|  |
| --- |
|  |
| **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC  NGÀNH CÔNG NGHỆ THÔNG TIN |
| **XÂY DỰNG WEBSITE QUẢN LÝ QUY TRÌNHTHỰC HIỆN DỰ ÁN CHO CÔNG TY ANNGO BẰNG PYTHON DJANGO** |
|  |
| **CBHD: *ThS. Ngô Thị Thanh Hòa*** |
| **Sinh viên:  *Nguyễn Ngô An*** |
| **Mã số sinh viên: *2020606739*** |
|  |
| Hà Nội - Năm 2024 |

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời tri ân sâu sắc đến toàn thể quý thầy cô trong khoa công nghệ thông tin, những người đã không ngừng hỗ trợ và khuyến khích em trong suốt quãng thời gian học tập và nghiên cứu tại trường. Đặc biệt, em muốn bày tỏ lòng biết ơn đặc biệt đến Ths. Ngô Thị Thanh Hòa, người đã dành thời gian và tâm huyết để chỉ dẫn, hướng dẫn và động viên em trong quá trình thực hiện đề tài này.

Em cũng muốn bày tỏ lòng biết ơn sâu sắc đến tất cả các thầy cô tại Trường Đại học Công nghiệp Hà Nội, những người đã chia sẻ những kiến thức, kinh nghiệm và phương pháp giúp em hoàn thiện đề tài này. Tuy nhiên, em nhận thức rằng với lượng lớn thông tin và yêu cầu phức tạp của một hệ thống web, vẫn còn nhiều hạn chế và thách thức mà em chưa thể vượt qua.

Trong quá trình nghiên cứu và thực hiện đề tài, em đã gặp phải nhiều khó khăn và thiếu sót. Em hi vọng sẽ tiếp tục nhận được sự đóng góp và hỗ trợ quý báu từ phía quý thầy cô và các bạn để hoàn thiện hơn nữa.

Em xin chân thành cảm ơn!

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc168050278)

[MỤC LỤC ii](#_Toc168050279)

[DANH MỤC CÁC HÌNH VẼ, CÁC HÌNH ẢNH iii](#_Toc168050280)

[DANH MỤC BẢNG BIỂU v](#_Toc168050281)

[MỞ ĐẦU vi](#_Toc168050282)

[CHƯƠNG 1. CÔNG NGHỆ SỬ DỤNG TRONG WEBSITE QUẢN LÝ QUY TRÌNH THỰC HIỆN DỰ ÁN 1](#_Toc168050283)

[1.1. Django 1](#_Toc168050284)

[1.2. Jinja 6](#_Toc168050285)

[1.3. Ngôn ngữ JavaScript 7](#_Toc168050286)

[1.4. Web socket 10](#_Toc168050287)

[1.5. WebRTC 13](#_Toc168050288)

[CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 18](#_Toc168050289)

[2.1. Yêu cầu phần mềm 18](#_Toc168050290)

[2.2. Biểu đồ usecase 19](#_Toc168050291)

[2.3. Đặc tả usecase 21](#_Toc168050292)

[2.4. Thiết kế cơ sở dữ liệu 40](#_Toc168050293)

[CHƯƠNG 3. KẾT QUẢ ĐỒ ÁN 49](#_Toc168050294)

[3.1. Kết quả đạt được 49](#_Toc168050295)

[3.2. Kiểm thử 72](#_Toc168050296)

[KẾT LUẬN 78](#_Toc168050297)

[TÀI LIỆU THAM KHẢO 79](#_Toc168050298)

DANH MỤC CÁC HÌNH VẼ, CÁC HÌNH ẢNH

[Hình 1.1: Mô hình MVT 1](#_Toc168009855)

[Hình 1.2: Model ORM 2](#_Toc168009856)

[Hình 1.3: Giao diện Admin tích hợp 2](#_Toc168009857)

[Hình 1.4: Biến cần thêm 11](#_Toc168009858)

[Hình 1.5: Ví dụ cấu hình 11](#_Toc168009859)

[Hình 1.6: Ví dụ AsyncWebsocketConsumer 12](#_Toc168009860)

[Hình 1.7: Ví dụ file routing 12](#_Toc168009861)

[Hình 1.8: Ví dụ gửi dữ liệu 13](#_Toc168009862)

[Hình 1.9: Ví dụ nhận dữ liệu 13](#_Toc168009863)

[Hình 1.10: Quy trình thiết lập kết nối 17](#_Toc168009864)

[Hình 2.1: Biểu đồ tổng quát 19](#_Toc168009865)

[Hình 2.2: Biểu đồ phân rã quản trị viên 20](#_Toc168009866)

[Hình 2.3: Phân quyền người dùng 20](#_Toc168009867)

[Hình 2.4: Biểu đồ phân rã lập lịch 21](#_Toc168009868)

[Hình 2.5: Biểu đồ ERD 40](#_Toc168009869)

[Hình 3.1: Sơ đồ màn hình 49](#_Toc168009870)

[Hình 3.2: Giao diện đăng nhập 49](#_Toc168009871)

[Hình 3.3: Giao diện đăng ký 50](#_Toc168009872)

[Hình 3.4: Giao diện trang chủ phía nhân viên dự án 50](#_Toc168009873)

[Hình 3.5: Giao diện trang chủ phía khách hàng 51](#_Toc168009874)

[Hình 3.6: Giao diện đổi mật khẩu 51](#_Toc168009875)

[Hình 3.7: Giao diện hỗ trợ phía người dùng 52](#_Toc168009876)

[Hình 3.8: Giao diện chọn người cần hỗ trợ 52](#_Toc168009877)

[Hình 3.9: Giao diện hỗ trợ bên quản trị viên 53](#_Toc168009878)

[Hình 3.10: Giao diện chỉnh sửa thông tin tài khoản 53](#_Toc168009879)

[Hình 3.11: Giao diện tạo dự án 54](#_Toc168009880)

[Hình 3.12: Giao diện trang chủ dự án 54](#_Toc168009881)

[Hình 3.13: Giao diện chỉnh sửa thông tin dự án 55](#_Toc168009882)

[Hình 3.14: Giao diện lập lịch 56](#_Toc168009883)

[Hình 3.15: Giao diện thêm giai đoạn 56](#_Toc168009884)

[Hình 3.16: Giao diện sửa giai đoạn 57](#_Toc168009885)

[Hình 3.17: Giao diện thêm nhiệm vụ 58](#_Toc168009886)

[Hình 3.18: Giao diện sửa nhiệm vụ 59](#_Toc168009887)

[Hình 3.19: Giao diện lập báo cáo tiến độ 59](#_Toc168009888)

[Hình 3.20: Giao diện duyệt tiến độ 60](#_Toc168009889)

[Hình 3.21: Giao diện quản lý nhân viên 61](#_Toc168009890)

[Hình 3.22: Giao diện thêm nhân viên 61](#_Toc168009891)

[Hình 3.23: Giao diện tạo nhóm 62](#_Toc168009892)

[Hình 3.24: Giao diện quản lý nhóm 62](#_Toc168009893)

[Hình 3.25: Giao diện trang chủ nhóm 63](#_Toc168009894)

[Hình 3.26: Giao diện lập báo cáo dự án 64](#_Toc168009895)

[Hình 3.27: Giao diện theo dõi tiến độ 64](#_Toc168009896)

[Hình 3.28: Giao diện chia sẻ file 65](#_Toc168009897)

[Hình 3.29: Giao diện họp video 66](#_Toc168009898)

[Hình 3.30: Giao diện kết thúc dự án 66](#_Toc168009899)

[Hình 3.31: Giao diện tổng kết dự án 67](#_Toc168009900)

[Hình 3.32: Mục tổng quan dự án 68](#_Toc168009901)

[Hình 3.33: Mục thành viên dự án 69](#_Toc168009902)

[Hình 3.34: Mục các bản báo cáo 69](#_Toc168009903)

[Hình 3.35: Mục chia sẻ file 69](#_Toc168009904)

[Hình 3.36: Giao diện bảo trì người dùng 70](#_Toc168009905)

[Hình 3.37: Giao diện bảo trì người dùng 71](#_Toc168009906)

[Hình 3.38: Kết quả test gọi video 77](#_Toc168009907)

DANH MỤC BẢNG BIỂU

[Bảng 2.1: Bảng người dùng 40](#_Toc168009956)

[Bảng 2.2: Bảng nhân viên 41](#_Toc168009957)

[Bảng 2.3: Bảng dự án 42](#_Toc168009958)

[Bảng 2.4: Bảng giai đoạn 42](#_Toc168009959)

[Bảng 2.5: Bảng nhiệm vụ 43](#_Toc168009960)

[Bảng 2.6: Bảng phân công 43](#_Toc168009961)

[Bảng 2.7: Bảng báo cáo tiến độ 44](#_Toc168009962)

[Bảng 2.8: Bảng báo cáo kiểm tra 44](#_Toc168009963)

[Bảng 2.9: Bảng báo cáo dự án 45](#_Toc168009964)

[Bảng 2.10: Bảng thành viên nhóm 45](#_Toc168009965)

[Bảng 2.11: Bảng nhóm 46](#_Toc168009966)

[Bảng 2.12: Bảng tin nhắn 46](#_Toc168009967)

[Bảng 2.13: Bảng chia sẻ file 46](#_Toc168009968)

[Bảng 2.14: Bảng họp video 47](#_Toc168009969)

[Bảng 2.15: Bảng video kết nối 47](#_Toc168009970)

[Bảng 2.16: Bảng kênh hỗ trợ 47](#_Toc168009971)

[Bảng 2.17: Bảng tin nhắn hỗ trợ 48](#_Toc168009972)

[Bảng 3.1: Kịch bản test đăng nhập 73](#_Toc168009973)

[Bảng 3.2: Các tài khoản sử dụng để test 73](#_Toc168009974)

[Bảng 3.3: Kết quả test 74](#_Toc168009975)

[Bảng 3.4: Kịch bản test gửi nội dung trò chuyện 75](#_Toc168009976)

[Bảng 3.5: Kết quả test gửi nội dung trò chuyện 76](#_Toc168009977)

[Bảng 3.6: Kịch bản test gọi video 76](#_Toc168009978)

[Bảng 3.7: Các tài khoản sử dụng để test 77](#_Toc168009979)

MỞ ĐẦU

* 1. Lý do chọn đề tài

Trong bối cảnh của cuộc cách công nghiệp 4.0 hiện nay, việc áp dụng máy móc điện tử hiện đại vào sản xuất và phát triển sản phẩm đã không còn là một điều gì quá xa lạ. Việc áp dụng tiến bộ khoa học hiện đại đã hỗ trợ con người rất nhiều trong công việc giúp tăng hiệu quả lao động cũng như làm giảm công việc tay chân. Trong đó em thấy các công việc liên quan đến quản lý yêu cầu phải có sự trao đổi, giao tiếp giữa các bên được hưởng lợi rất nhiều thông qua công nghệ thông tin.

Ngoài ra trong quá trình học tập tại trường em cũng có cơ hội được trải qua việc làm bài tập lớn, khi đó việc phân chia công việc để thực hiện diễn ra chủ yếu thông qua tin nhắn trên các nền tảng mạng xã hội như Zalo, Facebook. Việc này thường làm phát sinh một số yếu tố bất lợi như thành viên chưa nắm hết công việc phải làm, quên một số công việc cần làm, các thành viên nhóm thường không nắm được tiến độ của nhau khi có người chậm lại so với kế hoạch thì cũng không biết để giúp đỡ làm ảnh hưởng đến toàn dự án.

Từ đó, em nghĩ ra ý tưởng về một phần mềm giúp hỗ trợ quản lý quy trình thực hiện dự án, giúp các nhà quản lý nắm bắt được quá trình phát triển dự án. Phát triển phần mềm này trên nền tảng website sẽ giúp người dùng truy cập dễ dàng, thuận tiện và linh hoạt.

Lý do chọn Python Django để xây dựng website là vì Django là một framework rất nổi tiếng trong việc phát triển website bằng Python. Django cung cấp sẵn giao diện cho Admin, kiểu dữ liệu trong cơ sở dữ liệu đa dạng, đặc biệt dễ dàng trong việc thiết kế cơ sở dữ liệu.

* 1. Mục tiêu đề tài

### Đối tượng của ứng dụng web

Ứng dụng web được xây dựng phục vụ ba đối tượng chính là quản trị viên, nhân viên dự án và khách hàng:

* *Quản trị viên*: là người hỗ trợ nhân viên và khách hàng khi có yêu cầu từ họ.
* *Nhân viên dự án*: là những người trực tiếp tham gia xây dựng dự án. VD: quản lý dự án, đội ngũ phát triển, …
* *Khách hàng*: là những người tuy không trực tiếp tham gia dự án nhưng cũng cần nắm được quá trình phát triển. VD: chủ đầu tư, quan sát viên,..

### Mục tiêu

* Phát triển website giúp quản lý quá trình thực hiện dự án: Phát triển website có đầy đủ chức năng giúp người quản lý dự án quản lý quá trình thực hiện dự án: tạo lịch thực hiện, phân chia nhiệm vụ, theo dõi tiến độ hoàn thành, phân quyền những người tham gia dự án.
* Giao tiếp và trao đổi trong dự án:
* Phát triển kênh giao tiếp riêng giữa những người thực hiện dự án với nhau: nhắn tin, gọi video.
* Cho phép những người thực hiện dự án chia sẻ với nhau tài nguyên dữ liệu phần mềm.
* Xác thực và quản lý người dùng: Phát triển chức năng đăng nhập. Phân quyền người dùng theo vai trò khi đăng nhập.
  1. Bố cục của báo cáo

***Báo cáo gồm 3 chương:***

* *Chương 1:* Công nghệ sử dụng trong website quản lý quy trình thực hiện dự án: Tổng quan về Django và các công nghệ được sử dụng trong đề tài.
* *Chương 2:* Phân tích thiết kế hệ thống: Từ những kiến thức của mình và các yêu cầu đặt ra của công việc quản lý dự án. Em tiến hành vẽ các biểu đồ chức năng, biểu đồ quan hệ, thiết kế cơ sở dữ liệu.
* *Chương 3:* Kết quả đồ án: Kết quả thu được sau khi sau thời gian thực hiện đồ án và kiểm thử một số chức năng của website.

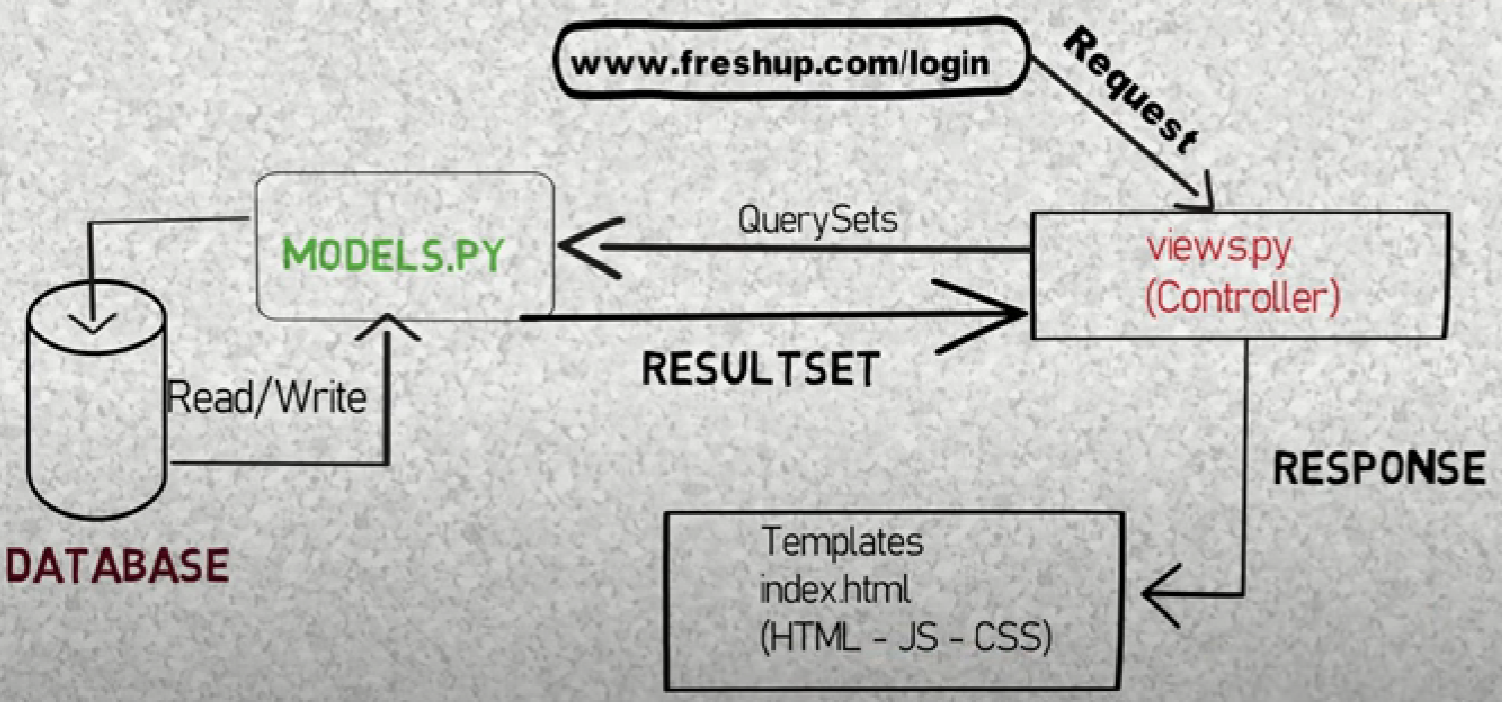
# CÔNG NGHỆ SỬ DỤNG TRONG WEBSITE QUẢN LÝ QUY TRÌNH THỰC HIỆN DỰ ÁN

## Django

*Django* là một framework web mạnh mẽ của ngôn ngữ lập trình Python, được thiết kế để giúp nhà phát triển xây dựng ứng dụng web nhanh chóng và dễ bảo trì.

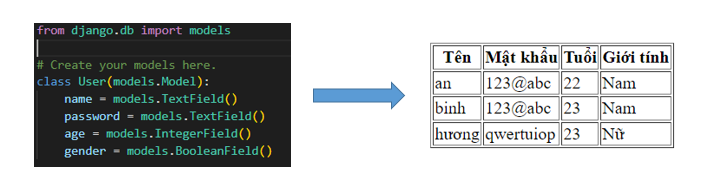
1. Một số đặc điểm của Django

* *Mô hình MVT (Model-View-Template):* Django sử dụng mô hình MVT, nơi Model đại diện cho dữ liệu và cơ sở dữ liệu, View xử lý logic xử lý và hiển thị, còn Template quản lý giao diện người dùng.



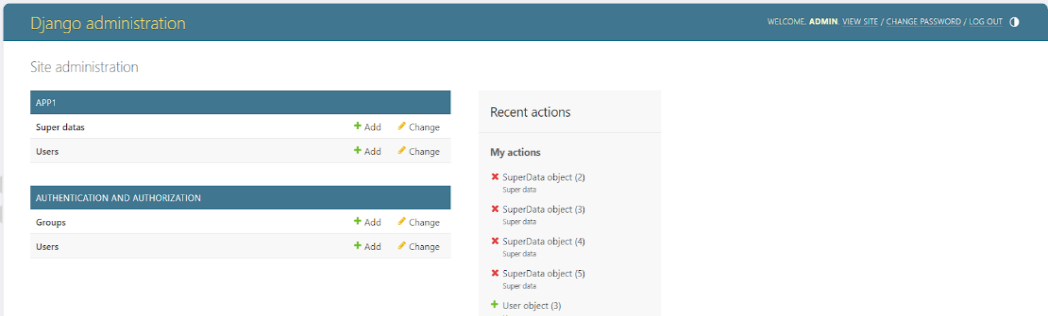
1. Mô hình MVT

* *ORM (Object-Relational Mapping):* Django cung cấp một ORM mạnh mẽ, giúp tương tác với cơ sở dữ liệu mà không cần viết SQL trực tiếp. Điều này giúp làm giảm độ phức tạp và tăng tính di động của ứng dụng.



1. Model ORM

* *Giao diện Admin tích hợp*: Django cung cấp một giao diện quản trị tích hợp sẵn, giúp quản lý dữ liệu của ứng dụng mà không cần phải xây dựng giao diện quản trị riêng.



1. Giao diện Admin tích hợp

* *Định tuyến URL:* Hệ thống định tuyến URL của Django cho phép bạn dễ dàng xác định cách các URL được ánh xạ vào các hàm xử lý (views).
* *Hỗ trợ biểu mẫu (form):* Django cung cấp các lớp biểu mẫu, giúp xử lý dữ liệu đầu vào từ người dùng một cách hiệu quả và an toàn.

1. Các bước cơ bản trong Django

* *Tạo dự án:* Dùng lệnh “django-admin” trong terminal để tạo 1 dự án mới, nếu chưa cài đặt django thì dùng lệnh “pip install django” để cài đặt.
* *Tạo ứng dụng:* Sử dụng “python manage.py startapp tên\_ứng\_dụng” để tạo một ứng dụng mới trong dự án. Nó sẽ tự tạo 1 folder có tên là tên ứng dụng bạn đặt, nó sẽ có sẵn các file:
  + admin.py: sử dụng để định nghĩa cấu hình liên quan đến giao diện admin của ứng dụng như chỉnh sửa database, tùy chỉnh hiển thị, bộ lọc, tìm kiếm, …
  + models.py: sử dụng để định nghĩa cấu trúc dữ liệu của ứng dụng.
  + views.py: sử dụng để viết các hàm xử lý logic và trả lại kết quả.
* *Xây dựng mô hình (Models*): Xác định các lớp mô hình để đại diện cho dữ liệu và cấu trúc cơ sở dữ liệu.
* *Tạo Views:* Viết hàm xử lý trong views để xử lý logic và trả về kết quả cho người dùng.
* *Xây dựng Templates*: Tạo folder đặt tên là Templates (lưu ý: phải đặt tên đúng là Templates) bên trong folder ứng dụng. Folder này sẽ chứa các file HTML, CSS được dùng trong ứng dụng.
* *Xác định URL:* Định nghĩa các mẫu URL để ánh xạ đến các views tương ứng.
* *Chạy Migrations:* Sử dụng “python manage.py makemigrations” và “python manage.py migrations” để áp dụng các thay đổi vào cơ sở dữ liệu. Lặp lại 2 lệnh này bất cứ khi nào có thay đổi về  cấu trúc cơ sở dữ liệu.
* *Chạy Server:* Sử dụng “python manage.py runserver” để chạy máy chủ và kiểm tra ứng dụng

1. Cơ sở dữ liệu của Django

Cơ sở dữ liệu của Django được xây dựng dựa trên ORM (Object-Relational Mapping), giúp tương tác với cơ sở dữ liệu bằng cách sử dụng đối tượng Python thay vì truy vấn SQL trực tiếp.

1. Một số khái niệm về cơ sở dữ liệu của Django

* *Models:* Models là các lớp Python đại diện cho các bảng trong cơ sở dữ liệu. Mỗi trường của mô hình tương ứng với một cột trong bảng.
* *Migration:* Migration là quá trình chuyển đổi các thay đổi trong models thành các bảng cơ sở dữ liệu thực tế.
* *Quan hệ:* Django hỗ trợ nhiều loại quan hệ giữa các models, chẳng hạn như ForeignKey (một-đến-nhiều), OneToOneField (một-một) và ManyToManyField (nhiều-đến-nhiều).
* *Django ORM:* Django ORM giúp thực hiện các thao tác cơ sở dữ liệu mà không cần phải viết SQL trực tiếp. Các thao tác tạo, sửa đổi, xóa và truy vấn dữ liệu được thực hiện một cách đơn giản dễ dàng.

1. Các giá trị dùng để định nghĩa các trường trong cơ sở dữ liệu

* *primary\_key:* dùng để biết có gán khóa chính cho trường này không.
* *null:* cho phép trường có nhận được giá trị null không.
* *blank:* cho phép trường được bỏ trống khi nhập form.
* *default:* gán giá trị mặc định cho trường.
* *unique:* đảm bảo giá trị của trường là duy nhất trong toàn bộ bảng.
* *max\_lenghth:* gán độ dài tối đa của chuỗi ký tự cho phép lưu trữ, chỉ sử dụng cho CharField.
* *choices:* gán trường chỉ nhận các giá trị đã được định nghĩa sẵn.
* *upload\_to:* gán vị trí lưu trữ file, chỉ dùng cho các trường như file, hình ảnh.
* *auto\_now* và *auto\_now\_add*: dùng để tự động cập nhật thời gian cho trường, chỉ dùng cho DateTimeField.
  + *auto\_now*: sẽ cập nhật thời gian khi bản ghi được tạo hoặc mỗi khi được cập nhật.
  + *auto\_now\_add*: chỉ cập nhật thời gian khi bản ghi được tạo.
* *on\_delete:* thường được dùng cho trường ForeignKey, cho biết hành động nào sẽ được thực hiện khi bản ghi liên kết với trường (đối tượng mẹ) bị xóa. Các giá trị mà *on\_delete* có thể nhận:
  + *models.CASCADE*: khi đối tượng mẹ bị xóa, tất cả các đối tượng con sẽ bị xóa.
  + *models.PROTECT*: ngăn việc xóa đối tượng  mẹ nếu còn tồn tại bất kỳ đối tượng con nào.
  + *models.SET\_NULL*: khi đối tượng mẹ bị xóa, tất cả các đối tượng con sẽ có giá trị là null, yêu cầu phải có null=True.
  + *models.SET\_DEFAULT*: khi đối tượng mẹ bị xóa, tất cả các đối tượng con sẽ có giá trị là giá trị default, yêu cầu gán giá trị default.
  + *models.SET*(set\_value): khi đối tượng mẹ bị xóa, tất cả các đối tượng con sẽ có giá trị là giá trị set\_value.
  + *models.DO\_NOTHING:* khi đối tượng mẹ bị xóa thì không làm gì cả. Lưu ý: chỉ sử dụng khi đảm bảo sẽ không phát sinh bất kỳ lỗi ràng buộc nào.

1. Các trường và model dữ liệu được sử dụng trong dự án

* *AutoField*: được sử dụng để đại diện cho một cột cơ sở dữ liệu tự động tăng, thường được sử dụng để làm khóa chính, định danh duy nhất cho mỗi bản ghi trong một bảng.
* *TextField*: được sử dụng để lưu trữ dữ liệu văn bản dài, không giới hạn.
* *CharField*: được sử dụng để lưu trữ dữ liệu dạng chuỗi ký tự, có giới hạn.
* *EmailField*: được sử dụng để lưu trữ dữ liệu kiểu chuỗi địa chỉ email, tự động kiểm tra xem địa chỉ có đúng định dạng không khi nhập, lưu trữ.
* *DateTimeField*: được sử dụng để lưu trữ dữ liệu kiểu ngày, thời gian được lấy theo múi giờ (timezone) được định nghĩa trong setting.py.
* *BooleanField*: được sử dụng để lưu trữ dữ liệu kiểu logic, True hoặc False.
* *ImageField*: được sử dụng để lưu trữ dữ liệu kiểu file hình ảnh.
* *FileField*: được sử dụng để lưu trữ dữ liệu kiểu file bất kỳ.
* *ForeignKey*: được dùng để gán liên kết khóa ngoài.
* *models.TextChoices:* là một lớp model cho phép định nghĩa các lựa chọn cho các trường CharField hoặc TextField, giúp mã hóa một số lựa chọn cụ thể. Một thuộc tính của lớp TextChoices có 2 giá trị là giá trị (value) và nhãn (label). Giá trị (value) sẽ là giá trị thật được lưu trong bản ghi cơ sở dữ liệu còn nhãn (label) thường sẽ được dùng để hiển thị trong giao diện.

## Jinja

*Jinja* là một hệ thống template engine cho Python. Nó được sử dụng để tách logic xử lý và hiển thị trong ứng dụng web, giúp code trở nên sáng sủa và dễ hiểu. Jinja được sử dụng rộng rãi trong các framework web như Flask và Django. Khi cài đặt Django, Jinja cũng sẽ được cài đặt theo.

1. Ứng dụng của Jinja

* *Tùy biến Template*: Jinja cho phép tùy biến template với nhiều tính năng mạnh mẽ như:
* Cho phép tái sử dụng các đoạn mã HTML
* Chuyển đổi dữ liệu trước khi hiển thị
* Hỗ trợ các biểu thức logic và vòng lặp phức tạp
* *Hiệu suất:* Jinja được thiết kế để có hiệu suất cao hơn trong một số trường hợp nhất định so với template engine mặc định của Django. Điều này có thể quan trọng đối với các ứng dụng web yêu cầu xử lý một lượng lớn dữ liệu hoặc có lưu lượng truy cập cao.
* *Tính năng mở rộng:* Jinja hỗ trợ nhiều tính năng mở rộng và tích hợp tốt với các thư viện Python khác. Điều này giúp dễ dàng mở rộng các tính năng của ứng dụng web.
* *Kế thừa template:* Jinja hỗ trợ kế thừa template, cho phép tạo ra các template cơ sở và sau đó mở rộng chúng, giúp quản lý mã HTML dễ dàng và giảm thiểu sự trùng lặp.
* *Khả năng tương thích:* Jinja có thể tích hợp với Django dễ dàng thông qua các thư viện có sẵn. Điều này giúp các lập trình viên có thể sử dụng các tính năng của Jinja mà không cần phải từ bỏ các tiện ích và cấu trúc của Django.
* *Hỗ trợ người dùng cuối:* Jinja giúp tạo ra các giao diện người dùng tùy biến và động hơn, giúp nâng cao trải nghiệm người dùng cuối. Các lập trình viên có thể tạo ra các trang web phong phú về mặt hình ảnh và chức năng.

1. Những kiến thức cơ bản trong Jinja

* *Giá trị và câu lệnh:* Các biểu thức được bao bọc trong cặp dấu {{ … }} để hiển thị giá trị, và các câu lệnh điều khiển như vòng lặp và điều kiện được sử dụng với cặp {% … %}.
  + {{ variable }} : dùng để hiển thị giá trị biểu thức
  + {% for giá\_trị in chuỗi\_giá\_trị %} code\_html {% endfor %} : vòng lặp
  + {% if điều\_kiện %} code\_html {% else %} code\_html {% endif %}: điều kiện
* *Kế thừa và mở rộng:* Jinja hỗ trợ cơ chế kế thừa, cho phép tái sử dụng các mẫu và mở rộng chúng để tạo ra các mẫu lớn hơn.
  + {% extends “tên file html dùng để mở rộng”%}
  + {% block tên\_block %} code\_html {% endblock %}: trong file html dùng để mở rộng cũng phải có block trùng tên thì mới truyền code html sang được.

## Ngôn ngữ JavaScript

*JavaScript* là một ngôn ngữ lập trình thông dịch (interpreted programming language) được sử dụng chủ yếu để tạo ra các trang web tương tác và động. JavaScript được phát triển ban đầu bởi Netscape và sau đó được tiêu chuẩn hóa bởi Ecma International trong tiêu chuẩn ECMA-262. JavaScript là một phần của các trình duyệt web phổ biến như Chrome, Firefox, Safari, và Edge, và được sử dụng để thêm các tính năng động vào các trang web, như hiển thị thông báo, thay đổi nội dung của trang web mà không cần tải lại, và xử lý các sự kiện như nhấp chuột và gõ phím. JavaScript cũng có thể được sử dụng để tạo ra các ứng dụng web độc lập, nhưng nó thường được sử dụng kết hợp với HTML và CSS để tạo ra các trang web.

1. Các kiểu dữ liệu trong JavaScript

* *Number*: Kiểu dữ liệu này biểu diễn các số, bao gồm cả số nguyên và số thập phân.
* *String*: Kiểu dữ liệu này biểu diễn các chuỗi ký tự. Chuỗi được bao quanh bởi dấu nháy đơn hoặc dấu nháy kép.
* *Boolean*: Kiểu dữ liệu này chỉ có hai giá trị: true và false
* *Undefined*: Kiểu dữ liệu này đại diện cho một biến chưa được gán giá trị.
* *Null*: Kiểu dữ liệu này đại diện cho một giá trị không tồn tại hoặc không hợp lệ.
* *Object*: Kiểu dữ liệu này đại diện cho một đối tượng, một nhóm các thuộc tính và phương thức.
* *Array*: Kiểu dữ liệu này đại diện cho một mảng các giá trị. Mảng có thể chứa bất kỳ kiểu dữ liệu nào.
* *Function*: Kiểu dữ liệu này đại diện cho một hàm.
* *Symbol*: Kiểu dữ liệu này đại diện cho một giá trị duy nhất không thể thay đổi được.
* *BigInt*: Kiểu dữ liệu này cho phép bạn làm việc với số nguyên lớn hơn 2^53 - 1.
* *Date*: Kiểu dữ liệu này đại diện cho một ngày và thời gian cụ thể.
* *RegExp*: Kiểu dữ liệu này đại diện cho một biểu thức chính quy.

Khi được sử dụng trong HTML, Java script được viết trong thẻ <script>…</script>

1. Ứng dụng của JavaScript

* *Tạo tương tác động cho trang web:* JavaScript cho phép trang web phản hồi tương tác của người dùng, chẳng hạn như nhấp chuột, nhập dữ liệu vào biểu mẫu, hoặc di chuột qua các phần tử. Ví dụ: hiển thị thông báo khi người dùng nhấn vào một nút, hoặc cập nhật nội dung trang web mà không cần tải lại toàn bộ trang.
* *Xử lý và xác thực dữ liệu:* JavaScript có thể được sử dụng để xác thực dữ liệu nhập vào từ phía người dùng trước khi gửi đến máy chủ. Điều này giúp giảm tải cho máy chủ và cải thiện trải nghiệm người dùng bằng cách cung cấp phản hồi ngay lập tức.
* *Tạo hiệu ứng và hoạt hình:* JavaScript có thể được sử dụng để tạo ra các hiệu ứng hình ảnh và hoạt hình trên trang web.
* *Lập trình phía máy khách:* JavaScript chạy trên trình duyệt của người dùng, giúp giảm tải công việc trên máy chủ và cải thiện tốc độ phản hồi của trang web. Các tác vụ như thay đổi nội dung trang, quản lý sự kiện, và thực hiện các phép tính có thể được thực hiện mà không cần liên lạc với máy chủ.
* *Tích hợp các API bên ngoài:* JavaScript có thể tương tác với các API (Application Programming Interface) để lấy và gửi dữ liệu từ các dịch vụ bên ngoài như Google Maps, Twitter, Facebook, và nhiều dịch vụ khác. Điều này mở rộng khả năng của trang web, giúp tích hợp nhiều tính năng phong phú.
* *Xây dựng ứng dụng một trang:* Với sự trợ giúp của các framework như React và Angular, JavaScript có thể được sử dụng để xây dựng các ứng dụng một trang, nơi mà toàn bộ ứng dụng chạy trên một trang duy nhất và chỉ thay đổi nội dung theo tương tác của người dùng, mang lại trải nghiệm mượt mà và nhanh chóng.

## Web socket

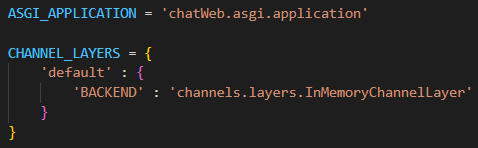
*WebSocket* là một công nghệ được sử dụng để thiết lập một kết nối liên tục, hai chiều giữa máy khách và máy chủ qua mạng Internet. Nó cho phép truyền tải dữ liệu trực tiếp và mạnh mẽ giữa các ứng dụng web.

1. Các đặc điểm của WebSocket

* *Kết nối liên tục*: WebSocket thiết lập một kết nối TCP hai chiều giữa máy khách và máy chủ, giúp truyền tải dữ liệu theo thời gian thực.
* *Giao thức dựa trên TCP*: WebSocket sử dụng giao thức TCP để thiết lập kết nối, giúp tránh được các vấn đề liên quan đến HTTP như các tiêu đề lặp đi lặp lại và chi phí liên kết.
* *Giao thức mở và tiêu chuẩn*: WebSocket là một tiêu chuẩn được định rõ bởi IETF (Internet Engineering Task Force), giúp đảm bảo tính tương thích giữa các ứng dụng trên các nền tảng và trình duyệt khác nhau.
* *Khả năng mở rộng*: WebSocket có khả năng mở rộng và tương thích tốt với các công nghệ và thư viện khác nhau như Node.js, Python, Ruby, và nhiều ngôn ngữ lập trình web khác.
* *Hiệu suất cao*: Do không cần thiết lập lại kết nối mỗi khi cần truyền tải dữ liệu, WebSocket giúp cải thiện hiệu suất và giảm độ trễ so với các phương pháp truyền tải truyền thống như long polling hay AJAX.
* *Bảo mật*: WebSocket hỗ trợ các lớp bảo mật như SSL/TLS để bảo vệ dữ liệu trong quá trình truyền tải, giúp ngăn chặn các cuộc tấn công như giả mạo hoặc đánh cắp thông tin.

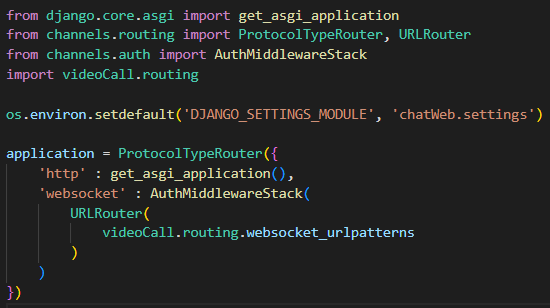
1. Tích hợp WebSocket vào dự án Django

* *Cài đặt channels và daphne*: channels và daphne là 2 thư viện hỗ trợ WebSocket của Python.
* Cài đặt chúng bằng lệnh pip installer channels và pip installer daphne trong terminal.
* Sau khi cài đặt thì ta cần vào file setting.py của dự án để khai báo sử dụng chúng. Thêm ‘channels’, ‘daphne’ và INSTALLED\_APPS. Ngoài ra cần thêm 1 số biến vào file setting. Sau khi thêm, thực hiện chạy python manage.py runserver mà thấy có thông báo Starting ASGI/Daphne thì đã cài đặt thành công.



1. Biến cần thêm

* Ngoài ra còn phải cấu hình lại file asgi.py để server có thể tiếp nhận tín hiệu từ WebSocket



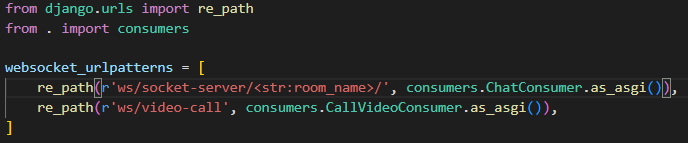
1. Ví dụ cấu hình

* *Cách sử dụng bên server*: ta cần tạo 2 file consumers.py và routing.py.
* File consumers.py sẽ chứa các lớp AsyncWebsocketConsumer hoặc WebsocketConsumer, để thực hiện các tác vụ khi có tín hiệu WebSocket từ client. Lớp AsyncWebsocketConsumer và WebsocketConsumer khá giống nhau, nhưng lớp AsyncWebsocketConsumer có thể thực hiện các hàm bất đồng bộ ‘async def’ giúp tăng hiệu suất so với việc sử dụng các hàm đồng bộ ‘def’ cổ điển của WebsocketConsumer.



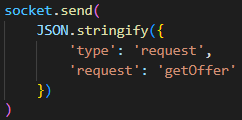
1. Ví dụ AsyncWebsocketConsumer

* File routing.py sẽ chứa các đường dẫn, mỗi đường dẫn sẽ kích hoạt 1 consumer cụ thể khi có tín hiệu từ client.



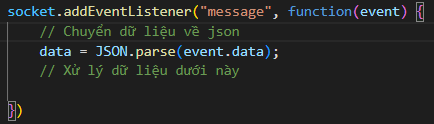
1. Ví dụ file routing

* *Cách sử dụng bên client*: trong JavaScript có sẵn lớp WebSocket hỗ trợ thực hiện gửi và nhận từ server.
* Tạo 1 đối tượng WebSocket bằng câu lệnh <tên đối tượng > = new WebSocket (<chuỗi URL kết nối đến máy chủ>); . Chuỗi URL kết nối gồm 3 thành phần: Giao thức WebSocket, URL của trang web, đường dẫn endpoint.
* Giao thức WebSocket: có 2 loại giao thức không bảo mật ‘ws://’ và giao thức có bảo mật ‘wss://’. Khi dùng giao thức bảo mật, dữ liệu được mã hóa trước khi được gửi đi và giải mã sau khi đến máy chủ, đảm bảo tính riêng tư và bảo mật của dữ liệu.
* URL của trang web: bao gồm tên miền và cổng (nếu có) của trang web, có thể dễ dàng lấy nó thông qua đối tượng window, câu lệnh ‘window.location.host’.
* Đường dẫn endpoint: là đường dẫn đã khai báo trong routing.py
* Sau khi đã tạo xong đối tượng ta có thể gửi và nhận dữ liệu với server mà không phải tải lại trang.
* Gửi dữ liệu: sử dụng hàm send của WebSocket



1. Ví dụ gửi dữ liệu

* Nhận dữ liệu: sử dụng hàm addEventListener của WebSocket



1. Ví dụ nhận dữ liệu

## WebRTC

*WebRTC*, viết tắt của Web Real-Time Communication, là một công nghệ cho phép trình duyệt web và ứng dụng truyền thông giao tiếp trực tuyến một cách trực quan và tiện lợi mà không cần cài đặt phần mềm bổ sung hay plugin. Được hỗ trợ bởi một số trình duyệt phổ biến như Google Chrome, Mozilla Firefox, và Opera, WebRTC cho phép truyền dữ liệu âm thanh, video, và dữ liệu trực tiếp giữa các máy tính mà không cần qua máy chủ trung gian.

Công nghệ này cung cấp các API (Application Programming Interfaces) để phát triển ứng dụng giao tiếp trực tuyến như video call, voice call, và chia sẻ dữ liệu trong thời gian thực.

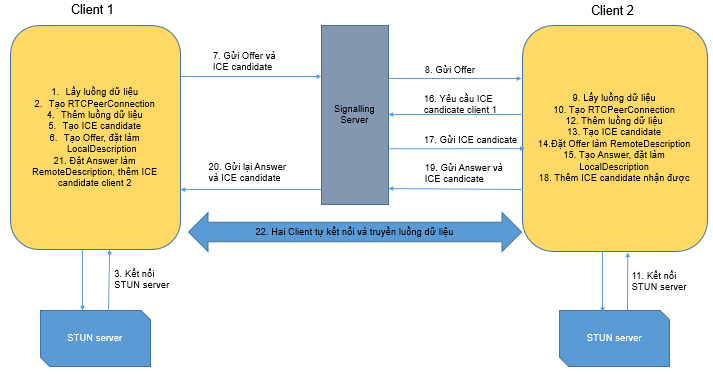
1. Một số tính năng chính của WebRTC

* *Giao tiếp ngang hàng*: WebRTC cho phép truyền dữ liệu trực tiếp giữa các máy tính, giảm thiểu độ trễ và tăng tính bảo mật bằng cách loại bỏ sự phụ thuộc vào máy chủ trung gian.
* *Điều khiển truy vấn phần cứng*: WebRTC cung cấp khả năng truy cập camera và microphone của người dùng thông qua trình duyệt, cho phép chúng ta tạo các ứng dụng giao tiếp trực tuyến mà không cần cài đặt phần mềm bổ sung.
* *Mã hóa và bảo mật*: WebRTC sử dụng các giao thức mã hóa tiêu chuẩn như DTLS (Datagram Transport Layer Security) và SRTP (Secure Real-time Transport Protocol) để bảo vệ dữ liệu trong quá trình truyền.
* *Tương thích đa nền tảng*: WebRTC được hỗ trợ trên nhiều nền tảng và trình duyệt, giúp ứng dụng phát triển trên các thiết bị và hệ điều hành khác nhau.

1. Một số thuật ngữ trong WebRTC

* *ICE (Interactive Connectivity Establishment)*: là một giao thức được sử dụng để thiết lập kết nối trực tiếp giữa các máy tính thông qua mạng Internet, bao gồm việc xác định địa chỉ IP và cổng phù hợp.
* *ICE candidate*: là một thành phần của giao thức ICE, mỗi máy tính trong một kết nối WebRTC sẽ tạo ra các ICE candidate, đại diện cho các cách tiếp cận khác nhau để truy cập địa chỉ IP và cổng của nó thông qua NAT hoặc firewall. Cụ thể, mỗi ICE candidate bao gồm một địa chỉ IP, một cổng và một giao thức (UDP, TCP hoặc SCTP).
* *NAT (Network Address Translation):* là một công nghệ được sử dụng trong mạng máy tính để chuyển đổi các địa chỉ IP của các thiết bị trong mạng nội bộ thành địa chỉ IP công cộng.
* *STUN (Session Traversal Utilities for NAT):* Là một dịch vụ hoặc máy chủ được sử dụng để tìm kiếm địa chỉ IP công cộng của một máy tính nằm sau NAT.
* *TURN (Traversal Using Relays around NAT):* Là một dịch vụ hoặc máy chủ được sử dụng khi các kỹ thuật khi STUN không thể thực hiện. Thay vì cố gắng thiết lập kết nối trực tiếp giữa hai máy tính, TURN sẽ tạo ra một "trạm" (relay) hoạt động trung gian. Dữ liệu sẽ được truyền qua trạm này, giúp vượt qua rào cản NAT.
* *SDP (Session Description Protocol):* Là một định dạng dữ liệu được sử dụng để mô tả thông tin về phiên trò chuyện hoặc kết nối giao tiếp, bao gồm các thông tin về các luồng phương tiện và các tham số kết nối.
* *Offer*: là 1 loại SDP mà các đối tác trao đổi với nhau, offer được tạo ra bởi bên yêu cầu kết nối để gửi đến bên muốn được kết nối.
* *Answer*: tương tự offer nhưng được tạo ra bởi bên được kết nối để trả lời lại cho bên yêu cầu.
* *Peer Connection (Kết nối đồng đối tác):* Đại diện cho kết nối trực tiếp giữa hai máy tính trong một ứng dụng WebRTC.
* *MediaStream (Luồng phương tiện):* Là một chuỗi dữ liệu liên tục của dữ liệu phương tiện như âm thanh hoặc video được tạo ra từ các nguồn như camera hoặc microphone.
* *DataChannel (Kênh dữ liệu):* Là một phần của API WebRTC cho phép truyền dữ liệu trực tiếp giữa các peer connection, không chỉ là dữ liệu phương tiện mà còn là dữ liệu văn bản hoặc nhị phân.
* *Signaling*: Là quá trình trao đổi thông tin giữa các máy tính để thiết lập một kết nối WebRTC, bao gồm việc truyền tải thông điệp như địa chỉ IP, cổng, và các tham số kết nối. Thường được thực hiện qua 1 server trung gian, gọi là Signalling Server.

1. Quy trình thiết lập kết nối giữa 2 máy Client 1 và Client 2 trong WebRTC gồm các bước thực hiện
2. Client 1 yêu cầu lấy luồng dữ liệu từ thiết bị như dữ liệu hình ảnh, âm thanh từ camera và microphone.
3. Client 1 tạo 1 đối tượng RTCPeerConnection
4. Kết nối đến STUN server để biết được địa chỉ IP công cộng và cổng kết nối hướng tới client 1 mà những thiết bị bên ngoài mạng nội bộ có thể truy cập để kết nối tới client 1.
5. Thêm các luồng dữ liệu đã lấy ở bước 1 vào RTCPeerConnection để có thể truyền đi.
6. Từ thông tin có được từ STUN, RTCPeerConnection tạo ra các ICE candidate.
7. Tạo 1 đối tượng Offer thông qua đối tượng RTCPeerConnection. Đặt Offer đã tạo làm LocalDescription của RTCPeerConnection.
8. Gửi Offer và các ICE candidate đã tạo tới Signalling Server.
9. Signalling Server gửi Offer cho Client 2
10. Client 2 yêu cầu lấy luồng dữ liệu dữ liệu từ camera, microphone của thiết bị.
11. Client 2 tạo 1 đối tượng RTCPeerConnection
12. Kết nối đến STUN server để biết được địa chỉ IP công cộng và cổng kết nối hướng tới client 2 mà những thiết bị bên ngoài mạng nội bộ có thể truy cập để kết nối tới client 2.
13. Thêm luồng dữ liệu đã lấy vào RTCPeerConnection để có thể truyền đi.
14. Từ thông tin có được từ STUN, RTCPeerConnection tạo ra các ICE candidate.
15. Client 2 đặt Offer Client 1 làm RemoteDescription của RTCPeerConnection.
16. Client 2 tạo Answer thông qua đối tượng RTCPeerConnection và đặt Answer đã tạo làm LocalDescription của RTCPeerConnection.
17. Client 2 yêu cầu các ICE candidate của Client 1 từ Signalling Server.
18. Signalling Server gửi lại các ICE candidate
19. Thêm các ICE candidate vào RTCPeerConnection của Client 2.
20. Gửi Answer và các ICE candidate đã tạo tới Signalling Server.
21. Signalling Server gửi lại Answer và các ICE candidate
22. Đặt Answer làm RemoteDescription và thêm các ICE candidate client 2 vào RTCPeerConnection
23. Client 1 và Client 2 bắt đầu tự kết nối và truyền dữ liệu cho với nhau.



1. Quy trình thiết lập kết nối

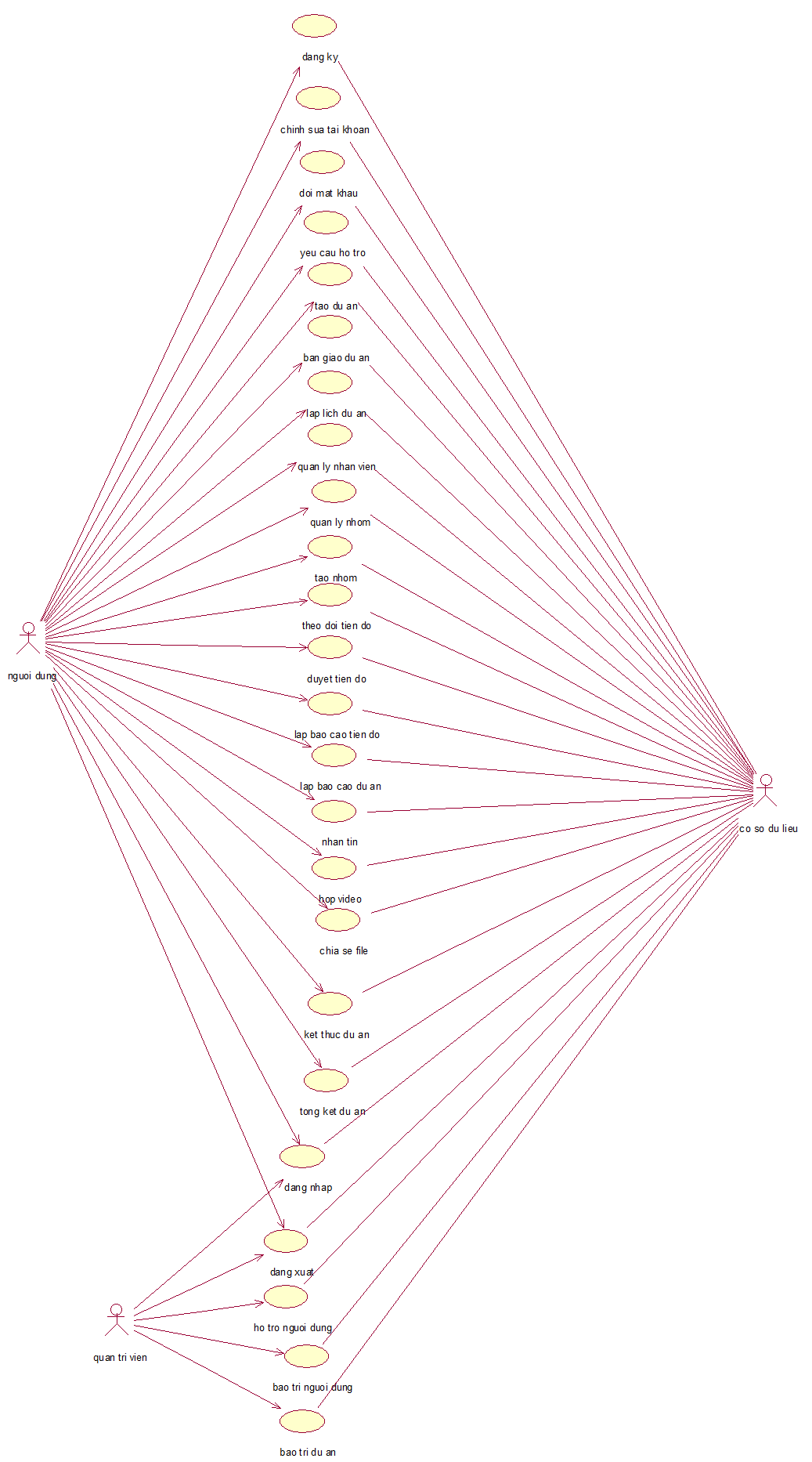
# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Yêu cầu phần mềm

* ***Quản lý công việc*:** Cho phép tạo, gán và theo dõi các công việc, công việc cụ thể cho các thành viên trong nhóm.
* ***Lập kế hoạch và theo dõi tiến độ*:** Cho phép lập kế hoạch dự án, thiết lập mốc thời gian và theo dõi tiến độ thực hiện.
* ***Giao tiếp và tương tác*:** Tạo kênh giao tiếp nội bộ trong nhóm, bao gồm cả thảo luận, trao đổi thông tin, kênh giao tiếp sẽ là nhắn tin và gọi video
* ***Quản lý tài nguyên*:** Theo dõi và quản lý các tài nguyên như tài liệu và ngân sách.
* ***Theo dõi chi phí và ngân sách*:** Giúp quản lý các chi phí liên quan đến dự án và theo dõi ngân sách được phân bổ.
* ***Quản lý rủi ro*:** Đánh giá các rủi ro có thể xảy ra trong quá trình thực hiện dự án.
* ***Báo cáo và phân tích*:** Cho phép tạo và phân tích báo cáo về tiến độ, hiệu suất và các chỉ số khác của dự án.
* ***Bảo mật thông tin*:** Đảm bảo an toàn cho thông tin và dữ liệu của dự án, bao gồm cả quản lý quyền truy cập và mã hóa dữ liệu.

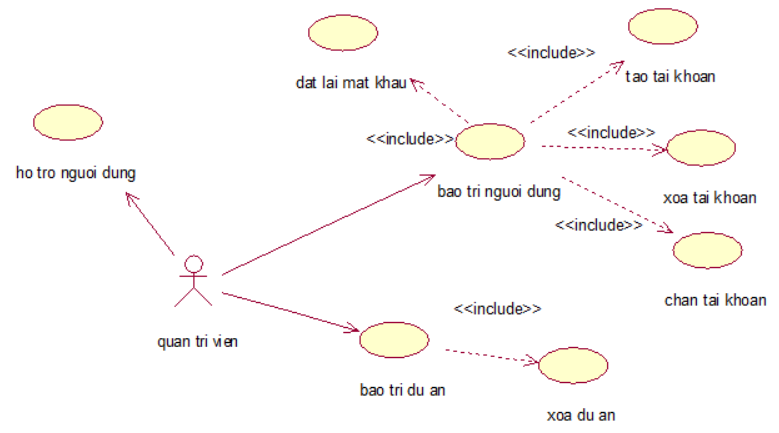
## Biểu đồ usecase

***Biểu đồ usecase tổng quát***



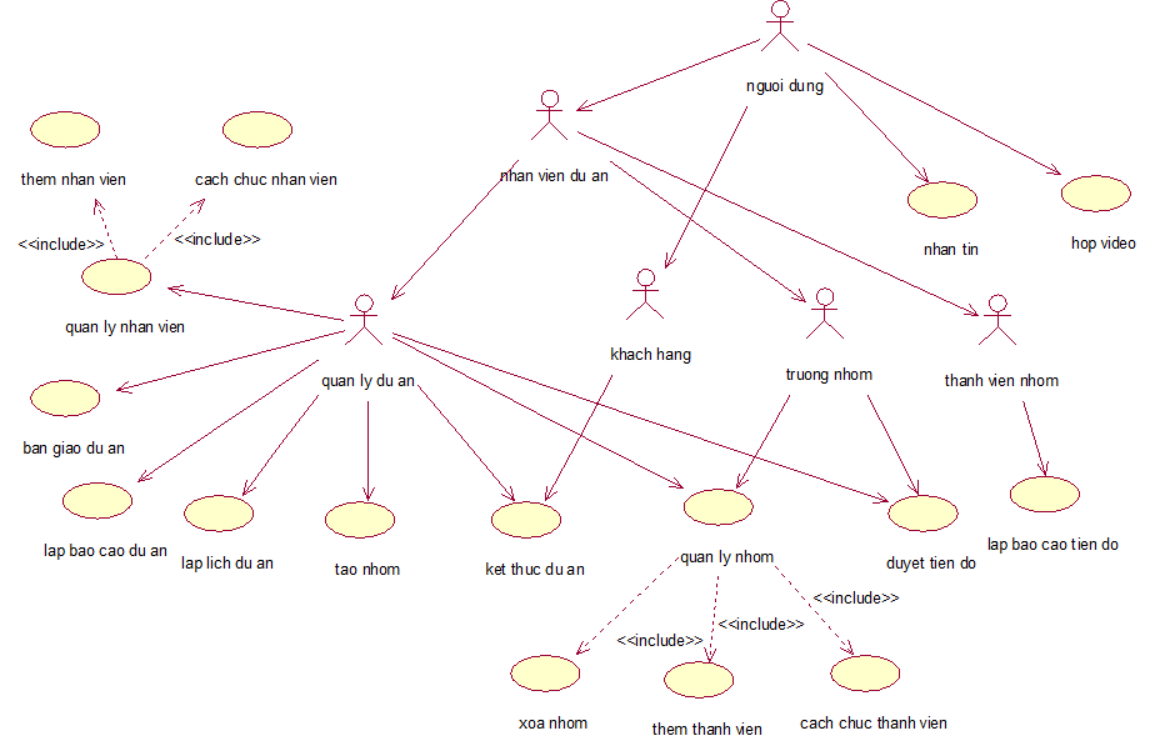
1. Biểu đồ tổng quát

***Biểu đồ usecase phân rã quản trị viên***

**

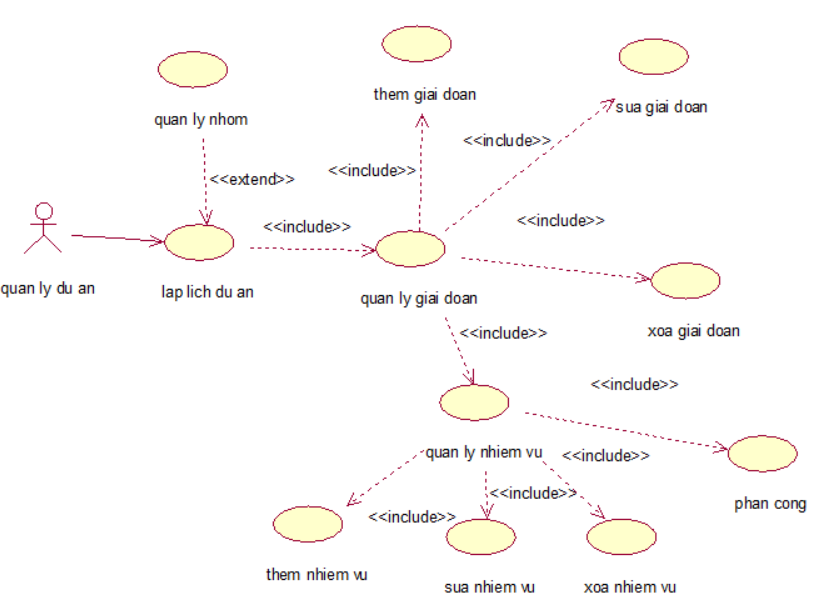
1. Biểu đồ phân rã quản trị viên

***Biểu đồ usecase phân rã người dùng***

**

1. Phân quyền người dùng

***Biểu đồ usecase phân rã lập lịch***

**

1. Biểu đồ phân rã lập lịch

## Đặc tả usecase

1. Usecase đăng ký

* *Mô tả*: cho phép người dùng đăng ký tạo tài khoản
* *Actor*: Người dùng
* *Kịch bản chính*:
* Bước 1: Hệ thống hiển thị trang đăng ký.
* Bước 2: Người dùng điền các thông tin vào form gồm: Tên tài khoản, mật khẩu, lựa chọn kiểu tài khoản. Rồi ấn “Đăng ký”
* Bước 3: Hệ thống kiểm tra xem tài khoản đã nhập có hợp lệ không. Nếu có, tạo và lưu tài khoản mới.
* Bước 4: Chuyển người dùng đến trang đăng nhập.
* *Kịch bản phụ*:
* Nhánh Bước 3: Tài khoản không hợp lệ
* Bước 3.1: Hiện thị lý do không hợp lệ và hiển thị lại trang đăng nhập, giữ nguyên dữ liệu người dùng nhập sai.

1. Usecase đăng nhập

* *Mô tả*: cho phép người dùng đăng nhập vào tài khoản đã tạo
* *Actor*: Người dùng
* *Kịch bản chính*:
* Bước 1: Hệ thống hiển thị trang đăng nhập.
* Bước 2: Người dùng điền các thông tin vào form gồm: Tên tài khoản, mật khẩu. Rồi ấn “Đăng nhập”
* Bước 3: Hệ thống tìm tài khoản phù hợp với thông tin đã nhập và đúng mật khẩu. Nếu tìm thấy
* Bước 4: Chuyển người dùng đến trang chủ của họ
* *Kịch bản phụ*:
* Nhánh Bước 2: Người dùng nhấn vào “Hỗ trợ”
* Bước 2.1: Thực hiện chức năng yêu cầu hỗ trợ, liên hệ tới quản trị viên để giải quyết vấn đề gặp phải
* Nhánh Bước 3: Không tìm thấy tài khoản hoặc mật khẩu nhập không đúng.
* Bước 3.1: Hiện thị lý do và hiển thị lại trang đăng ký, giữ nguyên dữ liệu người dùng đã nhập.

1. Usecase đăng xuất

* *Mô tả*: cho phép người dùng đăng xuất khỏi tài khoản đã đăng nhập
* *Actor*: Người dùng, quản trị viên (admin)
* *Kịch bản chính*:
  + Bước 1: Người dùng nhấn vào “Đăng xuất”
  + Bước 2: Chuyển hướng họ về lại trang đăng nhập
* *Kịch bản phụ*: không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công.

1. Usecase chỉnh sửa tài khoản

* *Mô tả*: Người dùng chỉnh sửa các thông tin cá nhân được lưu.
* *Actor*: Người dùng
* *Kịch bản chính*:
* Bước 1: Người dùng nhấn vào “Thông tin cá nhân”
* Bước 2: Hệ thống chuyển hướng người dùng đến trang chỉnh sửa thông tin cá nhân. Với form được điền sẵn các thông tin đã được lưu bởi hệ thống gồm: tên tài khoản, email, ảnh đại diện.
* Bước 3: Người dùng sửa thông tin trong form. Rồi ấn “Lưu”.
* Bước 4: Hệ thống kiểm tra các thông tin mới đã hợp lệ chưa. Nếu hợp lệ.
* Bước 5: Hệ thống lưu thông tin mới và chuyển hướng người dùng về trang chủ.
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công.

1. Usecase đổi mật khẩu

* *Mô tả*: Người dùng đổi mật khẩu cho tài khoản đang đăng nhập
* *Actor*: Người dùng
* *Kịch bản chính*:
* Bước 1: Người dùng nhấn vào “Đổi mật khẩu”
* Bước 2: Chuyển hướng họ đến trang đổi mật khẩu
* Bước 3: Người dùng điền các thông tin vào form gồm: Mật khẩu cũ, mật khẩu mới (cần nhập 2 lần để xác định). Rồi ấn “Đổi mật khẩu”
* Bước 4: Hệ thống kiểm tra xem thông tin form đã hợp lệ chưa. Nếu hợp lệ.
* Bước 5: Hệ thống lưu mật khẩu mới và chuyển hướng người dùng về trang chủ.
* *Kịch bản phụ:*
  + Nhánh B4: thông tin không hợp lệ
  + Bước 4.1: Chuyển lại về B2 nhưng kèm theo thông báo tại sao chưa hợp lệ, vẫn dữ nguyên thông tin đã nhập từ trước.

1. Usecase yêu cầu hỗ trợ

* *Mô tả*: người dùng nhắn tin trực tiếp với quản trị viên để yêu cầu 1 số vấn đề khi sử dụng web
* *Actor*: Người dùng
* *Kịch bản chính*:
* Bước 1: Người dùng nhấn vào “Hỗ trợ”
* Bước 2: Chuyển hướng họ đến trang hỗ trợ
* Bước 3: Người dùng nhập tin nhắn để gửi cho quản trị viên, ấn “Gửi”, đợi phản hồi từ quản trị viên.
* Bước 4: Chức năng kết thúc khi người dùng rời đi
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng từng đăng nhập thành công ít nhất 1 lần trước đây.

1. Usecase tạo dự án

* *Mô tả*: người dùng tạo 1 dự án
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng nhập vào “Tạo dự án”
* Bước 2: Hệ thống đưa người dùng đến trang tạo dự án.
* Bước 3: Người dùng nhập thông tin vào form thông tin dự án gồm: tên dự án, mô tả, ngân sách. Rồi ấn “Tạo”.
* Bước 4: Hệ thống kiểm tra thông tin đã nhập đúng chưa
* Bước 5: Nếu đã đúng, lưu lại và đưa người dùng đến trang chủ.
* *Kịch bản phụ*:
  + Nhánh Bước 4: nếu chưa, hiển thị lý do và yêu cầu người dùng nhập lại.
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase bàn giao dự án

* *Mô tả*: Người dùng bàn giao dự án mình đang quản lý cho 1 người khác
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng ấn vào “Chỉnh sửa thông tin dự án” ở trang chủ dự án.
* Bước 2: Hệ thống đưa người dùng đến trang chỉnh sửa thông tin dự án.
* Bước 3: Người dùng chọn người nhận bàn giao trong trường Quản lý dự án. Rồi nhấn “Lưu”
* Bước 4: Hệ thống lưu thông tin
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase thêm giai đoạn

* *Mô tả*: Người dùng thêm 1 giai đoạn vào dự án
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng đang ở trang lập lịch, nhấn vào “Thêm giai đoạn”
* Bước 2: Hệ thống hiển thị form thêm giai đoạn
* Bước 3: Người dùng điền vào form: tên giai đoạn, ngày bắt đầu, ngày kết thúc. Rồi ấn “Lưu”
* Bước 4: Hệ thống kiểm tra xem nhập liệu đã đúng chưa. Nếu đúng thì lưu lại thay đổi và dừng hiển thị form nhập.
* *Kịch bản phụ*:
* Nhánh Bước 1: Nếu người dùng chọn 1 giai đoạn trước khi nhấn nút thì, giai đoạn được tạo sẽ có thứ tự ngay đằng sau giai đoạn được chọn.
* Nhánh Bước 4: Nếu nhập liệu sai, đưa ra cảnh báo và yêu cầu nhập lại.
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase sửa giai đoạn

* *Mô tả*: Người dùng sửa 1 giai đoạn
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng chọn giai đoạn cần sửa rồi ấn nút “Sửa”
* Bước 2: Hệ thống hiển thị form với thông tin giai đoạn đã lưu được hiển thị
* Bước 3: Người dùng sửa thông tin rồi ấn “Lưu”
* Bước 4: Hệ thống kiểm tra xem nhập liệu đã đúng chưa. Nếu đúng thì lưu lại thay đổi và dừng hiển thị form.
* *Kịch bản phụ*:
* Nhánh Bước 4: Nếu nhập liệu sai, đưa ra cảnh báo và yêu cầu sửa lại.
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase xóa giai đoạn

* *Mô tả*: Người dùng xóa 1 giai đoạn đã tạo
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng chọn giai đoạn cần xóa, rồi ấn “Xóa”
* Bước 2: Hệ thống hiển thị form xác nhận xóa
* Bước 3: Người dùng xác nhận xóa
* Bước 4: Hệ thống xóa giai đoạn
* *Kịch bản phụ*:
* Nhánh Bước 3: Người dùng chọn không xóa, dừng chức năng.
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase thêm nhiệm vụ

* *Mô tả*: Người dùng thêm 1 nhiệm vụ vào giai đoạn
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng đang ở trang lập lịch, nhấn vào giai đoạn cần thêm nhiệm vụ rồi nhấn vào “Thêm”
* Bước 2: Hệ thống hiển thị form thêm nhiệm vụ
* Bước 3: Người dùng điền vào form: tên nhiệm vụ, ngày bắt đầu, ngày kết thúc, mô tả, chi phí. Rồi ấn “Lưu”
* Bước 4: Hệ thống kiểm tra xem nhập liệu đã đúng chưa. Nếu đúng thì lưu lại thay đổi và dừng hiển thị form nhập.
* *Kịch bản phụ*:
* Nhánh Bước 1: Nếu người dùng chọn 1 nhiệm vụ thay vì 1 giai đoạn thì nhiệm vụ được tạo sẽ có thứ tự ngay đằng sau nhiệm vụ được chọn.
* Nhánh Bước 4: Nếu nhập liệu sai, đưa ra cảnh báo và yêu cầu nhập lại.
* *Tiền điều kiện*: Người dùng đăng nhập thành công.

1. Usecase sửa nhiệm vụ

* *Mô tả*: Người dùng sửa 1 nhiệm vụ đã tạo.
* *Actor*: Người dùng (quản lý dự án).
* *Kịch bản chính*:
* Bước 1: Người dùng chọn nhiệm vụ cần sửa rồi ấn nút “Sửa”
* Bước 2: Hệ thống hiển thị form với thông tin nhiệm vụ đã lưu được hiển thị
* Bước 3: Người dùng sửa thông tin rồi ấn “Lưu”
* Bước 4: Hệ thống kiểm tra xem nhập liệu đã đúng chưa. Nếu đúng thì lưu lại thay đổi và dừng hiển thị form.
* *Kịch bản phụ*:
* Nhánh Bước 4: Nếu nhập liệu sai, đưa ra cảnh báo và yêu cầu sửa lại.
* *Tiền điều kiện*: Người dùng đăng nhập thành công.

1. Usecase xóa nhiệm vụ

* *Mô tả*: Người dùng xóa 1 nhiệm vụ đã tạo
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng chọn nhiệm vụ cần xóa, rồi ấn “Xóa”
* Bước 2: Hệ thống hiển thị form xác nhận xóa
* Bước 3: Người dùng xác nhận xóa
* Bước 4: Hệ thống xóa giai đoạn
* *Kịch bản phụ*:
* Nhánh Bước 3: Người dùng chọn không xóa, dừng chức năng.
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase phân công

* *Mô tả*: Người dùng phân công cho người dùng khác để hoàn thành 1 nhiệm vụ
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng chọn 1 nhiệm vụ, rồi ấn “Phân công”
* Bước 2: Hệ thống hiển thị form phân công gồm phân công cho nhân viên hoặc phân cho thành viên nhóm
* Bước 3: Nếu người dùng chọn phân công cho nhân viên, người dùng chọn người muốn phân công, rồi ấn “Lưu”
* Bước 4: Hệ thống lưu thông tin đã nhập
* *Kịch bản phụ*:
* Nhánh Bước 2: Nếu người dùng chọn phân công cho thành viên nhóm, người dùng phải chọn nhóm có thành viên đó và chọn thành viên đó trong nhóm rồi ấn “Lưu”. Nếu làm vầy thì nhóm trưởng của nhóm đó có thể duyệt tiến độ cho nhiệm vụ này thay cho quản lý dự án.
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase thêm nhân viên

* *Mô tả*: Người dùng thêm nhân viên vào dự án
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng ở trang chủ dự án, ấn “Nhân viên”
* Bước 2: Hệ thống hiển thị trang nhân viên
* Bước 3: Người dùng chọn vào “Thêm thành viên”
* Bước 4: Hệ thống hiển thị trang thêm nhân viên
* Bước 5: Người dùng sử dụng công cụ tìm để tìm và chọn những người muốn thêm vào dự án rồi ấn “Thêm”
* Bước 6: Hệ thống lưu lựa chọn
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase cách chức nhân viên

* *Mô tả*: Người dùng cách chức nhân viên ra khỏi dự án
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng ở trang chủ dự án, ấn “Nhân viên”
* Bước 2: Hệ thống hiển thị trang nhân viên
* Bước 3: Người dùng chọn những người muốn cách chức ra khỏi dự án rồi ấn “Cách chức”
* Bước 4: Hệ thống lưu lựa chọn
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase tạo nhóm

* *Mô tả*: Người dùng chia nhóm cho các nhân viên trong dự án
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng ở trang chủ dự án, rồi ấn “Nhóm”
* Bước 2: Hệ thống hiển thị trang chủ nhóm
* Bước 3: Người dùng ấn “Tạo nhóm”
* Bước 4: Hệ thống hiển thị trang tạo nhóm
* Bước 5: Người dùng nhập tên nhóm và chọn số lượng thành viên tối đa trong nhóm rồi ấn “Tạo”
* Bước 6: Hệ thống kiểm tra xem nhập liệu đã đúng chưa. Nếu đã đúng thì lưu lại.
* *Kịch bản phụ*:
* Nhánh Bước 6: Nếu nhập liệu sai, đưa ra cảnh báo và yêu cầu nhập lại.
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase xóa nhóm

* *Mô tả*: Người dùng xóa nhóm đã có
* *Actor*: Người dùng (quản lý dự án, trưởng nhóm)
* *Kịch bản chính*:
* Bước 1: Người dùng đang ở trang chủ nhóm
* Bước 2: Người dùng chọn nhóm cần xóa, rồi ấn “Xóa nhóm”
* Bước 3: Hệ thống hiển thị yêu cầu xác nhận xóa
* Bước 4: Người dùng chọn xóa
* Bước 5: Hệ thống xóa nhóm
* *Kịch bản phụ*:
* Nhánh Bước 5: Người dùng chọn không xóa, dừng chức năng
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase thêm thành viên

* *Mô tả*: Người dùng thêm thành viên vào nhóm
* *Actor*: Người dùng (quản lý dự án, trưởng nhóm)
* *Kịch bản chính*:
* Bước 1: Người dùng ở trang chủ nhóm, chọn 1 nhóm cần thêm thành viên, rồi ấn “Cài đặt nhóm”
* Bước 2: Hệ thống hiển thị trang cài đặt nhóm.
* Bước 3: Người dùng chọn “Thành viên nhóm”
* Bước 4: Hệ thống hiển thị trang thành viên nhóm
* Bước 5: Người dùng tìm những người cần thêm trong danh sách nhân viên dự án, rồi ấn “Thêm”
* Bước 6: Hệ thống lưu người được thêm vào nhóm
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase cách chức thành viên

* *Mô tả*: Người dùng cách chức thành viên ra khỏi nhóm
* *Actor*: Người dùng (quản lý dự án, trưởng nhóm)
* *Kịch bản chính*:
* Bước 1: Người dùng ở trang chủ nhóm, chọn 1 nhóm cần cách chức thành viên, rồi ấn “Cài đặt nhóm”
* Bước 2: Hệ thống hiển thị trang cài đặt nhóm
* Bước 3: Người dùng chọn “Thành viên nhóm”
* Bước 4: Hệ thống hiển thị trang thành viên nhóm
* Bước 5: Người tìm người cần cách chức rồi ấn vào nút “Cách chức” của thành viên đó
* Bước 6: Hệ thống lưu lại những thay đổi
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase theo dõi tiến độ

* *Mô tả*: Người dùng xem tiến độ hoàn thành của dự án
* *Actor*: Người dùng
* *Kịch bản chính*:
* Bước 1: Người dùng ở trang chủ dự án, rồi ấn “Tiến độ”
* Bước 2: Hệ thống hiển thị trang tiến độ gồm thông tin mức độ hoàn thành các giai đoạn của dự án và danh sách các báo cáo dự án.
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase duyệt tiến độ

* *Mô tả*: Người dùng duyệt tiến độ hoàn thành của 1 nhiệm vụ
* *Actor*: Người dùng (quản lý dự án, trưởng nhóm)
* *Kịch bản chính*:
* Bước 1: Người dùng vào trang lập lịch, chọn vào nhiệm vụ cần duyệt, rồi ấn “Tiến độ”
* Bước 2: Hệ thống hiển thị trang tiến độ nhiệm vụ gồm danh sách các báo cáo tiến độ và form tiến độ.
* Bước 3: Dựa theo đánh giá cá nhân người dùng nhập vào mức độ hoàn thành của nhiệm vụ đồng thời viết kết quả kiểm tra nhiệm vụ của mình rồi ấn “Lưu”.
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công
* *Lưu ý*: Trưởng nhóm chỉ duyệt được nhiệm vụ khi nhiệm vụ đó được giao cho thành viên trong nhóm còn quản lý dự án duyệt được tất cả nhiệm vụ.

1. Usecase lập báo cáo dự án

* *Mô tả*: Người dùng lập báo cáo tiến độ thực hiện của dự án
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Người dùng ở trang chủ dự án, người dùng chọn “Lập báo cáo dự án”
* Bước 2: Hệ thống đưa đến trang lập báo cáo. Hiển thị form tạo báo cáo. Phần tiến độ trong báo cáo được tạo tự động một phần, thông tin ứng với tiến độ dự án lúc tạo báo cáo.
* Bước 3: Người dùng nhập vào rủi ro đang gặp phải và kiến nghị, rồi ấn “Lưu”
* Bước 4: Hệ thống lưu báo cáo
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase lập báo tiến độ

* *Mô tả*: Người được giao nhiệm vụ lập báo cáo quá trình hoàn thành nhiệm vụ
* *Actor*: Người dùng
* *Kịch bản chính*:
* Bước 1: Người dùng vào trang lập lịch, chọn vào nhiệm vụ cần báo cáo, rồi ấn “Báo cáo”
* Bước 4: Hệ thống hiển thị trang báo cáo
* Bước 5: Người dùng nhập vào form báo cáo gồm: mô tả quá trình hoàn thành, kết quả hoàn thành, thời gian bắt đầu thực tế, thời gian kết thúc thực tế, rồi ấn “Lưu”.
* Bước 6: Hệ thống kiểm tra xem nhập liệu đã đúng chưa. Nếu đã đúng thì lưu lại.
* *Kịch bản phụ*:
* Nhánh Bước 6: Nếu nhập liệu sai, đưa ra cảnh báo và yêu cầu nhập lại.
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase chia sẻ file

* *Mô tả*: Người dùng tải file lên để những người dùng khác có thể xem và tải xuống
* *Actor*: Người dùng
* *Kịch bản chính*:
* Bước 1: Người dùng ở trang chủ dự án, rồi ấn “File”
* Bước 2: Hệ thống hiển thị trang chia sẻ file
* Bước 3: Người dùng chọn “Tải file lên”
* Bước 4: Hiển thị dialog chọn file để người dùng chọn
* Bước 5: Người dùng chọn file rồi ấn “Open”
* Bước 6: Hệ thống lưu file tải lên
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase nhắn tin

* *Mô tả*: Người dùng nhắn tin trong nhóm
* *Actor*: Người dùng
* *Kịch bản chính*:
* Bước 1: Người dùng ở trang chủ nhóm, chọn 1 nhóm để nhắn tin
* Bước 2: Hệ thống hiển thị tin nhắn đã có
* Bước 3: Người dùng viết tin nhắn vào hộp tin nhắn rồi ấn “Gửi”
* Bước 4: Hệ thống gửi tin nhắn đến những người dùng khác trong nhóm
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase kết thúc dự án

* *Mô tả*: Người dùng kết thúc 1 dự án
* *Actor*: Người dùng (quản lý dự án, khách hàng)
* *Kịch bản chính*:
* Bước 1: Người dùng (khách hàng) nhấn vào “Kết thúc dự án” tại trang chủ của dự án, khi tất cả người dùng (khách hàng) của dự án đồng ý cho kết thúc dự án. Người dùng (quản lý dự án) có thể nhấn vào “Kết thúc dự án” tại trang chủ của dự án của mình.
* Bước 2: Hệ thống hiển thị trang kết thúc dự án
* Bước 3: Người dùng (quản lý dự án) nhận vào các thông tin bao gồm: kết quả dự án, đánh giá sản phẩm của dự án, kết quản bàn giao sản phẩm cho khách hàng, rồi ấn “Lưu”.
* Bước 4: Hệ thống kiểm tra thông tin nhập rồi lưu lại
* *Kịch bản phụ*:
* Nhánh Bước 4: Nếu nhập liệu sai, đưa ra cảnh báo và yêu cầu nhập lại.
* *Tiền điều kiện*: Người dùng đăng nhập thành công, dự án phải thỏa mãn các yêu cầu: tất cả nhiệm vụ đã đạt 100% tiến độ, mỗi nhiệm vụ phải có ít nhất 1 báo cáo thực hiện.

1. Usecase tổng kết dự án

* *Mô tả*: Người dùng xem thông tin 1 dự án mà mình có tham gia đã kết thúc
* *Actor*: Người dùng
* *Kịch bản chính*:
* Bước 1: Người dùng ở trang chủ rồi chọn 1 dự án trong mục “Các dự án đã hoàn thành”
* Bước 2: Hệ thống hiển thị trang tổng kết dự án đó, người dùng có thể xem các thông tin gồm: thời gian thực hiện dự án, nhân sự tham gia, các báo cáo được tạo trong quá trình thực hiện dự án, file chia sẻ.
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Người dùng đăng nhập thành công, dự án được chọn đã kết thúc

1. Usecase họp video

* *Mô tả*: Người dùng tổ chức họp video
* *Actor*: Người dùng (quản lý dự án)
* *Kịch bản chính*:
* Bước 1: Ở trang chủ dự án người dùng chọn vào “Họp”
* Bước 2: Hệ thống hiển thị trang họp video
* Bước 3: Những người dùng có tham gia dự án lúc này có thể ấn vào “Họp” để tham gia phòng họp
* Bước 4: Phòng họp duy trì đến khi người dùng là quản lý dự án rời đi.
* *Kịch bản phụ*:
* Nhánh Bước 3: nếu quản lý dự án chưa mở phòng họp thì người dùng khác ấn vào sẽ không có tác dụng.
* *Tiền điều kiện*: Người dùng đăng nhập thành công

1. Usecase hỗ trợ người dùng

* *Mô tả*: Quản trị viên nhắn tin hỗ trợ người dùng vấn đề khi sử dụng web
* *Actor*: Quản trị viên (Admin)
* *Kịch bản chính*:
* Bước 1: Người dùng gửi tin nhắn đến quản trị viên thông qua chức năng yêu cầu hỗ trợ.
* Bước 2: Quản trị viên tiếp nhận, thực hiện yêu cầu rồi trả lời lại.
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Quản trị viên đăng nhập thành công

1. Usecase tạo tài khoản

* *Mô tả*: Quản trị viên tạo 1 tài khoản người dùng mới
* *Actor*: Quản trị viên (Admin)
* *Kịch bản chính*:
* Bước 1: Quản trị viên chọn “Tạo tài khoản”
* Bước 2: Hệ thống hiển thị form tạo tài khoản
* Bước 3: Quản trị viên nhập thông tin vào form, rồi ấn “Lưu”
* Bước 4: Hệ thống kiểm tra nhập liệu đã hợp lý chưa, rồi lưu lại
* *Kịch bản phụ*:
* Nhánh Bước 4: Nếu chưa, hiển thị thông báo
* *Tiền điều kiện*: Quản trị viên đăng nhập thành công

1. Usecase xóa tài khoản

* *Mô tả*: Quản trị viên xóa tài khoản đã tạo
* *Actor*: Quản trị viên (Admin)
* *Kịch bản chính*:
* Bước 1: Quản trị viên click vào checkbox các tài khoản cần xóa, chọn hành động “Xóa”, rồi ấn “Go”
* Bước 2: Hệ thống đưa ra yêu cầu xác nhận xóa
* Bước 3: Quản trị viên đồng ý xóa
* Bước 4: Hệ thống xóa các tài khoản đã chọn
* *Kịch bản phụ*:
  + Nhánh Bước 3: Quản trị viên không đồng ý, dừng chức năng
* *Tiền điều kiện*: Quản trị viên đăng nhập thành công

1. Usecase chặn tài khoản

* *Mô tả*: Quản trị viên chặn tài khoản đã tạo
* *Actor*: Quản trị viên (Admin)
* *Kịch bản chính*:
* Bước 1: Quản trị viên click vào checkbox các tài khoản cần chặn, chọn hành động “Chặn”, rồi ấn “Go”
* Bước 2: Hệ thống chặn các tài khoản đã chọn
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Quản trị viên đăng nhập thành công

1. Usecase đặt lại mật khẩu

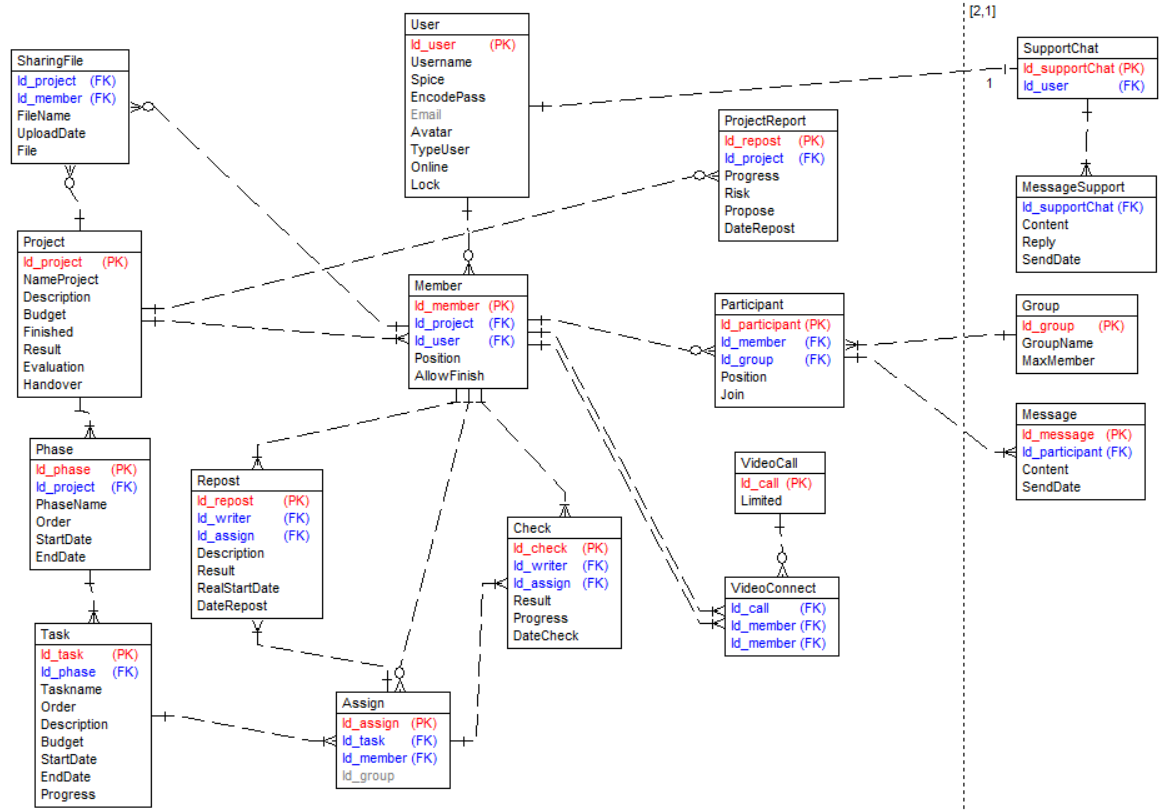
* *Mô tả*: Quản trị viên đặt lại mật khẩu cho tài khoản
* *Actor*: Quản trị viên (Admin)
* *Kịch bản chính*:
* Bước 1: Quản trị viên click vào checkbox các tài khoản cần đặt lại mật khẩu, chọn hành động “Reset password”, rồi ấn “Go”
* Bước 2: Hệ thống đặt lại mật khẩu các tài khoản đã chọn thành “123@Abc”.
* *Kịch bản phụ*: Không có
* *Tiền điều kiện*: Quản trị viên đăng nhập thành công

1. Usecase xóa dự án

* *Mô tả*: Quản trị viên xóa các dự án
* *Actor*: Quản trị viên (Admin)
* *Kịch bản chính*:
* Bước 1: Quản trị viên click vào checkbox các dự án cần xóa, chọn hành động “Xóa”, rồi ấn “Go”.
* Bước 2: Hệ thống đưa ra yêu cầu xác nhận xóa
* Bước 3: Quản trị viên đồng ý xóa
* Bước 4: Hệ thống xóa các dự án đã chọn
* *Kịch bản phụ*:
* Nhánh Bước 3: Quản trị viên không đồng ý, dừng chức năng
* *Tiền điều kiện*: Quản trị viên đăng nhập thành công

## Thiết kế cơ sở dữ liệu

1. Biểu đồ quan hệ thực thể



1. Biểu đồ ERD
2. Bảng thiết kế cơ sở dữ liệu

* *Bảng người dùng (User)*

1. Bảng người dùng

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 11 | Khóa chính | Id\_user | integer | Not null | Mã người dùng |
| 22 |  | Username | text | Not null | Tên người dùng |
| 33 |  | Spice | char(10) | Not null | Mã hỗ trợ mã hóa mật khẩu |
| 44 |  | EncodePass | char(64) | Not null | Mật khẩu sau khi mã hóa |
| 55 |  | Email | email | Null | Email người dùng |
| 66 |  | Avatar | image | Not null | Ảnh đại diện |
| 77 |  | TypeUser | char(1) | Not null | Kiểu người dùng: thành viên dự án, khách hàng |
| 88 |  | Online | boolean | Not null | True: nếu người dùng đang online, False: nếu không online |
| 99 |  | Lock | boolean | Not null | True: khi tài khoản bị khóa False: nếu không bị khóa |

* *Bảng nhân viên (Member)*

1. Bảng nhân viên

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_member | integer | Not null | Mã nhân viên |
| 2 | Khóa ngoài | Id\_project | integer | Not null | Mã dự án |
| 3 | Khóa ngoài | Id\_user | integer | Not null | Mã người dùng |
| 4 |  | Position | char(1) | Not null | Chức vụ thành viên trong dự án: quản lí dự án, nhân viên, khách hàng |
| 5 |  | AllowFinish | boolean | Not null | Cho biết khách hàng có đồng ý kết thúc dự án chưa: True nếu đã đồng ý, False nếu chưa đồng ý |

* *Bảng dự án (Project)*

1. Bảng dự án

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_project | integer | Not null | Mã dự án |
| 2 |  | NameProject | text | Not null | Tên dự án |
| 3 |  | Description | text | Not null | Mô tả dự án |
| 4 |  | Budget | integer | Not null | Ngân sách của dự án |
| 5 |  | Finished | boolean | Not null | Cho biết dự án đã kết thúc chưa: False nếu chưa kết thúc, True nếu đã kết thúc |
| 6 |  | Result | text | Not null | Kết quả cuối cùng của dự án |
| 7 |  | Evaluation | text | Not null | Đánh giá sản phẩm của dự án |
| 8 |  | Handover | text | Not null | Tình trạng bàn giao sản phẩm cho khách hàng |

* *Bảng giai đoạn (Phase)*

1. Bảng giai đoạn

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_phase | integer | Not null | Mã giai đoạn |
| 2 | Khóa ngoài | Id\_project | integer | Not null | Mã dự án |
| 3 |  | PhaseName | text | Not null | Tên giai đoạn |
| 4 |  | Order | integer | Not null | Số thứ tự giai đoạn trong dự án |
| 5 |  | StartDate | date | Not null | Ngày bắt đầu dự kiến |
| 6 |  | EndDate | date | Not null | Ngày kết thúc dự kiến |

* *Bảng nhiệm vụ (Task)*

1. Bảng nhiệm vụ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_task | integer | Not null | Mã nhiệm vụ |
| 2 | Khóa ngoài | Id\_phase | integer | Not null | Mã giai đoạn |
| 3 |  | TaskName | text | Not null | Tên nhiệm vụ |
| 4 |  | Order | integer | Not null | Số thứ tự nhiệm vụ trong giai đạn |
| 5 |  | Description | text | Not null | Mô tả nhiệm vụ |
| 6 |  | Budget | integer | Not null | Ngân sách cho nhiệm vụ |
| 7 |  | StartDate | dateTime | Not null | Ngày bắt đầu dự kiến |
| 8 |  | EndDate | dateTime | Not null | Ngày kết thúc dự kiến |
| 9 |  | Progress | integer | Not null | Tiến độ thực hiện từ 0-100% |

* *Bảng phân công (Assign)*

1. Bảng phân công

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_assign | integer | Not null | Mã phân công |
| 2 | Khóa ngoài | Id\_task | integer | Not null | Mã nhiệm vụ |
| 3 | Khóa ngoài | Id\_member | integer | Not null | Mã nhân viên |
| 4 |  | Id\_group | integer | Null | Nếu nhiệm vụ được phân cho thành viên trong nhóm thì sẽ là khóa chính bảng nhóm, nếu là phân cá nhân thì để trống |

* *Bảng báo cáo tiến độ (Repost)*

1. Bảng báo cáo tiến độ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_repost | integer | Not null | Mã báo cáo |
| 2 | Khóa ngoài | Id\_assign | integer | Not null | Mã phân công |
| 3 | Khóa ngoài | Id\_writer | integer | Not null | Mã thành viên viết báo cáo |
| 4 |  | Description | text | Not null | Mô tả quá trình thực hiện nhiệm vụ |
| 5 |  | Result | text | Not null | Kết quả thực hiện |
| 6 |  | RealStartDate | date | Not null | Ngày bắt đầu thực tế |
| 7 |  | DateRepost | dateTime | Not null | Ngày viết báo cáo |

* *Bảng báo cáo kiểm tra (Check)*

1. Bảng báo cáo kiểm tra

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_check | integer | Not null | Mã báo cáo kiểm tra |
| 2 | Khóa ngoài | Id\_writer | integer | Not null | Mã thành viên viết báo cáo |
| 3 | Khóa ngoài | Id\_assign | integer | Not null | Mã phân công |
| 4 |  | Result | text | Not null | Kết quả kiểm tra nhiệm vụ phân công lúc kiểm tra |
| 5 |  | Progress | integer | Not null | Tiến độ nhiệm vụ sau khi kiểm tra |
| 6 |  | DateCheck | dateTime | Not null | Ngày viết báo cáo kiểm tra |

* *Bảng báo cáo dự án (ProjectRepost)*

1. Bảng báo cáo dự án

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_repost | integer | Not null | Mã báo cáo dự án |
| 2 | Khóa ngoài | Id\_project | integer | Not null | Mã dự án |
| 3 |  | Progress | text | Not null | Thông tin tiến độ hiện tại của dự án |
| 4 |  | Risk | text | Not null | Rủi ro đang gặp phải |
| 5 |  | Propose | text | Not null | Đề xuất đưa ra cho dự án |
| 6 |  | DateRepost | dateTime | Not null | Ngày viết báo cáo |

* *Bảng thành viên nhóm (Participant)*

1. Bảng thành viên nhóm

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_participant | integer | Not null | Mã thành viên |
| 2 | Khóa ngoài | Id\_member | integer | Not null | Mã nhân viên |
| 3 | Khóa ngoài | Id\_group | integer | Not null | Mã nhóm |
| 4 |  | Position | char(1) | Not null | Chức vụ thành viên trong nhóm: trưởng nhóm, thành viên |
| 5 |  | Join | boolean | Not null | Giá trị là False nếu người dùng đã là thành viên nhóm nhưng đang bị cách chức, còn không giá trị là True |

* *Bảng nhóm (Group)*

1. Bảng nhóm

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_group | integer | Not null | Mã nhóm |
| 2 |  | GroupName | text | Not null | Tên nhóm |
| 3 |  | MaxNumber | integer | Not null | Số thành viên tối đa trong nhóm |

* *Bảng tin nhắn (Message)*

1. Bảng tin nhắn

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_message | integer | Not null | Mã tin nhắn |
| 2 | Khóa ngoài | Id\_participant | integer | Not null | Mã thành viên |
| 3 |  | Content | text | Not null | Nội dung tin nhắn |
| 4 |  | SendDate | dateTime | Not null | Thời gian gửi tin |

* *Bảng chia sẻ file (SharingFile)*

1. Bảng chia sẻ file

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa ngoài | Id\_project | integer | Not null | Mã dự án |
| 2 | Khóa ngoài | Id\_member\_share | integer | Not null | Mã nhân viên, id người tải lên |
| 3 |  | FileName | text | Not null | Tên file |
| 4 |  | UploadDate | dateTime | Not null | Ngày tải lên |
| 5 |  | File | file | Not null | File tải lên |

* *Bảng họp video (VideoCall)*

1. Bảng họp video

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_call | integer | Not null | Mã cuộc gọi |
| 2 |  | Limited | text | Not null | Chưa giá trị mã dự án, chỉ các thành viên của dự án mới được tham gia cuộc gọi |

* *Bảng video kết nối (VideoConnect)*

1. Bảng video kết nối

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa ngoài | Id\_call | integer | Not null | Mã cuộc gọi |
| 2 | Khóa ngoài | Id\_caller | integer | Not null | Mã nhân viên, id của người gọi caller |
| 3 | Khóa ngoài | Id\_callee | integer | Not null | Mã nhân viên, id của người trả lời callee |

* *Bảng kênh hỗ trợ (SupportChat)*

1. Bảng kênh hỗ trợ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa chính | Id\_supportChat | integer | Not null | Mã hỗ trợ |
| 2 | Khóa ngoài | Id\_user | integer | Not null | Mã người dùng |

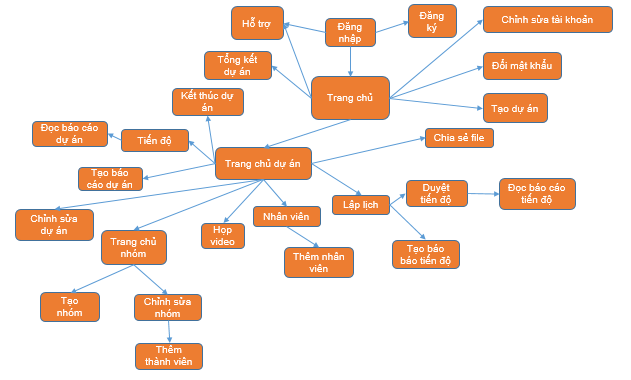
* *Bảng tin nhắn hỗ trợ (SupportMessage)*

1. Bảng tin nhắn hỗ trợ

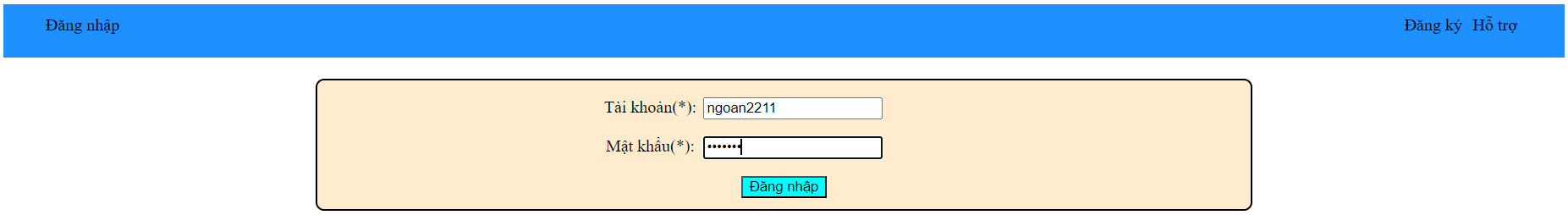
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Khóa | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | Khóa ngoài | Id\_supportChat | integer | Not null | Mã hỗ trợ |
| 2 |  | Content | integer | Not null | Nội dung tin nhắn |
| 3 |  | Reply | boolean | Not null | True: người viết tin là quản trị viên, False: người viết tin là người dùng |
| 4 |  | SendDate | dateTime | Not null | Ngày gửi tin nhắn |

# KẾT QUẢ ĐỒ ÁN

## Kết quả đạt được



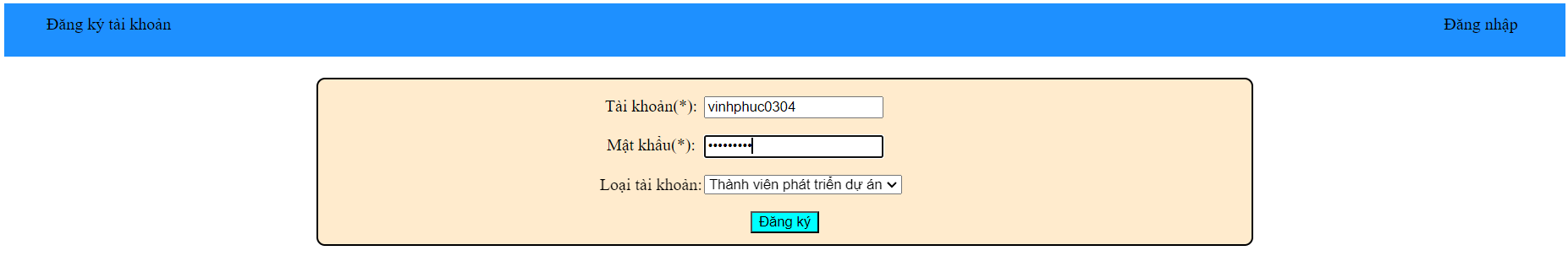
1. Sơ đồ màn hình
2. Giao diện đăng nhập

****

1. Giao diện đăng nhập

* Để đăng nhập vào website người dùng cần nhập đầy đủ tên tài khoản và mật khẩu rồi nhấn “Đăng nhập”.
* Khi muốn tạo 1 tài khoản mới, người dùng có thể chọn “Đăng ký” để thực hiện chức năng đăng ký.
* Nếu người dùng từng đăng nhập thành công trước đây sẽ có thêm nút “Hỗ trợ” để kích hoạt chức năng hỗ trợ người dùng, sử dụng khi người dùng gặp vấn đề cần hỗ trợ từ quản trị viên.

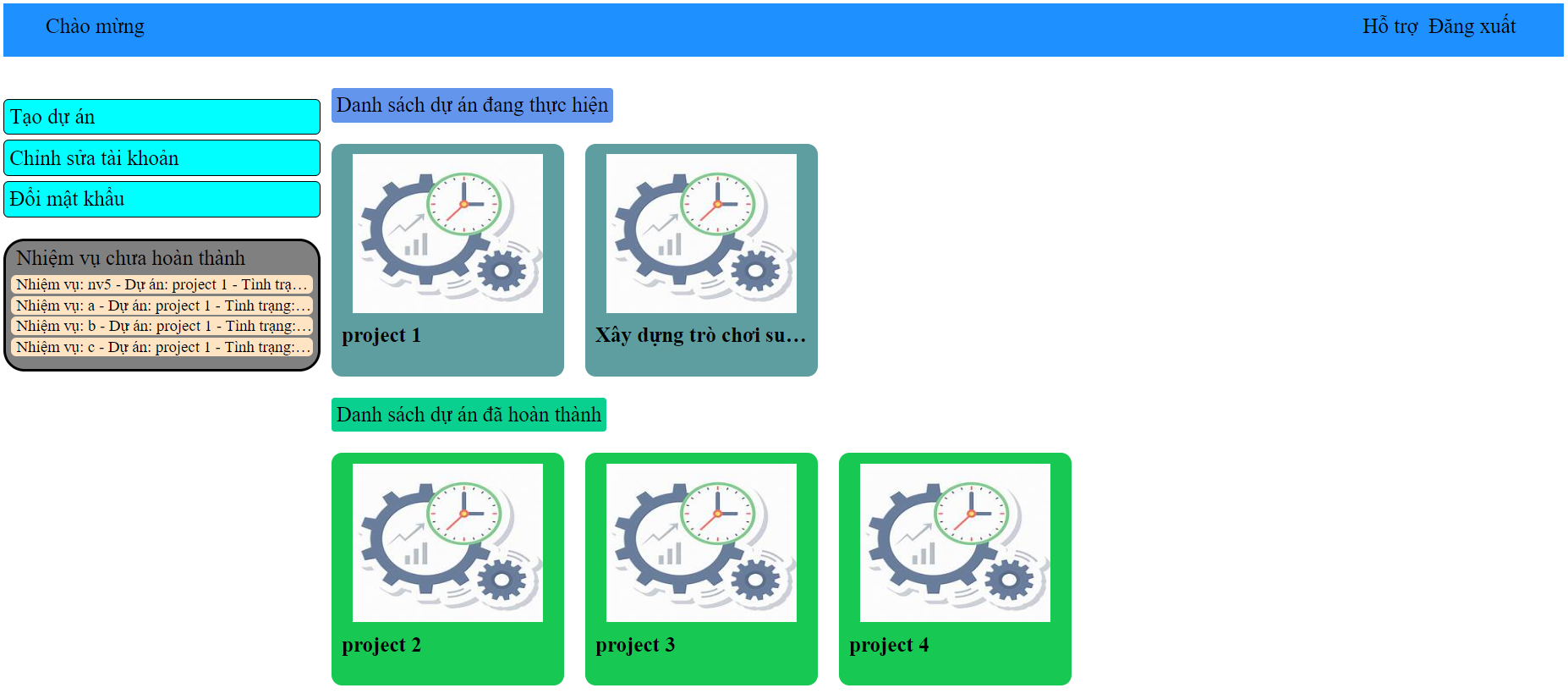
1. Giao diện đăng ký

****

1. Giao diện đăng ký

* Để đăng ký tài khoản mới, người dùng cần nhập đầy đủ tên tài khoản, mật khẩu và chọn loại tài khoản phù hợp với nhu cầu của mình. Rồi ấn “Đăng ký” để tạo tài khoản. Yêu cầu: Tên tài khoản cần có tối thiểu 8 ký tự. Mật khẩu phải có cả số, ký tự được biệt và chữ in hoa.
* Nếu người dùng muốn quay lại trang đăng nhập thì có thể chọn “Đăng nhập”, hệ thống sẽ đưa người dùng quay lại trang đăng nhập.

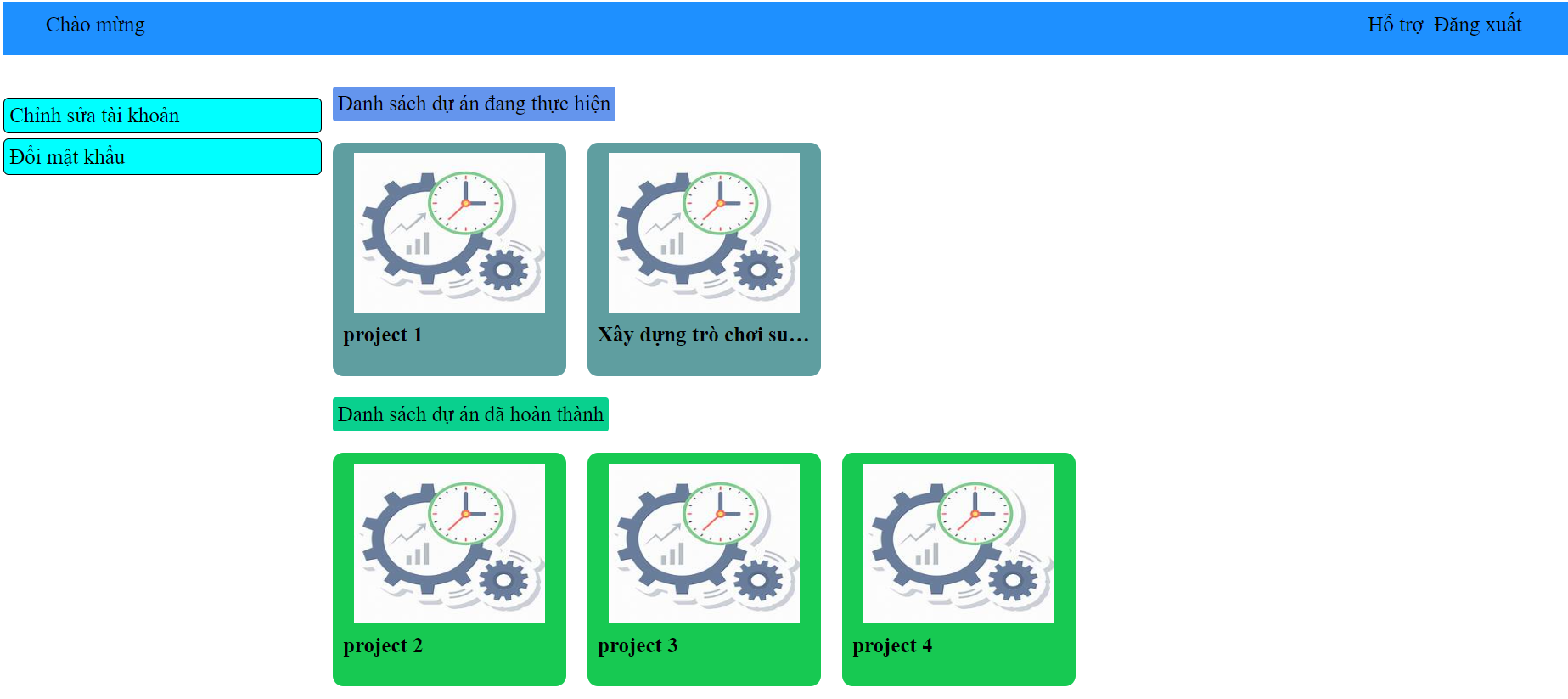
1. Giao diện trang chủ phía nhân viên dự án

****

1. Giao diện trang chủ phía nhân viên dự án

* Giao diện gồm các nút để người dùng kích hoạt các chức năng: Tạo dự án, chỉnh sửa tài khoản, đổi mật khẩu.
* Bên dưới có danh sách các nhiệm vụ mà người dùng được phân công đã đến lịch cần thực hiện.
* Các dự án được chia làm 2 mục: dự án đang thực hiện và dự án đã hoàn thành. Khi người dùng chọn vào 1 dự án đang thực hiện thì sẽ tới trang chủ dự án đó. Còn khi chọn 1 dự án đã kết thúc sẽ đi đến trang tổng kết dự án đó.

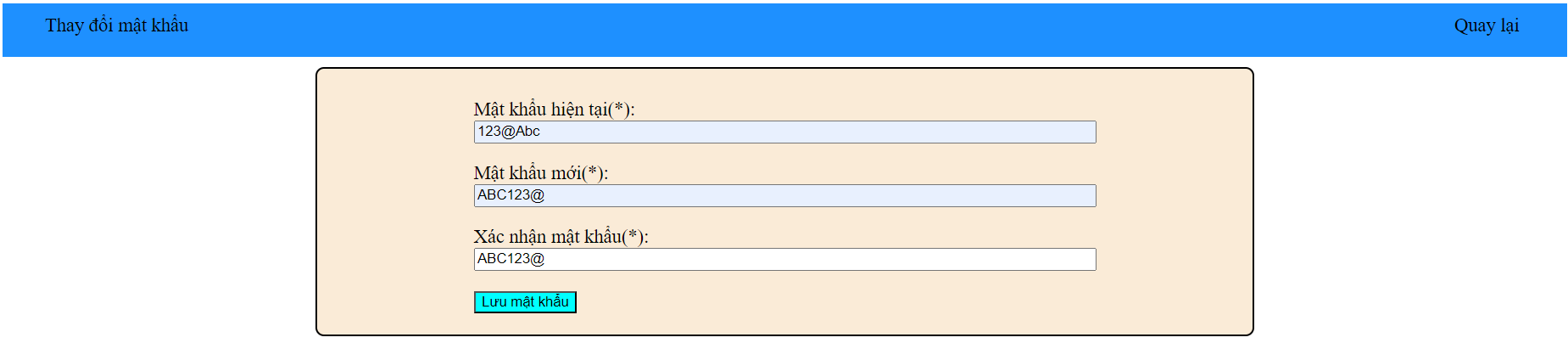
1. Giao diện trang chủ phía khách hàng

****

1. Giao diện trang chủ phía khách hàng

* Giao diện tượng tự như bên phía nhân viên dự án nhưng sẽ không có nút “Tạo dự án” hay bảng nhiệm vụ cần thực hiện

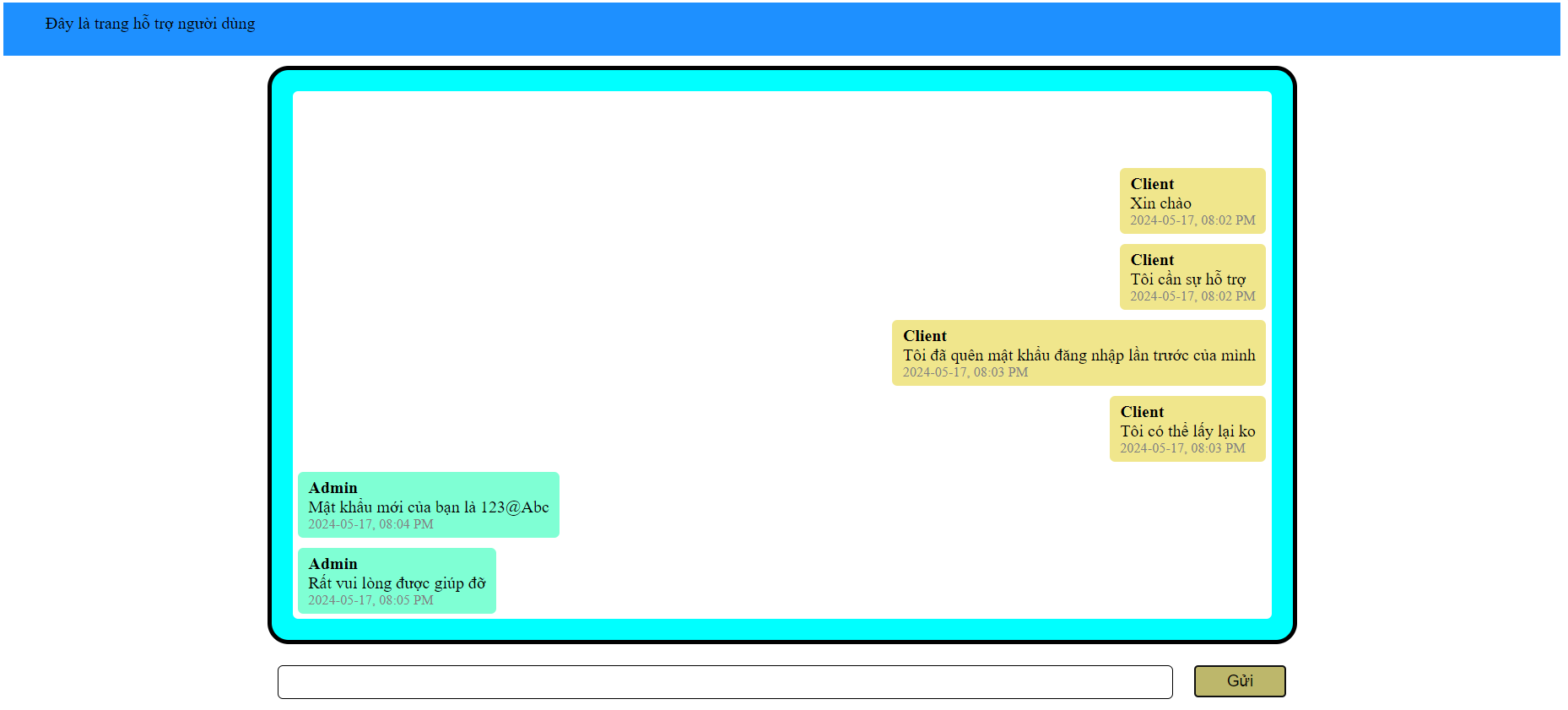
1. Giao diện đổi mật khẩu

****

1. Giao diện đổi mật khẩu

* Để thay đổi mật khẩu của tài khoản đang dùng người dùng cần nhập đầy đủ mật khẩu hiện tại đang dùng, mật khẩu mới muốn dùng. Mật khẩu mới cần nhập 2 lần để xác nhận. Rồi ấn “Lưu mật khẩu”. Yêu cầu: mật khẩu mới phải có cả số, ký tự được biệt và chữ in hoa.

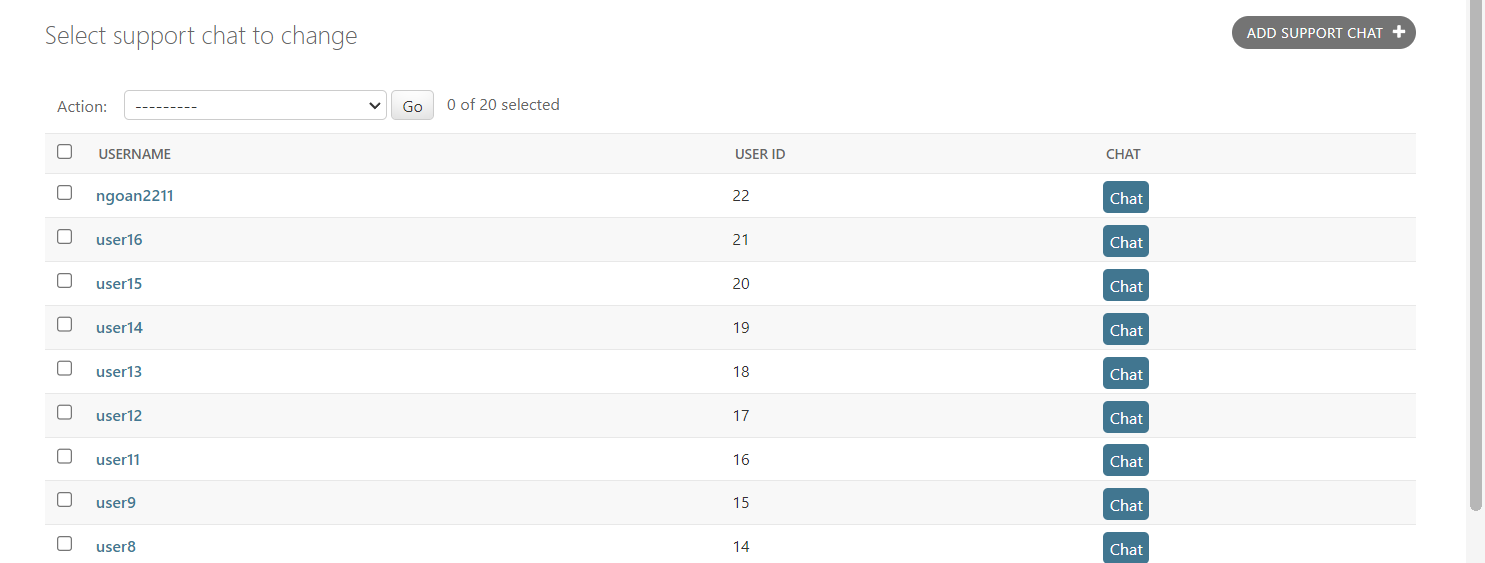
1. Giao diện hỗ trợ phía người dùng

****

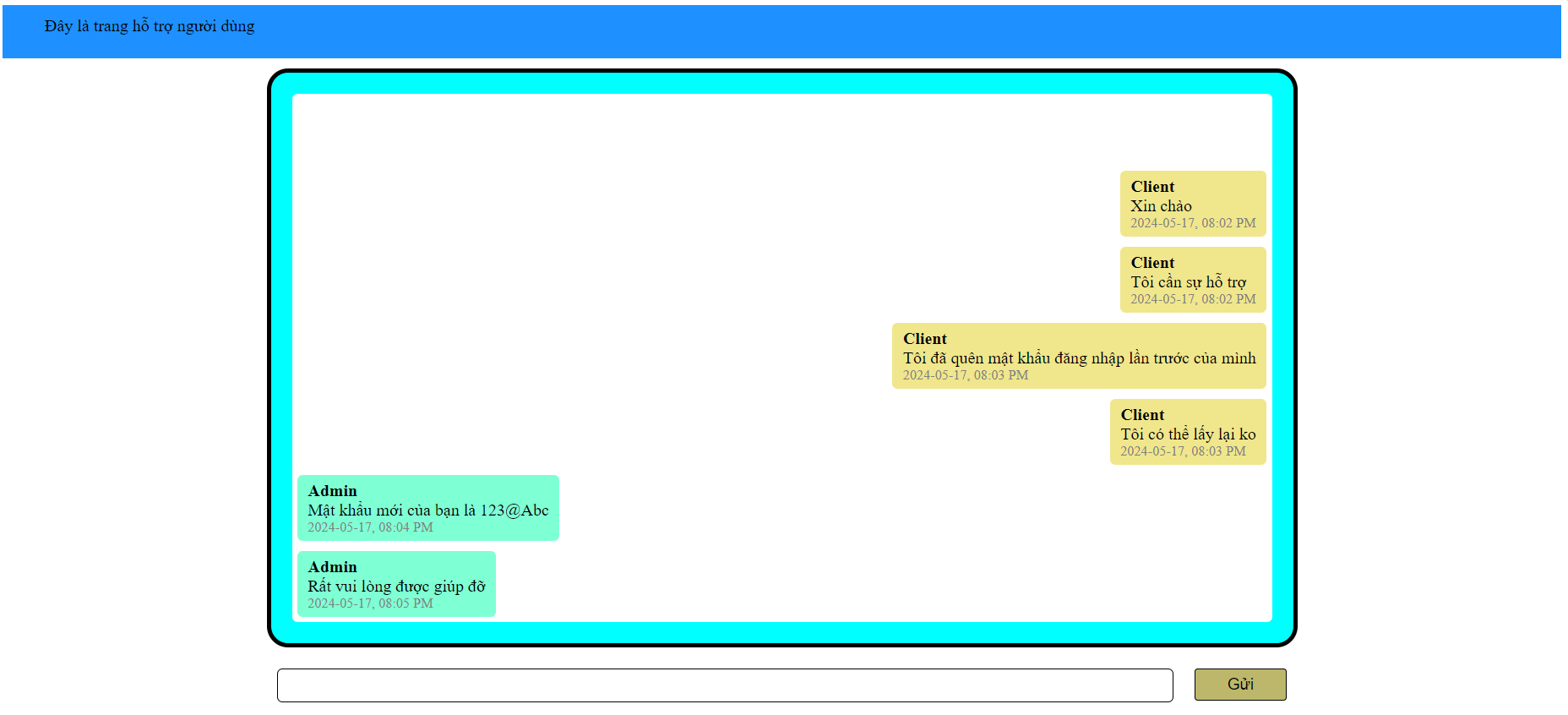
1. Giao diện hỗ trợ phía người dùng

* Để yêu cầu hỗ trợ từ quản trị viên, người dùng có thể nhập yêu cầu của mình vào thanh văn bản phía dưới rồi ấn “Gửi” để gửi tin cho quản trị viên. Sau đó quản trị viên sẽ nhắn lại để giải quyết vấn đề.

1. Giao diện hỗ trợ phía quản trị viên

**

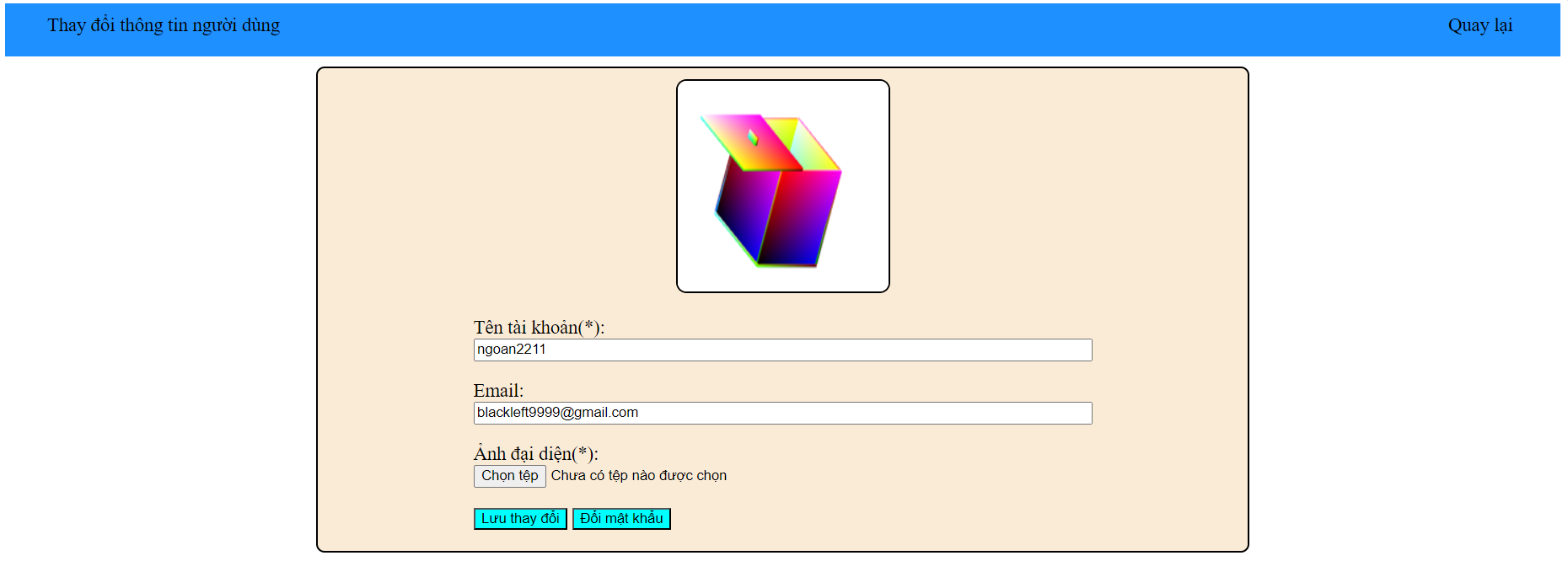
1. Giao diện chọn người cần hỗ trợ

**

1. Giao diện hỗ trợ bên quản trị viên

* Quản trị viên khi hỗ trợ người dùng thì cần ấn vào “Chat” ứng với tài khoản của người dùng để mở giao diện trò chuyện hỗ trợ.
* Khi ở giao diện trò chuyện hỗ trợ, quản trị viên cần nhập vào thanh văn bản phía dưới và ấn “Gửi” để gửi tin cho người dùng cần hỗ trợ.

1. Giao diện chỉnh sửa thông tin tài khoản

****

1. Giao diện chỉnh sửa thông tin tài khoản

* Để chỉnh sửa thông tin tài khoản, người dùng có thể chọn thay đổi tên, email đã lưu từ trước. Nếu muốn đổi ảnh đại diện có thể ấn vào “Chọn tệp”, lúc đó sẽ mở ra hộp thoại để chọn file ảnh làm ảnh đại diện. Sau khi đã chỉnh sửa xong, người dùng cần nhấn “Lưu thay đổi” để lưu các chỉnh sửa của mình.
* Người dùng có thể chọn “Đổi mật khẩu” để chuyển sang chức năng đổi mật khẩu nếu cần.

1. Giao diện tạo dự án

****

1. Giao diện tạo dự án

* Để tạo 1 dự án mới, người dùng cần nhập đầy đủ tên dự án, mô tả cho dự án, ngân sách dùng trong dự án. Rồi ấn “Tạo dự án” để lưu lại.

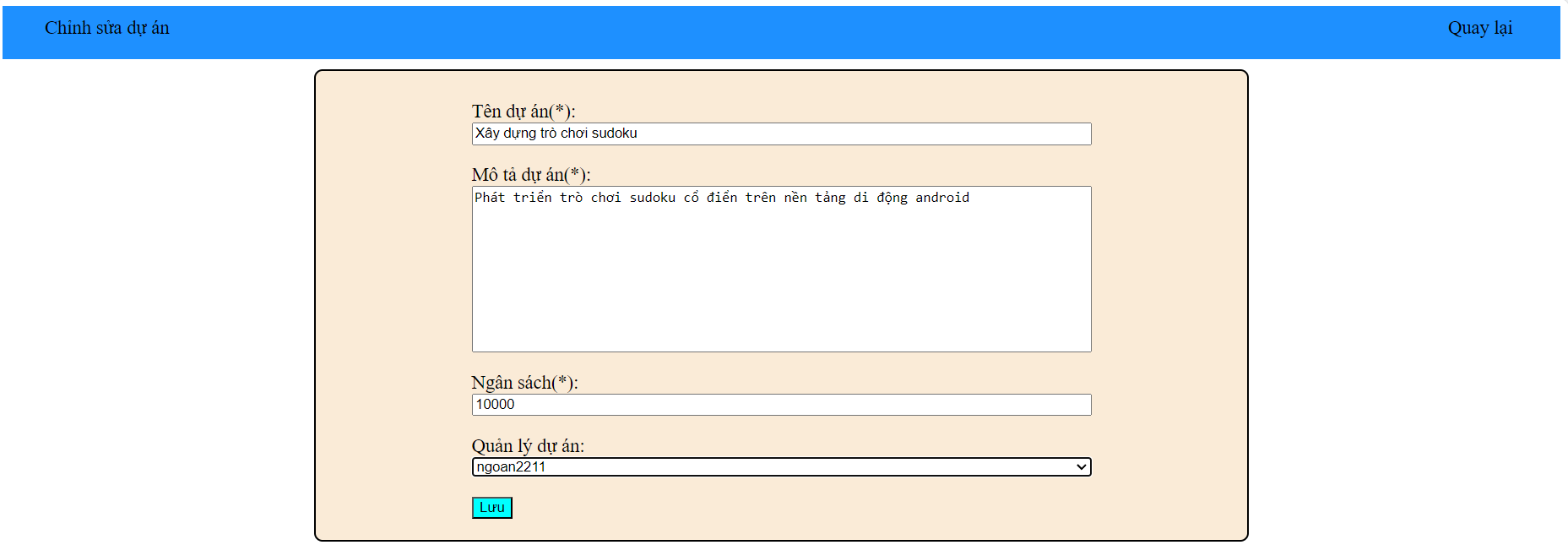
1. Giao diện trang chủ dự án

****

1. Giao diện trang chủ dự án

* Trang chủ dự án bao gồm bảng mô tả dự án, bên dưới là nút để kích hoạt chức năng chỉnh sửa thông tin dự án, khi dự án đủ điều kiện để kết thúc (các nhiệm vụ đạt 100% tiến độ, mỗi nhiệm vụ có ít nhất 1 báo cáo tiến độ, các khách hàng của dự án đồng ý kết thúc dự án) sẽ có thêm nút kích hoạt chức năng kết thúc dự án phía dưới
* Bên dưới còn có bảng thể hiện các nhiệm vụ mà người dùng được phân công và đã đến lịch cần thực hiện
* Bên phải là các lựa chọn để kích hoạt các chức năng: chọn “Lịch trình” để thực hiện chức năng lập lịch, chọn “Nhân viên” để thực hiện chức năng quản lý nhân viên, chọn “Lập báo cáo dự án” để thực hiện chức năng lập báo cáo dự án, chọn “Tiến độ” để thực hiện chức năng xem tiến độ thực hiện, chọn “File chia sẻ” để thực hiện chức năng chia sẻ file, chọn “Nhóm” để đến trang chủ nhóm, chọn “Họp” để thực hiện chức năng họp online.

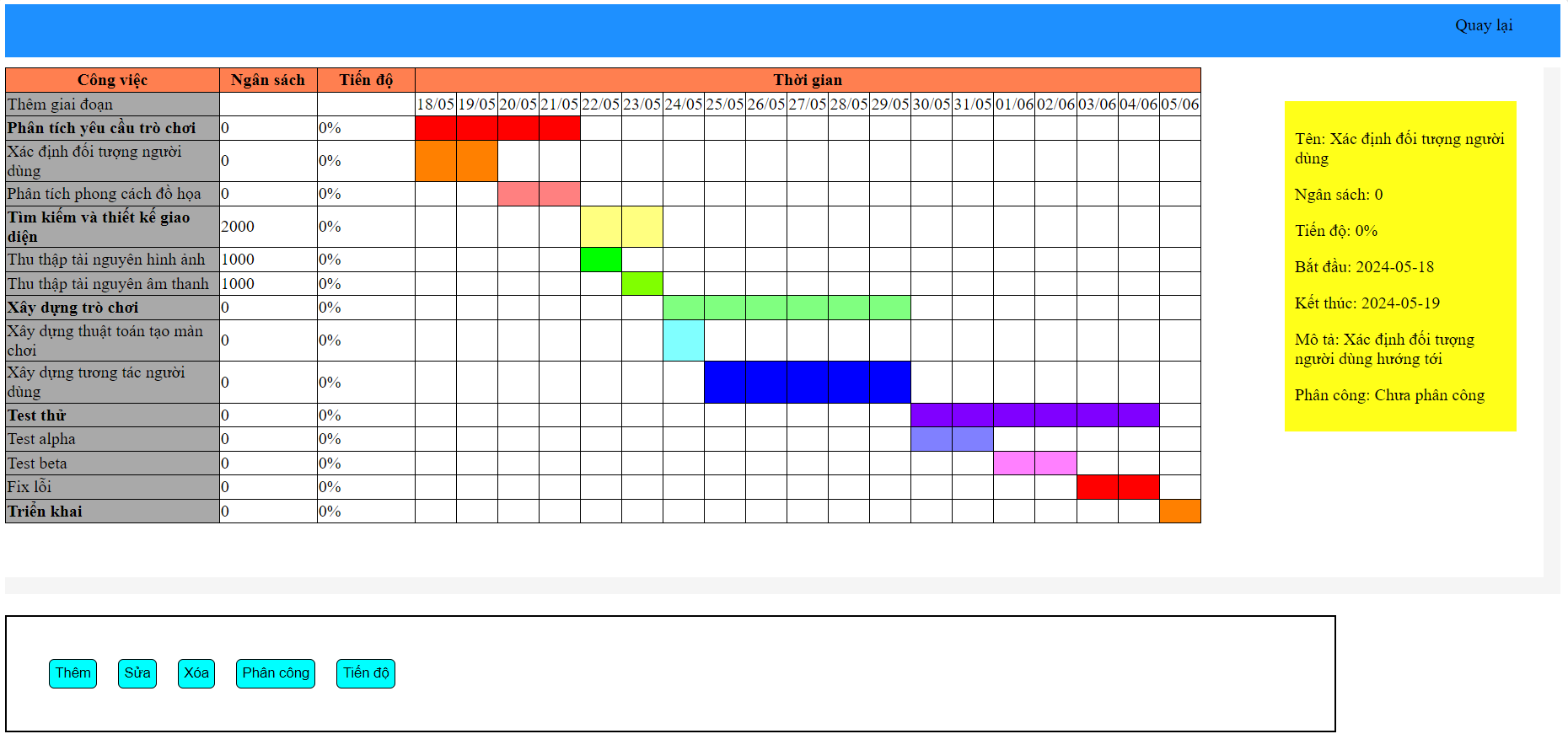
1. Giao diện chỉnh sửa thông tin dự án

****

1. Giao diện chỉnh sửa thông tin dự án

* Để chỉnh sửa thông tin dự án đã tạo, người dùng có thể tùy chọn sửa lại tên dự án, mô tả, ngân sách dùng trong dự án. Nếu người dùng chọn 1 người khác trong quản lý dự án thì người được chọn sẽ được bàn giao cho dự án hiện tại. Rồi ấn “Lưu” để lưu lại thay đổi.

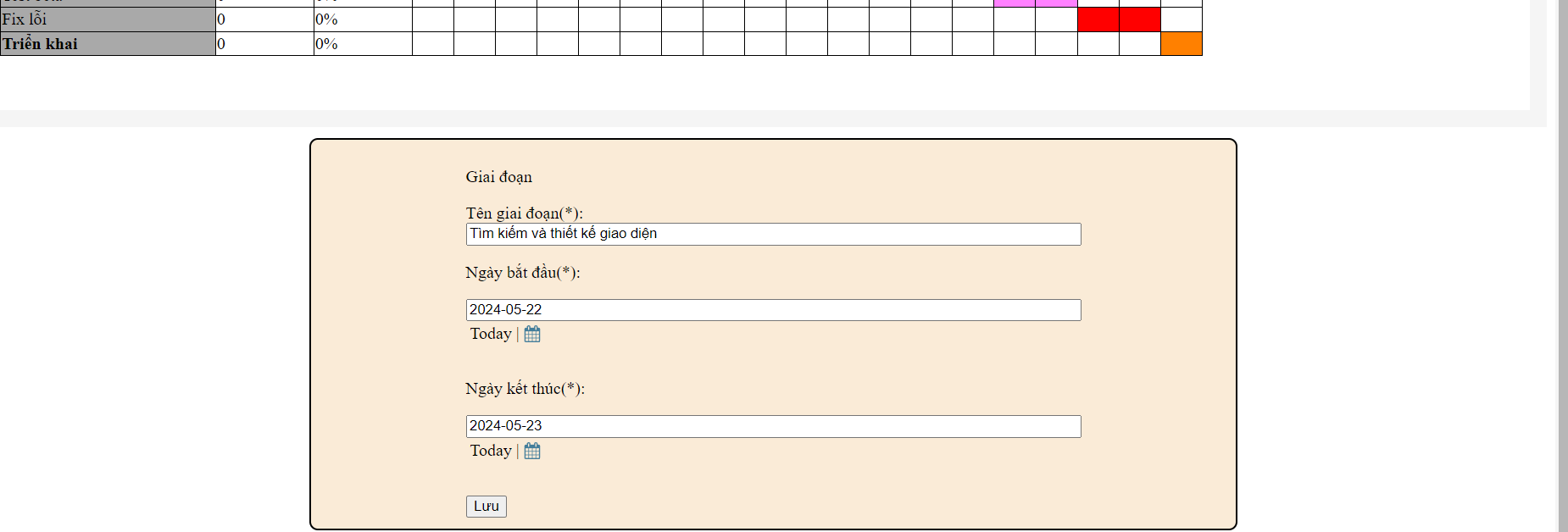
1. Giao diện lập lịch

****

1. Giao diện lập lịch

* Khi người dùng chọn vào 1 dòng ứng với công việc giai đoạn hoặc nhiệm vụ cụ thể thì sẽ có 1 hộp thông tin màu vàng hiển thị ở góc phải, hiển thị thông tin công việc đó. Khi này người dùng có thể ấn các nút “Thêm”, “Sửa”, “Xóa”, “Phân công”, “Duyệt”, “Báo cáo” để hiện chức năng tương ứng với công việc được chọn. Với mỗi dạng công việc khác nhau và người dùng khác nhau mà các nút chức năng sẽ hiển thị khác nhau.

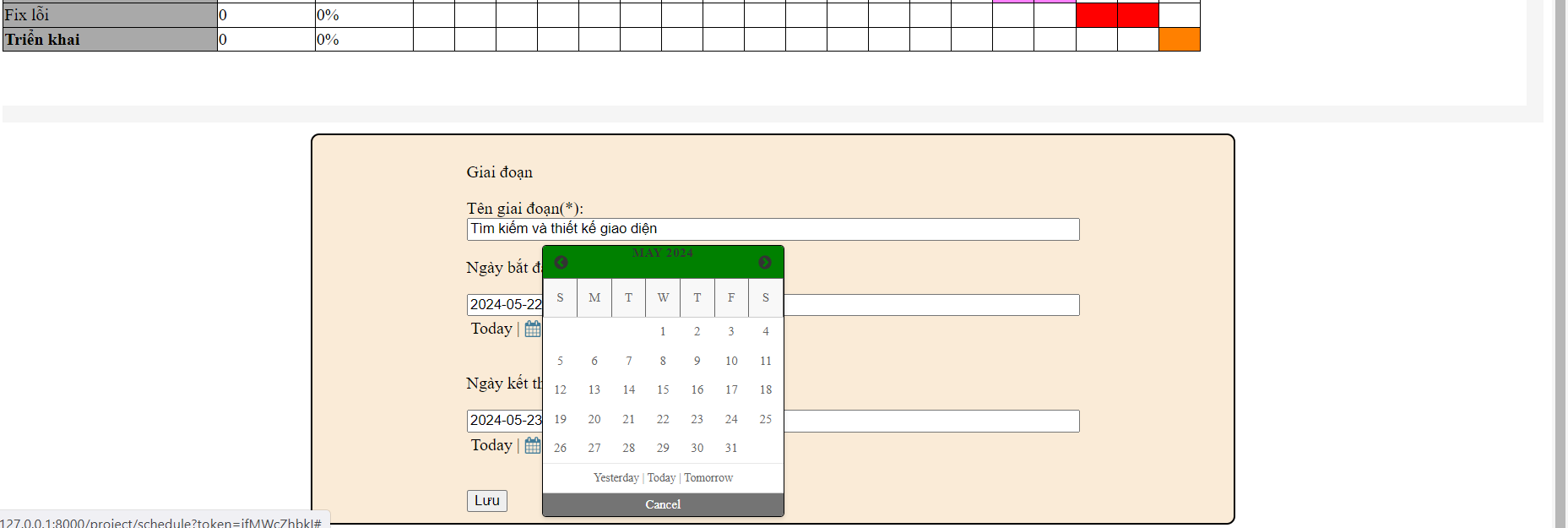
1. Giao diện thêm giai đoạn

****

1. Giao diện thêm giai đoạn

* Kích hoạt khi người dùng ấn vào “Thêm giai đoạn” trên bảng lập lịch hoặc là chọn 1 giai đoạn rồi ấn “Thêm”.
* Để thêm giai đoạn, người dùng cần nhập đầy đủ tên giai đoạn, ngày bắt đầu, ngày kết thúc. Rồi ấn “Lưu” để lưu giai đoạn đã tạo.
* Nếu người dùng tạo giai đoạn bằng nút “Thêm giai đoạn” thì giai đoạn mới sẽ có thứ tự đầu tiên từ trên xuống trong bảng còn nếu làm bằng cách thứ 2 thì giai đoạn mới sẽ nằm ở vị trí ngay sau giai đoạn đã chọn lúc trước.

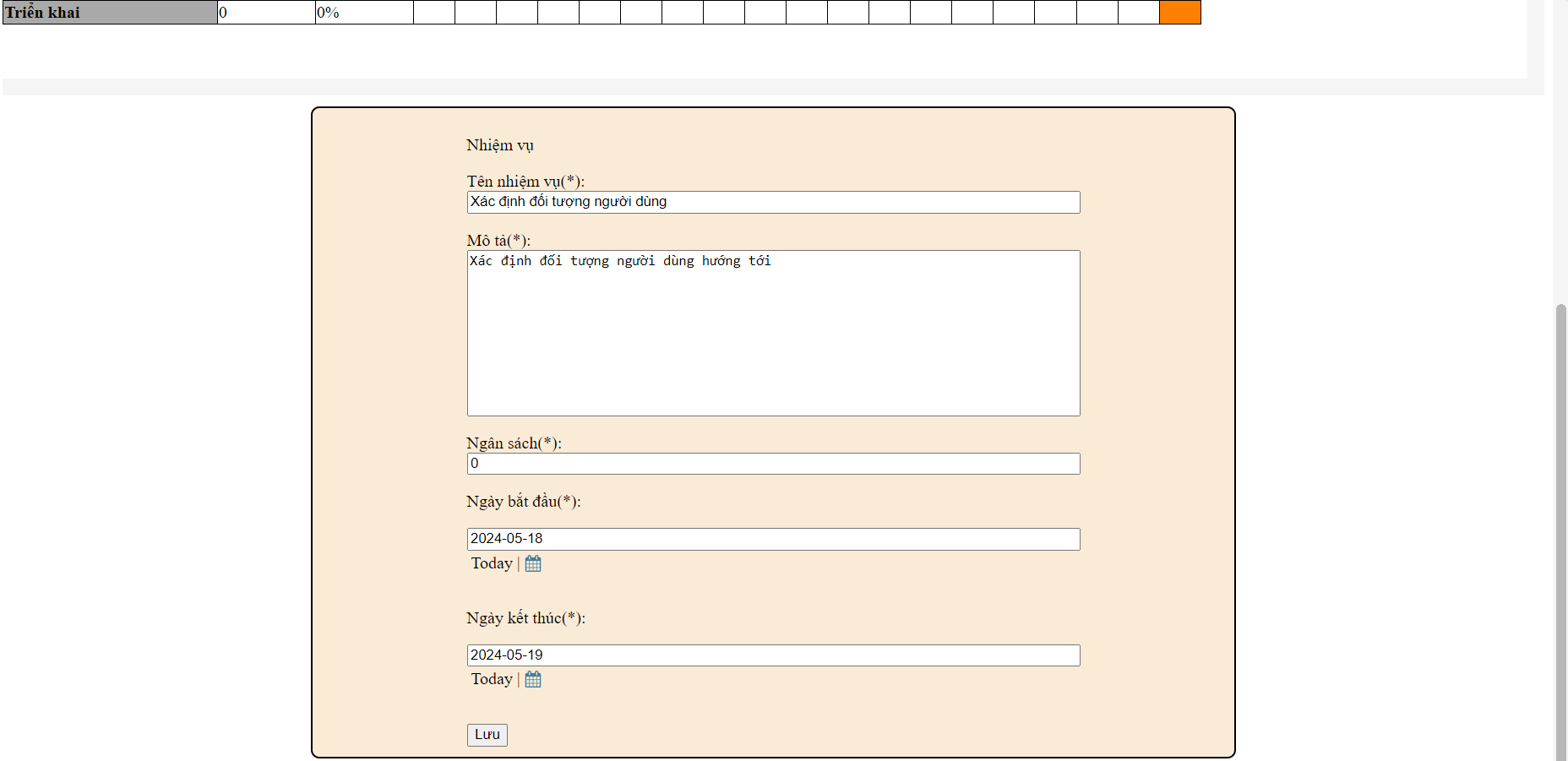
1. Giao diện sửa giai đoạn

****

1. Giao diện sửa giai đoạn

* Để sửa 1 giai đoạn đã tạo, người dùng cần chọn 1 giai đoạn trên bảng rồi ấn vào nút “Sửa”
* Khi này người dùng có thể lựa chọn thay đổi lại tên giai đoạn, ngày bắt đầu, ngày kết thúc. Rồi ấn “Lưu” để lưu thay đổi.

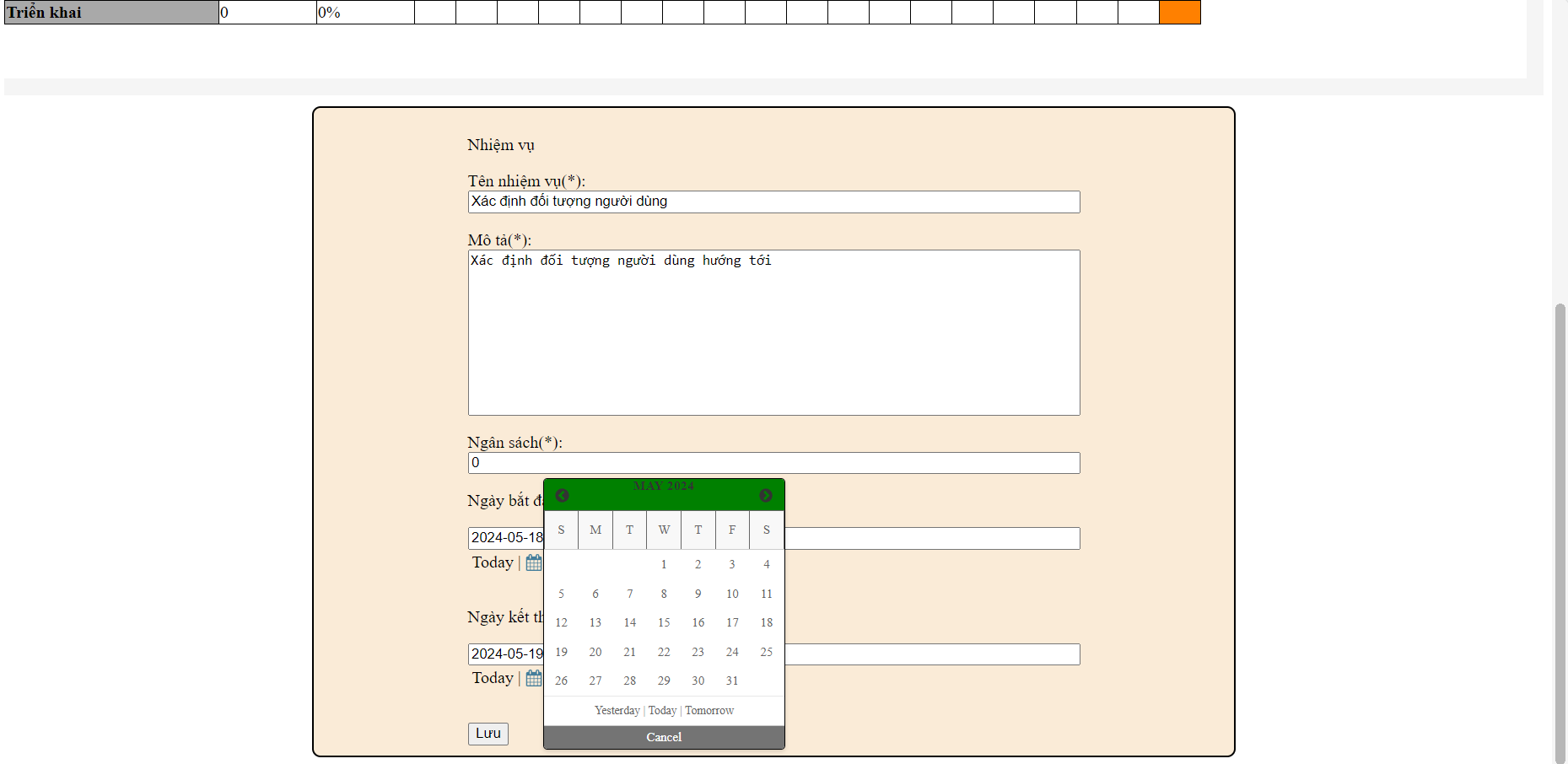
1. Giao diện thêm nhiệm vụ

****

1. Giao diện thêm nhiệm vụ

* Kích hoạt bằng cách người dùng chọn 1 giai đoạn rồi ấn “Thêm nhiệm vụ” hoặc chọn 1 nhiệm vụ đã có rồi ấn “Thêm”
* Để tạo nhiệm vụ mới, người dùng cần nhập đầy đủ tên nhiệm vụ, mô tả, ngân sách dùng, ngày bắt đầu, ngày kết thúc. Rồi ấn “Lưu” để lưu nhiệm vụ đã tạo.
* Nếu nhiệm vụ mới được tạo thông qua cách chọn giai đoạn thì nhiệm vụ đó sẽ có thứ tự đầu tiên trong giai đoạn đó. Còn nếu được tạo thông qua cách chọn nhiệm vụ thì nhiệm vụ mới sẽ có thứ tự ngay sau nhiệm vụ đã chọn, trong cùng 1 giai đoạn.

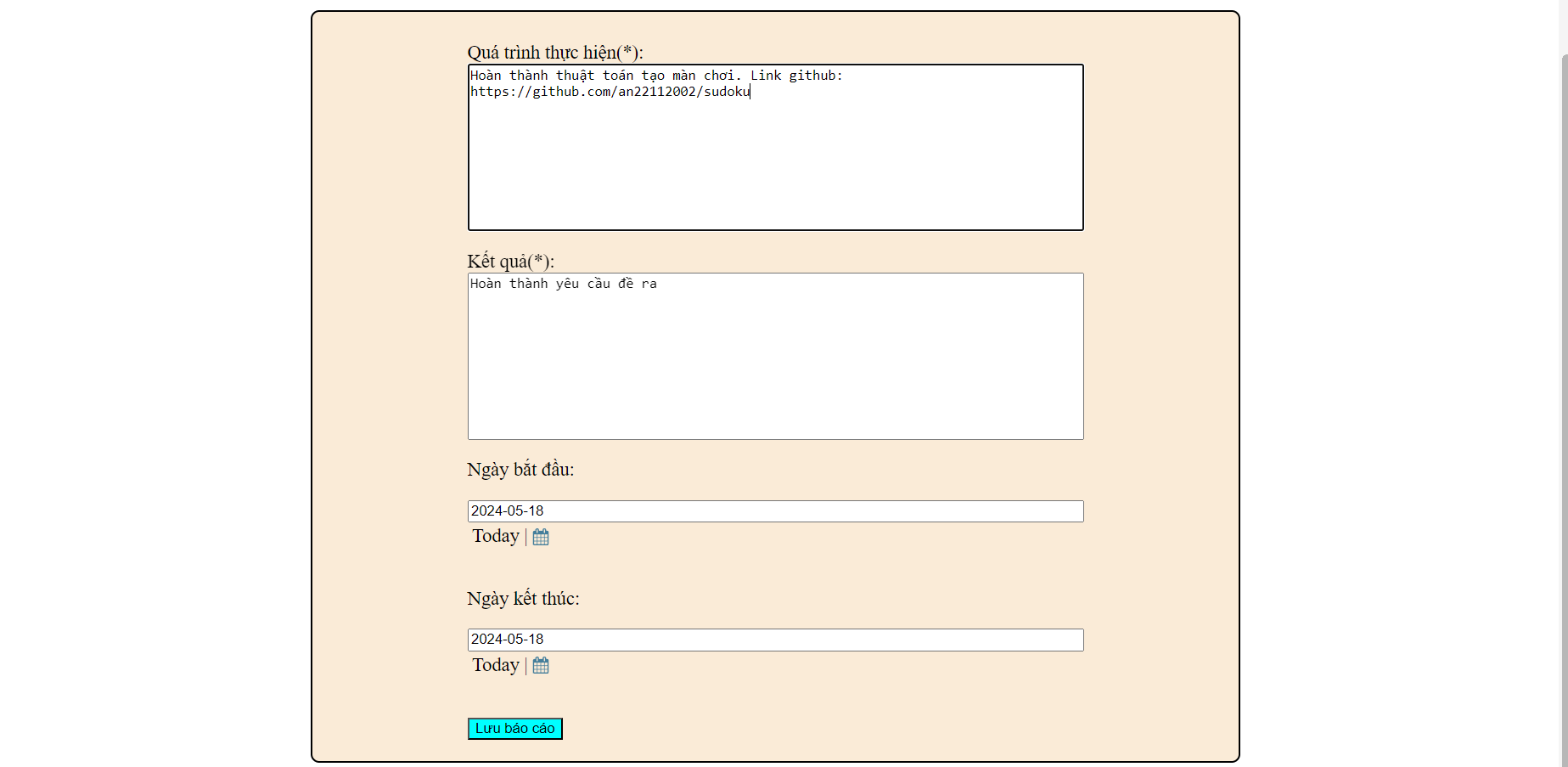
1. Giao diện sửa nhiệm vụ

****

1. Giao diện sửa nhiệm vụ

* Để sửa 1 nhiệm vụ đã tạo, người dùng cần chọn vào nhiệm vụ cần sửa rồi ấn “Sửa”
* Khi này người dùng có thể tùy chọn sửa lại tên nhiệm vụ, mô tả, ngân sách dùng, ngày bắt đầu, ngày kết thúc. Rồi ấn “Lưu” để lưu lại các thay đổi.

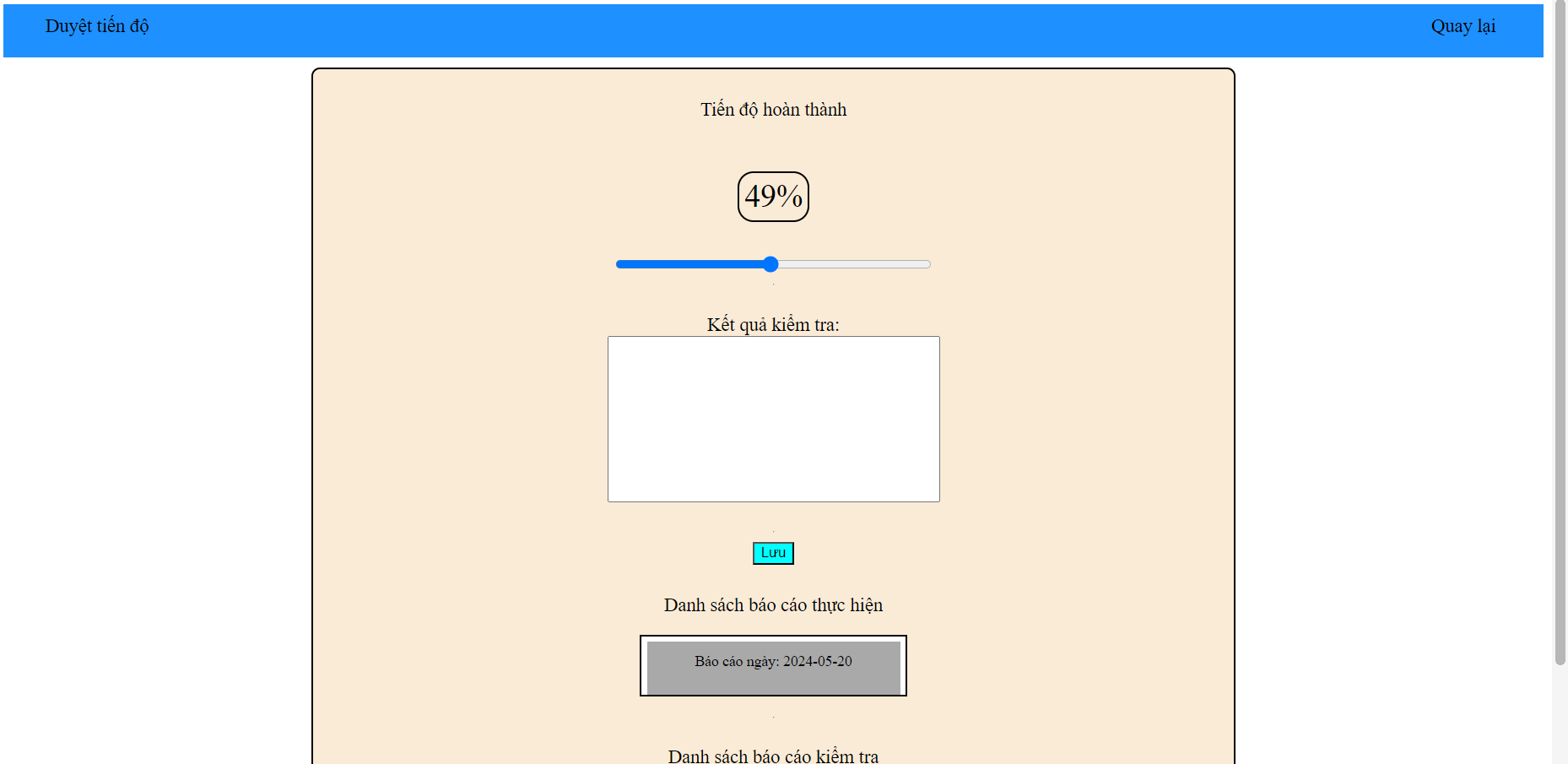
1. Giao diện lập báo cáo tiến độ

****

1. Giao diện lập báo cáo tiến độ

* Để lập báo cáo tiến độ, người dùng cần nhập đầy đủ quá trình thực hiện, kết quả cuối cùng, ngày bắt đầu thực hiện thực tế, ngày kết thúc thực tế. Rồi ấn “Lưu báo cáo” để lưu lại.

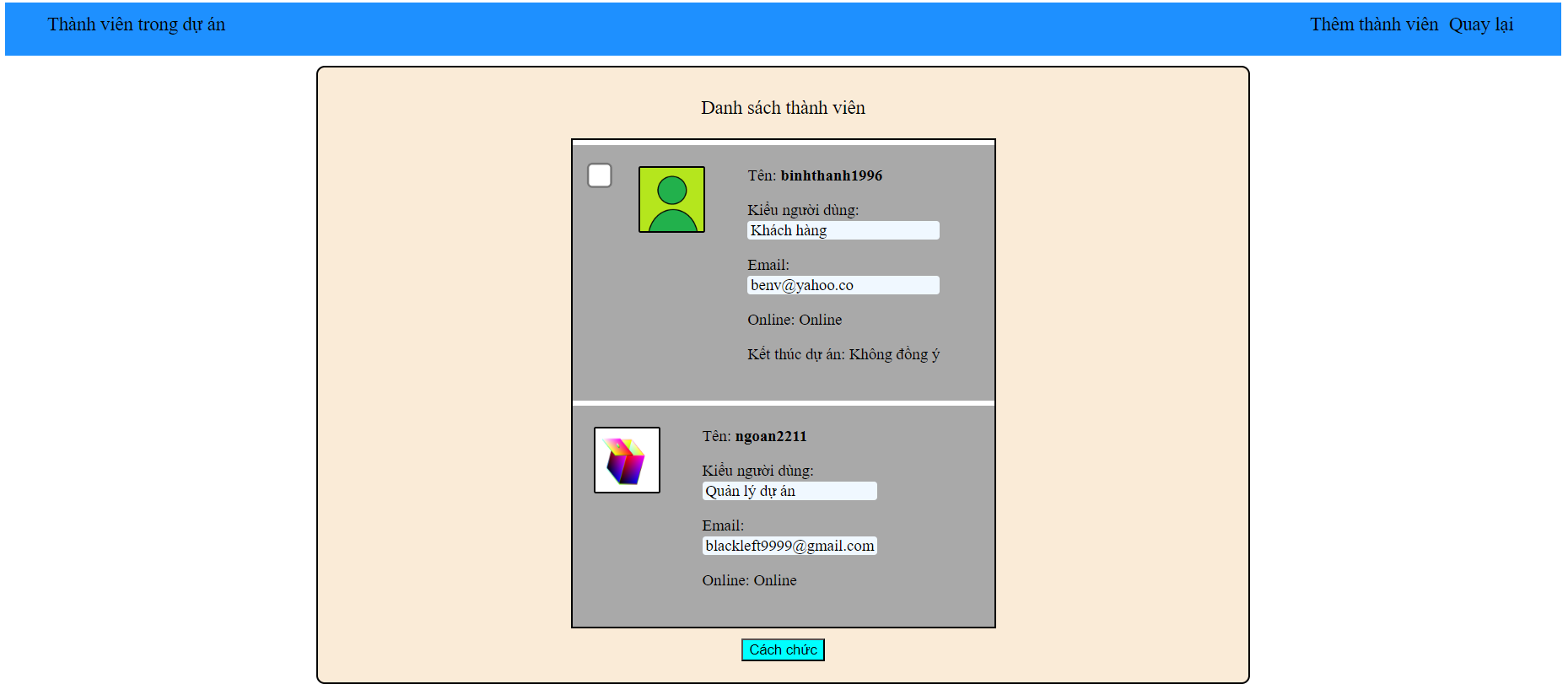
1. Giao diện duyệt tiến độ

****

1. Giao diện duyệt tiến độ

* Để duyệt báo cáo tiến độ, người dùng có thể kéo thả thanh trượt để thay đổi tiến độ hoàn thành, giá trị tùy thuộc vào đánh giá cá nhân của người duyệt dựa trên các báo cáo tiến độ nhiệm vụ và các báo cáo kiểm tra trước, ngoài ra người dùng cần nhập kết quả kiểm tra của mình đối với công việc của người làm. Sau đó người dùng ấn “Lưu” sẽ tạo ra 1 báo cáo kiểm tra mới và thay đổi tiến độ hoàn thành của nhiệm vụ.

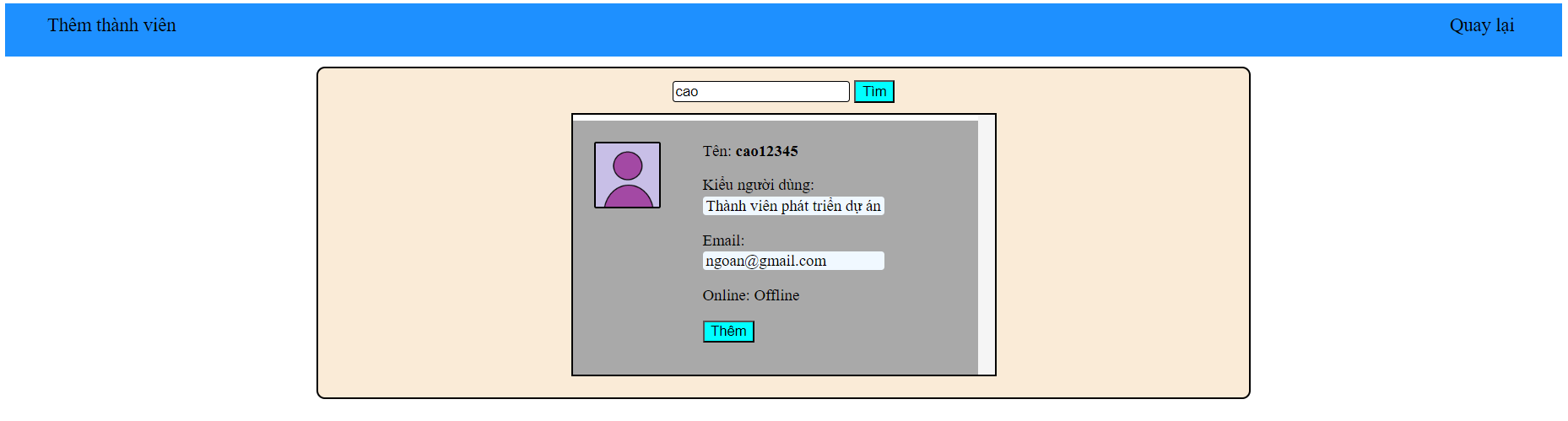
1. Giao diện quản lý nhân viên dự án

****

1. Giao diện quản lý nhân viên

* Để thêm thành viên mới vào dự án, người dùng cần ấn vào “Thêm thành viên”
* Để cách chức thành viên ra khỏi dự án, người dùng cần chọn vào người dùng đó bằng cách tích vào check box người dùng tương ứng rồi ấn vào “Cách chức” để cách chức người dùng đó.
* Không thể cách chức người đang là quản lý dự án.

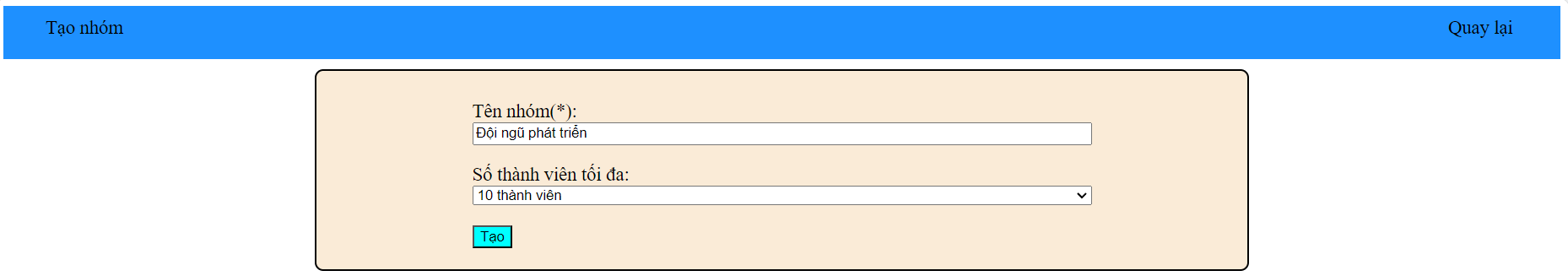
1. Giao diện thêm nhân viên dự án

****

1. Giao diện thêm nhân viên

* Để thêm nhân viên vào dự án, người dùng cần tìm người dùng đó bằng cách nhập tên hoặc email người đó (không cần nhập đầy đủ nhưng sẽ tìm chính xác hơn khi nhập đầy đủ) vào thanh tìm kiếm và ấn “Tìm”. Danh sách người dùng phù hợp sẽ hiển thị bên dưới. Khi tìm thấy người cần thêm thì ấn vào nút “Thêm” của người đó để thêm người dùng vào dự án.

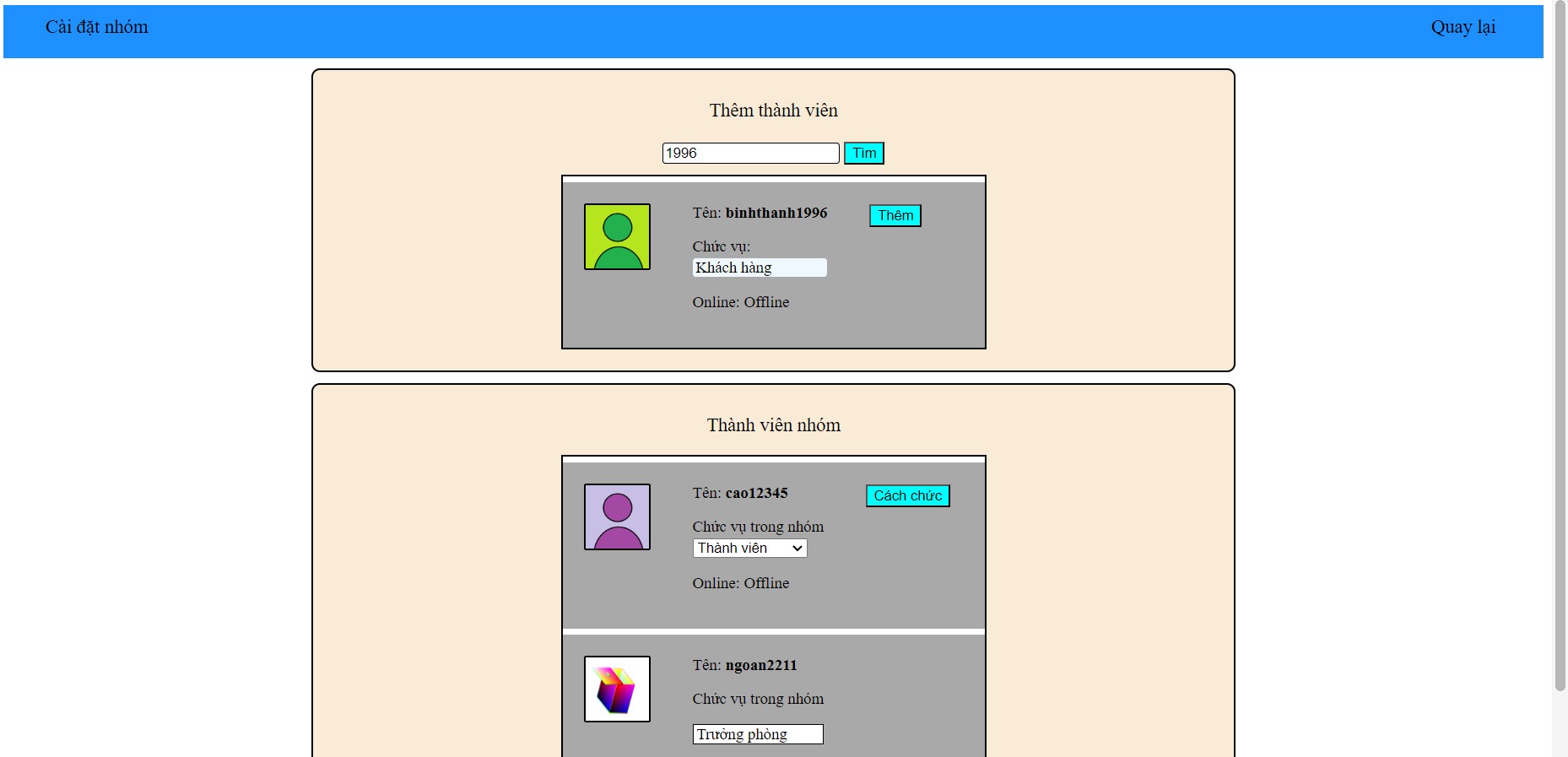
1. Giao diện tạo nhóm

****

1. Giao diện tạo nhóm

* Để tạo nhóm mới, người dùng cần nhập đầy đủ tên nhóm và chọn số lượng thành viên tối đa. Rồi ấn “Tạo” để tạo nhóm.

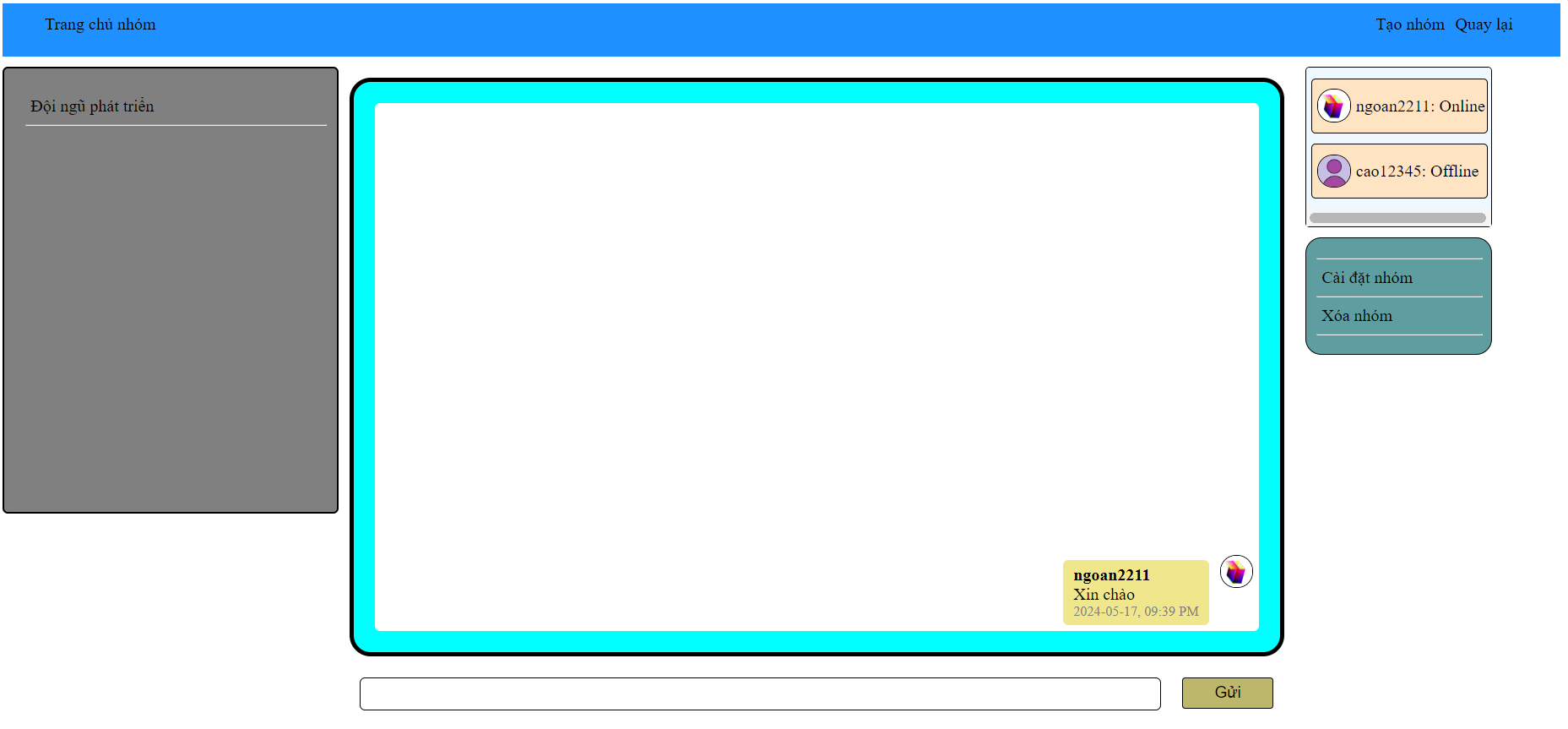
1. Giao diện quản lý nhóm

****

1. Giao diện quản lý nhóm

* Người dùng có thể nhập vào thanh tìm kiếm để tìm các thành viên dự án. Ấn “Thêm” nếu muốn thêm thành viên đó vào nhóm hiện tại.
* Nếu muốn nâng thành viên lên trưởng phòng hoặc thành viên thì chỉnh sửa chức vụ trong nhóm của người đó.
* Nếu muốn cách chức 1 người thì chỉ cần ấn vào “Cách chức” của người đó. Người đó sẽ không còn là thành viên của nhóm nữa.

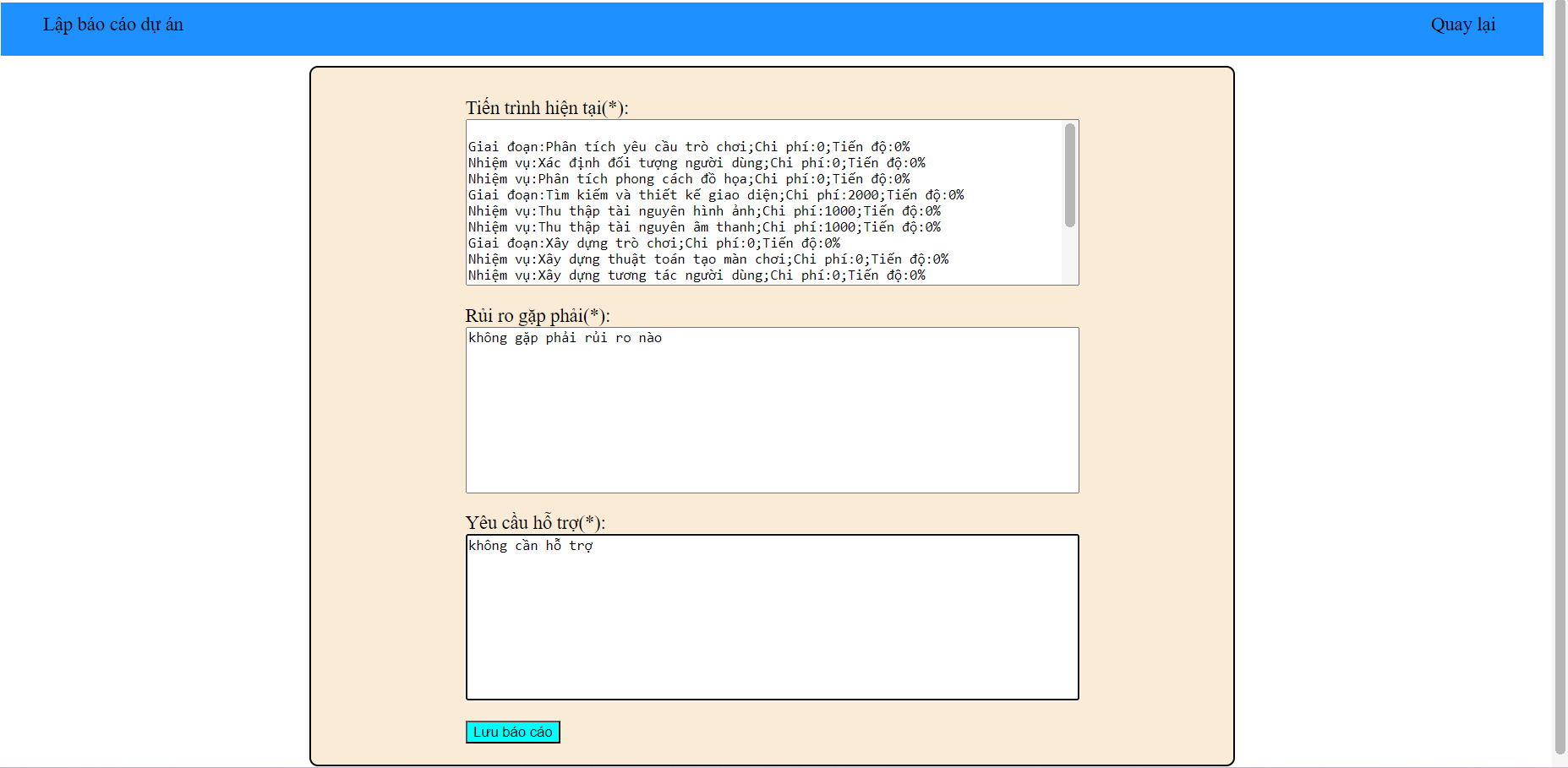
1. Giao diện trang chủ nhóm

****

1. Giao diện trang chủ nhóm

* Ở trang chủ nhóm, người dùng cần chọn 1 nhóm để trò chuyện trong danh sách nhóm, phía bên trái. Sau khi chọn 1 nhóm cụ thể, người dùng có thể nhập tin nhắn vào thanh tin nhắn rồi ấn “Gửi” để gửi tin đi trong nhóm.
* Nếu người dùng là trưởng nhóm thì sau khi chọn nhóm sẽ có thêm danh sách các chức năng có thể kích hoạt gồm “Cài đặt nhóm” và “Xóa nhóm”.
* Khi cần tạo nhóm mới thì cần chọn vào “Tạo nhóm” sẽ kích hoạt chức năng tạo nhóm.

1. Giao diện lập báo cáo dự án



1. Giao diện lập báo cáo dự án

* Để tạo báo cáo dự án, người dùng cần nhập đầy đủ tiến trình hiện tại, rủi ro đang gặp phải, yêu cầu hỗ trợ. Rồi ấn “Lưu báo cáo” để lưu lại.

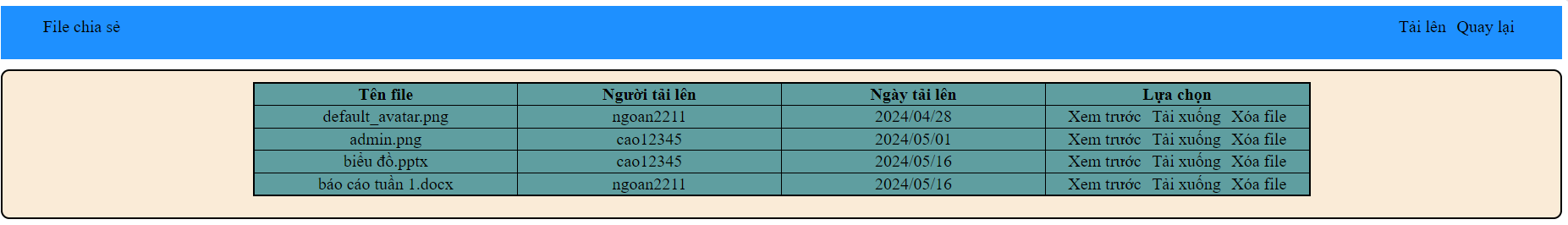
1. Giao diện theo dõi tiến độ

****

1. Giao diện theo dõi tiến độ

* Ở đây người dùng có thể xem thông tin cụ thể tiến độ hiện tại của dự án

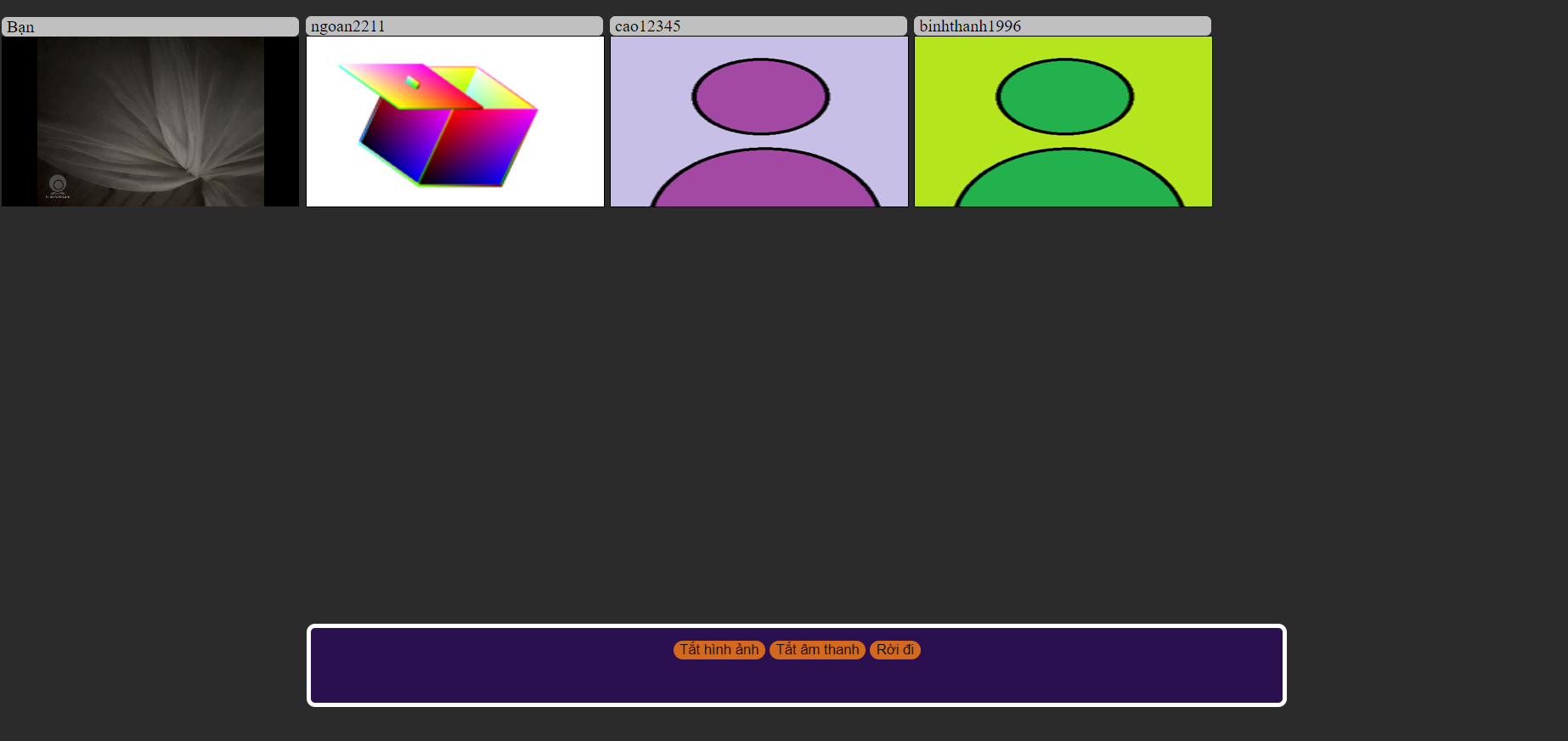
1. Giao diện chia sẻ file

****

1. Giao diện chia sẻ file

* Ở trang chia sẻ file sẽ hiển thị danh sách file đang được chia sẻ trong dự án.
* Khi người dùng nhấn “Tải lên” sẽ mở hộp thoại để người dùng chọn file tải lên. Giới hạn kích thước file có thể tải lên là 5MB
* Nếu người dùng chọn “Tải xuống” người dùng sẽ tải xuống file tương ứng.
* Người dùng có thể chọn “Xem trước” để xem file đó ở định dạng pdf, lưu ý chỉ hoạt động với các file có định dạng docx, file hình ảnh và file pdf.
* Nếu người dùng là quản lý dự án thì sẽ có thêm lựa chọn “Xóa file”, khi chọn “Xóa file” người dùng sẽ xóa file tương ứng.

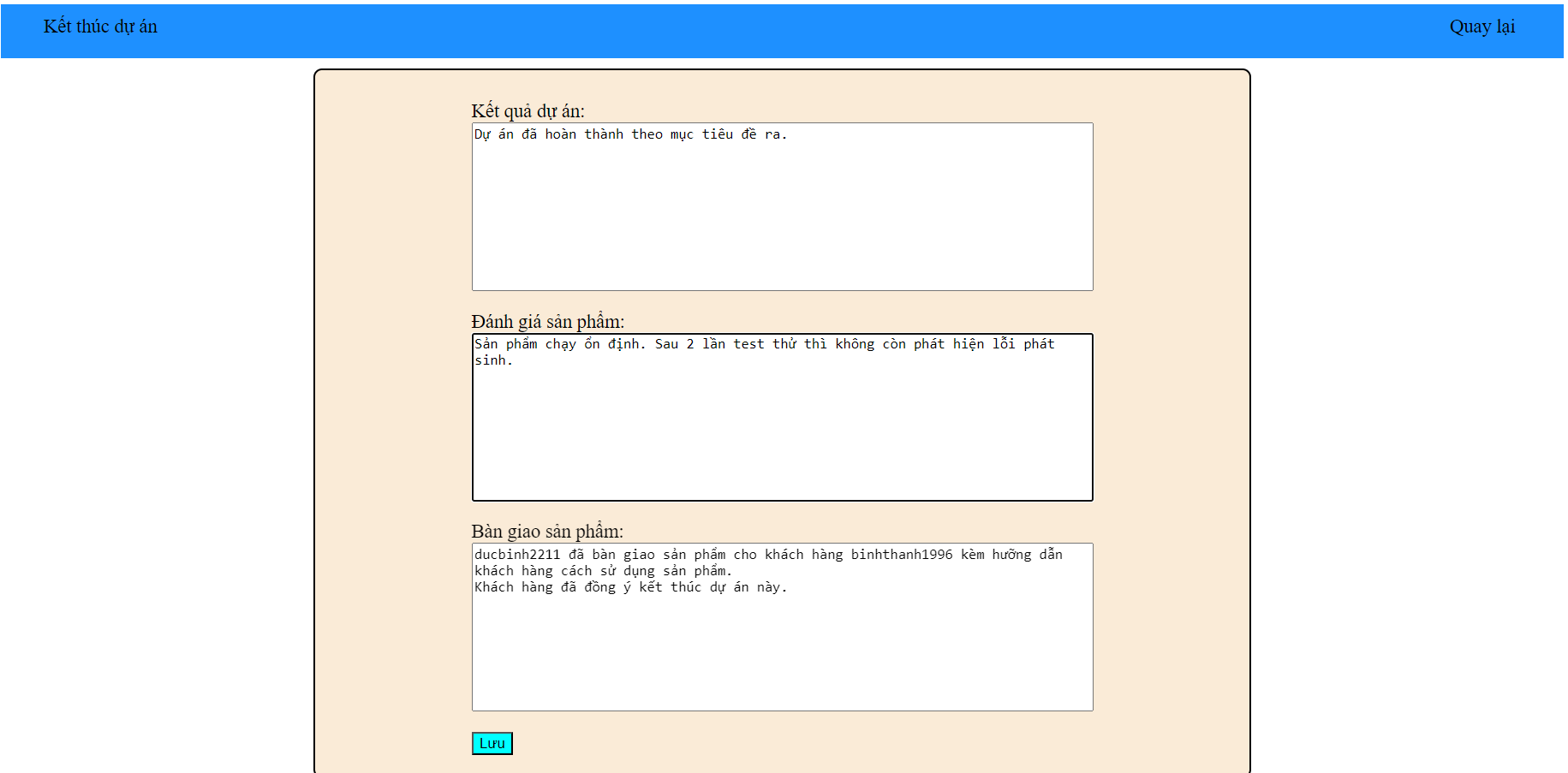
1. Giao diện họp video

****

1. Giao diện họp video

* Khi người dùng nhấn vào họp, nếu người dùng là quản lý dự án thì sẽ tạo 1 phòng mới. Nếu không sẽ tìm xem có cuộc gọi nào ứng với dự án của hiện tại không để tham gia.
* Khi tham gia thành công người dùng có thể tắt/bật hình ảnh bản thân, tắt/bật âm thanh bản thân hoặc rời đi. Nếu quản lý dự án rời đi thì phòng họp kết thúc và tất cả mọi người đều thoát.

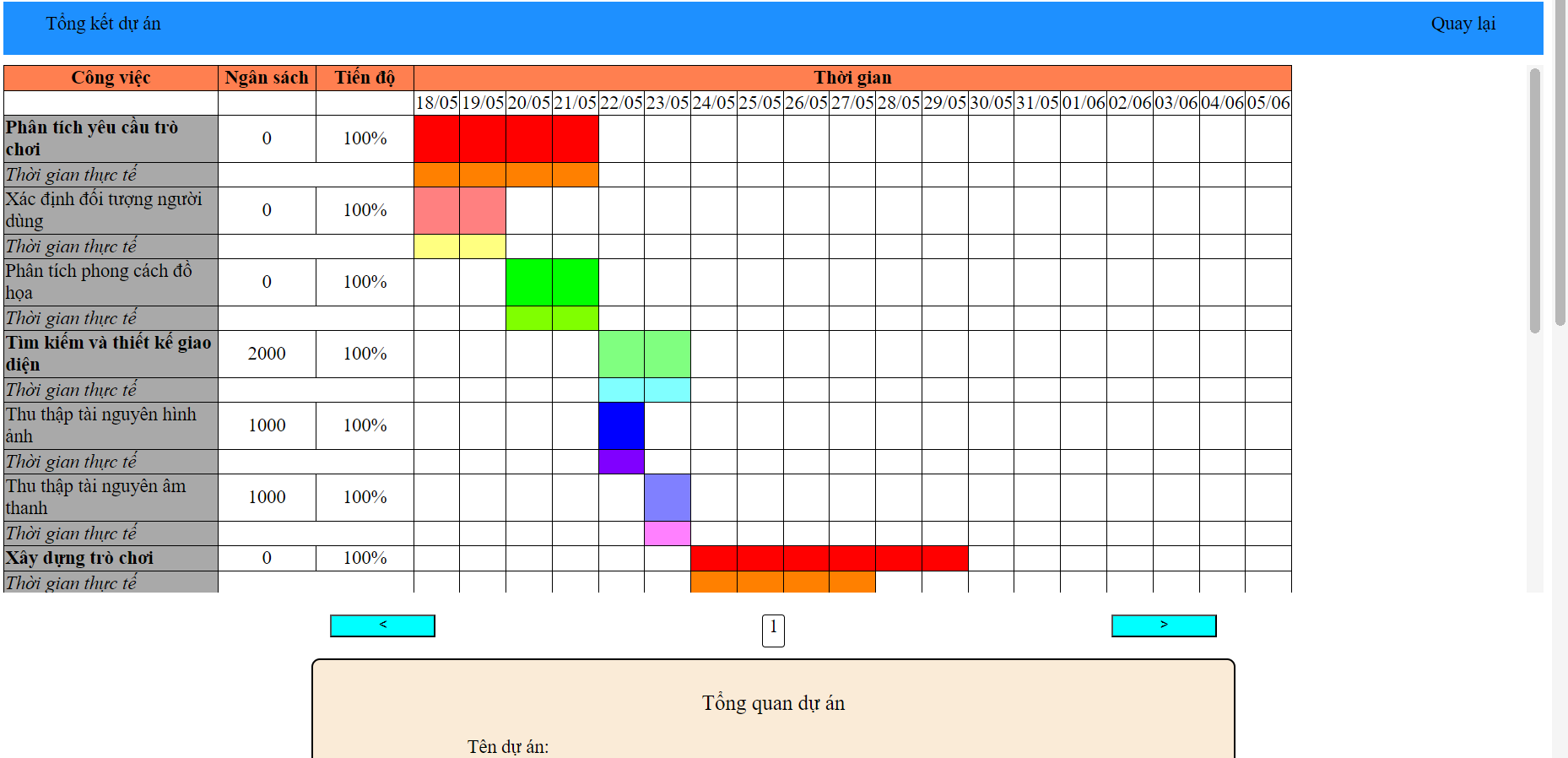
1. Giao diện kết thúc dự án

****

1. Giao diện kết thúc dự án

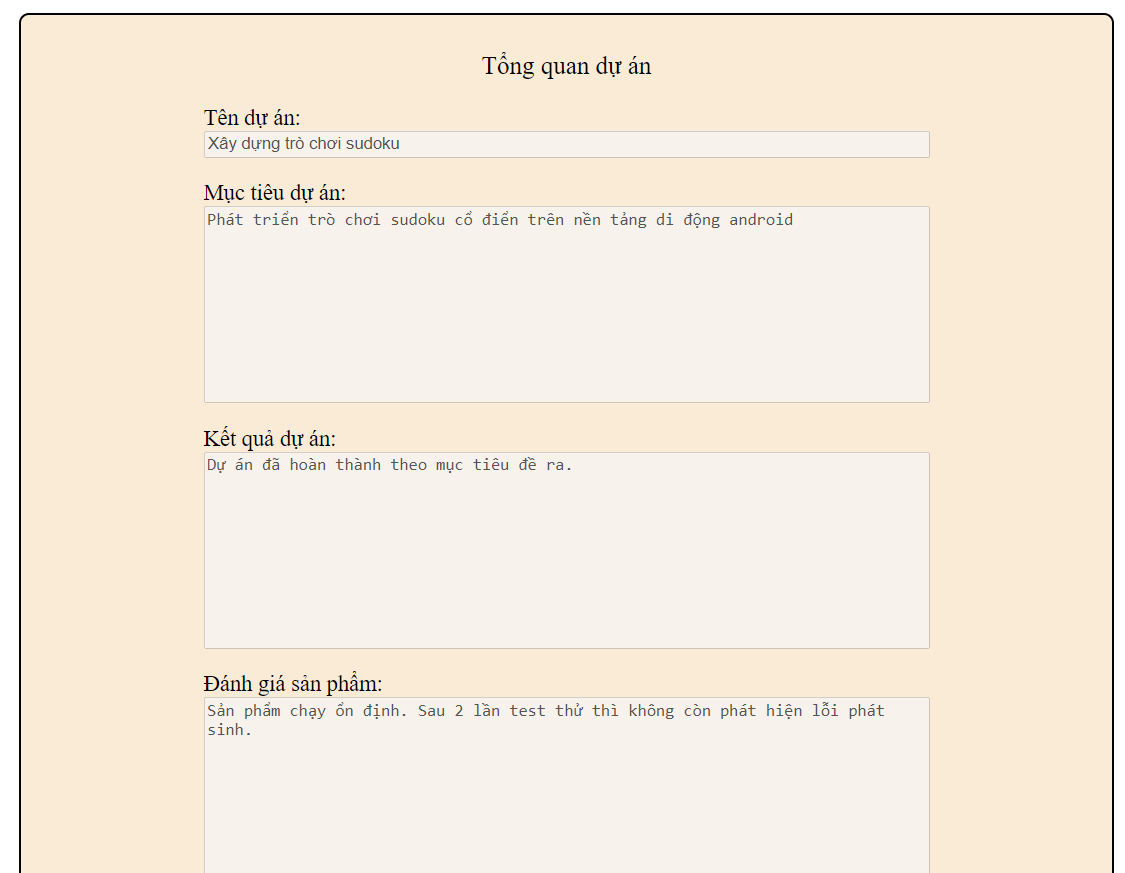
* Để kết thúc dự án người dùng cần nhập vào đầy đủ: Kết quả dự án, đánh giá sản phẩm của dự án, kết quả bàn giao sản phẩm. Rồi ấn nút “Lưu”.

1. Giao diện tổng kết dự án

****

1. Giao diện tổng kết dự án

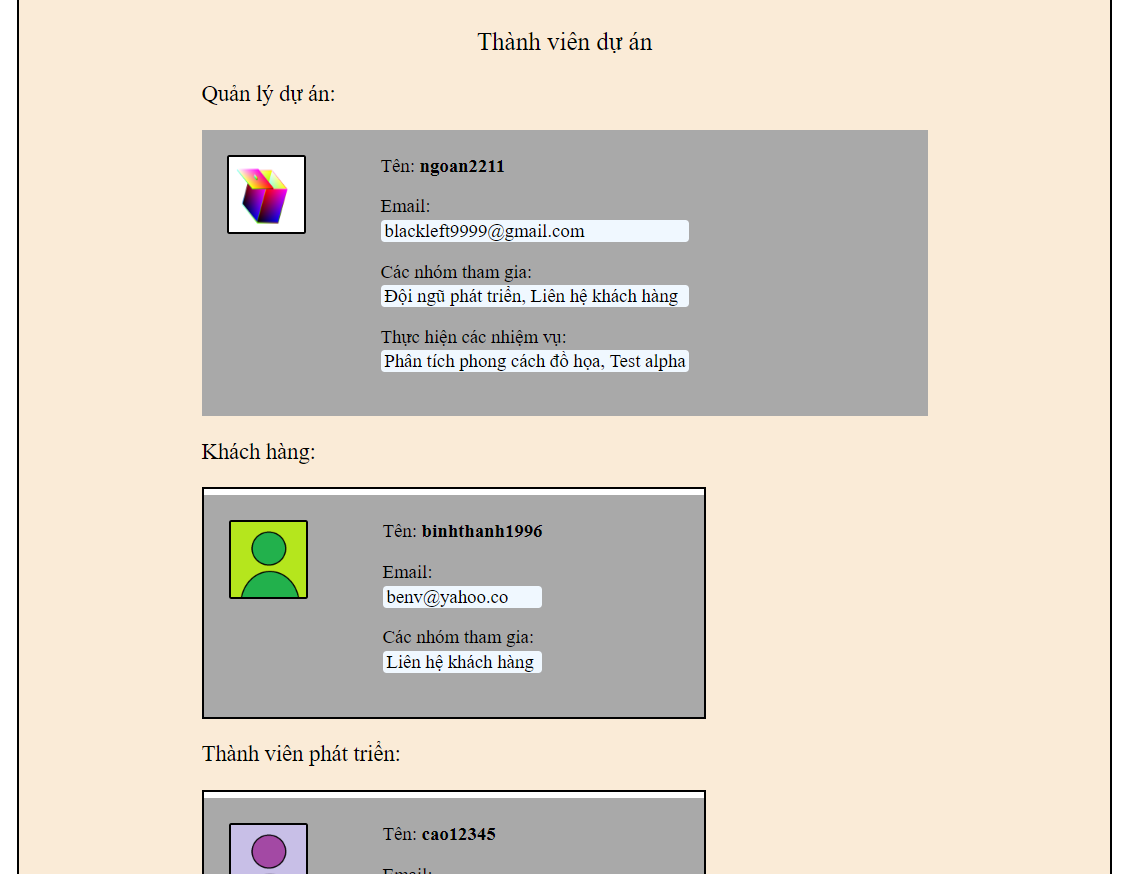
* Giao diện gồm bảng thời gian thể hiện thời gian các giai đoạn, nhiệm vụ được thực hiện theo kế hoạch và thời gian thực hiện chúng trong thực tế dựa theo các báo cáo thực hiện được tạo.
* Bên dưới gồm 4 mục: tổng quan dự án, thành viên dự án, các bản báo cáo, file chia sẻ trong dự án.

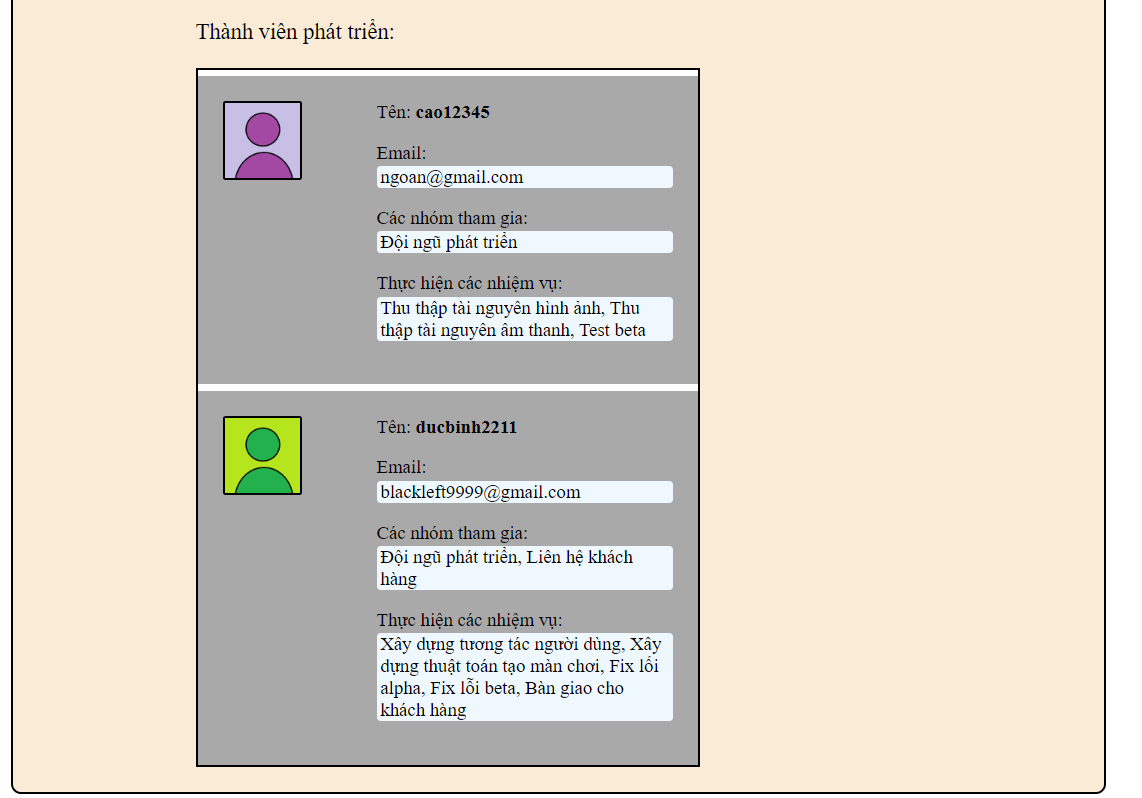




1. Mục tổng quan dự án

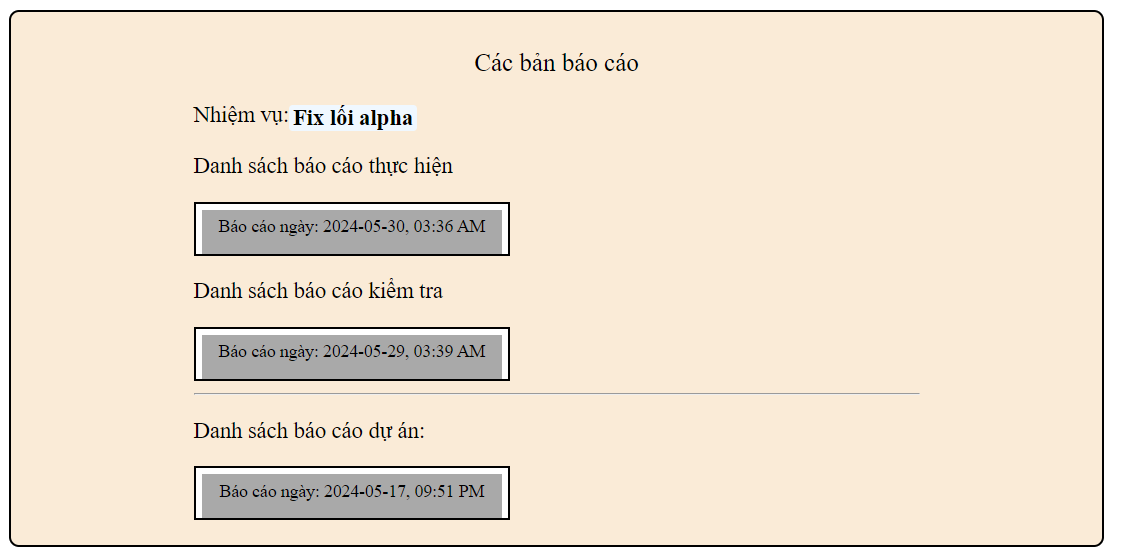
* Gồm các thông tin dự án: tên dự án, mục tiêu đề ra, kết quả dự án, đánh giá sản phẩm, kết quả bàn giao, ngân sách sử dụng trong dự án.





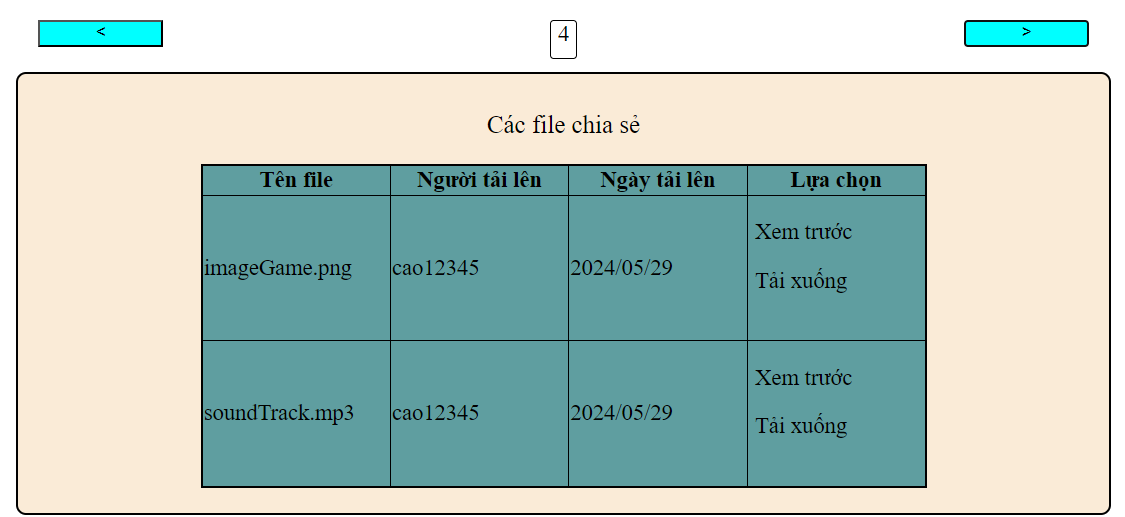
1. Mục thành viên dự án

* Gồm các nhân viên đã tham gia trong quá trình phát triển dự án.



1. Mục các bản báo cáo

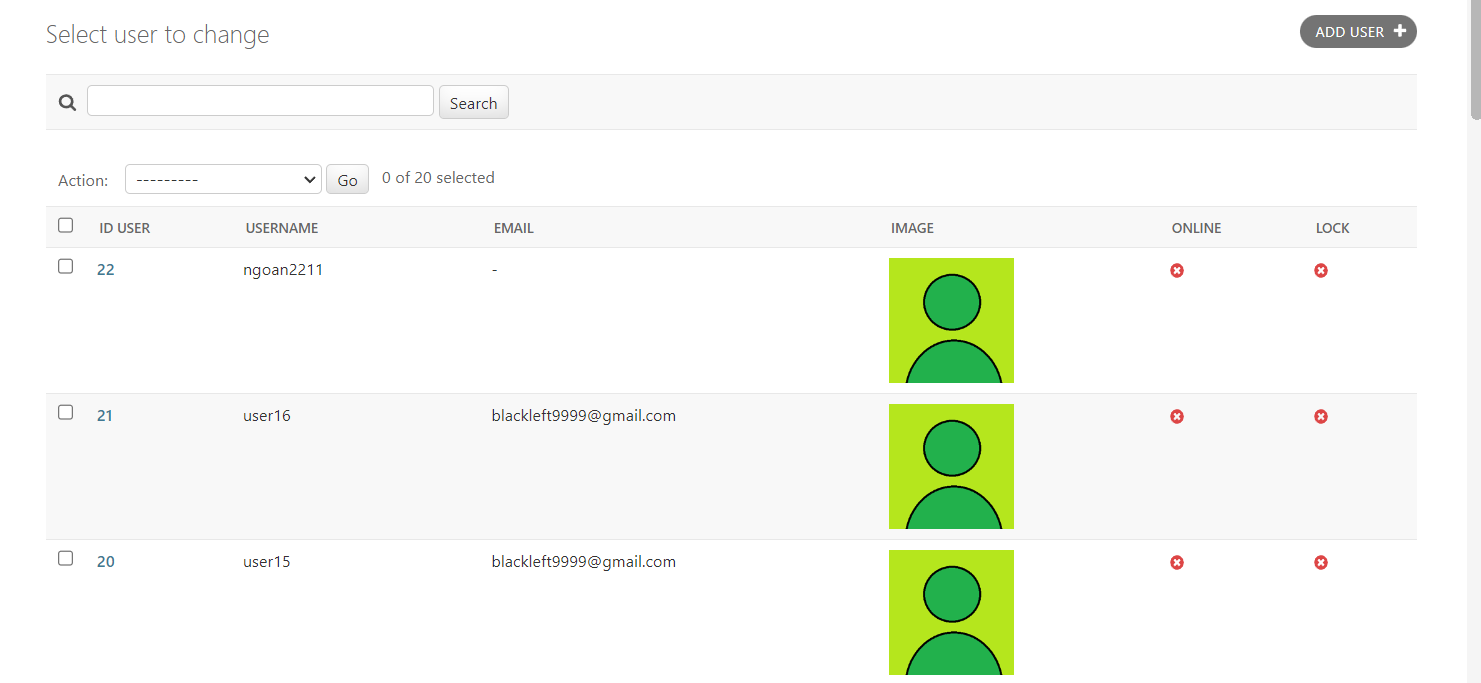
* Gồm các báo cáo dự án được tạo bởi quản lý dự án. Khi người dùng chọn 1 nhiệm vụ cụ thể trên bảng thời gian sẽ hiển thị thêm các báo cáo thực hiện và báo cáo kiểm tra nhiệm vụ tương ứng.

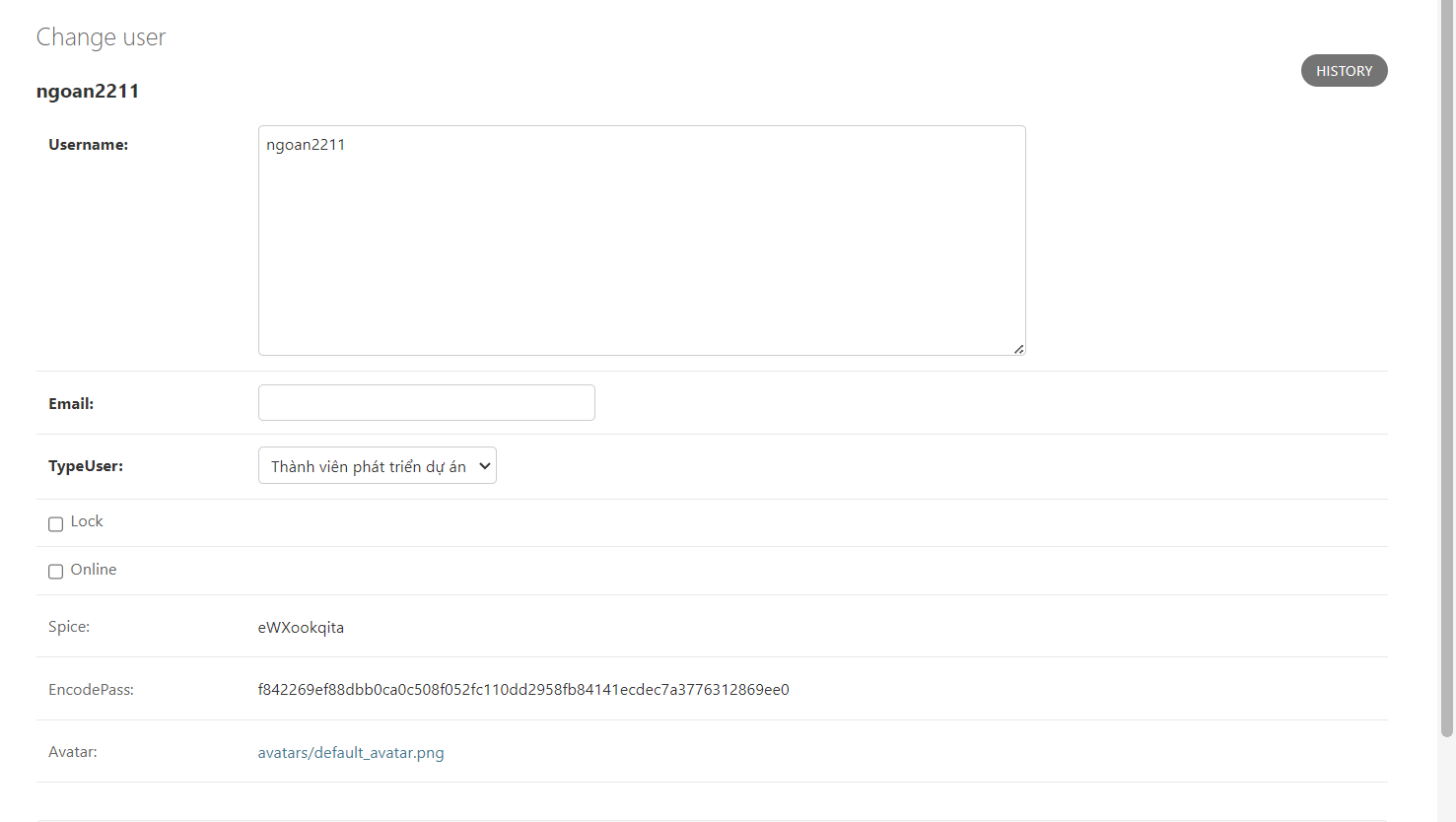


1. Mục chia sẻ file

* Gồm các file được chia sẻ trong dự án

1. Giao diện bảo trì người dùng

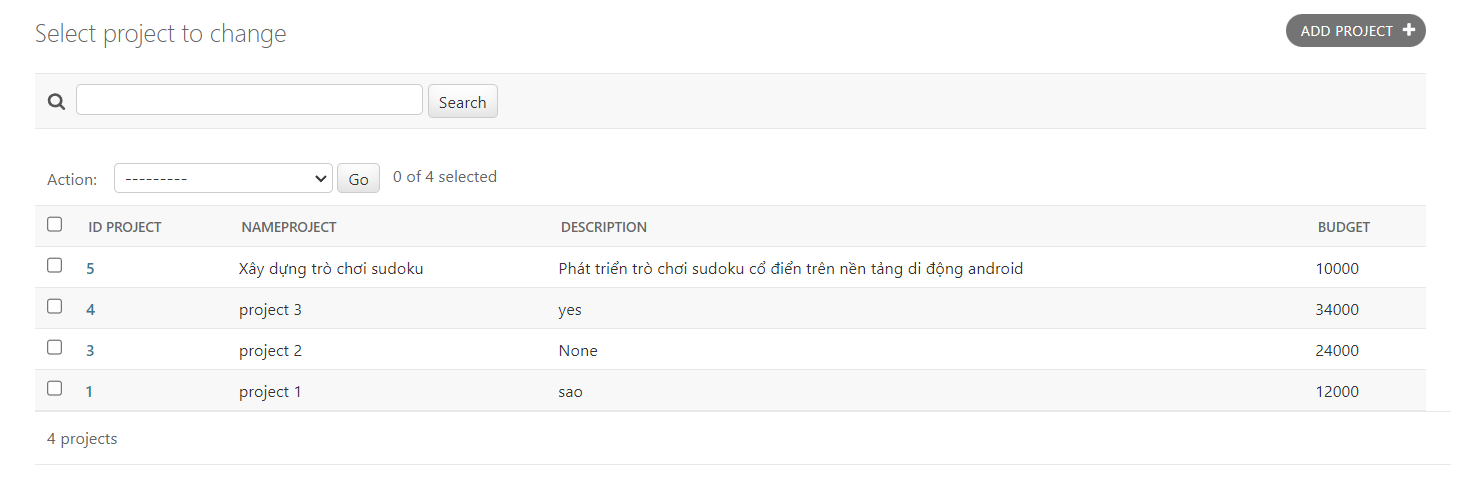
****

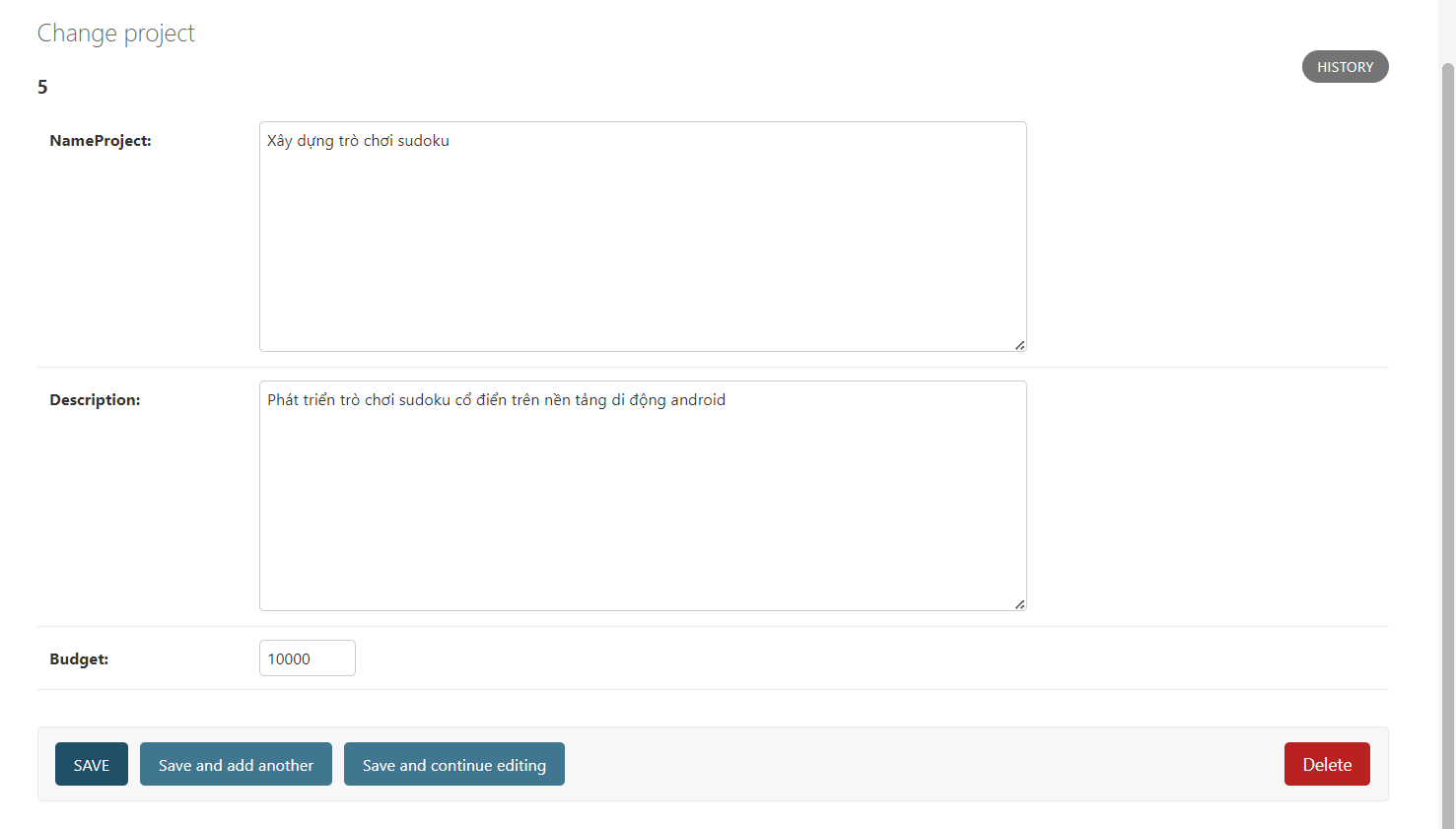
****

1. Giao diện bảo trì người dùng

* Bảo trì viên có thể dùng thanh tìm kiếm để tìm kiếm người dùng, chọn hành động rồi click checkbox các người dùng cần thực hiện hành động để thực hiện hành động. Các hành động có thể thực hiện gồm: xóa tài khoản, đặt lại mật khẩu và khóa tài khoản. Nhấn “Add user” để thêm tài khoản mới.
* Khi ấn vào Id người dùng sẽ mở giao diện giúp chỉnh sửa chi tiết thông tin người dùng

1. Giao diện bảo trì dự án

****

****

1. Giao diện bảo trì người dùng

* Bảo trì viên có thể dùng thanh tìm kiếm để tìm kiếm dự án, chọn action rồi click checkbox các dự án cần thực hiện hành động để thực hiện hành động. Các hành động có thể thực hiện gồm: xóa dự án. Nhấn “Add project” để thêm tài khoản mới.
* Khi ấn vào Id dự án sẽ mở giao diện giúp chỉnh sửa chi tiết thông tin dự án.

## Kiểm thử

*Kiểm thử* là quá trình đánh giá một hệ thống hoặc ứng dụng để xác định xem nó hoạt động như mong đợi hay không và đưa ra các sửa đổi cần thiết để cải thiện chất lượng và độ tin cậy của nó. Quá trình này thường bao gồm việc thiết lập các điều kiện thử nghiệm, thực hiện các bước kiểm tra và so sánh kết quả với các dự đoán trước. Mục tiêu của kiểm thử là phát hiện các lỗi, vấn đề hoặc rủi ro có thể gây ra các vấn đề trong hoạt động của hệ thống và đảm bảo rằng nó hoạt động theo cách mong muốn của người dùng.

1. Các loại kiểm thử thường dùng

* *Kiểm thử chức năng*: Đảm bảo rằng các chức năng của ứng dụng hoạt động đúng theo yêu cầu chức năng đã được xác định trước.
* *Kiểm thử hiệu suất*: Đánh giá hiệu suất của ứng dụng dưới tải công việc khác nhau để đảm bảo rằng nó đáp ứng được yêu cầu về tốc độ, thời gian phản hồi và khả năng chịu tải.
* *Kiểm thử tải*: Kiểm tra hiệu suất của hệ thống khi có tải công việc cao, thường làm bằng cách tăng dần số lượng người dùng hoặc tải công việc.
* *Kiểm thử căn bản*: Kiểm tra các chức năng cơ bản của ứng dụng sau khi xây dựng để đảm bảo rằng nó hoạt động như mong đợi trước khi tiến hành kiểm thử chi tiết hơn.
* *Kiểm thử phục hồi*: Kiểm tra khả năng của hệ thống để phục hồi sau sự cố, như mất kết nối mạng hoặc sự cố phần cứng.
* *Kiểm thử bảo mật*: Đánh giá cơ sở hạ tầng và ứng dụng để xác định các lỗ hổng bảo mật và ngăn chặn các cuộc tấn công.
* *Kiểm thử tương thích*: Kiểm tra xem phần mềm hoạt động đúng trên các nền tảng, trình duyệt và thiết bị khác nhau.
* *Kiểm thử dịch vụ web*: Kiểm tra giao diện lập trình ứng dụng (API) để đảm bảo rằng nó hoạt động đúng và trả về kết quả mong đợi.
* *Kiểm thử tự động*: Sử dụng công cụ và kịch bản để tự động hóa việc kiểm thử, giúp tăng cường hiệu quả và tiết kiệm thời gian.
* *Kiểm thử người dùng cuối*: Thực hiện bởi người dùng cuối để đảm bảo rằng ứng dụng đáp ứng được nhu cầu và mong đợi của họ.

1. Kiểm thử chức năng đăng nhập

Trong đồ án, kiểm thử được sử dụng là kiểm thử chức năng

* *Kịch bản kiểm thử*

1. Kịch bản test đăng nhập

|  |  |  |
| --- | --- | --- |
| Tên test case | | Đăng nhập tài khoản |
| Mục tiêu | | Kiểm tra khả năng thực hiện của chức năng Đăng nhập |
| Người thực hiện | | Nguyễn Ngô An |
| Các bước thực hiện | | |
| STT | Bước thực hiện | |
| 1 | Mở giao diện trang khởi đầu | |
| 2 | Nhấn nút “Đăng nhập” để sử dụng chức năng | |
| 3 | Nhận dữ liệu | |
| 4 | Nhấn nút “Đăng nhập” | |
| Kết quả mong đợi | | Chức năng thực hiện đúng những gì đã mô tả |

* *Các tài khoản sử dụng để test*

1. Các tài khoản sử dụng để test

|  |  |  |
| --- | --- | --- |
| STT | Tên tài khoản | Mật khẩu |
| 1 | ngoan2211 | ABC123@ |
| 2 | binhthanh1996 | P@$$vv0rd |
| 3 | caothanh34 | C@o1h@nh |

* *Kết quả kiểm thử*

1. Kết quả test

|  |  |  |  |
| --- | --- | --- | --- |
| Test | Thông tin nhập | Kết quả | Trạng thái |
| 1 | Tên tài khoản: “ngoan2211”  Mật khẩu: “ABC123@” | Được đưa về Trang chủ của tài khoản | Thành công |
| 2 | Tên tài khoản: “”  Mật khẩu: “” | Hiển thị thông báo “Vui lòng điền vào trường này” chỉ vào trường tên tài khoản | Thành công |
| 3 | Tên tài khoản: “dd”  Mật khẩu: “123” | Hiển thị thông báo “Tài khoản không tồn tại” | Thành công |
| 4 | Tên tài khoản: “an”  Mật khẩu: “” | Hiển thị thông báo “Vui lòng điền vào trường này” chỉ vào trường mật khẩu | Thành công |
| 5 | Tên tài khoản: “binhthanh1996”  Mật khẩu: “P@$$vv0rd” | Được đưa về Trang chủ của tài khoản | Thành công |
| 6 | Tên tài khoản: “caothanh34”  Mật khẩu: “C@01h@nh” | Hiển thị thông báo mật khẩu sai | Thành công |
| 7 | Tên tài khoản: “”  Mật khẩu: “qq” | Hiển thị thông báo “Vui lòng điền vào trường này” chỉ vào trường tên tài khoản | Thành công |
| 8 | Tên tài khoản: “cc”  Mật khẩu: “” | Hiển thị thông báo “Vui lòng điền vào trường này” chỉ vào trường mật khẩu | Thành công |

1. Kiểm thử chức năng gửi tin nhắn

* *Kịch bản kiểm thử*

1. Kịch bản test gửi nội dung trò chuyện

|  |  |  |
| --- | --- | --- |
| Tên test case | | Gửi nội dung trò chuyện |
| Mục tiêu | | Kiểm tra khả năng thực hiện của chức năng Gửi nội dung trò chuyện |
| Người thực hiện | | Nguyễn Ngô An |
| Các bước thực hiện | | |
| STT | Bước thực hiện | |
| 1 | Thực hiện đăng nhập vào 2 tài khoản riêng biệt đã có sẵn và đều là thành viên của 1 nhóm trò chuyện chung. Gọi 2 tài khoản này là bên A và bên B. | |
| 2 | Bên A thực hiện nhập và gửi cho bên B | |
| Kết quả mong đợi | | Chức năng thực hiện đúng những gì đã mô tả |

* *Kết quả kiểm thử*

1. Kết quả test gửi nội dung trò chuyện

|  |  |  |  |
| --- | --- | --- | --- |
| Test | Thông tin nhập | Kết quả | Trạng thái |
| 1 | Text: “hello world” | Bên B nhận được tin nhắn “hello world” | Thành công |
| 2 | Text: “tin nhắn thử nghiệm” | Bên B nhận được tin nhắn “tin nhắn thử nghiệm” | Thành công |
| 3 | Text: “” | Bên A không thấy có tin gửi, bên B cũng không nhận được tin mới | Thành công |
| 4 | Text: “ ” | Bên A không thấy có tin gửi, bên B cũng không nhận được tin mới | Thành công |
| 5 | Text: “https://www.youtube.com/” | Bên B nhận được tin nhắn “https://www.youtube.com/” | Thành công |

1. Kiểm thử chức năng gọi video

* *Kịch bản kiểm thử*

1. Kịch bản test gọi video

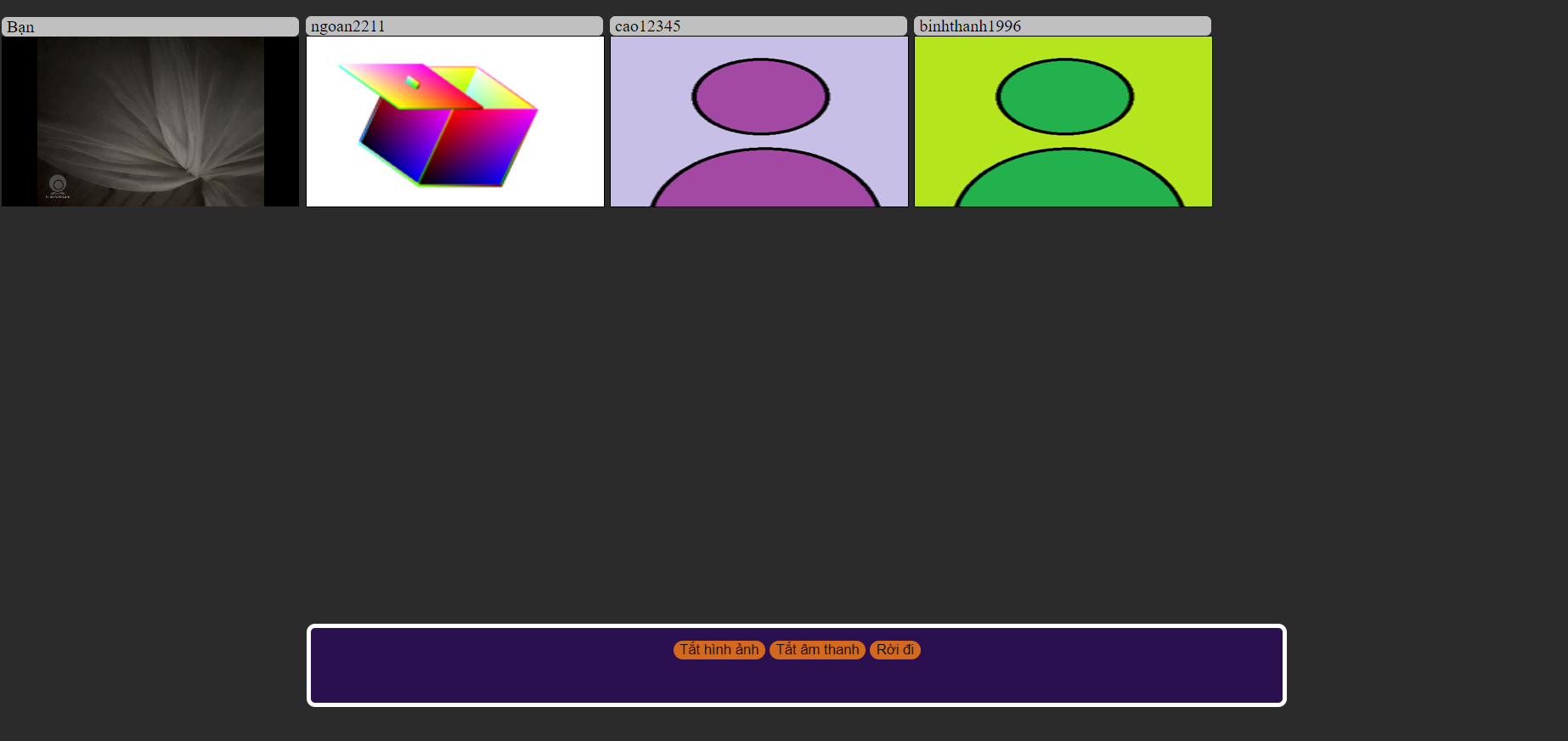
|  |  |  |
| --- | --- | --- |
| Tên test case | | Gọi video |
| Mục tiêu | | Kiểm tra khả năng thực hiện cuộc gọi video |
| Người thực hiện | | Nguyễn Ngô An |
| Các bước thực hiện | | |
| STT | Bước thực hiện | |
| 1 | Đăng nhập vào 4 tài khoản bằng 4 site riêng biệt | |
| 2 | Cả 4 tài khoản cùng tham gia gọi video của dự án ‘project 1’ | |
| Kết quả mong đợi | | Chức năng thực hiện đúng những gì đã mô tả |

* *Các tài khoản sử dụng*

1. Các tài khoản sử dụng để test

|  |  |  |
| --- | --- | --- |
| STT | Tên tài khoản | Mật khẩu |
| 1 | ngoan2211 | ABC123@ |
| 2 | binhthanh1996 | ABC123@ |
| 3 | cao12345 | 123@ABC |
| 4 | ducbinh2211 | 123@ABC |

* *Kết quả đạt được:*

****

1. Kết quả test gọi video

* Kết nối ổn đỉnh, dữ liệu truyền hình ảnh, âm thanh hiển thị đồng bộ giữa các người dùng. Tuy nhiên có hiện tượng tiếng vọng khi bật âm thanh.

KẾT LUẬN

***Trong quá trình thực hiện đồ án, em đã đạt được các kết quả sau:***

* Tìm hiểu được quy trình tạo ra sản phẩm bằng Django.
* Nắm được kiến thức sử dụng Jinja, HTML, CSS, Java script trong dự án Django.
* Hiểu biết và vận dụng để tạo model cơ sở dữ liệu của Django. Đồng thời biết cách xuất, nhập, sửa, xóa dữ liệu.
* Tìm hiểu được cách mã hóa đơn giản thông qua hàm băm SHA-256 để bảo mật mật khẩu người dùng.
* Hiểu được cách tạo Rest API bằng Django để trao đổi thông tin giữa người dùng (client) với máy chủ (server).
* Biết cách sử dụng web socket để truyền và nhận dữ liệu giữa người dùng với máy chủ.
* Biết thêm được cách sử dụng webRTC để tạo ra cuộc gọi video.

***Hạn chế***

* Chức năng gọi video có hiện tượng âm thanh bị vọng lại khi nói.
* Chức năng xem trước file trong chia sẻ file chưa hỗ trợ hết các định dạng file. File .docx có văn bản tiếng việt khi xem trước bị lỗi chữ cái.
* Giao diện chưa được thẩm mỹ còn đơn giản và chưa được bắt mắt.

***Hướng phát triển***

* Giải quyết các hạn chế đã đề cập.
* Cải thiện thêm các chức năng hiện có
* Thêm tính năng khi gọi video như chia sẻ màn hình, quay lại cuộc gọi, ..

TÀI LIỆU THAM KHẢO

**Tiếng Việt**

1. Nguyễn Trung Phú, Trần Phương Nhung, Đỗ Thị Minh Nguyệt, (2019),“*Giáo trình thiết kế Web*”, Trường đại học công nghiệp Hà Nội.
2. Nguyễn Thị Thanh Huyền, Ngô Thị Bích Thúy, Phạm Kim Phượng (2011), “*Giáo trình phân tích thiết kế hệ thống*”, Nhà xuất bản giáo dục Việt Nam.
3. Võ Duy Tuấn, (2015), “*Python cơ bản … rất là cơ bản*”, SEDU.

**Web**

1. “w3schools” <https://www.w3schools.com/> Trang web cung cấp các kiến thức cốt về HTML, CSS, Python
2. “stackoverflow” <https://stackoverflow.com/> Trang web hỏi đáp các vấn đề người lập trình gặp phải.
3. VNLAB TECH SHARE - Tìm hiểu về WebRTC và ứng dụng video chat - https://www.youtube.com/watch?v=GRztdZ3X7C8&t=1079s
4. Good Morning Developers - Want to make a video chat app? Watch this video for WebRTC - <https://www.youtube.com/watch?v=g42>y

NO\_dxWQ&t=1375s