" in col.lower():

                name\_col = col

            elif "mã sinh viên" in col.lower():

                msv\_col = col

        if name\_col and msv\_col:

            excel\_data = []

            for index, row in df.iterrows():

                name = row[name\_col]

                msv = row[msv\_col]

                if pd.notna(name) and pd.notna(msv):

                    name = unidecode(str(name)).upper()

                    excel\_data.append(f"{name} - {msv}")

            return excel\_data

        else:

            print("Không tìm thấy cột 'Họ tên' và 'Mã sinh viên'.")

            return []

    except Exception as e:

        print(f"Lỗi khi đọc file Excel: {e}")

        return []

Hình 3.x: Mã nguồn Mô-đun đọc danh sách từ file Excel

* + - 1. Mô-đun so khớp thông tin sinh viên với danh sách phòng thi

**Mục đích:** Hàm này kiểm tra mã sinh viên (MSV) được trích xuất từ thông tin OCR có tồn tại trong danh sách sinh viên hay không.

|  |  |
| --- | --- |
| **Thành phần** | **Mô tả chi tiết** |
| Tham số đầu vào | |  | | --- | |  |  |  | | --- | | extracted\_info (dict): Thông tin được trích xuất từ OCR, chứa mã sinh viên (key 'MSV').  students\_list (list): Danh sách sinh viên dưới dạng chuỗi có định dạng "HỌ TÊN - MÃ SINH VIÊN". | |
| Quy trình hoạt động | 1. Kiểm tra nếu thông tin trích xuất từ OCR có chứa mã sinh viên ('MSV').  2. Tách danh sách sinh viên thành từng phần tử với định dạng (Tên, MSV).  3. Duyệt qua từng sinh viên trong danh sách và so sánh mã sinh viên với dữ liệu từ OCR.  4. Nếu tìm thấy sinh viên trùng khớp, in ra thông tin sinh viên và trả về True.  5. Nếu không tìm thấy mã sinh viên hoặc dữ liệu OCR không có mã, trả về False. |
| Kết quả trả về | True: Khi mã sinh viên từ OCR trùng với mã trong danh sách sinh viên.  False: Khi không tìm thấy mã sinh viên hoặc không có mã trong dữ liệu OCR. |

Bảng 3.x: Bảng mô tả Mô-đun so khớp thông tin sinh viên

def compare\_with\_student\_list(extracted\_info, students\_list):

if 'MSV' in extracted\_info:

msv\_from\_ocr = extracted\_info['MSV']

students\_data = [student.split(' - ') for student in students\_list]

for student in students\_data:

student\_name, student\_msv = student[0], student[1]

if msv\_from\_ocr == student\_msv:

print(f"Thông tin sinh viên khớp: {student\_name} - {student\_msv}”)

return True

print("Không tìm thấy mã sinh viên trong danh sách.")

return False

else:

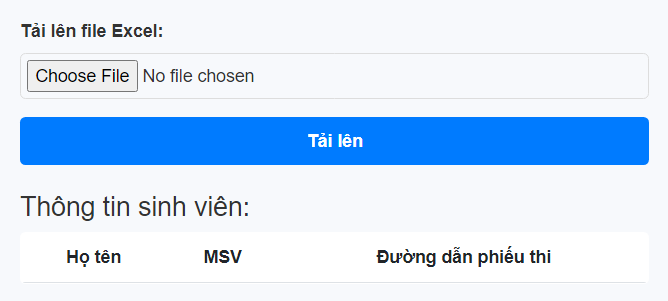
print("Không có mã sinh viên trong dữ liệu OCR.")

return False

Hình 3.x: Mã nguồn Mô-đun so khớp thông tin sinh viên

* + 1. Giao diện chức năng so khớp thông tin sinh viên với danh sách

Giao diện hệ thống được thiết kế đơn giản và dễ sử dụng, bao gồm ba thành phần chính: khu vực tải file Excel, nơi người dùng có thể tải file và kiểm tra đường dẫn file đã chọn; khu vực kết quả trích xuất, hiển thị thông tin sinh viên như họ tên, mã số, thông tin phòng thi cùng trạng thái xử lý (ví dụ: "Trích xuất thành công"); và các nút chức năng, cho phép thực hiện thao tác trích xuất dữ liệu sau khi tải file lên. Sự bố trí hợp lý này giúp người dùng thao tác nhanh chóng và theo dõi hiệu quả quá trình xử lý thông tin.



Hình 3.x: Giao diện chức năng so khớp thông tin sinh viên

**Hướng dẫn thực hiện với giao diện**

1. Khu vực tải file Excel

Nút tải file Excel:

Chức năng: Cho phép người dùng tải file Excel chứa danh sách sinh viên và thông tin phòng thi lên hệ thống để tiến hành xử lý.

Cách thực hiện:

* Nhấn vào nút "Tải file Excel" (thường có biểu tượng "Tải lên" hoặc "Chọn file").
* Cửa sổ chọn file sẽ mở ra. Người dùng cần chọn file Excel có định dạng .xlsx (hoặc .xls) từ máy tính của mình.
* Sau khi chọn xong, nhấn "Mở" để tải lên file.

Hiển thị đường dẫn tải lên:

Chức năng: Sau khi người dùng tải thành công file Excel, hệ thống sẽ hiển thị đường dẫn của file đã tải lên trên giao diện.

Cách thực hiện:

* Sau khi tải file lên, hệ thống sẽ tự động hiển thị đường dẫn hoặc tên của file trên giao diện.
* Người dùng có thể kiểm tra lại tên và đường dẫn của file để đảm bảo file đã được tải lên chính xác.

1. Khu vực kết quả trích xuất

Kết quả trích xuất thông tin:

Chức năng: Hiển thị các thông tin đã được trích xuất từ file Excel sau khi người dùng nhấn nút "Trích xuất thông tin".

Cách thực hiện:

* Sau khi file Excel được tải lên, nhấn nút "Trích xuất thông tin".
* Hệ thống sẽ tiến hành đọc và phân tích nội dung file Excel, trích xuất các thông tin như mã sinh viên, họ tên, thông tin phòng thi (nếu có), và thông tin giám thị.
* Các thông tin này sẽ được hiển thị trên giao diện trong khu vực kết quả trích xuất. Ví dụ: Mã sinh viên, tên sinh viên, phòng thi, giám thị, và trạng thái trích xuất.

Khu vực thông báo trạng thái:

Chức năng: Hiển thị trạng thái của quá trình trích xuất thông tin từ file Excel, giúp người dùng biết liệu quá trình này có thành công hay không.

Cách thực hiện: Sau khi nhấn nút "Trích xuất thông tin", hệ thống sẽ thông báo trạng thái trích xuất.

Các thông báo có thể là:

* "Trích xuất thành công": Thông tin đã được trích xuất và hiển thị đúng trên giao diện.
* "Không nhận diện được thông tin": Nếu hệ thống không thể trích xuất thông tin (do lỗi file, dữ liệu không hợp lệ, v.v.), thông báo này sẽ hiển thị.
* "Lỗi trong quá trình xử lý": Thông báo khi có lỗi hệ thống xảy ra trong quá trình đọc file.

1. Các nút chức năng

**Trích xuất thông tin:**

Chức năng: Sau khi tải file Excel lên, người dùng có thể nhấn nút "Trích xuất thông tin" để hệ thống tiến hành trích xuất thông tin từ file.

Cách thực hiện:

* Sau khi file Excel đã được tải lên và hiển thị đường dẫn, nhấn vào nút "Trích xuất thông tin".
* Hệ thống sẽ bắt đầu quá trình đọc dữ liệu từ file và trích xuất thông tin như mã sinh viên, tên sinh viên, phòng thi và giám thị.
* Sau khi hoàn thành, các thông tin sẽ được hiển thị trong khu vực kết quả trích xuất.

**Các nút chức năng khác:**

Thoát hoặc Đóng: Để đóng giao diện hoặc thoát khỏi ứng dụng.

Tải lại: Nếu người dùng muốn tải lại file mới hoặc bắt đầu lại quá trình xử lý.

* + 1. Kiểm thử chức năng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Mục tiêu kiểm thử** | **Kịch bản kiểm thử** | **Kết quả mong đợi** | **Kết quả thực tế** |
| 1 | Đảm bảo hệ thống có thể tải và xử lý file Excel hợp lệ. | 1. Nhấn nút "Tải file Excel". 2. Chọn file Excel hợp lệ từ máy tính. 3. Nhấn "Mở". | 1. Hệ thống hiển thị đường dẫn của file đã tải lên. 2. Đường dẫn file xuất hiện chính xác trên giao diện. | Như kết quả mong đợi |
| 2 | Đảm bảo hệ thống xử lý đúng các file không phải Excel. | 1. Nhấn nút "Tải file Excel". 2. Chọn file không phải file Excel (ví dụ: .txt, .pdf). 3. Nhấn "Mở". | Hệ thống hiển thị thông báo lỗi, yêu cầu người dùng chọn file Excel hợp lệ. | Thông báo định dạng không hợp lệ |
| 3 | Đảm bảo hệ thống trích xuất thông tin chính xác từ file Excel hợp lệ. | 1. Tải file Excel hợp lệ lên hệ thống. 2. Nhấn nút "Trích xuất thông tin". | 1. Hệ thống hiển thị các thông tin trích xuất từ file: mã sinh viên, họ tên, phòng thi, giám thị. 2. Trạng thái trích xuất thành công được hiển thị. | Các thông tin được trích xuất đúng với thực tế |
| 4 | Đảm bảo hệ thống xử lý khi không có dữ liệu trong file Excel. | 1. Tải file Excel không chứa dữ liệu hợp lệ (ví dụ, chỉ có tiêu đề cột mà không có thông tin). 2. Nhấn nút "Trích xuất thông tin". | Hệ thống hiển thị thông báo "Không nhận diện được thông tin". | Lỗi không thông báo, trả về giao diện trống |
| 5 | Đảm bảo hệ thống thông báo lỗi khi trích xuất dữ liệu sai. | 1. Tải file Excel có dữ liệu sai lệch (ví dụ: mã sinh viên sai, thiếu thông tin phòng thi). 2. Nhấn nút "Trích xuất thông tin". | Hệ thống hiển thị thông báo lỗi hoặc thông báo "Lỗi trong quá trình xử lý" nếu không thể trích xuất thông tin chính xác. | Thông báo lỗi như mong đợi |
| 6 | Đảm bảo hệ thống hiển thị thông báo trạng thái chính xác sau khi trích xuất thông tin thành công. | 1. Tải file Excel hợp lệ lên hệ thống. 2. Nhấn nút "Trích xuất thông tin". | Hệ thống hiển thị thông báo "Trích xuất thành công" trong khu vực thông báo trạng thái. | Chưa thông báo lỗi |
| 7 | Kiểm thử giao diện người dùng (UI) | Kiểm tra tính năng các nút “Tải file”, “Xử lý” trên giao diện người dùng | Giao diện dễ sử dụng, các nút chức năng hoạt động đúng và không gặp lỗi. | Giao diện dễ sử dụng, không gặp lỗi |
| 8 | Đảm bảo hệ thống hiển thị thông báo khi không có thông tin phòng thi trong file. | 1. Tải file Excel hợp lệ nhưng không có thông tin phòng thi. 2. Nhấn nút "Trích xuất thông tin". | Hệ thống hiển thị thông báo "Không có thông tin phòng thi" trong khu vực kết quả trích xuất. | Lỗi không thông báo, trả về giao diện trống |

Bảng 3.x: Kiểm thử chức năng so khớp thông tin sinh viên

* 1. Chức năng đối chiếu khuôn mặt trên thẻ sinh viên với ảnh thực tế

Chức năng này giúp xác thực danh tính sinh viên thông qua việc đối chiếu khuôn mặt trên thẻ sinh viên với hình ảnh thu được từ camera trong thời gian thực, nhằm đảm bảo tính chính xác và hạn chế gian lận.

* + 1. Xây dựng các Mô-đun liên quan
       1. Mô-đun nhận diện và cắt khuôn mặt có trên thẻ sinh viên

**Mục đích:** Mô-đun này có nhiệm vụ phát hiện và cắt khuôn mặt từ ảnh thẻ sinh viên, sau đó chuyển khuôn mặt sang dạng base64 để dễ dàng truyền qua API hoặc lưu trữ.

|  |  |
| --- | --- |
| **Thành phần** | **Mô tả chi tiết** |
| Tham số đầu vào | |  | | --- | |  | | img\_path (str): Đường dẫn đến ảnh cần xử lý (ảnh sinh viên hoặc ảnh từ tài liệu). | | |
| Thư viện sử dụng | - cv2 (OpenCV): Dùng để xử lý ảnh, phát hiện khuôn mặt và thực hiện các phép toán hình ảnh. - logging: Dùng để ghi lại các thông báo và lỗi trong quá trình xử lý. - base64: Chuyển đổi ảnh sang chuỗi base64 để trả về cho API. - PIL: Dùng để chuyển đổi ảnh từ OpenCV sang định dạng có thể mã hóa thành base64. |
| Quy trình hoạt động | 1. Đọc ảnh từ đường dẫn img\_path sử dụng OpenCV. 2. Tiến hành phát hiện khuôn mặt trong ảnh bằng CascadeClassifier (Haar Cascade). 3. Nếu phát hiện khuôn mặt, cắt và phóng to khuôn mặt theo tỉ lệ xác định. 4. Chuyển ảnh khuôn mặt đã cắt thành định dạng base64. 5. Trả về chuỗi base64 của khuôn mặt nếu tìm thấy khuôn mặt, hoặc None nếu không tìm thấy khuôn mặt. |
| Kết quả trả về | str: Chuỗi base64 của khuôn mặt đã được cắt và mã hóa (nếu phát hiện khuôn mặt). Nếu không phát hiện khuôn mặt, trả về None. |

Bảng 3.x: Bảng mô tả Mô-đun nhận diện và cắt khuôn mặt có trên thẻ sinh viên

def process\_student\_id(*img\_path*):

    try:

        img = cv2.imread(img\_path)

        if img is None:

            raise ValueError(f"Không thể đọc được ảnh từ đường dẫn {img\_path}")

        logging.info("Đã đọc thành công ảnh.")

        face\_img = detect\_face(img)

        if face\_img is not None:

   pil\_img = Image.fromarray(cv2.cvtColor(face\_img, cv2.COLOR\_BGR2RGB))

            buffered = io.BytesIO()

            pil\_img.save(buffered, *format*="JPEG")

            img\_str = base64.b64encode(buffered.getvalue()).decode('utf-8')

            logging.info("Đã cắt và mã hóa khuôn mặt thành công.")

            return img\_str

        else:

            logging.warning("Không tìm thấy khuôn mặt.")

            return None

    except Exception as e:

        logging.error(f"Lỗi trong quá trình xử lý: {e}")

        return None

Hình 3.x: Mã nguồn Mô-đun nhận diện và cắt khuôn mặt có trên thẻ sinh viên

* + - 1. Mô-đun đối chiếu khuôn mặt đã cắt trên thẻ với khuôn mặt thực tế

**Mục đích:** Mô-đun này có nhiệm vụ phát hiện và cắt khuôn mặt từ ảnh thẻ sinh viên, sau đó chuyển khuôn mặt sang dạng base64 để dễ dàng truyền qua API hoặc lưu trữ. Mô-đun này sẽ thực hiện các bước xử lý ảnh để tìm khuôn mặt, cắt khuôn mặt ra khỏi ảnh gốc, và cuối cùng chuyển ảnh khuôn mặt sang định dạng base64 để có thể sử dụng trong các ứng dụng web hoặc các dịch vụ API.

|  |  |
| --- | --- |
| **Thành phần** | **Mô tả chi tiết** |
| Tham số đầu vào | |  | | --- | |  | | - uploaded\_image (np.array): Ảnh tải lên dưới dạng ma trận NumPy. - face\_image\_base64 (str): Ảnh khuôn mặt đã được mã hóa dạng base64. - threshold (float): Ngưỡng so sánh (mặc định là 0.6). | | |
| Thư viện sử dụng | - base64: Dùng để giải mã chuỗi base64 thành ảnh. - numpy: Sử dụng cho các phép toán với vector và tính khoảng cách Euclidean. - cv2 (OpenCV): Dùng để xử lý ảnh, chuyển đổi ảnh giữa các định dạng. - PIL: Dùng để mở ảnh từ byte stream. - dlib: Dùng để phát hiện khuôn mặt, dự đoán các đặc trưng của khuôn mặt và trích xuất đặc trưng khuôn mặt. |
| Quy trình hoạt động | 1. Giải mã ảnh từ base64: Chuyển ảnh khuôn mặt từ chuỗi base64 về dạng ảnh. 2. Phát hiện khuôn mặt: Sử dụng dlib để phát hiện khuôn mặt trong cả ảnh tải lên và ảnh base64. 3. Trích xuất vector đặc trưng: Trích xuất vector đặc trưng của khuôn mặt bằng mô hình face\_recognition\_model\_v1 của dlib. 4. So sánh khuôn mặt: Tính khoảng cách Euclidean giữa hai vector đặc trưng, nếu khoảng cách nhỏ hơn ngưỡng threshold, kết luận là khuôn mặt giống nhau. Nếu không, kết luận là khác nhau. 5. Trả kết quả: Trả về kết quả so sánh (cùng một người hoặc khác nhau) và khoảng cách Euclidean. |
| Kết quả trả về | str: Kết quả so sánh hai khuôn mặt dưới dạng chuỗi, ví dụ: "Cùng 1 người (Khoảng cách Euclidean: 0.45)" hoặc "2 người khác nhau (Khoảng cách Euclidean: 1.20)". |

Bảng 3.x: Bảng mô tả Mô-đun đối chiếu khuôn mặt đã cắt với khuôn mặt thực tế

def compare\_faces(uploaded\_image, face\_image\_base64, threshold=0.6):

    image1 = cv2.cvtColor(uploaded\_image, cv2.COLOR\_RGB2BGR)

    image2 = decode\_base64\_image(face\_image\_base64)

    embedding1 = get\_face\_embedding(image1)

    embedding2 = get\_face\_embedding(image2)

    if embedding1 is None:

        return "Ảnh tải lên không tìm thấy khuôn mặt"

    if embedding2 is None:

        return "Ảnh khuôn mặt từ thẻ không tìm thấy khuôn mặt"

    distance = np.linalg.norm(embedding1 - embedding2)

    if distance < threshold:

        return f"Cùng 1 người (Khoảng cách Euclidean: {distance:.2f})"

    else:

        return f"2 người khác nhau (Khoảng cách Euclidean: {distance:.2f})"

Hình 3.x: Mã nguồn Mô-đun đối chiếu khuôn mặt đã cắt với khuôn mặt thực tế

* + 1. Giao diện chức năng đối chiếu khuôn mặt trên thẻ với ảnh thực tế

Giao diện người dùng sẽ là công cụ trực quan giúp cán bộ giám sát và sinh viên dễ dàng thực hiện quy trình đối chiếu khuôn mặt nhanh chóng, chính xác trong các kỳ thi.

A screenshot of a computer

Description automatically generated

* + - 1. Giới thiệu giao diện chính chức năng

Giao diện sẽ gồm các thành phần như sau:

1. Khu vực tải ảnh thẻ sinh viên và ảnh cá nhân sinh viên vào hệ thống:

* Nút tải file ảnh: Cho phép người dùng tải file ảnh lên để hệ thống xử lý.
* Hiển thị đường dẫn tải lên: Sau khi tải ảnh, hệ thống sẽ hiển thị đường dẫn file trên giao diện, giúp người dùng kiểm tra file cần xử lý.

1. Khu vực kết quả trích xuất:

* Kết quả trích xuất thông tin: Hiển thị các thông tin trích xuất từ kết quả so sánh 2 khuôn mặt của hệ thống và thông tin trên thẻ.
  + - 1. Hướng dẫn thực hiện với giao diện

Giao diện sẽ bao gồm các thành phần như sau:

1. Khu vực đưa ảnh thẻ sinh viên và ảnh cá nhân sinh viên vào hệ thống

**Nút tải file ảnh:**

Chức năng: cho phép người dùng nhập ảnh và hiển thị 2 hình ảnh thẻ sinh viên và ảnh chụp sinh viên thực tế

Cách thực hiện:

* Nhấn vào nút "Tải file" theo 2 khu vực cho ảnh thẻ sinh viên và ảnh cá nhân (thường có biểu tượng "Tải lên" hoặc "Chọn file").
* Cửa sổ chọn file sẽ mở ra. Người dùng cần chọn file ảnh có định dạng .png (hoặc .jpg) từ máy tính của mình.
* Sau khi chọn xong, nhấn "Mở" để tải lên file.

**Hiển thị đường dẫn tải lên:**

Chức năng: Sau khi người dùng tải thành công 2 file ảnh, hệ thống sẽ hiển thị đường dẫn của file đã tải lên trên giao diện.

Cách thực hiện:

* Sau khi tải file lên, hệ thống sẽ tự động hiển thị đường dẫn hoặc tên của file trên giao diện.
* Người dùng có thể kiểm tra lại tên và đường dẫn của file để đảm bảo file đã được tải lên chính xác.

1. Khu vực kết quả trích xuất

**Kết quả trích xuất thông tin**

Chức năng: Hiển thị các thông tin đã được trích xuất từ file Excel sau khi người dùng nhấn nút "Trích xuất thông tin".

Cách thực hiện:

* Sau khi 2 file ảnh được tải lên, nhấn nút "Gửi".
* Hệ thống sẽ tiến hành đọc và xử lý 2 ảnh, sử dụng các thuật toán và chương trình đã được cài sẵn để tiến hành so sánh.
* Kết quả sẽ được hiển thị xem có phải cùng 1 người không cùng với thông tin thẻ sinh viên bên cạnh.
  + 1. Kiểm thử chức năng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Mục tiêu kiểm thử** | **Kịch bản kiểm thử** | **Kết quả mong đợi** | **Kết quả thực tế** |
| 1 | Đảm bảo hệ thống có thể tải và xử lý 2 file ảnh hợp lệ. | 1. Nhấn nút "Tải file". 2. Chọn file JPG, PNG, hay các file định dạng hình ảnh hợp lệ từ máy tính. 3. Nhấn "Mở". | 1. Hệ thống hiển thị đường dẫn của file đã tải lên. 2. Đường dẫn file xuất hiện chính xác trên giao diện. | Như kết quả mong đợi |
| 2 | Đảm bảo hệ thống xử lý đúng các file không thuộc định dạng hình ảnh | 1. Nhấn nút "Tải file". 2. Chọn file không phải file ảnh (ví dụ: .txt, .mp3). 3. Nhấn "Mở". | Hệ thống hiển thị thông báo lỗi, yêu cầu người dùng chọn file Excel hợp lệ. | Thông báo định dạng không hợp lệ |
| 3 | Đảm bảo hệ thống hiển thị kết quả chính xác sau khi so sánh thành công. | 1. Tải file ảnh hợp lệ lên hệ thống. 2. Nhấn nút "Gửi ảnh". | Hệ thống hiển thị ngay kết quả so sánh và thông tin thẻ sinh viên. | Chưa thông báo lỗi |
| 4 | Kiểm thử giao diện người dùng (UI) | Kiểm tra tính năng các nút “Tải file”, “Xử lý” trên giao diện người dùng | Giao diện dễ sử dụng, các nút chức năng hoạt động đúng và không gặp lỗi. | Giao diện dễ sử dụng, không gặp lỗi |
| 5 | Đảm bảo hệ thống hiển thị thông báo khi chương trình gặp lỗi | 1. Nhấn nút “Tải file”. 2. Nhấn nút "Gửi ảnh". | Hệ thống hiển thị thông báo "Không xử lý được" trong khu vực kết quả trích xuất. | Lỗi không thông báo, trả về giao diện trống |

Bảng 3.x: Kiểm thử chức năng đối chiếu khuôn mặt trên thẻ sinh viên với ảnh thực tế

* 1. Chức năng xuất phiếu và giám sát quy trình thi

Chức năng xuất phiếu và giám sát quy trình thi đóng vai trò quan trọng trong việc quản lý kỳ thi của sinh viên, đảm bảo tính minh bạch và công bằng trong suốt quá trình thi. Hệ thống sẽ tự động tạo phiếu thi cho từng sinh viên, theo dõi việc tham gia thi và giám sát hành vi của sinh viên trong phòng thi để phát hiện kịp thời các hành vi gian lận. Các mô-đun sẽ được triển khai để kết nối và xử lý thông tin liên quan đến sinh viên, kỳ thi, giám thị, và các hành vi trong phòng thi.

* + 1. Xây dựng các Mô-đun liên quan
       1. Mô-đun in phiếu thi

Mô-đun in phiếu thi có nhiệm vụ tạo ra các phiếu dự thi cho sinh viên dưới dạng file PDF. Chức năng này hỗ trợ việc cấp phát phiếu thi, bao gồm các thông tin cần thiết như tên sinh viên, mã sinh viên, môn thi, mã khóa và vị trí ngồi. Hệ thống sử dụng thư viện FPDF để tạo file PDF, kết hợp với font chữ tùy chỉnh để đảm bảo giao diện đẹp mắt và dễ đọc. Phiếu thi được lưu trữ vào thư mục định sẵn và trả về kết quả với thông tin file PDF đã tạo. Quy trình hoàn thiện mô-đun này giúp đơn giản hóa công tác cấp phát phiếu thi và bảo mật thông tin thi cử.

|  |  |
| --- | --- |
| **Thành phần** | **Mô tả chi tiết** |
| Tham số đầu vào | |  | | --- | |  | | student\_name (str): Tên sinh viên, student\_msv (str): Mã sinh viên, exam\_name (str): Tên môn thi, exam\_code (str): Mã khóa, seat\_position (str): Vị trí ngồi, output\_dir (str): Thư mục xuất file (mặc định là "tickets") | | |
| Thư viện sử dụng | FPDF: Tạo file PDF. |
| Quy trình hoạt động | 1. Kiểm tra thư mục font và tạo thư mục xuất file nếu cần.  2. Tạo file PDF, thêm các thông tin cần thiết vào phiếu thi.  3. Lưu file PDF vào thư mục đã định và trả về thông tin kết quả. |
| Kết quả trả về | Một từ điển chứa status (thành công hay lỗi), message (thông báo), ticket\_file (đường dẫn file PDF) và ticket\_info (danh sách thông tin phiếu thi). |

Bảng 3.x: Bảng mô tả Mô-đun in phiếu thi cho sinh viên

def generate\_exam\_ticket\_pdf(student\_name, student\_msv, exam\_name, exam\_code, seat\_position, output\_dir="tickets"):

    BASE\_DIR = os.path.dirname(os.path.abspath(\_\_file\_\_))

    FONT\_DIR = os.path.join(BASE\_DIR, "tickets”)

    font\_path\_regular = os.path.join(FONT\_DIR, "Roboto-Regular.ttf")

    font\_path\_bold = os.path.join(FONT\_DIR, "Roboto-Bold.ttf")

    if not os.path.exists(font\_path\_regular) or not os.path.exists(font\_path\_bold):

        return {"status": "error", "message": "Font files not found"}

    TICKET\_DIR = os.path.join(BASE\_DIR, output\_dir)

    os.makedirs(TICKET\_DIR, exist\_ok=True)

    ticket\_filename = f"{student\_msv}\_exam\_ticket.pdf"

    ticket\_path = os.path.join(TICKET\_DIR, ticket\_filename)

    pdf = FPDF()

    pdf.add\_page()

    try:

        pdf.add\_font("Roboto", "", font\_path\_regular, uni=True)

        pdf.add\_font("Roboto", "B", font\_path\_bold, uni=True)

    except Exception as e:

        print(f"Không thể tải font: {e}")

        pdf.set\_font("Arial", "", 12)

    pdf.set\_font("Roboto", "B", 16)

    pdf.cell(0, 10, "PHIẾU DỰ THI", ln=True, align="C")

    pdf.ln(10)

    pdf.set\_font("Roboto", "", 12)

    pdf.cell(0, 10, f"Tên sinh viên   : {student\_name}", ln=True)

    pdf.cell(0, 10, f"Mã sinh viên    : {student\_msv}", ln=True)

    pdf.cell(0, 10, f"Tên môn thi     : {exam\_name}", ln=True)

    pdf.cell(0, 10, f"Mã khóa         : {exam\_code}", ln=True)

    pdf.cell(0, 10, f"Vị trí ngồi     : {seat\_position}", ln=True)

    pdf.ln(10)

    pdf.output(ticket\_path)

    return {

        "status": "success",

        "message": "Phiếu thi đã được tạo thành công dưới dạng PDF",

        "ticket\_file": ticket\_path,

        "ticket\_info": {

            "Tên sinh viên": student\_name,

            "Mã sinh viên": student\_msv,

            "Tên môn thi": exam\_name,

            "Mã khóa": exam\_code,

            "Vị trí ngồi": seat\_position

        }

    }

Hình 3.x: Mã nguồn Mô-đun in phiếu thi cho sinh viên

* + - 1. Mô-đun giám sát quy trình thi

Text+img

* + 1. Giao diện chức năng in phiếu và giám sát quy trình thi

Text+img

* + 1. Kiểm thử chức năng

Text+img

* 1. Các chức năng quản lý của quản trị viên
     1. Chức năng quản lý sinh viên
     2. Chức năng quản lý giám thị
     3. Chức năng quản lý phòng thi

KẾT LUẬN

Kết quả đạt được

Những tồn tại và hạn chế

TÀI LIỆU THAM KHẢO

[1] Lê Văn Phùng, *Kỹ nghệ phần mềm*, NXB Thông tin và Truyền thông, 2014.

[2] Lê Văn Phùng, Lê Hương Giang, *Kỹ nghệ phần mềm nâng cao*, NXB Thông tin và truyền thông, 2015.

[3] Đoàn Văn Ban, Nguyễn Thị Tĩnh, *Giáo trình phân tích thiết kế hệ thống hướng đối tượng bằng UML*, NXB Đại học sư phạm, 2011.

[4] Ian Sommerville*, Software Engineering*, Ninth Edition, Addison-Wesley, 2011.

[5] Scott Tilley, Harry J. Rosenblatt, *Systems Analys and Design*, Shelly Cashman Series, 11th Edition, 2016.

**<Thêm một vài tài liệu liên quan đến công cụ ở đây> theo định dạng sau:**

[#] Tên tác giả*, Tên sách*, phiên bản (nếu có), nhà xuất bản, năm xuất bản.