**ĐẠI HỌC ĐÀ NẴNG**

**ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO**

**PBL4: DỰ ÁN HỆ ĐIỀU HÀNH & MẠNG MÁY TÍNH**

**ĐỀ TÀI:**

**DIỄN ĐÀN TRAO ĐỔI KIẾN THỨC**

**Danh sách thành viên: Phan Trần Nhật Hạ - 102210159**

**Nguyễn Đức Huy - 102210164**

**Trần Lê Như Quỳnh - 102210183**

**Giảng viên hướng dẫn: Mai Văn Hà**

**Đà Nẵng, 01/2024**

**LỜI CẢM ƠN**

Trước tiên, chúng em xin gửi lời cảm ơn đến thầy Mai Văn Hà – giảng viên hướng dẫn chúng em thực hiện đề tài. Cảm ơn thầy đã chỉ bảo, giúp đỡ nhóm trong suốt thời gian thực hiện và cũng là người giúp nhóm chúng em mở rộng những ý tưởng cũng như góp ý chỉnh sửa cho đề tài được hoàn thiện.

Mặc dù đã cố gắng hoàn thành báo cáo đề tài trong phạm vi và khả năng tuy nhiên sẽ không thể tránh khỏi những thiếu sót. Chúng em rất mong nhận được sự cảm thông và tận tình chỉ bảo của thầy cô.

Chúng em xin chân thành cảm ơn!

**MỤC LỤC**

[LỜI MỞ ĐẦU 5](#_Toc155196348)

[CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI 6](#_Toc155196349)

[1.1 Giới thiệu đề tài 6](#_Toc155196350)

[1.2 Mục tiêu 6](#_Toc155196351)

[1.3 Mục đích 6](#_Toc155196352)

[1.4 Quản lý dự án 6](#_Toc155196353)

[1.5 Phân công nhiệm vụ 7](#_Toc155196354)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 8](#_Toc155196355)

[2.1 Mô hình MVC 8](#_Toc155196356)

[2.2 Công nghệ Ajax 8](#_Toc155196357)

[2.3 Mã hóa Bcrypt 9](#_Toc155196358)

[2.4 Giao thức HTTP 10](#_Toc155196359)

[2.5 Giao thức SMTP 10](#_Toc155196360)

[CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 12](#_Toc155196361)

[3.1 Yêu cầu chức năng 12](#_Toc155196362)

[3.2 Yêu cầu phi chức năng 13](#_Toc155196363)

[3.3 Thiết kế cơ sở dữ liệu 13](#_Toc155196364)

[3.3.1 Class diagram 13](#_Toc155196365)

[3.3.2 Cơ sở dữ liệu 14](#_Toc155196366)

[CHƯƠNG 4: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ 15](#_Toc155196367)

[4.1 Mô hình triển khai hệ thống 15](#_Toc155196368)

[4.2 Giao diện hệ thống 15](#_Toc155196369)

[4.2.1 Giao diện trang chủ 15](#_Toc155196370)

[4.2.2 Giao diện đăng kí, đăng nhập 16](#_Toc155196371)

[4.2.3 Giao diện Admin 18](#_Toc155196372)

[4.3 Đánh giá kết quả 27](#_Toc155196373)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 28](#_Toc155196374)

[1. Kết quả đạt được 28](#_Toc155196375)

[2. Hướng phát triển 28](#_Toc155196376)

[Tài liệu tham khảo 29](#_Toc155196377)

**PHỤ LỤC HÌNH ẢNH**

Hình 1.1: Mô hình MVC 8

Hình 1.2: Cách thức hoạt động AJAX 9

Hình 1.3: Cách thức hoạt động của Bcrypt 9

Hình 1.4: Minh họa sử dụng SMTP 11

[Hình 3.1: Class Diagram 13](#_Toc155196424)

[Hình 3.2: Cơ sở dữ liệu 14](#_Toc155196425)

[Hình 4.1: Giao diện trang chủ 15](#_Toc155176805)

[Hình 4.2: Giao diện xác thực Email 16](#_Toc155176806)

[Hình 4.3: Giao diện đăng kí tài khoản 16](#_Toc155176807)

[Hình 4.4: Giao diện đăng nhập 17](#_Toc155176808)

[Hình 4.5: Giao diện đăng nhập khi tài khoản bị khóa 17](#_Toc155176809)

[Hình 4.6: Giao diện trang thông tin Admin 18](#_Toc155176810)

[Hình 4.7: Giao diện duyệt các bài viết 18](#_Toc155176811)

[Hình 4.8: Giao diện chọn lý do không duyệt bài viết 19](#_Toc155176812)

[Hình 4.9: Giao diện các bài viết "được duyệt" 19](#_Toc155176813)

[Hình 4.10: Giao diện các bài viết "không được duyệt" 20](#_Toc155176814)

[Hình 4.11: Giao diện quản lý tài khoản 20](#_Toc155176815)

[Hình 4.12: Giao diện tìm kiếm của Admin 21](#_Toc155176816)

[Hình 4.13: Giao diện quản lý lĩnh vực 21](#_Toc155176817)

[Hình 4.14: Giao diện trang thông tin người dùng 22](#_Toc155176818)

[Hình 4.15: Giao diện thay đổi mật khẩu 22](#_Toc155176819)

[Hình 4.16: Giao diện bài viết cá nhân 23](#_Toc155176820)

[Hình 4.17: Giao diện tạo bài viết mới 23](#_Toc155176821)

[Hình 4.18: Giao diện trang chủ người dùng 24](#_Toc155176822)

[Hình 4.19: Giao diện tìm kiếm của người dùng 24](#_Toc155176823)

[Hình 4.20: Giao diện báo cáo người dùng 25](#_Toc155176824)

[Hình 4.21: Giao diện bình luận bài viết 25](#_Toc155176825)

# LỜI MỞ ĐẦU

Trong thời đại số hóa hiện nay, Internet đã trở thành một phần không thể thiếu trong cuộc sống và công việc của chúng ta. Internet không chỉ mang lại sự kết nối toàn cầu mà còn cung cấp nền tảng cho việc trao đổi thông tin và kiến thức. Với sự bùng nổ của thông tin, việc trao đổi và chia sẻ kiến thức trở nên cần thiết hơn bao giờ hết. Tuy nhiên, việc tìm kiếm và tiếp cận thông tin kiến thức chất lượng vẫn là một thách thức lớn.

Hiểu được vấn đề này, các trang web và các ứng dụng khác nhau lần lượt được ra đời. Các nhà phát triển luôn tìm cách để áp dụng những công nghệ mới, kĩ thuật mới vào sản phẩm của mình nhằm tạo ra một môi trường thân thiện và hơn hết có thể hỗ trợ người dùng mạng trong việc tìm hiểu và cập nhật thông tin thế giới.

Với mong muốn đóng góp vào sự phát triển của môi trường mạng và ngành công nghệ thông tin, nhóm chúng em đã tìm hiểu và nghiên cứu xây dựng một diễn đàn trao đổi kiến thức nhằm hỗ trợ mọi người có một môi trường an toàn để nêu ra các thắc mắc của mình trong mọi lĩnh vực cuộc sống. Từ đó, người dùng sẽ nhận được các câu trả lời từ các khía cạnh, góc nhìn khác nhau của những người sử dụng khác cho thắc mắc của mình.

Nhờ sự hướng dẫn của thầy Mai Văn Hà, chúng em có thể ứng dụng những kiến thức đã học cũng như hiểu hơn về quy trình phát triển một dự án liên quan đến hệ điều hành và mạng máy tính để có thể hoàn thiện đồ án này một cách tốt nhất. Nhưng vì đây là đồ án môn học nên không thể tránh khỏi những thiếu sót, chúng em rất mong nhận được sự thông cảm và chỉ bảo của thầy cô để có thể hoàn thiện sản phầm tốt hơn trong tương lai.

## CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI

### 1.1 Giới thiệu đề tài

Đề tài “Diễn đàn trao đổi kiến thức” được ra đời với mong muốn tạo dựng cộng đồng, nơi mà người dùng ở mọi lứa tuổi có thể đặt câu hỏi và giải đáp thắc mắc cùng nhau. Diễn đàn sẽ là một nền tảng cho mọi người có thể chia sẻ và trao đổi kiến thức một cách hiệu quả và thuận tiện.

Với sự phát triển không ngừng của công nghệ, việc xây dựng các diễn đàn trực tuyến đã trở thành một xu hướng phổ biến, tạo ra một không gian chia sẻ và trao đổi kiến thức đa dạng. Nhận biết được điều đó Diễn đàn trao đổi kiến thức HAHYWU được xây dựng với ý nghĩa thực tiễn cao và sẽ là một công cụ hữu ích cho mọi người trong việc học tập, nghiên cứu và chia sẻ kiến thức. Cùng với đó là một môi trường học tập lành mạnh, nơi mọi người có thể tự do nêu lên các thắc mắc và lời giải của mình cho các câu hỏi của người dùng khác. Để sử dụng, người dùng có thể đăng kí tài khoản dựa trên email của mình sau đó đăng nhập bằng tài khoản.

Trong tương lai, với sự đổi mới của công nghệ, diễn đàn sẽ ngày càng phát triển và ứng dụng các công nghệ mới để phù hợp hơn với thời đại và thị hiếu của người sử dụng.

### 1.2 Mục tiêu

Ngày nay, trong thời kì Internet ngày càng phát triển, cùng với các thông tin kiến thức ngày càng đổi mới trong rất nhiều lĩnh vực khác nhau. Nhóm chúng em quyết định chọn đề tài “Xây dựng diễn đàn trao đổi kiến thức” với mong muốn xây dựng một môi trường có thể rút ngắn khoảng cách của người dùng mạng với nhau và hơn cả là khoảng cách của người dùng mạng đến các kiến thức mới của thời đại dưới các khía cạnh và góc nhìn khác nhau.

### 1.3 Mục đích

* Giúp sinh viên ứng dụng các kiến thức được học vào giải quyết các bài toán thực tế.
* Rèn luyện sự sáng tạo trong việc thiết kế giao diện trang web..
* Hiểu rõ quá trình tạo nên một sản phẩm mạng.
* Nâng cao tư duy lập trình và tìm ra các giải pháp, thuật toán phù hợp và tối ưu hơn.

### 1.4 Quản lý dự án

* Công cụ quản lý mã nguồn: Github
* Công cụ thiết kế giao diện: Figma
* Công cụ quản lý thư viện: Maven
* Ngôn ngữ lập trình: Java, JavaScript
* Công cụ lập trình: Eclipse
* Công cụ quản lý cơ sở dữ liệu: MySQL

### 

### 1.5 Phân công nhiệm vụ

|  |  |
| --- | --- |
| **Thành viên** | **Nhiệm vụ** |
| Phan Trần Nhật Hạ | - Khảo sát và phân tích các chức năng.  - Thiết kế cơ sở dữ liệu.  - Thiết kế giao diện.  - Lập trình giao diện  - Lập trình chức năng Admin và một vài chức năng phía User.  - Testing ứng dụng.  - Viết báo cáo. |
| Nguyễn Đức Huy | - Khảo sát và phân tích các chức năng  - Thiết kế cơ sở dữ liệu.  - Thiết kế giao diện.  - Lập trình giao diện.  - Testing ứng dụng.  - Viết báo cáo. |
| Trần Lê Như Quỳnh | - Khảo sát và phân tích các chức năng.  - Thiết kế cơ sở dữ liệu.  - Thiết kế giao diện.  - Lập trình chức năng User  - Testing ứng dụng.  - Viết báo cáo. |

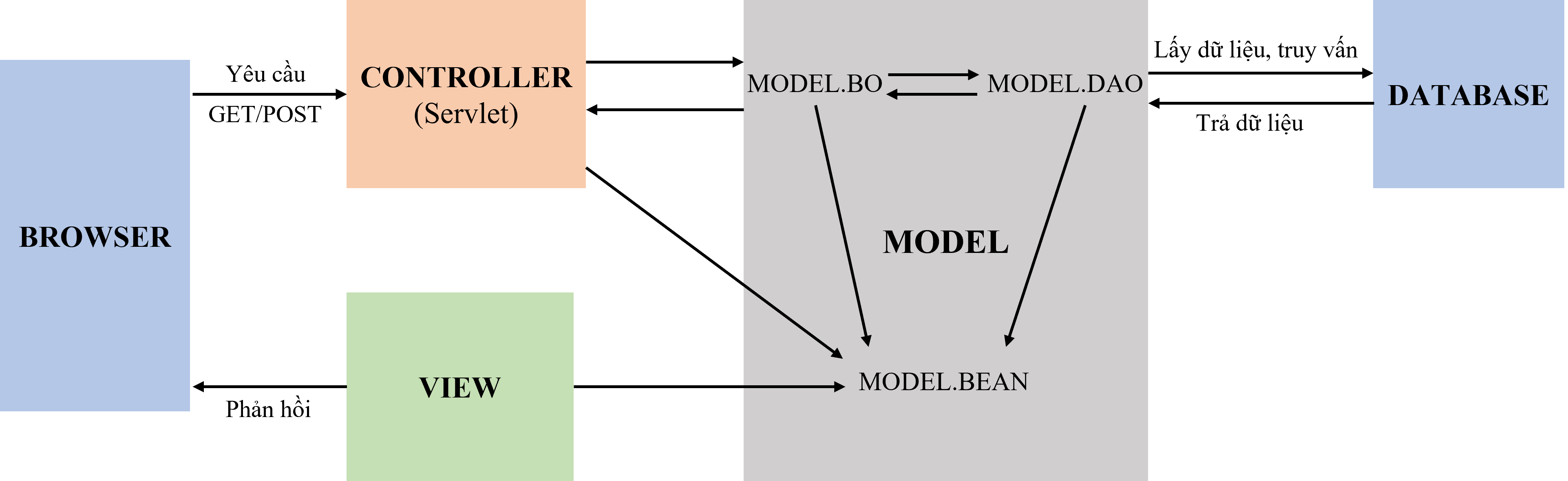
## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

### 2.1 Mô hình MVC

Mô hình MVC (Model-View-Controller) là một mẫu kiến trúc phần mềm giúp tách ứng dụng thành ba phần logic: Model, View và Controlle. Mỗi thành phần có nhiệm vụ riêng, hoạt động độc lập với các thành phần khác.

* Model: đại diện cho dữ liệu và các thao tác liên quan đến dữ liệu.
  + BEAN: phần chứa những class java thường được xây dựng mô phỏng lại 1 cấu trúc bản dữ liệu trong csdl gồm các cấu trúc getter và setter.
  + BO: phần xử lý logic của chương trình, là trung gian giữa Controller và DAO, các lớp khác muốn giao tiếp với database phải thông qua trung giang BO.
  + DAO: phần làm việc trực tiếp với database, việc giao tiếp database chỉ duy nhất xuất hiện ở lớp DAO, các lớp khác muốn làm việc với database  phải thông qua nó.
* View: đại diện cho giao diện người dùng.
* Controller: đại diện cho các thao tác xử lý sự kiện và điều khiển luồng dữ liệu giữa Model và View.

Mô hình MVC giúp cho việc phát triển ứng dụng web trở nên dễ dàng hơn bằng cách chia ứng dụng thành các phần cụ thể, tất cả đều có mục đích riêng. Nó cung cấp sự phân tách rõ ràng giữa logic nghiệp vụ, logic Ul và logic đầu vào, giúp code ít phức tạp hơn và dễ làm việc hơn.



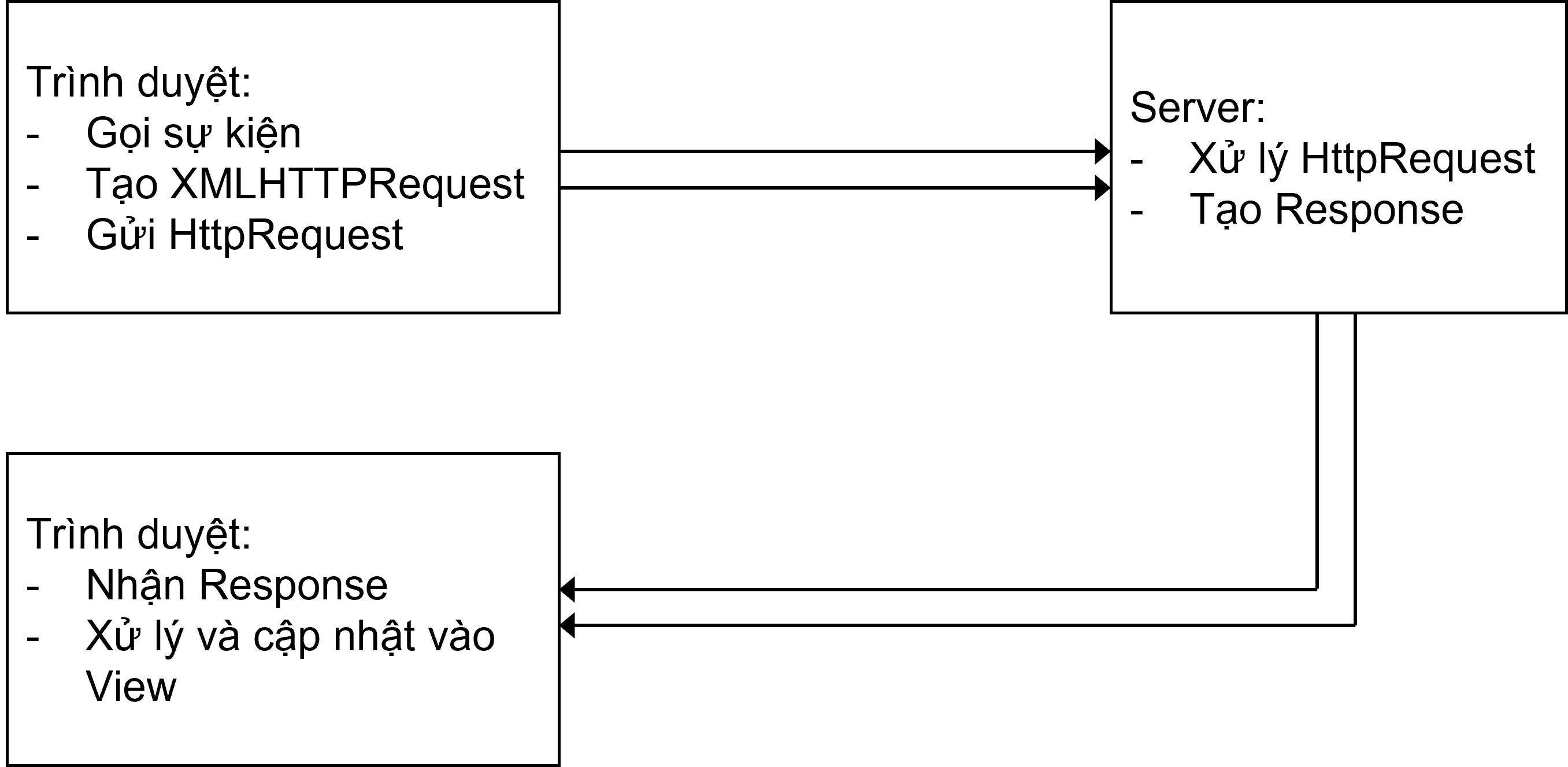
##### Hình 1.1: Mô hình MVC

### 2.2 Công nghệ Ajax

AJAX (Asynchronous Javascript & XML) là phương thức trao đổi dữ liệu với máy chủ và cập nhật một hay nhiều phần của trang web, hoàn toàn không reload lại toàn bộ trang.

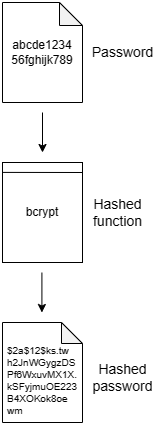
Những lợi ích mà AJAX mang lại:

* AJAX được sử dụng để thực hiện một callback. Được dùng để thực hiện việc truy xuất dữ liệu hoặc lưu trữ dữ liệu mà không cần phải reload lại toàn bộ trang web. Với những server nhỏ thì việc này cũng tiết kiệm được băng thông cho chúng ta hơn.
* Cần gì thì chỉ gửi dữ liệu phần đó, load lại 1 phần nhỏ để cập nhật thông tin chứ không load cả trang. Bằng cách này thì có thể giảm thiểu được tốc độ tải trang giúp người dùng có trải nghiệm tốt hơn.
* Trang web bạn tạo ra cũng sẽ đa dạng và sinh động hơn.



##### Hình 1.2: Cách thức hoạt động AJAX

### 2.3 Mã hóa Bcrypt

Bcrypt là thuật toán mã hóa mật khẩu được sử dụng để bảo vệ mật khẩu người dùng trong các hệ thống thông tin và ứng dụng. Nó được thiết kế để chống lại các cuộc tấn công dò mật khẩu thông qua việc sử dụng kỹ thuật “slow and expensive”. Bcrypt sử dụng hàm băm (hash function) có thời gian tính toán chậm để mã hóa mật khẩu. Quá trình mã hóa này được thực hiện thông qua việc lặp đi lặp lại một số lần (số lần lặp lại được gọi là “factors” hoặc “cost”). Bằng cách thực hiện nhiều lần lặp, bcrypt tạo ra một chuỗi mã băm dài và khó đoán, gây khó khăn cho kẻ tấn công khi tìm ra mật khẩu ban đầu từ mã băm.

Bcrypt cung cấp một phương thức đơn giản để sử dụng trong các ngôn ngữ lập trình khác nhau. Thông thường, khi lưu trữ mật khẩu trong cơ sở dữ liệu, ta sẽ sử dụng Bcrypt để băm mật khẩu người dùng trước khi lưu trữ nó. Khi người dùng đăng nhập, hệ thống sẽ băm mật khẩu đã nhập và so sánh với giá trị băm trong cơ sở dữ liệu để xác nhận tính hợp lệ của mật khẩu. Với các tính năng an toàn và độ tin cậy của nó, bcrypt đã trở thành một trong những lựa chọn hàng đầu để bảo vệ mật khẩu trong các ứng dụng và hệ thống hiện đại.

##### Hình 1.3: Cách thức hoạt động của Bcrypt

### 2.4 Giao thức HTTP

HTTP (HyperText Transfer Protocol) - Giao thức truyền tải siêu văn bản: được dùng trong world wide web với mục đích tạo nên nền tảng kết nối giữa client và server. Client ở đây đại diện cho bất kỳ loại thiết bị hoặc chương trình nào, có thể là PC, smartphone… Còn Server được dùng để chỉ những máy tính trên đám mây. HTTP là một giao thức cho phép trao đổi và sử dụng các nguồn tài nguyên khác nhau, chẳng hạn như HTML doc. Một doc hoàn chỉnh sẽ được tạo nên từ nhiều doc con bao gồm văn bản, layout, media, video, script…

Sự kết nối bằng HTTP giữa Web Browser và Web Server qua 4 giai đoạn:

* Tạo kết nối: Web Browser giao tiếp với Web Server nhờ địa chỉ Internet và số cổng (ngầm định là 80) được đặc trả trong URL.
* Thực hiện yêu cầu: Web Browser gửi thông tin tới Web Server để yêu cầu phục vụ. Việc gửi và nhận thông tin ở đây theo phương thức POST/GET.
* Phản hồi: Web Server gửi một phản hồi về Web Browser nhằm đáp ứng yêu cầu của Web Browser.
* Kết thúc kết nối: Khi kết thúc quá trình trao đổi giữa Web Browser và Web Server thì sự kết nối chấm dứt.

### 2.5 Giao thức SMTP

Giao thức SMTP (Simple Mail Transfer Protocol – giao thức truyền tải thư tín đơn giản) là một chuẩn truyền tải thư điện tử qua mạng Internet. Nó thiết lập kênh kết nối giữa mail client và mail server, và thiết lập kênh liên lạc giữa mail server gửi và mail server nhận.

SMTP là hệ thống giao thức có nhiệm vụ nhận hay truyền tải dữ liệu trong email của người dùng. Hệ thống chỉ nhận và gửi thư điện tử email thông qua thiết bị có kết nối mạng Internet. Những thiết bị nhận và gửi email được gọi là máy chủ SMTP, mỗi máy chủ đều liên kết tới cổng mạng Internet 25 (mặc định) – cổng TCP.



##### Hình 1.4: Minh họa sử dụng SMTP

## 

## CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 3.1 Yêu cầu chức năng

* Đăng kí:
  + Người dùng nhập Email để đăng kí tài khoản. Một mã OTP sẽ được gửi về Email vừa nhập. Người dùng phải nhập đúng mã OTP để sang bước tiếp theo.
  + Sau khi nhập đúng OTP, người dùng sẽ điền thông tin (Tên, tên đăng nhập, mật khẩu) để đăng kí tài khoản cá nhân.
* Đăng nhập:
  + Người dùng sẽ đăng nhập bằng tài khoản cá nhân của mình.
  + Admin sẽ đăng nhập bằng tài khoản được cấp sẵn.
  + Nếu quên mật khẩu thì người dùng và admin có thể bấm vào dòng “Forget password?”. Sau đó sẽ được yêu cầu xác thực Email, nếu thành công sẽ xuất hiện giao diện nhập mật khẩu mới.
* Quản lý thông tin cá nhân:
  + Sau khi đăng nhập, người dùng và admin có thể xem và chỉnh sửa thông tin cá nhân của mình.
* Xem thông tin người dùng khác:
  + Người dùng có thể xem thông tin và bình luận câu trả lời các bài viết/câu hỏi của nhau.
  + Admin có thể xem thông tin và bài viết/câu hỏi của các người dùng.
  + Người dùng có thể báo cáo những người dùng khác, admin sẽ xem xét khóa tài khoản người dùng đó nếu số lượng bị báo cáo quá nhiều.
* Đăng bài viết:
  + Người sử dụng sau khi đăng nhập có thể đăng bài viết để nêu lên các thắc mắc của mình, người dùng vẫn có thể bình luận trao đổi trong bài viết của mình.
  + Sau khi bấm đăng bài, bài viết đó sẽ được gửi qua Admin để chờ duyệt.
* Duyệt bài viết:
  + Bài viết của người dùng sau khi được đăng sẽ ở trong trạng thái “chờ duyệt”.
  + Admin sẽ duyệt các bài viết sau khi người dùng đăng. Những bài viết “được duyệt” sẽ xuất hiện ở trang chủ của tất cả người dùng và những bài viết “không được duyệt” admin sẽ chọn lý do sau đó sẽ bị ẩn đối với người dùng khác ngoại trừ chủ bài viết.
  + Admin có thể xem xét lại những bài viết “đã duyệt”.
* Tìm kiếm thông tin:
  + Người dùng có thể tìm kiếm người dùng khác hoặc những bài viết/câu hỏi tương tự với thắc mắc của mình.
  + Admin có thể tìm kiếm người dùng diễn đàn và các bài viết khác.
* Quản lý bài viết:
  + Người dùng có thể xóa các bài viết của mình.
  + Chỉnh sửa bài viết/câu hỏi chỉ được thực hiện khi bài viết đó đang trong trạng thái “chờ duyệt”.
* Phân loại bài viết:
  + Admin có thể thêm các mục nhằm phân loại bài viết dễ dàng hơn
  + Người dùng có thể lựa chọn phân loại phù hợp với bài viết của mình rút ngắn phạm vi câu hỏi để dễ dàng nhận được câu trả lời phù hợp hơn.

### 3.2 Yêu cầu phi chức năng

* Giao diện thân thiện, phù hợp với mọi lứa tuổi.
* Hệ thống đảm bảo tính dễ sử dụng cho người dùng.
* Bảo mật thông tin người dùng.

### 3.3 Thiết kế cơ sở dữ liệu

#### 3.3.1 Class diagram

A black screen with white text

Description automatically generated

##### Hình 3.1: Class Diagram

#### 3.3.2 Cơ sở dữ liệu

A screenshot of a computer

Description automatically generated

##### Hình 3.2: Cơ sở dữ liệu

## CHƯƠNG 4: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ

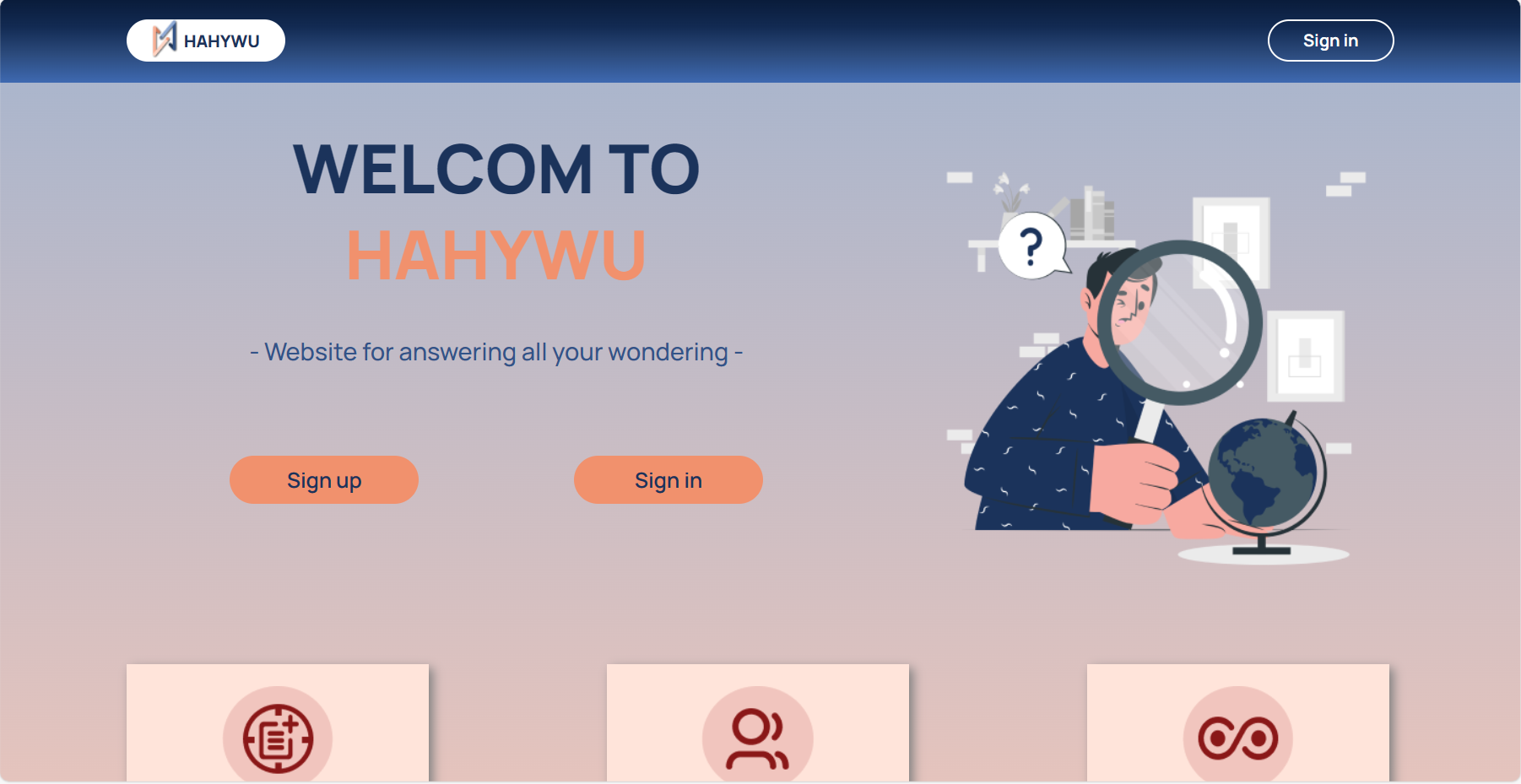
### 4.1 Mô hình triển khai hệ thống

Hệ thống được triển khai theo mô hình MVC (Model-View-Controller):

* Model: đại diện cho dữ liệu và các thao tác liên quan đến dữ liệu.
  + BEAN: phần chứa những class java thường được xây dựng mô phỏng lại 1 cấu trúc bảng dữ liệu trong database gồm các cấu trúc getter và setter.
  + BO: phần xử lý logic của chương trình, là trung gian giữa Controller và DAO, các lớp khác muốn giao tiếp với database phải thông qua trung giang BO.
  + DAO: phần làm việc trực tiếp với database, việc giao tiếp database chỉ duy nhất xuất hiện ở lớp DAO, các lớp khác muốn làm việc với database phải thông qua nó.
* View: đại diện cho giao diện người dùng.
* Controller: đại diện cho các thao tác xử lý sự kiện và điều khiển luồng dữ liệu giữa Model và View.

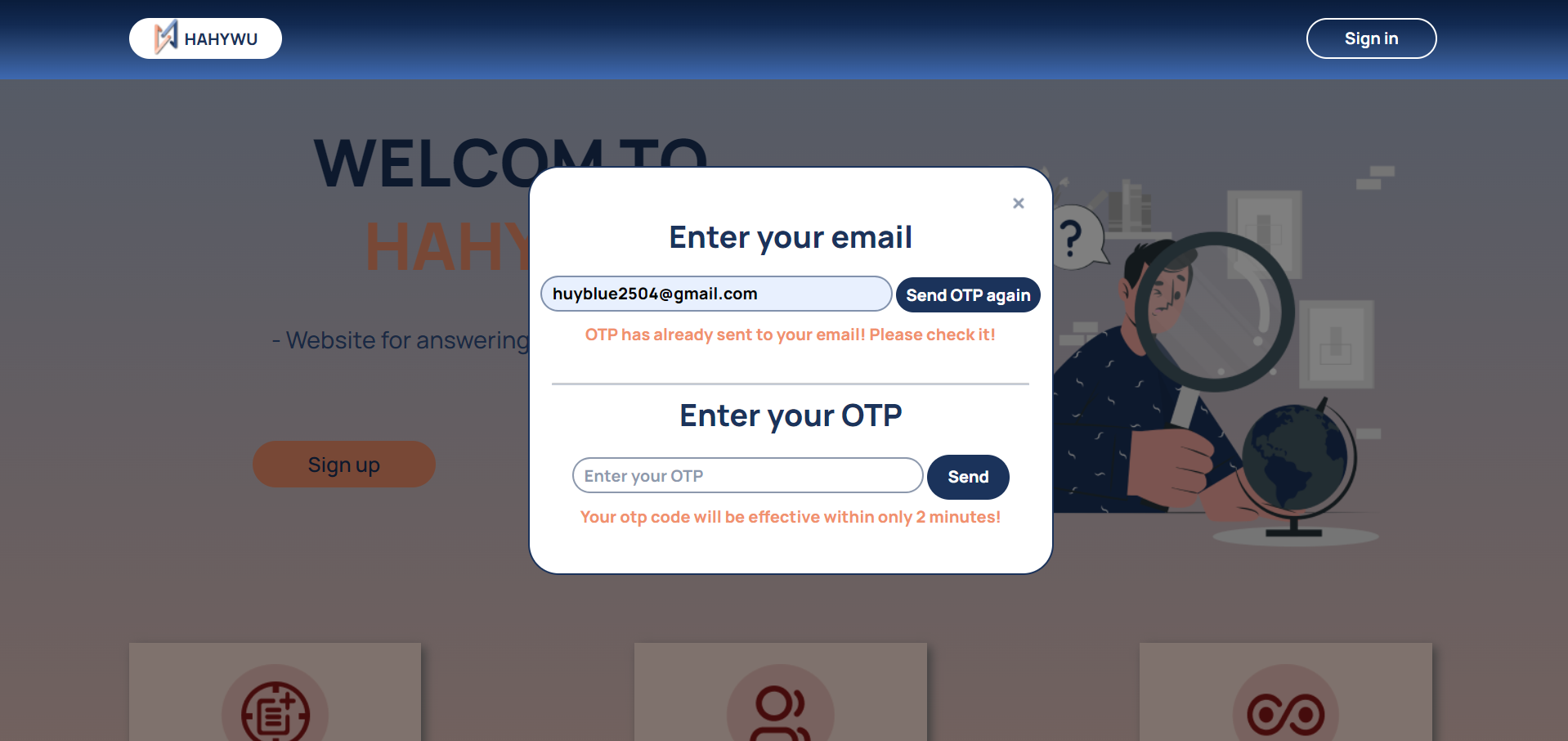
### 4.2 Giao diện hệ thống

#### Giao diện trang chủ

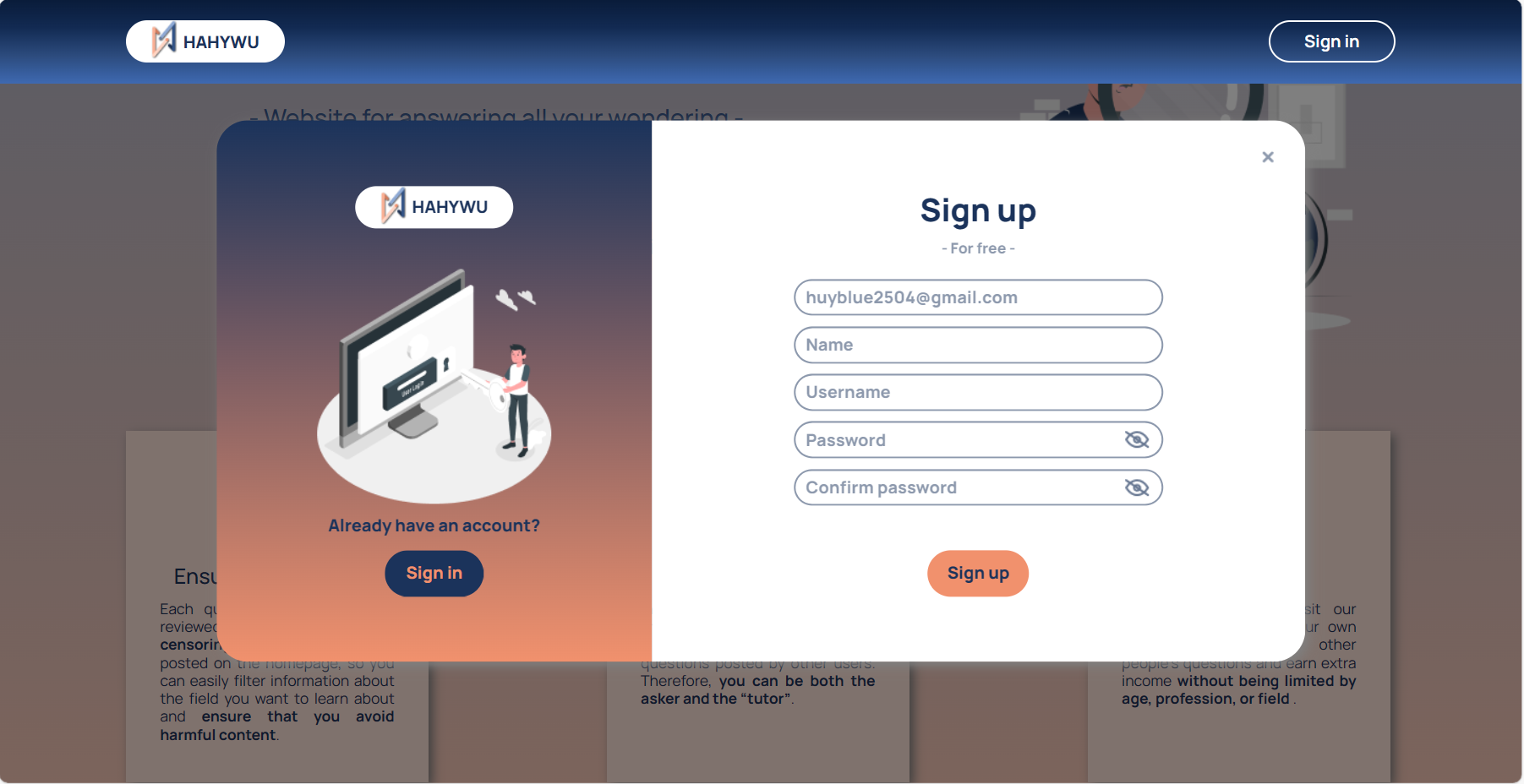


##### Hình 4.1: Giao diện trang chủ

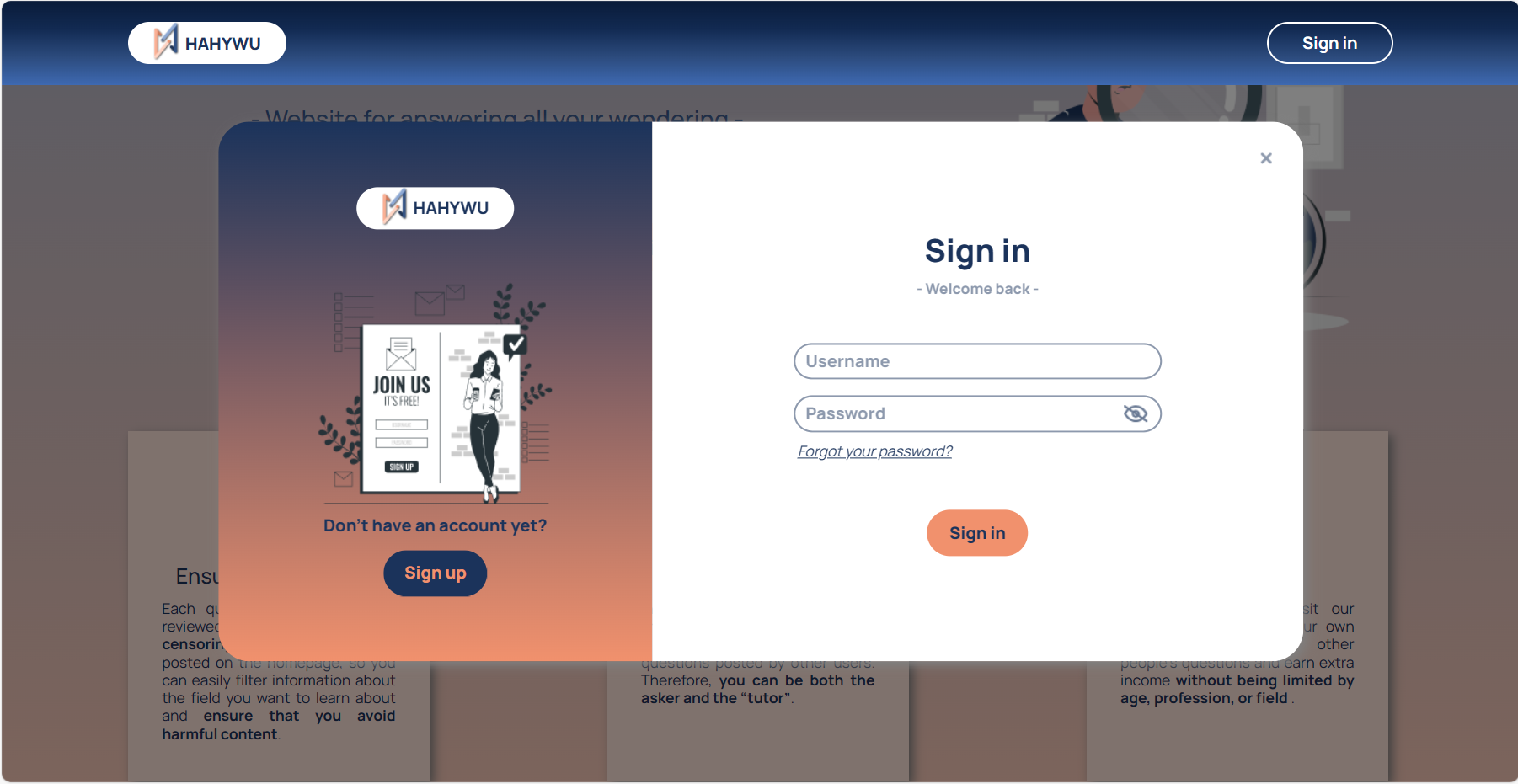
#### Giao diện đăng kí, đăng nhập



##### Hình 4.2: Giao diện xác thực Email



##### Hình 4.3: Giao diện đăng kí tài khoản



##### Hình 4.4: Giao diện đăng nhập

A screenshot of a computer

Description automatically generated

##### Hình 4.5: Giao diện đăng nhập khi tài khoản bị khóa

#### Giao diện Admin

A screenshot of a computer

Description automatically generated

##### Hình 4.6: Giao diện trang thông tin Admin

A screenshot of a computer

Description automatically generated

##### Hình 4.7: Giao diện duyệt các bài viết

A screenshot of a chat

Description automatically generated

##### Hình 4.8: Giao diện chọn lý do không duyệt bài viết

A screenshot of a computer

Description automatically generated

##### Hình 4.9: Giao diện các bài viết "được duyệt"

A screenshot of a computer

Description automatically generated

##### Hình 4.10: Giao diện các bài viết "không được duyệt"

A screenshot of a computer

Description automatically generated

##### Hình 4.11: Giao diện quản lý tài khoản

A screenshot of a computer

Description automatically generated

##### Hình 4.12: Giao diện tìm kiếm của Admin

**A screenshot of a computer

Description automatically generated**

##### Hình 4.13: Giao diện quản lý lĩnh vực

* + 1. **Giao diện người dùng**

**A screenshot of a computer

Description automatically generated**

##### Hình 4.14: Giao diện trang thông tin người dùng

**A screenshot of a login form

Description automatically generated**

##### Hình 4.15: Giao diện thay đổi mật khẩu

**A screenshot of a computer

Description automatically generated**

##### Hình 4.16: Giao diện bài viết cá nhân

**A screenshot of a computer

Description automatically generated**

##### Hình 4.17: Giao diện tạo bài viết mới

**A screenshot of a computer

Description automatically generated**

##### Hình 4.18: Giao diện trang chủ người dùng

**A screenshot of a computer

Description automatically generated**

##### Hình 4.19: Giao diện tìm kiếm của người dùng

A screenshot of a computer

Description automatically generated

##### Hình 4.20: Giao diện báo cáo người dùng

A screenshot of a computer

Description automatically generated

##### Hình 4.21: Giao diện bình luận bài viết

A screenshot of a computer

Description automatically generated

##### Hình 4.21: Giao diện hiển thị thông báo khi có người bình luận bài đăng của mình

A screenshot of a computer

Description automatically generated

##### Hình 4.22: Giao diện hiển thị chi tiết bài đăng khi bấm vào thông báo

### 4.3 Đánh giá kết quả

* Giao diện thân thiện, dễ sử dụng.
* Hệ thống có đầy đủ các chức năng cơ bản của một diễn đàn trao đổi kiến thức.
* Hệ thống còn gặp vấn đề về tốc độ truy cập tùy thuộc vào lượng người dùng nhiều hay ít.

## 

## KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 1. Kết quả đạt được

Sau một khoảng thời gian nghiên cứu, thảo luận, tìm tòi và học hỏi từ các gợi ý cũng như hướng dẫn của giảng viên phụ trách. Nhóm chúng em đã bắt tay vào triển khai dự án, qua đó cũng đã đạt được một số yêu cầu trong việc xây dựng website trên môi trường mạng nhằm giúp người dùng có thể giải đáp mọi thắc mắc của mình.

* **Về mặt lý thuyết:**
* Chia dự án thành hai phần FrontEnd và BackEnd mang lại hiệu quả cao.
* Áp dụng được các kiến thức về xây dựng và thiết kế cơ sở dữ liệu một cách tối ưu.
* Ứng dụng được mô hình MVC trong việc quản lý và phân công nhiệm vụ để xây dựng hệ thống logic hơn.
* Thiết kế giao diện thân thiện, bắt mắt và có bố cục hợp lí, phù hợp với đối tượng người dùng ở mọi lứa tuổi.
* **Về mặt thực tiễn:** Dự án đã đạt được một số chức năng nổi bật như:
* Về phần Admin:
  + Admin có thể kiểm duyệt qua các bài đăng được người dùng đăng lên qua đó giúp cho người dùng có thể tiếp cận được với những thông tin phù hợp và tránh gặp phải những thông tin độc hại.
  + Bên cạnh đó, admin đã có thể quản lý được các tài khoản người dùng và có quyền khoá bất kì tài khoản nào bị báo cáo sau khi đã xem xét.
* Về phần người dùng:
  + Bên cạnh việc đăng bài thì người dùng còn có thể bình luận câu trả lời vào các bài đăng của người khác.
  + Ngoài ra, người dùng còn có thể tìm kiếm được những bài đăng theo lĩnh vực, thời gian. Nếu nhập thông tin vào thanh tìm kiếm, thì kết quả sẽ hiển thị cả những người dùng và bài đăng có nội dung tương tự trong thanh tìm kiếm.

### 2. Hướng phát triển

Với mục tiêu ứng dụng công nghệ vào các lĩnh vực trong đời sống, cụ thể là việc xây dựng diễn đàn trao đổi kiến thức, nhóm chúng em sẽ cố gắng tìm hiểu kĩ hơn cũng như cải thiện và nâng cấp một số chức năng trong hệ thống chẳng hạn như là: thêm chức năng chỉnh sửa câu trả lời để người dùng có thể dễ dàng chỉnh sửa nếu chẳng may viết nhầm nội dung nào đó hay chức năng kết nối giữa các người dùng với nhau để mọi người có thể dễ dàng trao đổi thông tin hơn thông qua các mục boxchat,… Việc cải tiến những chức năng đó nhằm hoàn thiện tốt đề tài hơn trong tương lai.

# Tài liệu tham khảo

[1] Mai Văn Hà , Slide bài giảng môn Công nghệ Web, Đại học bách khoa Đà Nẵng.

[2] Mai Văn Hà , Slide bài giảng môn Lập trình mạng, Đại học bách khoa Đà Nẵng.