TRƯỜNG ĐẠI HỌC QUY NHƠN

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO THỰC HÀNH CHUYÊN NGÀNH CÔNG NGHỆ PHẦN MỀM**

**ĐỀ TÀI: TÍCH HỢP MODULE TRÍ TUỆ NHÂN TẠO DALL-E ĐỂ TẠO ỨNG DỤNG WEB XỬ LÝ ẢNH KỸ THUẬT SỐ**

**Sinh viên: TRỊNH CHẤN PHÚC**

**Chuyên ngành: CÔNG NGHỆ PHẦN MỀM**

**Lớp: CNTT K43D**

**Giảng viên hướng dẫn: PHẠM TRẦN THIỆN**

**Khóa học: 2020 – 2024**

**Quy Nhơn, ngày 30 tháng 05 năm 2024**

**LỜI CẢM ƠN**

Lời đầu tiên, em xin cảm ơn chân thành đến quý thầy cô trong Bộ môn CNTT, Công Nghệ Thông Tin cùng toàn thể thầy cô trong trường Đại học Quy Nhơn đã tận tình giảng dạy, hướng dẫn và trang bị cho em những kiến thức bổ ích trong suốt quá trình học tập và nghiên cứu tại trường.

Em cũng xin chân thành gửi lời cảm ơn sâu sắc đến thầy PHẠM TRẦN THIỆN người đã tận tình hướng dẫn, trực tiếp chỉ bảo và tạo mọi điều kiện giúp đỡ em trong suốt quá trình làm báo cáo Thực Hành Chuyên Ngành Công Nghệ Phần Mềm.

Sau cùng em xin gửi lời cảm ơn chân thành tới gia đình, bạn bè đã động viên, cổ vũ và đóng góp ý kiến cho em.

Trong quá trình thực hiện đề tài, do kiến thức và kinh nghiệm của bản thân còn nhiều hạn chế nên không thể tránh được sai sót, em mong nhận được sự thông cảm và góp ý từ quý thầy cô để chuyên đề của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

Quy Nhơn, ngày … tháng … năm 2024

**Sinh viên**

Trịnh Chấn Phúc

MỤC LỤC

[DANH MỤC CÁC KÝ HIỆU VIẾT TẮT vi](#_Toc167994377)

[DANH MỤC HÌNH ẢNH vii](#_Toc167994378)

[DANH MỤC BẢNG viii](#_Toc167994379)

[ĐẶT VẤN ĐỀ 1](#_Toc167994380)

[1. Lý do chọn đề tài 1](#_Toc167994381)

[CƠ SỞ LÝ LUẬN 2](#_Toc167994382)

[NỘI DUNG VÀ PHƯƠNG PHÁP NGHIÊN CỨU 3](#_Toc167994383)

[1. Đối tượng nghiên cứu 3](#_Toc167994384)

[2. Phạm vi nghiên cứu 3](#_Toc167994385)

[3. Nội dung nghiên cứu 4](#_Toc167994386)

[4. Phương pháp nghiên cứu 5](#_Toc167994387)

[CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN 6](#_Toc167994388)

[1.1 Tổng quan về quy trình tạo ảnh nghệ thuật của người dùng 6](#_Toc167994389)

[1.2 Tổng quan về ReactJS 6](#_Toc167994390)

[1.3 Tổng quan về Dall-E AI 7](#_Toc167994391)

[1.4 Tổng quan về framework xử lý ảnh react-filerobot-image-editor 9](#_Toc167994392)

[1.5 Tổng quan về hệ quản trị cơ sở dữ liệu MongoDb 10](#_Toc167994393)

[1.6 Tổng quan về ngôn ngữ lập trình Nodejs và framework Express 12](#_Toc167994394)

[CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 14](#_Toc167994395)

[2.1 Phân tích yêu cầu 14](#_Toc167994396)

[2.1.1 Yêu cầu chức năng 14](#_Toc167994397)

[2.1.2 Yêu cầu phi chức năng 14](#_Toc167994398)

[2.2 Thiết kế hệ thống 15](#_Toc167994399)

[2.2.1 Sơ đồ phân cấp chức năng hệ thống 15](#_Toc167994400)

[2.2.2 Sơ đồ luồng dữ liệu 16](#_Toc167994401)

[2.2.2.1 Biểu đồ mức ngữ cảnh 16](#_Toc167994402)

[2.2.3 Sơ đồ Use case 17](#_Toc167994403)

[2.2.3.1 Sơ đồ Use case tổng quan 17](#_Toc167994404)

[2.2.3.2 Sơ đồ Use case dịch vụ ảnh 18](#_Toc167994405)

[2.2.4 Thiết kế sơ đồ hoạt động 19](#_Toc167994406)

[2.2.4.1 Đăng nhập 19](#_Toc167994407)

[2.2.4.2 Đăng ký tài khoản 20](#_Toc167994408)

[2.2.4.5 Tạo ảnh 21](#_Toc167994409)

[2.3 Thiết kế cơ sở dữ liệu 23](#_Toc167994410)

[2.3.1 Từ điển dữ liệu 23](#_Toc167994411)

[2.4 Giao diện và xử lý 24](#_Toc167994412)

[2.4.1 Thiết kế giao diện 24](#_Toc167994413)

[2.4.2 Thiết kế xử lý 27](#_Toc167994414)

[2.4.2.1 Xử lý đăng ký tài khoản 27](#_Toc167994415)

[2.4.2.2 Xử lý đăng nhập 27](#_Toc167994416)

[2.4.2.3 Xử lý chức năng hiển thị thông tin 27](#_Toc167994417)

[2.4.2.4 Xử lý chức năng cập nhật ảnh 27](#_Toc167994418)

[CHƯƠNG 3: TÍCH HỢP MODULE TRÍ TUỆ NHÂN TẠO DALL-E ĐỂ TẠO ỨNG DỤNG WEB SỬ LÝ ẢNH KỸ THUẬT SỐ 28](#_Toc167994419)

[3.1 Xây dựng hệ thống 28](#_Toc167994420)

[3.1.1 Cấu trúc mã lệnh 28](#_Toc167994421)

[3.1.2 Hệ quản trị cơ sở dữ liệu 29](#_Toc167994422)

[3.2 Giao diện phần mềm 30](#_Toc167994423)

[3.2.1 Giao diện trang chủ 30](#_Toc167994424)

[3.2.2 Đăng ký 31](#_Toc167994425)

[3.2.3 Đăng nhập 32](#_Toc167994426)

[3.2.4 Chỉnh sửa ảnh 32](#_Toc167994427)

[3.3 Kết quả và thảo luận 34](#_Toc167994428)

[3.4 Kết luận và kiến nghị 35](#_Toc167994429)

[TÀI LIỆU THAM KHẢO 36](#_Toc167994430)

[PHỤ LỤC 37](#_Toc167994431)

# **DANH MỤC CÁC KÝ HIỆU VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| Từ viết tắt | Từ đầy đủ | Nghĩa của từ |
| CNTT | Công nghệ thông tin | Công nghệ thông tin |
| CSDL | Cơ sở dữ liệu | Cơ sở dữ liệu |
| HTTP | HyperText Transfer Protocol | Giao thức truyền tải siêu văn bản |
| API | Application Programming Interface | Giao diện lập trình ứng dụng |
| HTML | Hypertext Markup Language | Ngôn ngữ đánh dấu siêu văn bản |
| UI | User Interface | Giao diện người dùng |
| DOM | Document Object Model | Mô hình các đối tượng tài liệu |
| URL | A Uniform Resource Locator | Đường dẫn của trang web |

# **DANH MỤC HÌNH ẢNH**

[Hình 1. 1 Cấu trúc thư mục ReactJS 8](#_Toc166030538)

[Hình 2. 1 Sơ đồ phân cấp chức năng 13](#_Toc166030539)

[Hình 2. 2 Biểu đồ mức ngữ cảnh 14](#_Toc166030540)

[Hình 2. 4 Sơ đồ Use case tổng quát 15](#_Toc166030541)

[Hình 2. 5 Sơ đồ Use case quản lý dịch vụ ảnh 15](#_Toc166030542)

[Hình 2. 8 Sơ đồ hoạt động đăng nhập 17](#_Toc166030543)

[Hình 2. 9 Sơ đồ hoạt động đăng ký tài khoản 18](#_Toc166030544)

[Hình 2. 12 Sơ đồ hoạt động tạo ảnh 19](#_Toc166030545)

[Hình 2. 17 Thiết kế giao diện trang chủ 20](#_Toc166030546)

[Hình 2. 18 Thiết kế giao diện trang chỉnh sửa ảnh 21](#_Toc166030547)

[Hình 2. 19 Thiết kế giao diện trang đăng ký 22](#_Toc166030548)

[Hình 2. 19 Thiết kế giao diện trang đăng nhập 22](#_Toc166030549)

[Hình 3. 1 Cấu trúc thư mục của project 24](#_Toc166030550)

[Hình 3. 2 Cơ sở dữ liệu Mongodb 26](#_Toc166030551)

[Hình 3. 3 Giao diện trang chủ 28](#_Toc166030552)

[Hình 3. 4 Giao diện trang đăng ký 28](#_Toc166030553)

[Hình 3. 6 Giao diện form đăng nhập 29](#_Toc166030554)

[Hình 3. 8 Giao diện trang chỉnh sửa ảnh 30](#_Toc166030555)

[Hình 4. 1 Biểu mẫu đăng nhập 35](#_Toc166030556)

[Hình 4. 2 Biểu mẫu đăng ký tài khoản 36](#_Toc166030557)

# **DANH MỤC BẢNG**

[Bảng 2. 1 Bảng mô tả sơ đồ Use case quản lý dịch vụ ảnh 16](#_Toc166030423)

[Bảng 2. 4 Bảng users 19](#_Toc166030424)

[Bảng 2. 6 Bảng image 20](#_Toc166030425)

[Bảng 4. 1 API Docs 55](#_Toc136778467)

**ĐẶT VẤN ĐỀ**

1. **Lý do chọn đề tài**

Với việc Internet ngày càng phát triển. Nhu cầu làm đẹp cũng như về về nghệ thuật của con người càng cao. Ai cũng muốn có thể tạo ra các bức ảnh cho riêng mình với các chủ đề do chính mình nghĩ ra. Nhưng khả năng hội hoạ của mỗi người lại khác nhau. Nên việc tạo ra các bức ảnh tự động từ máy tính mà không cần đến kỹ năng hội hoạ là một nhu cầu rất cao.

Ngày nay, việc dùng internet để tìm kiếm thông tin và xử lý những vấn đề trong đời sống trở nên phổ biến hơn bao giờ hết. Đặt biệt lĩnh vực AI ngày càng phát triển và giải quyết được nhiều nhu cầu hàng ngày. Cụ thể ở đây là ngày càng có nhiều công cụ AI có thể giúp người dùng tạo ra các bức ảnh nghệ thuật từ những nội dung mô tả của người dùng. Có thể kể đến các công cụ AI phổ biến: Dall-e, text2image.

Việc xây dựng website tạo ảnh nghệ thuật bằng công nghệ là AI một giải pháp hữu hiệu để giải quyết vấn đề này, giúp cho người dung có dễ dàng tiếp cận và nhanh chóng tạo được các bức ảnh ưng ý theo ý mình.

Sau một quá trình học tập, nghiên cứu em nhận thấy được tiềm năng phát triển của lĩnh vực ảnh nghệ thuật, nhu cầu tạo ảnh nghệ thuật ngày càng cao. Chình vì thế, được sự đồng ý của GVHD em chọn đề tài “**Tích hợp module trí tuệ nhân tạo Dall-e để tạo ứng dụng Web sử lý ảnh kỹ thuật số**”.

**CƠ SỞ LÝ LUẬN**

Hiện nay, trên thị trường Việt Nam cũng như trên toàn thế giới có khá nhiều trang web về sàn giao dịch việc làm khác nhau. Một số website sàn giao dịch việc làm phổ biến như:

1. Tính sáng tạo và độc đáo của hình ảnh:

* DALL-E AI có khả năng tạo ra hình ảnh độc đáo và không giới hạn từ mô tả văn bản.
* Tích hợp công nghệ này vào một trang web tạo ảnh nghệ thuật có thể tạo ra những tác phẩm mới mẻ và sáng tạo mà người dùng chưa từng trải nghiệm trước đó.

1. Trải nghiệm người dùng tương tác:

* Sử dụng nền tảng ReactJS để xây dựng trang web có trải nghiệm người dùng mượt mà và tương tác.
* Người dùng có thể nhập mô tả văn bản và chứng kiến quá trình sinh ra hình ảnh nghệ thuật ngay trước mắt thông qua giao diện tương tác.

1. Phát triển dễ dàng và mở rộng:

* ReactJS cung cấp một cơ sở codebase linh hoạt và dễ dàng mở rộng, cho phép phát triển và bảo trì dự án một cách hiệu quả.
* Điều này cung cấp cơ sở cho việc tích hợp các tính năng mới và cải thiện trải nghiệm người dùng theo thời gian.

1. Khả năng tích hợp và mở rộng:

* ReactJS cho phép tích hợp các thư viện và API bên ngoài một cách dễ dàng.
* Công nghệ DALL-E AI có thể được tích hợp thông qua các API hoặc thư viện JavaScript, giúp tạo ra một trải nghiệm đa dạng và phong phú cho người dùng.

1. Tính ứng dụng thực tiễn:

* Trang web có thể cung cấp cho người dùng một cách mới để thể hiện sự sáng tạo của họ thông qua việc tạo ra hình ảnh nghệ thuật từ mô tả văn bản.
* Điều này có thể hỗ trợ cả những người làm nghệ thuật và người dùng bình thường muốn khám phá sức sáng tạo của mình.

Tóm lại, việc xây dựng một trang web tạo ảnh nghệ thuật tích hợp công nghệ DALL-E AI trên nền tảng ReactJS mang lại nhiều tiềm năng và cơ hội phát triển, đồng thời cung cấp một trải nghiệm sáng tạo và độc đáo cho người dùng.

**NỘI DUNG VÀ PHƯƠNG PHÁP NGHIÊN CỨU**

* 1. **Đối tượng nghiên cứu**
* Thư viện Javascript ReactJS.
* Ngôn ngữ lập trình Nodejs.
* Dall-e API
* Hệ quản trị cơ sở dữ liệu mongodb.
  1. **Phạm vi nghiên cứu**
* Nghiên cứu về công nghệ tảo ảnh bằng Dall-e API.
* Nghiên cứu về quy trình và các chức năng chỉnh sửa ảnh.
* Nghiên cứu các chức năng của người dùng:

+ Đăng ký tài khoản, đăng nhập vào website.

+ Tạo ảnh bằng Dall-e api

+ Chỉnh sửa ảnh.

+ Tương tác giữa người dùng.

* 1. **Nội dung nghiên cứu**

**CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN**

* 1. Tổng quan về quy tạo ảnh nghệ thuật
  2. Tổng quan về ReactJS
  3. Tổng quan về Dal-E AI
  4. Tổng quan về framework xử lý ảnh react-filerobot-image-editor
  5. Tổng quan về hệ quản trị cơ sở dữ liệu mongodb
  6. Tổng quan về ngôn ngữ lập trình nodejs và framework express

**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

2.1 Phân tích yêu cầu

2.1.1 Yêu cầu chức năng

2.1.2 Yêu cầu phi chức năng

2.2 Thiết kế hệ thống

2.2.1 Sơ đồ phân cấp chức năng hệ thống

2.2.2 Sơ đồ luồng dữ liệu

2.2.3 Sơ đồ Usecase

2.2.4 Sơ đồ hoạt động

2.3 Thiết kế cơ sở dữ liệu

2.3.1 Từ điển dữ liệu

2.3.2 Mô hình quan hệ

2.3.3 Mô hình vật lý

2.4 Giao diện và xử lý

2.4.1 Thiết kế giao diện

2.4.2 Thiết kế xử lý

**CHƯƠNG 3: TÍCH HỢP MODULE TRÍ TUỆ NHÂN TẠO DALL-E ĐỂ TẠO ỨNG DỤNG WEB SỬ LÝ ẢNH KỸ THUẬT SỐ**

3.1 Xây dựng hệ thống

3.1.1 Cấu trúc mã lệnh

3.1.2 Hệ quản trị cơ sở dữ liệu

3.2 Giao diện phần mềm

3.3 Kết quả và thảo luận

3.4 Kết luận và kiến nghị

* 1. **Phương pháp nghiên cứu**
* Phương pháp nghiên cứu tài liệu:

+ Tổng hợp các kiến thức đã học về ngôn ngữ lập trình, cơ sở dữ liệu, phân tích và thiết kế hệ thống, lập trình web cơ bản và nâng cao.

+ Nghiên cứu về ngôn ngữ lập trình Nodejs, hệ quản trị cơ sở dữ liệu mongodb, Javascript và thư viện ReactJS.

+ Nghiên cứu tạo ảnh bằng Dall-e api

+ Nghiên cứu về các vấn đề liên quan đến tạo ảnh nghệ thuật và một số chức năng chỉnh sửa ảnh

* Phương pháp thực nghiệm:

+ Tìm hiểu các chức năng của các công cụ để phân tích và thiết kế hệ thống, xây dựng cơ cở dữ liệu, thiết kế, xây dựng giao diện và các chức năng của website.

+ Triển khai thử nghiệm website.

# **CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN**

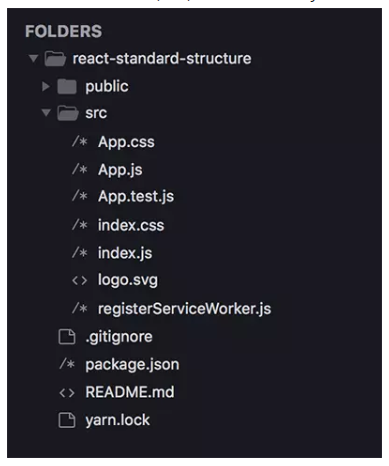
* 1. **Tổng quan về quy trình tạo ảnh nghệ thuật của người dùng**
* Quy của người dùng khi tạo và chỉnh sửa ảnh nghệ thuật bước sau:
* Đăng ký tài khoản: Khách hàng và người làm đăng ký tài khoản trên trang web.
* Đăng nhập vào trang web bằng tài khoản của người dùng.
* Nhập mô tả bức ảnh cần tạo.
* Bấm tạo ảnh.
* Ảnh được tạo sẽ hiển thị trong dách sách phía tay phải
* Sau khi bức ảnh đã tạo. chọn một bức ảnh và ấn chỉnh sửa.
* Sau khi chỉnh sửa ảnh nhấn lưu để lưu lại.
  1. **Tổng quan về ReactJS**

ReactJS là một thư viện JavaScript mã nguồn mở, được phát triển bởi Facebook để xây dựng các ứng dụng web động, tương tác và có hiệu suất cao. ReactJS cho phép tạo các thành phần UI động, có thể được tái sử dụng, cũng như quản lý trạng thái ứng dụng và các sự kiện tương tác với người dùng.

ReactJS sử dụng cách tiếp cận "virtual DOM" để quản lý các thành phần của ứng dụng. Thay vì cập nhật trực tiếp các thành phần khi có thay đổi, ReactJS tạo ra một bản sao ảo của DOM và chỉ cập nhật lại các thành phần có sự thay đổi. Điều này giúp cải thiện hiệu suất ứng dụng và giảm thiểu thời gian render.

ReactJS được thiết kế để tạo ra các ứng dụng đơn trang (single-page applications) và có thể tích hợp với các thư viện và framework khác như Redux, AngularJS và BackboneJS.

Với cộng đồng lớn và sự phát triển nhanh chóng của ReactJS, nó trở thành một trong những công cụ phổ biến nhất để xây dựng các ứng dụng web động hiện nay.



Hình 1. 1 Cấu trúc thư mục ReactJS

* 1. **Tổng quan về Dall-E AI**

DALL-E là một hệ thống trí tuệ nhân tạo do OpenAI phát triển, nổi bật với khả năng tạo ra hình ảnh từ văn bản mô tả. Dưới đây là một số điểm chính về DALL-E:

* **Khả năng Tạo Hình Ảnh**:

DALL-E có khả năng tạo ra hình ảnh từ các mô tả văn bản chi tiết. Chẳng hạn, bạn có thể yêu cầu DALL-E tạo một hình ảnh của “một con mèo mặc áo vest và đội mũ” và hệ thống sẽ tạo ra một hình ảnh phản ánh mô tả đó một cách đáng kinh ngạc.

* **Kiến Trúc và Cơ Chế Hoạt Động**

DALL-E được xây dựng dựa trên kiến trúc GPT-3, một mô hình ngôn ngữ lớn cũng do OpenAI phát triển. Thay vì chỉ xử lý ngôn ngữ, DALL-E được đào tạo để hiểu và tạo ra hình ảnh từ văn bản. Quá trình này liên quan đến việc mã hóa văn bản thành một định dạng mà mô hình có thể sử dụng để tạo ra hình ảnh tương ứng.

* **Ứng Dụng và Tiềm Năng**

DALL-E có rất nhiều ứng dụng tiềm năng, từ thiết kế đồ họa, nghệ thuật, quảng cáo cho đến trò chơi và giáo dục. Khả năng tạo ra hình ảnh theo yêu cầu giúp giảm bớt thời gian và công sức so với việc phải tự tạo hình ảnh từ đầu.

* **Hạn Chế và Thách Thức**

Dù có tiềm năng lớn, DALL-E cũng đối mặt với một số hạn chế và thách thức:

Chất lượng Hình Ảnh: Dù các hình ảnh do DALL-E tạo ra rất ấn tượng, đôi khi chất lượng và độ chi tiết của chúng vẫn chưa đạt mức tối ưu.

Đạo Đức và Sử Dụng Sai Mục Đích: Như với bất kỳ công nghệ AI nào, việc sử dụng DALL-E cũng đặt ra các vấn đề về đạo đức và khả năng bị lạm dụng để tạo ra nội dung giả mạo hoặc không phù hợp.

* **Phát Triển và Tương Lai**

OpenAI liên tục cải tiến và phát triển DALL-E. Phiên bản đầu tiên của DALL-E đã được giới thiệu vào năm 2021, và phiên bản DALL-E 2 sau đó đã ra mắt với khả năng tạo ra hình ảnh có độ phân giải cao hơn và chi tiết hơn. Tương lai của DALL-E hứa hẹn sẽ mang đến nhiều cải tiến và ứng dụng mới, mở rộng hơn nữa khả năng của trí tuệ nhân tạo trong việc tạo ra nội dung.

* **Tổng Kết**

DALL-E đại diện cho một bước tiến lớn trong lĩnh vực AI, minh họa cho tiềm năng kết hợp giữa ngôn ngữ và hình ảnh để tạo ra những công cụ sáng tạo mạnh mẽ và linh hoạt.

* 1. **Tổng quan về framework xử lý ảnh react-filerobot-image-editor**

React-filerobot-image-editor là một framework xử lý ảnh được xây dựng trên nền tảng React. Nó cung cấp một bộ công cụ mạnh mẽ và dễ sử dụng để chỉnh sửa và xử lý ảnh trực tiếp trên trình duyệt. Dưới đây là một tổng quan chi tiết về framework này:

* **Giới Thiệu Chung**

React-filerobot-image-editor là một thư viện mã nguồn mở do Scaleflex phát triển, cho phép người dùng chỉnh sửa hình ảnh trong các ứng dụng web. Thư viện này được thiết kế để tích hợp dễ dàng vào các dự án React, cung cấp các tính năng chỉnh sửa ảnh phong phú và giao diện người dùng thân thiện.

* **Các Tính Năng Chính**

Chỉnh Sửa Cơ Bản: Bao gồm cắt, xoay, lật, thay đổi kích thước, và chỉnh sửa độ sáng, độ tương phản, độ bão hòa, và nhiệt độ màu.

Bộ Lọc và Hiệu Ứng: Cung cấp nhiều bộ lọc và hiệu ứng hình ảnh để tạo ra các phong cách khác nhau.

Chèn Văn Bản và Hình Dán: Cho phép người dùng thêm văn bản và hình dán vào hình ảnh, với khả năng tùy chỉnh về màu sắc, kích thước, và vị trí.

Vẽ và Làm Mờ: Công cụ vẽ cho phép người dùng tự do vẽ trên hình ảnh, cũng như công cụ làm mờ để chỉnh sửa và cải thiện hình ảnh.

* **Ưu Điểm**

Dễ Dàng Tích Hợp: Được thiết kế để dễ dàng tích hợp vào các dự án React, giúp tiết kiệm thời gian và công sức cho các nhà phát triển.

Giao Diện Thân Thiện: Cung cấp giao diện người dùng thân thiện, dễ sử dụng ngay cả với những người không chuyên về kỹ thuật.

Tùy Biến Cao: Cho phép tùy chỉnh giao diện và tính năng để phù hợp với nhu cầu cụ thể của từng ứng dụng.

* **Hạn Chế**

Phụ Thuộc Vào Internet: Một số tính năng yêu cầu kết nối internet để sử dụng dịch vụ đám mây.

Giới Hạn Tính Năng: So với các phần mềm chỉnh sửa ảnh chuyên nghiệp như Photoshop, các tính năng của React-Filerobot-Image-Editor có thể bị giới hạn.

* **Tổng kết**

React-Filerobot-Image-Editor là một công cụ mạnh mẽ và tiện dụng cho việc chỉnh sửa ảnh trong các ứng dụng web React. Với bộ công cụ phong phú và giao diện người dùng trực quan, nó là lựa chọn lý tưởng cho các dự án yêu cầu tính năng chỉnh sửa ảnh nhanh chóng và dễ dàng.

* 1. **Tổng quan về hệ quản trị cơ sở dữ liệu MongoDb**

MongoDB là một hệ quản trị cơ sở dữ liệu (DBMS) phổ biến, thuộc loại NoSQL, được phát triển bởi MongoDB Inc. Nó sử dụng mô hình lưu trữ dữ liệu dạng tài liệu (document-oriented) và cấu trúc dữ liệu linh hoạt để lưu trữ và truy xuất thông tin.

* Kiến trúc:
  + MongoDB sử dụng mô hình tài liệu (document model), trong đó mỗi bản ghi được biểu diễn dưới dạng một tài liệu JSON (JavaScript Object Notation).
  + Tài liệu MongoDB được tổ chức thành các bộ sưu tập (collections), tương tự như các bảng trong hệ quản trị cơ sở dữ liệu quan hệ (RDBMS).
* Tính linh hoạt:
  + MongoDB không yêu cầu schema cố định cho các tài liệu. Mỗi tài liệu trong cùng một bộ sưu tập có thể có các trường (fields) khác nhau.
  + Điều này mang lại tính linh hoạt cao cho việc lưu trữ dữ liệu, cho phép dễ dàng thay đổi cấu trúc của dữ liệu mà không cần phải thay đổi schema.
* Tính tương thích với các ngôn ngữ lập trình:
  + MongoDB cung cấp các thư viện và driver cho nhiều ngôn ngữ lập trình phổ biến như Python, Java, JavaScript (Node.js), và nhiều ngôn ngữ khác.
  + Điều này giúp cho việc phát triển ứng dụng trên nền tảng MongoDB trở nên dễ dàng và linh hoạt.
* Hỗ trợ cho các truy vấn phức tạp:
  + MongoDB cung cấp một ngôn ngữ truy vấn mạnh mẽ và linh hoạt (MongoDB Query Language) để truy vấn dữ liệu từ bộ sưu tập.
  + Nó hỗ trợ các truy vấn phức tạp như truy vấn dưới dạng biểu thức chính quy, truy vấn dựa trên điều kiện logic, và các phép toán tự nhiên.
* Tính cao cấp và mở rộng:
  + MongoDB hỗ trợ các tính năng cao cấp như replica set để đảm bảo tính sẵn sàng và sao lưu dữ liệu, và sharding để mở rộng or phân tán cơ sở dữ liệu trên nhiều server.
  + Điều này cho phép MongoDB phù hợp với các ứng dụng có yêu cầu khối lượng dữ liệu lớn và tính sẵn sàng cao.
* ***Tóm lại MongoDB là một hệ quản trị cơ sở dữ liệu phổ biến, linh hoạt và mở rộng, được sử dụng rộng rãi trong nhiều ứng dụng khác nhau từ ứng dụng web đến các hệ thống phân tích dữ liệu.***
  1. **Tổng quan về ngôn ngữ lập trình Nodejs và framework Express**

Node.js là một môi trường thực thi mã nguồn mở, được xây dựng dựa trên JavaScript, cho phép chạy mã JavaScript ở phía máy chủ. Được phát triển bởi Ryan Dahl vào năm 2009, Node.js đã trở thành một trong những công nghệ phổ biến nhất cho việc phát triển ứng dụng web real-time và hệ thống mạng. Dưới đây là một tổng quan về Node.js và framework Express:

* Node.js:
  + Node.js cho phép viết mã JavaScript chạy phía máy chủ, không chỉ phía máy khách (trình duyệt).
  + Với Node.js, bạn có thể xây dựng các ứng dụng web real-time như ứng dụng chat, trò chơi trực tuyến, streaming video, và các ứng dụng có khả năng xử lý đa luồng cao.
  + Ưu điểm của Node.js:
  + Hiệu suất cao: Node.js được xây dựng dựa trên một kiến ​​trúc không đồng bộ (asynchronous), giúp tối ưu hóa hiệu suất và sử dụng tài nguyên hiệu quả.
  + Cộng đồng lớn: Node.js có một cộng đồng phát triển rộng lớn, với nhiều thư viện và framework phổ biến, hỗ trợ việc phát triển ứng dụng nhanh chóng và hiệu quả.
* Express.js:
  + Express.js là một framework Node.js phổ biến, được sử dụng để xây dựng các ứng dụng web và API.
  + Nó cung cấp một lớp trừu tượng lên trên Node.js để giúp quản lý và tổ chức ứng dụng web một cách dễ dàng và hiệu quả.
  + Ưu điểm của Express.js:
  + Dễ học và sử dụng: Express.js có cú pháp đơn giản và dễ hiểu, giúp cho việc phát triển ứng dụng nhanh chóng và dễ dàng.
  + Linh hoạt: Express.js cho phép bạn tự do tổ chức cấu trúc ứng dụng của mình và sử dụng các middleware để mở rộng chức năng của nó.
  + Hỗ trợ mạnh mẽ: Express.js có một cộng đồng lớn và phong phú, với nhiều middleware và module mở rộng, giúp giải quyết nhiều vấn đề khác nhau trong quá trình phát triển.

Tóm lại, Node.js và Express.js cung cấp một nền tảng mạnh mẽ cho việc phát triển các ứng dụng web và API hiệu suất cao, linh hoạt và dễ dàng sử dụng. Sự kết hợp giữa Node.js và Express.js thường được sử dụng rộng rãi trong cộng đồng phát triển web để tạo ra các ứng dụng mạnh mẽ và linh hoạt.

# **CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

## Phân tích yêu cầu

### 2.1.1 Yêu cầu chức năng

+ Đăng ký tài khoản.

+ Đăng nhập, đăng xuất.

+ Tạo ảnh

+ Hiển thị danh sách ảnh đã tạo của user

+ Hiển thị danh sách ảnh yêu thích

+ Chỉnh sửa ảnh

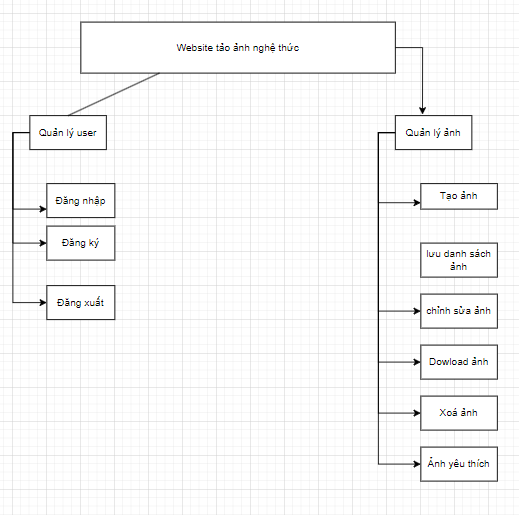
+ Dowload ảnh

### 2.1.2 Yêu cầu phi chức năng

* Tốc độ xử lý nhanh, chính xác.
* Bảo đảm an toàn dữ liệu thông tin người dùng.
* Giao diện, bố cục trang web hợp lý, thân thiện với người dùng.
* Hệ thống ổn định, dễ nâng cấp.

## Thiết kế hệ thống

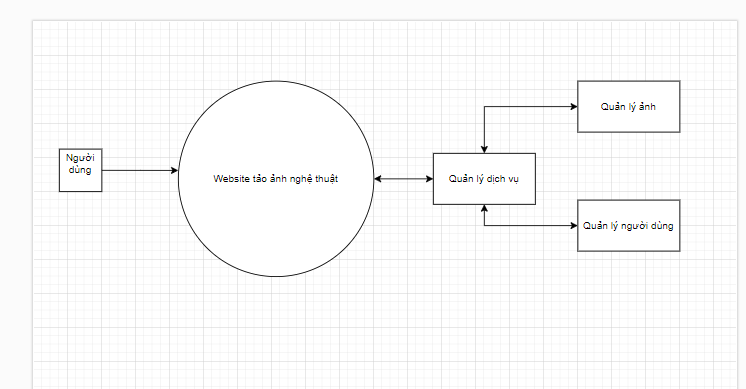
### 2.2.1 Sơ đồ phân cấp chức năng hệ thống



Hình 2. 1 Sơ đồ phân cấp chức năng

### 2.2.2 Sơ đồ luồng dữ liệu

### 2.2.2.1 Biểu đồ mức ngữ cảnh

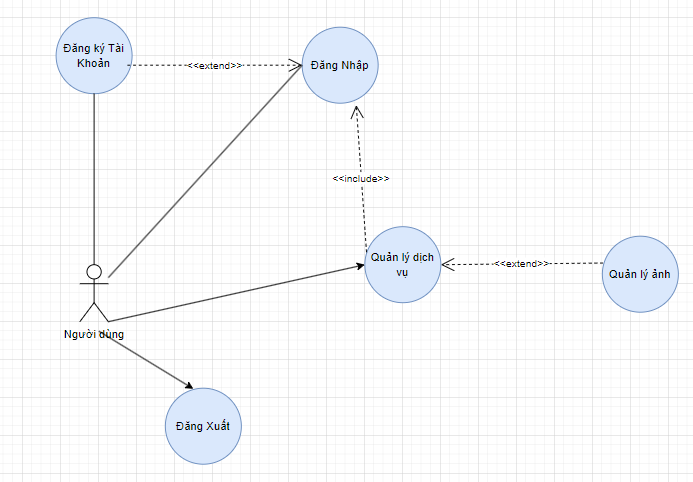


Hình 2. 2 Biểu đồ mức ngữ cảnh

### 2.2.3 Sơ đồ Use case

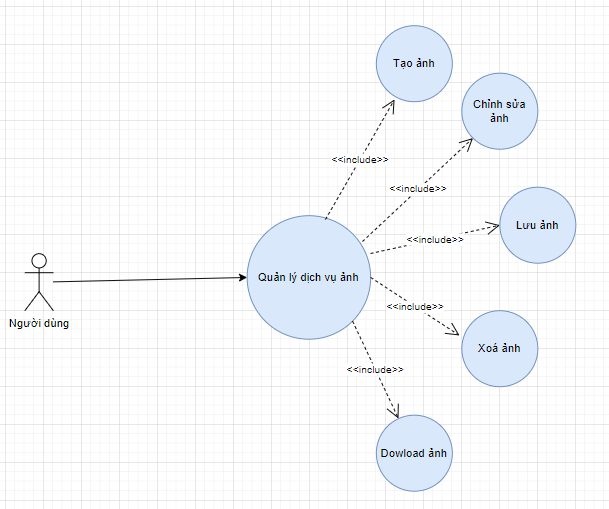
### 2.2.3.1 Sơ đồ Use case tổng quan

Tác nhân chính:

- Người dùng: Là người sử dụng các chức năng trên website.

Hình 2. 4 Sơ đồ Use case tổng quát

### 2.2.3.2 Sơ đồ Use case dịch vụ ảnh



Hình 2. 5 Sơ đồ Use case quản lý dịch vụ ảnh

* Mô tả sơ đồ Use case quản lý dịch vụ ảnh:

Bảng 2. 1 Bảng mô tả sơ đồ Use case quản lý dịch vụ ảnh

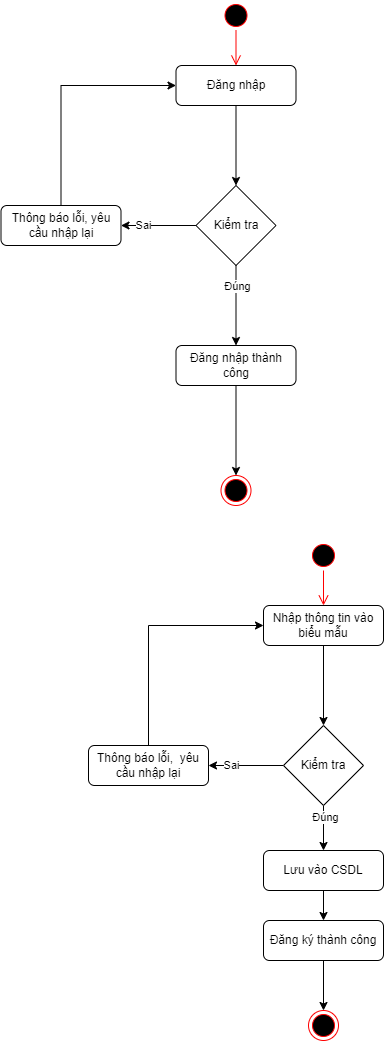
|  |  |
| --- | --- |
| **Tên Use case** | **Use case quản lý dịch vụ** |
| Tác nhân | Người dùng. |
| Mô tả | Cho phép tạo, sửa, xóa, download ảnh. |
| Điều kiện khởi tạo | Người dùng mở trang web và đăng nhập thành công. |
| Dòng sự kiện | Mở trang web và đăng nhập thành công. |
| Điều kiện sau | Hệ thống gửi thông tin cập nhật lên server lưu lại trên CSDL và trả về thông báo cho người dùng. |
| Điều kiện thoát | Khi chức năng thực hiện thành công.  Khi người dùng đăng xuất. |

### 2.2.4 Thiết kế sơ đồ hoạt động

### 2.2.4.1 Đăng nhập

Mô tả sơ đồ hoạt động đăng nhập:

* Chức năng đăng nhập bắt đầu khi người dùng có tài khoản nhưng chưa đăng nhập vào hệ thống truy cập vào trang đăng nhập.
* Người dùng thực hiện đăng nhập bằng cách điền đầy đủ thông tin trên form hiển thị bao gồm tên đăng nhập và mật khẩu và nhấn nút đăng nhập.
* Hệ thống sẽ kiểm tra thông tin đăng nhập, nếu sai hiển thị thông báo lỗi, nếu thành công sẽ lưu dữ liệu người dùng đăng nhập vào local Storage và chuyển hướng đến trang chủ.

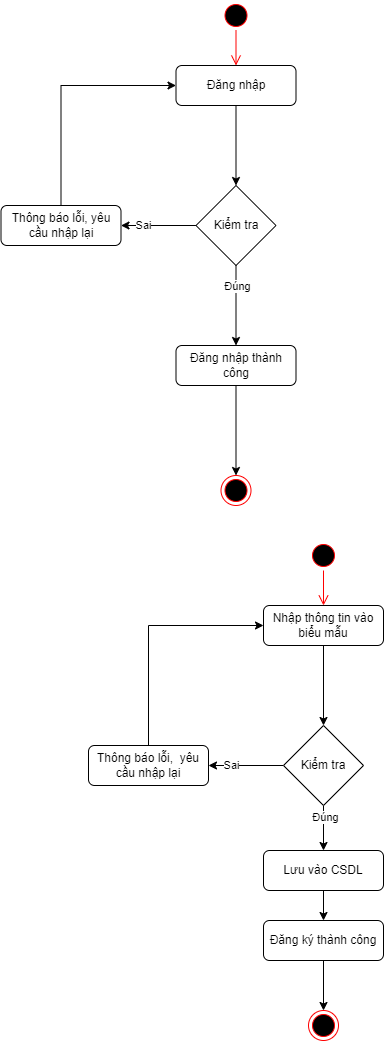


Hình 2. 8 Sơ đồ hoạt động đăng nhập

### 2.2.4.2 Đăng ký tài khoản

Mô tả sơ đồ hoạt động đăng ký tài khoản:

* Chức năng đăng ký bắt đầu khi người dùng chưa có tài khoản và truy cập vào trang đăng ký.
* Người dùng nhập các thông tin cần thiết trong form đăng ký và nhấn nút đăng ký để gửi thông tin.
* Hệ thống sẽ kiểm tra tính hợp lệ của thông tin đăng ký, bao gồm tên người dùng đã được sử dụng chưa. Nếu thông tin hợp lệ, lưu thông tin đăng ký của người dùng vào CSDL và chuyển hướng đến trang đăng nhập. Ngược lại xuất ra thông báo lỗi và yêu cầu nhập lại.

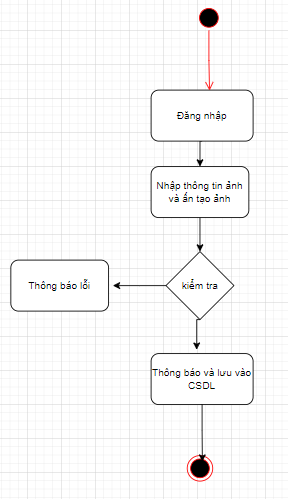


Hình 2. 9 Sơ đồ hoạt động đăng ký tài khoản

### 2.2.4.5 Tạo ảnh

Mô tả sơ đồ hoạt động đặt dịch vụ:

* Chức năng bắt đầu khi người dùng truy cập vào trang
* Người dùng nhập thông tin của bức ảnh
* Hệ thống kiểm tra thông tin, nếu hợp lệ, tạo ảnh thành công, lưu thông tin vào CSDL và trả data về cho người dùng hiển thị



Hình 2. 12 Sơ đồ hoạt động tạo ảnh

## Thiết kế cơ sở dữ liệu

### 2.3.1 Từ điển dữ liệu

* Bảng users:

Bảng 2. 2 Bảng users

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ý nghĩa** |
| 1 | \_id | string | Khóa, tự động tăng |
| 2 | userName | string | Tên đăng nhập |
| 3 | password | string | Mật khẩu |
| 4 | email | string | email |
| 5 | images | Array(string) | Danh sách image id của user |

* Bảng images:

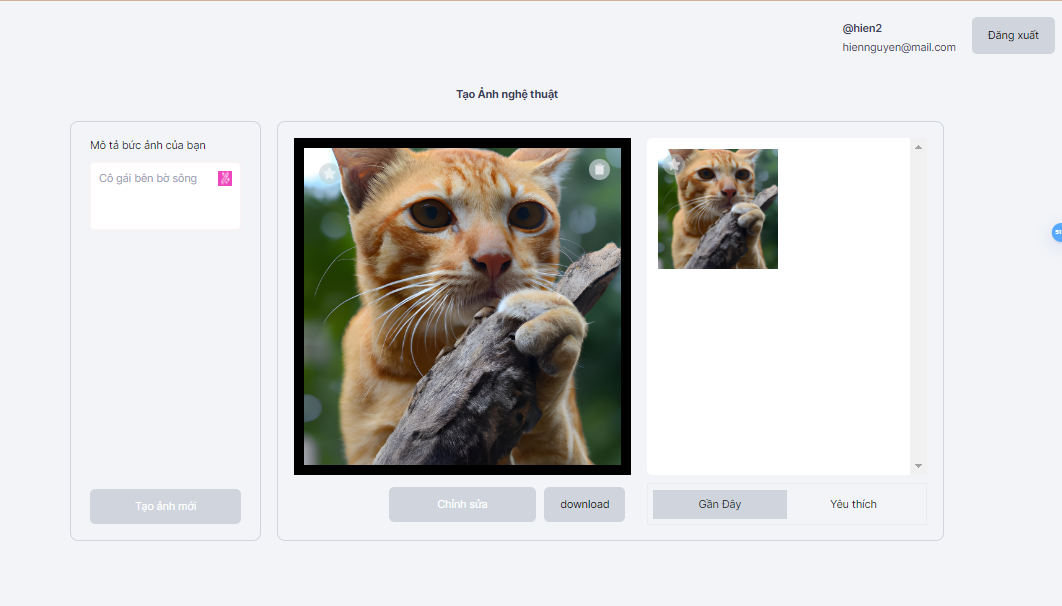
Bảng 2. 3 Bảng image

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ý nghĩa** |
| 1 | \_id | string | Khóa, tự động tăng |
| 2 | name | string | Tên ảnh |
| 3 | url | string | Data của ảnh |
| 4 | isFavorite | boolean | Biến kiểm tra là ảnh favorite |
| 5 | user\_id | string | Khóa ngoại, id user cung cấp |

## Giao diện và xử lý

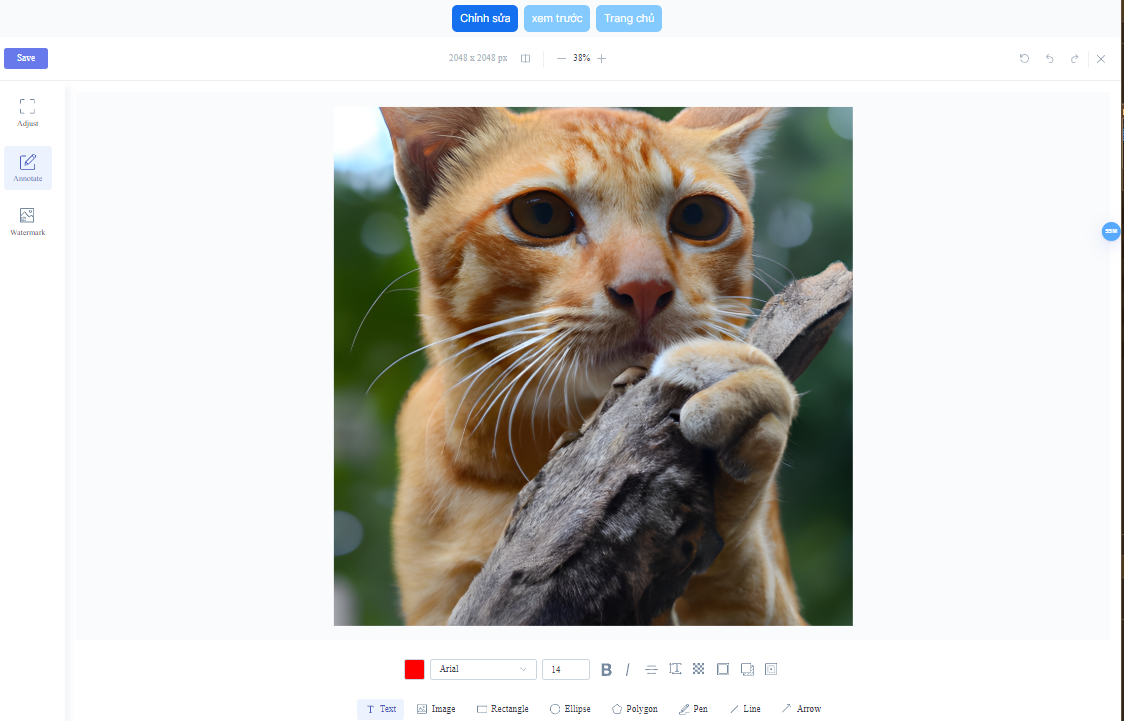
### 2.4.1 Thiết kế giao diện

* Giao diện trang chủ:

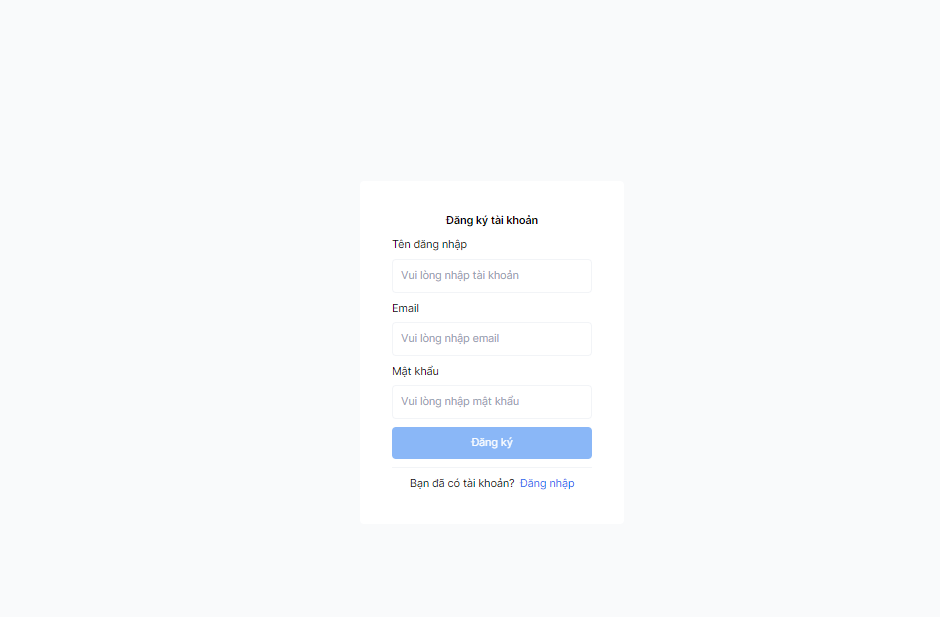


Hình 2. 17 Thiết kế giao diện trang chủ

* Giao diện trang chỉnh sửa ảnh:

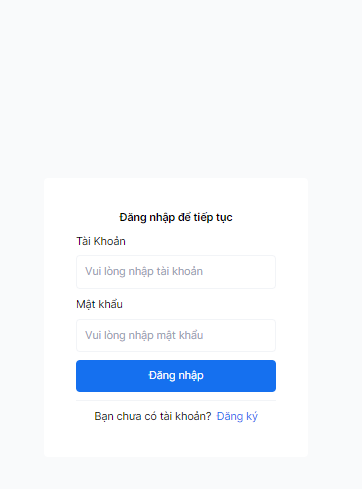


Hình 2. 18 Thiết kế giao diện trang chỉnh sửa ảnh

* Giao diện trang đăng ký:
* 

Hình 2. 19 Thiết kế giao diện trang đăng ký

* Giao diện trang đăng đăng nhập:



Hình 2. 19 Thiết kế giao diện trang đăng nhập

### 2.4.2 Thiết kế xử lý

### 2.4.2.1 Xử lý đăng ký tài khoản

* Kiểm tra thông tin người dùng nhập vào từ form đăng ký. Nếu thông tin trống thì báo lỗi.
* Sau khi người dùng click vào nút đăng ký, hệ thống gọi API đăng ký và gửi dữ liệu lên server với phương thức POST. Nếu thông tin đăng ký hợp lệ, server sẽ lưu thông tin người dùng vào CSDL và gửi thông báo thành công tới client, sau đó hệ thống sẽ điều hướng người dùng tới trang đăng nhập. Ngược lại, server sẽ trả về lỗi nếu thông tin không hợp lệ.

### 2.4.2.2 Xử lý đăng nhập

* Kiểm tra thông tin người dùng nhập từ ô tên người dùng và mật khẩu.
* Sau khi người dùng click vào nút đăng nhập, hệ thống sẽ gửi dữ liệu lên server qua API đăng nhập với phương thức POST. Server kiểm tra dữ liệu người dùng đã được lưu trong hệ thống chưa. Nếu có client sẽ điều hướng người dùng vào trang chủ. Thông tin của người dùng được trả về từ lệnh gọi API sẽ được lưu trong local Storage để duy trì trạng thái đăng nhập. Ngược lại, nếu thông tin người dùng không hợp lệ, thông báo lỗi cho người dùng.

### 2.4.2.3 Xử lý chức năng hiển thị thông tin

* Khi người dùng truy cập vào trang web, hệ thống sẽ gọi API đến server với phương thức GET để lấy dữ liệu của người dùng và hiển thị ra giao diện. Bao gồm các dữ liệu về hình ảnh đã tạo của người dùng
* Các thông tin về trang cá nhân của người dùng sẽ được gọi API đến server lấy dữ liệu hiển thị trên giao diện khi người dùng đăng nhập thành công.

### 2.4.2.4 Xử lý chức năng cập nhật ảnh

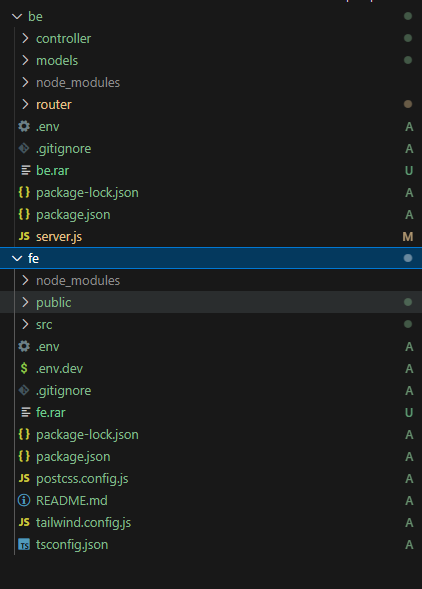
* Khi người dùng thực hiện một cập nhật, hệ thống sẽ gọi API và gửi dữ liệu cập nhật đến server với phương thức POST. Server thực hiện lưu dữ liệu vào CSDL và trả về thông báo thành công hoặc thất bại cho client. Sau đó, hệ thống sẽ hiển thị lại giao diện cho người dùng nếu cập nhật thành công, ngược lại thông báo ra lỗi.

# **CHƯƠNG 3: TÍCH HỢP MODULE TRÍ TUỆ NHÂN TẠO DALL-E ĐỂ TẠO ỨNG DỤNG WEB SỬ LÝ ẢNH KỸ THUẬT SỐ**

## Xây dựng hệ thống

### Cấu trúc mã lệnh

* Cấu trúc thư mục:



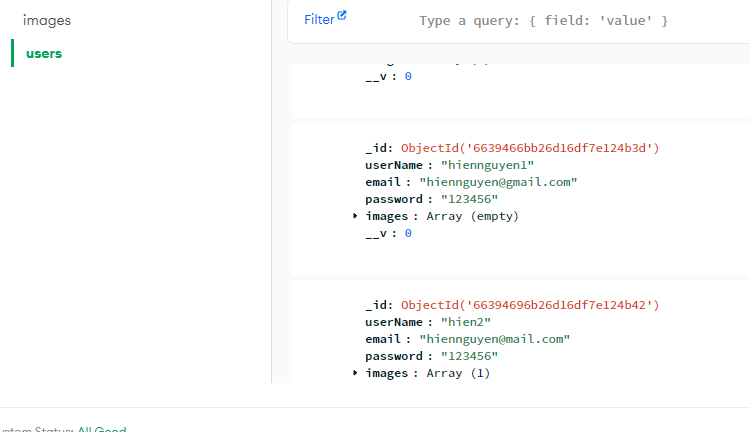
Hình 3. 1 Cấu trúc thư mục của project

* Thư mục be: xử lý các yêu cầu từ phía người dùng, thực hiện các tính toán, truy vấn cơ sở dữ liệu, tương tác với hệ thống tệp tin và trả về dữ liệu được yêu cầu cho phía front-end.
* Thư mục fe: xây dựng giao diện người dùng: gồm các UI và chức năng xử lý logic

### 3.1.2 Hệ quản trị cơ sở dữ liệu

* Sử dụng hệ quản trị cơ sở dữ liệu mongoDB.



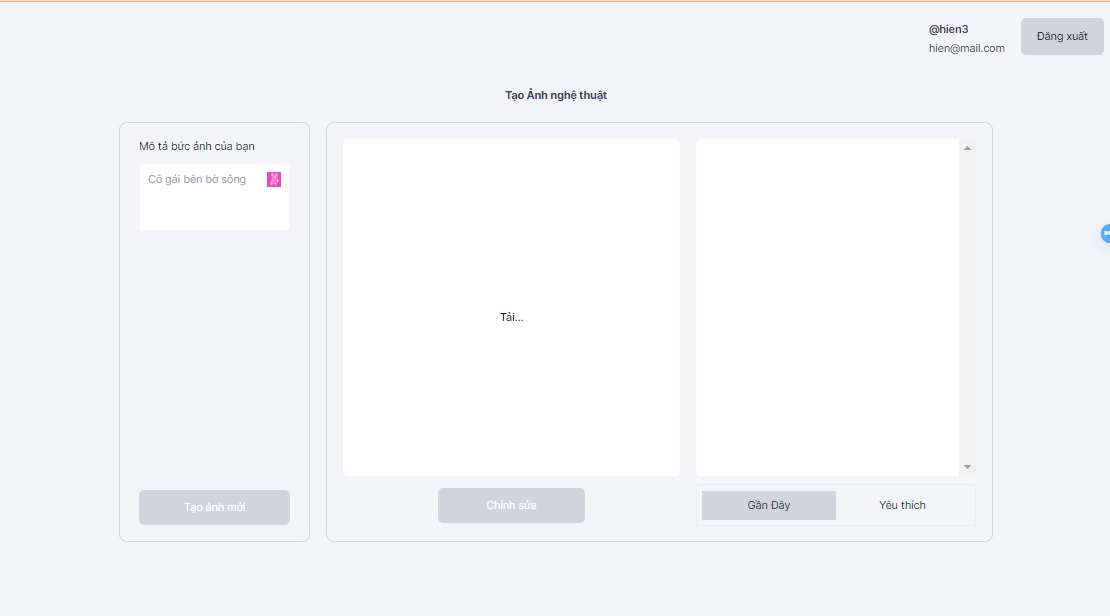


Hình 3. 2 Cơ sở dữ liệu Mongodb

## Giao diện phần mềm

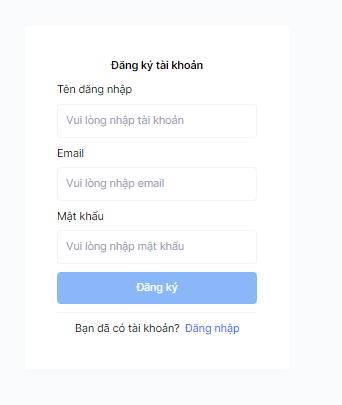
### 3.2.1 Giao diện trang chủ

* Trang chủ bao gồm phần tạo ảnh và danh sách hiển thị ảnh của người dùng, thông tin đăng nhập của người dùng



Hình 3. 3 Giao diện trang chủ

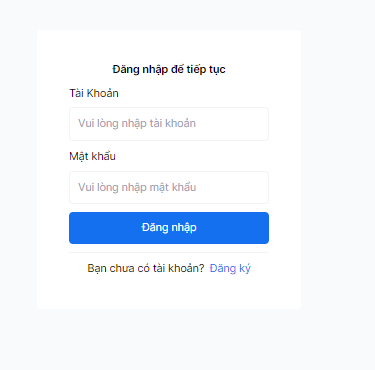
### 3.2.2 Đăng ký



Hình 3. 4 Giao diện trang đăng ký

* Người dùng nhập dữ liệu vào form để thực hiện việc đăng ký tài khoản.
* Nếu thông tin đăng ký hợp lệ, hệ thống lưu thông tin người dùng vào CSDL và chuyển hướng người dùng đến trang đăng nhập.

### 3.2.3 Đăng nhập

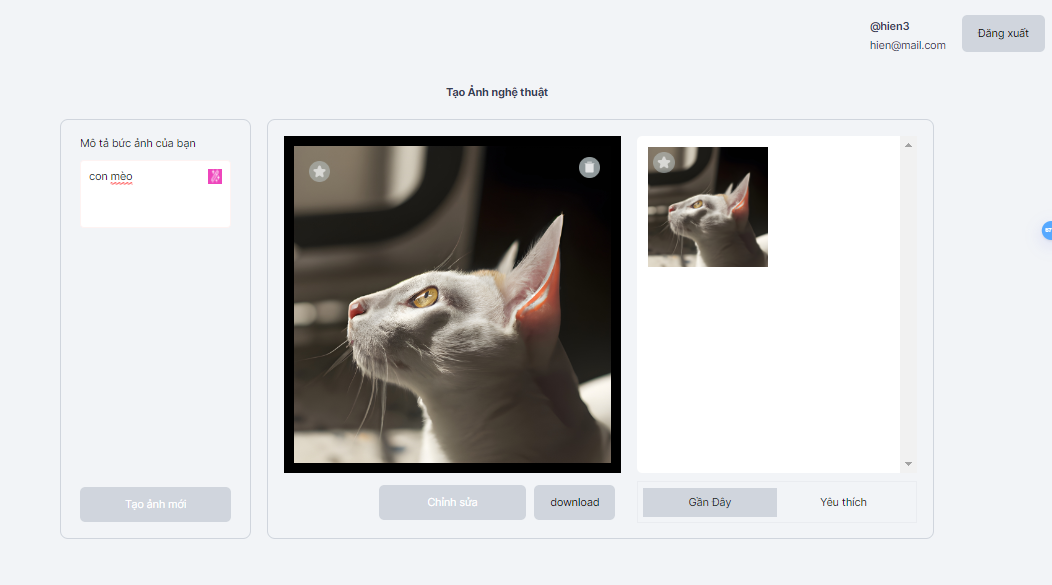


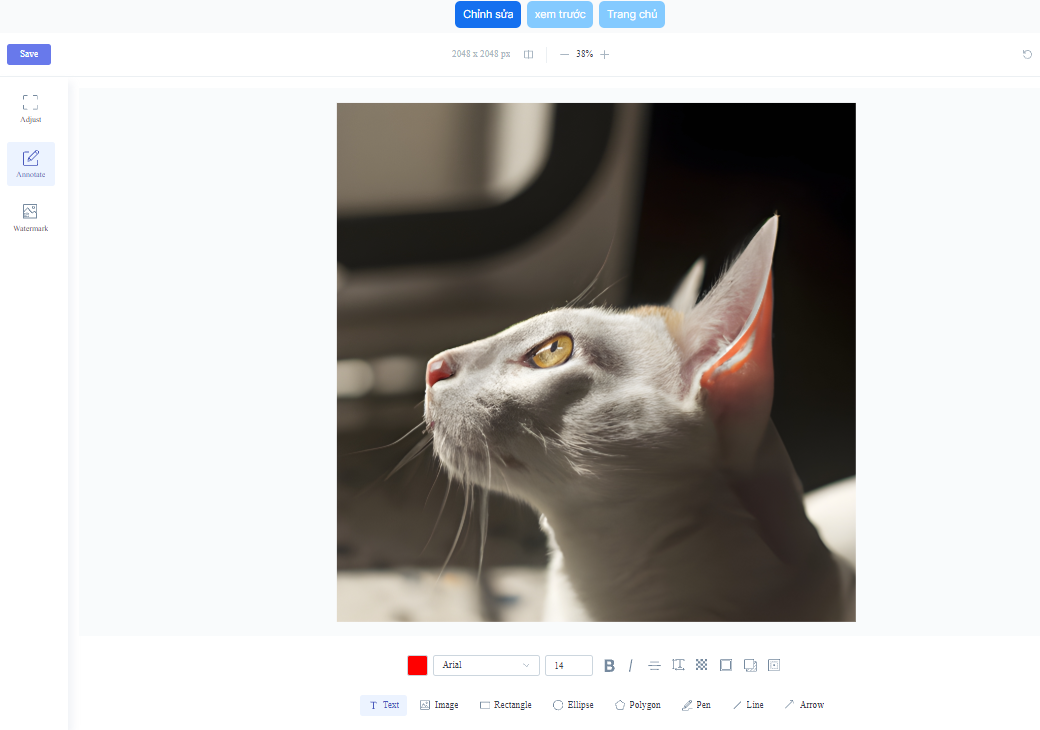
Hình 3. 6 Giao diện form đăng nhập

* Thông tin người dùng và token được lưu vào local Storage của trình duyệt để duy trì trạng thái đăng nhập.

### 3.2.4 Chỉnh sửa ảnh

* Sau khi đăng nhập thành công, người dùng tạo ảnh và chọn 1 bức ảnh. Click vào chỉnh sửa để đến trang chỉnh sửa ảnh





Hình 3. 8 Giao diện trang chỉnh sửa ảnh

* Người dùng chọn chỉnh sửa để cập nhật lại thay đổi bức ảnh. Sau khi người dùng nhấn save thay đổi, hệ thống sẽ gọi API cập nhật thông tin người dùng với phương thức POST và gửi dữ liệu lên server. Server sẽ lưu lại dữ liệu được cập nhật vào CSDL và trả về dữ liệu mới đã được cập nhật để hệ thống hiển thị trên giao diện.

## Kết quả và thảo luận

* **Một số kết quả đạt được**: Trong suốt quá trình thực hiện đề tài em đã tiếp thu được nhiều kiến thức mới từ lý thuyết đến kỹ năng xây dựng ứng dụng. Ngoài ra, đây cũng là cơ hội để em vận dụng, tổng hợp những kiến thức đã được truyền đạt trong suốt quá trình học tập vừa qua. Cụ thể:

+ Tìm hiểu ReactJS, express Framework, Mongodb và ứng dụng vào xây dựng website.

+ Hiểu được quy trình làm việc của một trang web tảo ảnh nghệ thuật

+ Hiểu thêm về cách xây dựng API bằng expressjs và Mongodb.

+ Hiểu thêm về cách xử lý giao diện trong ReactJS.

* **Kết quả**: Trong quá trình nghiên cứu em đã đạt được một số kết quả sau:

+ Xây dựng API và cơ sở dữ liệu ổn định.

+ Xây dựng giao diện thân thiện.

+ Hoàn thành các chức năng đã đề ra.

* **Hạn chế**: Trong suốt quá trình thực hiện thì đề tài cơ bản đã hoàn thành nhưng còn một số hạn chế:

+ Chưa xây dựng được chức năng thông báo đến người dùng.

+ Khả năng bảo mật còn thiếu sót, hiệu suất còn chậm.

+ Một số chức năng chưa hoàn thiện và cần được sửa lỗi.

+ Chưa xây dựng được các chức năng cho người dùng: đổi mật khẩu, thay đổi thông tin…

+ chưa deploy website lên host

1. **Kết luận và kiến nghị**

* Sau khi tìm hiểu và nghiên cứu em đã: “**TÍCH HỢP MODULE TRÍ TUỆ NHÂN TẠO DALL-E ĐỂ TẠO ỨNG DỤNG WEB SỬ LÝ ẢNH KỸ THUẬT SỐ**” với các chức năng ban đầu đã đề ra.
* Kết quả của chuyên đề là một website hỗ trợ người dùng có thể tạo ra các bức ảnh dựa theo mô tả của chính họ. Ngoài ra họ có thể đa dạng chỉnh sửa các bức ảnh bằng các tool và lưu.
* Tuy nhiên vẫn còn một số điểm hạn chế, em sẽ cố gắng hoàn thiện trong thời gian tới. Rất mong nhận được sự quan tâm và góp ý từ quý thầy cô và các bạn.
* **Hướng phát triển**:

+ Tiếp tục xây dựng website hoàn thiện hơn.

+ Khắc phục những hạn chế đã nêu.

+ Xây dựng thêm tính năng thông báo cho người dùng.

+ Xây dựng thêm chức năng đăng nhập nhanh bằng tài khoản Facabook, Google… và chức năng quên mật khẩu.

+ Xây dựng them các chức năng cho người dùng

**TÀI LIỆU THAM KHẢO**

[Tiếng Anh]

[1]. ReactJS document: <https://react.dev/>

[2]. Nodejs document: https://nodejs.org/docs/latest/api/

[3]. React router document: <https://reactrouter.com/en/main>

[4]. Express document: <https://devdocs.io/express/>

[5]. Dall-e api document: https://platform.openai.com/docs/guides/images/image-generation

[6]. MongoDb document: https://www.mongodb.com/docs

[7]. HTTP response status codes <https://developer.mozilla.org/en-US/docs/Web/HTTP/Status>

[8]. CORS — Cross-Origin Resource Sharing https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS

[Tiếng Việt]

**PHỤ LỤC**

**Website:**

**API Docs:**

Bảng 4. 1 API Docs

|  |  |  |  |
| --- | --- | --- | --- |
| User | Get all user | GET | /api/user/get |
| Get user by id | GET | /api/user/:id |
| Create new user | POST | /api/user/create |
| Signin | POST | /api/user/signin |
| Signout | POST | /api/user/signout |
| Delete image of user | POST | /api/user/delete-image |
| Image | Create image for user | POST | /api/image/generate |
| Get all image by user | POST | /api/image/getImageByUser |
| Toggle is favorite of image | POST | /api/image/toggle-favorite |
| Update url image | POST | /api/image/update-image-data |
| Update image for user | POST | /api/image/update |

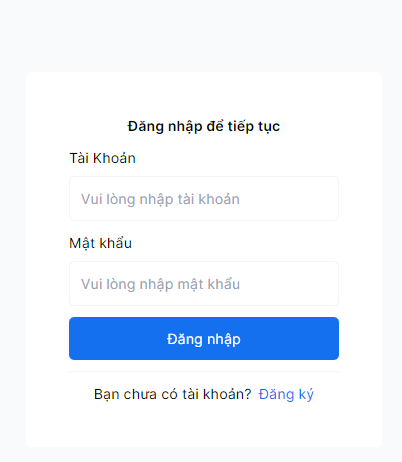
1. **Hướng dẫn sử dụng:**

* **Yêu cầu hệ thống:**

+ Trình duyệt web(Google Chrome, Firefox, Safari…) phiên bản mới nhất.

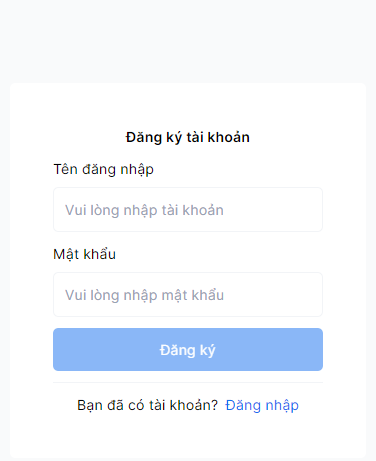
+ Kết nối Internet ổn định.

* **Các bước để truy cập vào website:**
  + Bước 1: Truy cập vào trang web với địa chỉ: <http://localhost:4300/>
  + Bước 2:
* Đăng nhập vào website với tên đăng nhập và mật khẩu có sẵn:
* Tên đăng nhập: phuc
* Mật khẩu: 123456



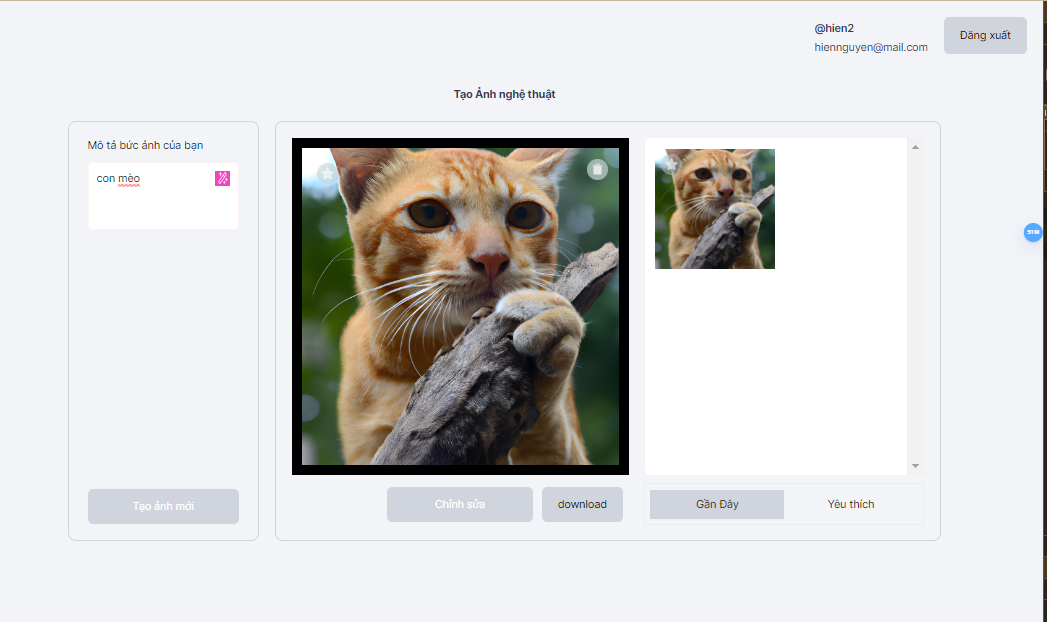
Hình 4. 1 Biểu mẫu đăng nhập

Hoặc lựa chọn đăng ký tài khoản, điền các thông tin vào biểu mẫu đăng ký, lựa chọn vai trò là khách hàng hoặc cộng tác viên và nhấn nút đăng ký.

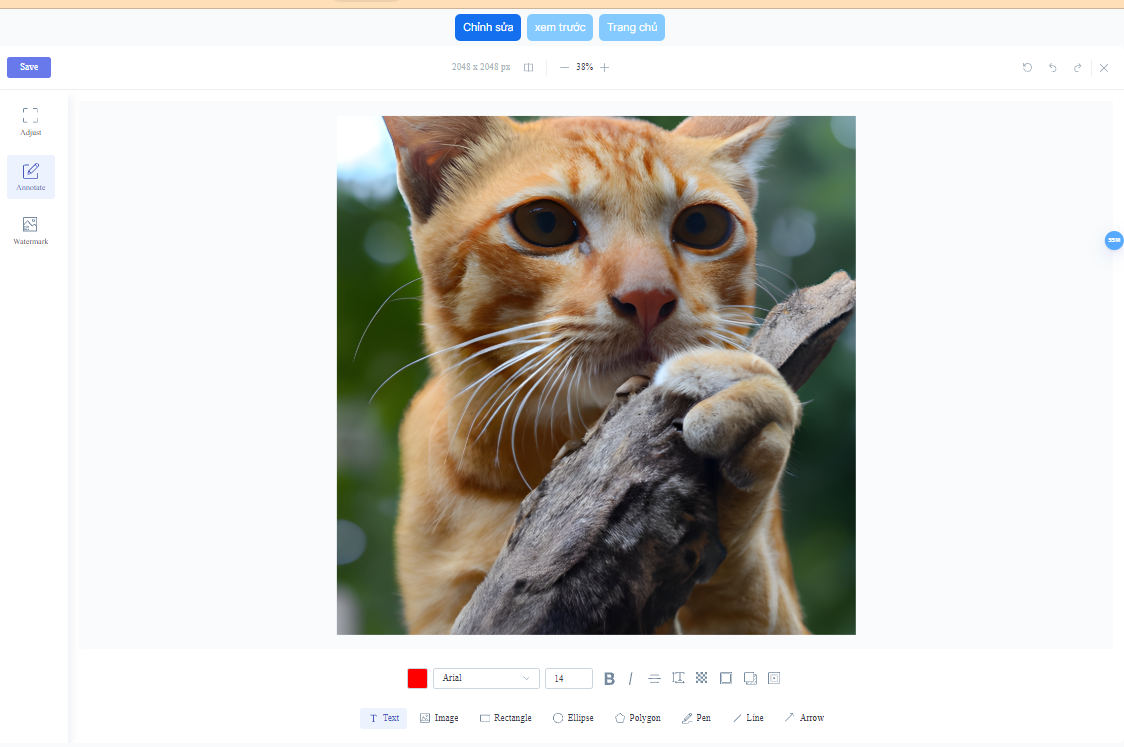


Hình 4. 2 Biểu mẫu đăng ký tài khoản

* + Bước 3: Sau khi đăng ký thành công, hệ thống sẽ chuyển người dùng đến trang đăng nhập, đăng nhập vào website bằng tên đăng nhập và mật khẩu mới đăng ký.
* **Giao diện:**



Hình 4. 3 Giao diện trang chủ



Hình 4. 4 Giao diện trang chỉnh sửa ảnh

+ Trang chủ hiển thị giao diện chính của website: bao gồm phần tạo ảnh và danh sách ảnh.

+ Trang chỉnh sửa ảnh hiển thị giao diện chỉnh sửa ảnh tích hợp các module chỉnh sửa của react-filerobot-image-editor: cắt ảnh theo khung hình, chèn ảnh, chèn chữ

+ Trang đăng ký: hiển thị form đăng ký để người dùng đăng ký tài khoản

+ Trang đăng nhập: hiển thị form đăng nhập để người dùng đăng nhập tài khoản đã đăng ký và truy cập vào trang chủ.

* **Chức năng và tính năng:**
  + Tạo ảnh
  + Đăng xuất, đăng ký, đăng nhập
  + Danh sách tất cả các ảnh đã tạo của user
  + Danh sách yêu thích
  + Lựa chọn hình ảnh chỉnh sửa
  + Dowload bức ảnh được chọn
  + Chỉnh sửa ảnh tích hợp các module của react-filerobot-image-editor: cắt ảnh theo khung, chèn hình, chèn chữ

**Ý KIẾN CỦA NGƯỜI HƯỚNG DẪN**

**Nhận xét:**

|  |  |
| --- | --- |
|  | Quy Nhơn, ngày ... tháng … năm 2024  **NGƯỜI HƯỚNG DẪN**  (*Ký và ghi rõ họ tên*) |

Họ và tên sinh viên: **Trịnh Chấn Phúc**.

Cán bộ hướng dẫn:

Tên đề tài: “Tích hợp module trí tuệ nhân tạo Dall-e để tạo ứng dụng Web sử lý ảnh kỹ thuật số ”.

**Điểm:** ……………..(……………………………………….)

|  |  |
| --- | --- |
|  | Quy Nhơn, ngày … tháng … năm 2024  **SINH VIÊN**  *(Ký và ghi rõ họ tên)* |
| **XÁC NHẬN** | |
| **CÁN BỘ HƯỚNG DẪN**  *(Ký và ghi rõ họ tên)* | **CHỦ TỊCH HỘI ĐỒNG**  *(Ký và ghi rõ họ tên)* |