

ĐỒ ÁN CHUYÊN NGÀNH
WEBSITE PHÂN TÍCH VÀ DỰ ĐOÁN CHI TIÊU CÁ
NHÂN

Ngành: **CÔNG NGHỆ THÔNG TIN**

Chuyên ngành: **CÔNG NGHỆ PHẦN MỀM**

Giảng viên hướng dẫn: **Nguyễn Huy Cường**

Sinh viên thực hiện:

MSSV:

Họ và tên:

Lớp:

2011065095

Phan Thị Anh

20DTHB1

HCM, Ngày 29/12/2023

LỜI CAM ĐOAN

Trong quá trình thực hiện Đồ án chuyên ngành Công nghệ phần mềm do khả năng tiếp thu thực tế còn nhiều hạn chế, kiến thức chưa sâu rộng. Mặc dù nhóm chúng em đã cố gắng hết sức nhưng chắc chắn bài báo cáo này khó tránh khỏi những thiếu sót, nhưng những gì được trình bày trong báo cáo này là sự thể hiện những kết quả đạt được dưới sự hướng dẫn của giảng viên – thầy Nguyễn Huy Cường đã nhiệt tình hướng dẫn, khuyến khích và tạo điều kiện cho chúng em hoàn thành tốt đề tài của mình.

Chúng em xin cam đoan rằng nội dung trình bày trong báo cáo đồ án, những số liệu và kết quả nghiên cứu là trung thực, hoàn toàn được thực hiện không sao chép bất kỳ nguồn nào khác. Ngoài ra, trong bài báo cáo có sử dụng một số nguồn tài liệu tham khảo đã được trích dẫn nguồn và chú thích rõ ràng. Nếu không đúng sự thật, chúng em xin chịu mọi trách nhiệm.

Sinh viên đồng thực hiện

Phan Thị Anh

MỤC LỤC

CHƯƠNG 1. TỔNG QUAN	1
1.1. Giới thiệu đề tài.....	1
1.2. Nhiệm vụ đồ án	1
1.2.1. Lý do chọn đề tài:.....	1
1.2.2. Ý nghĩa khoa học và thực tiễn:	2
1.2.3. Mục tiêu nghiên cứu:	2
1.2.4. Đối tượng và phạm vi giới hạn:	3
1.3. Cấu trúc đồ án:	3
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT	4
2.1. Frontend	4
2.1.1. HTML:	4
2.1.2. CSS.....	5
2.1.3. BOOTSTRAP:	7
2.1.4. JAVASRIPT (JS):	9
2.2. Backend.....	11
2.2.1. Python	11
2.2.2. Cơ sở dữ liệu MySQL	12
2.2.3. FastAPI.....	14
2.2.4. KNN (K-Nearest Neighbors)	15
2.2.5. Boto3	16
Ngoài ra, sử dụng các thư viện trong Python như: PIL, bcrypt, pandas, sqlite3,... ..	18
CHƯƠNG 3. KẾT QUẢ THỰC NGHIỆM	19
3.1. Phân tích thiết kế hệ thống.....	19
3.1.1. Mô hình thực thể quan hệ (ERD):.....	19

3.1.2. Sơ đồ Usecase:	19
3.1.3. Mô hình cơ sở dữ liệu:	22
3.2. Giao diện Website	25
3.2.1. Các giai đoạn hoàn thành website:.....	25
3.2.2. Giao diện người dùng:.....	25
3.2.3. Giao diện admin:	31
CHƯƠNG 4. KẾT LUẬN VÀ KIẾN NGHỊ	37
4.1. Kết luận.....	37
4.2. Hướng phát triển:	37

DANH MỤC CÁC KÝ HIỆU, CHỮ VIẾT TẮT

STT	Ký hiệu	Chữ viết đầy đủ
1	HTML	Hypertext Markup Language
2	CSS	Cascading Style Sheets
3	SQL	Structured Query Language
4	KNN	K-nearest neighbors

DANH MỤC BẢNG

<i>Bảng 1: Bảng Bill</i>	24
<i>Bảng 2: User</i>	24

DANH MỤC HÌNH VẼ, ĐỒ THỊ

<i>Hình 1: Mô hình thực thể quan hệ ERD.....</i>	<i>19</i>
<i>Hình 2: Usecase Tổng quát</i>	<i>20</i>
<i>Hình 3: Usecase đăng nhập</i>	<i>21</i>
<i>Hình 4: Usecase quản lý thông tin tài khoản</i>	<i>21</i>
<i>Hình 5: Usecase chỉnh sửa thông tin tài khoản</i>	<i>22</i>
<i>Hình 6: Usecase đổi mật khẩu.....</i>	<i>22</i>
<i>Hình 7: Sơ đồ Diagram trong MSSQL</i>	<i>23</i>
<i>Hình 8: Giao diện Trang chủ</i>	<i>26</i>
<i>Hình 9: Giao diện trang Đăng Nhập.....</i>	<i>27</i>
<i>Hình 10: Giao diện trang Đăng ký.....</i>	<i>28</i>
<i>Hình 11: Giao diện trang upload hóa đơn.....</i>	<i>29</i>
<i>Hình 12: Giao diện trang phân tích dữ liệu.....</i>	<i>30</i>
<i>Hình 13: Giao diện về chúng tôi</i>	<i>31</i>
<i>Hình 14: Giao diện trang Đăng nhập tài khoản Admin.....</i>	<i>32</i>
<i>Hình 15: Giao diện trang chủ Admin</i>	<i>33</i>
<i>Hình 16: Trang chỉnh sửa thông tin người dùng.....</i>	<i>33</i>
<i>Hình 17: Giao diện trang quản lý người dùng.....</i>	<i>34</i>
<i>Hình 18: Giao diện trang quản lý hóa đơn</i>	<i>34</i>
<i>Hình 19: Giao diện trang quản lý số liệu phân tích của từng user.....</i>	<i>35</i>
<i>Hình 20: Trang phân tích hóa đơn người dùng quyền admin.....</i>	<i>36</i>

CHƯƠNG 1. TỔNG QUAN

1.1. Giới thiệu đề tài

Trong bối cảnh nền kinh tế hiện đại, quản lý tài chính cá nhân đang trở thành một thách thức ngày càng lớn đối với mọi người. Sự đa dạng về nguồn thu nhập, cùng với việc thị trường ngày càng biến động, đặt ra những yêu cầu cao về khả năng kiểm soát và dự đoán chi tiêu cá nhân.

Đề tài "Website Phân Tích và Dự Đoán Chi Tiêu Cá Nhân" nhằm mục đích xây dựng một nền tảng trực tuyến giúp người sử dụng hiểu rõ hơn về mô hình chi tiêu cá nhân của mình thông qua các công cụ phân tích và dự đoán. Website sẽ không chỉ là một công cụ theo dõi chi tiêu, mà còn là một hệ thống thông minh có khả năng tư vấn và đưa ra các gợi ý cụ thể để quản lý tài chính hiệu quả.

Các chức năng chính của website sẽ bao gồm việc tự động thu thập dữ liệu chi tiêu từ các nguồn khác nhau, phân loại chi tiêu theo các hạng mục cụ thể, và xây dựng các mô hình dự đoán dựa trên các dữ liệu lịch sử. Ngoài ra, website cũng sẽ tích hợp các công nghệ trí tuệ nhân tạo để dự đoán số tiền chi tiêu trong tương lai của mỗi cá nhân.

Thông qua việc kết hợp sức mạnh của công nghệ thông tin và phân tích dữ liệu, đề tài này không chỉ hỗ trợ người sử dụng trong việc theo dõi chi tiêu một cách tự động và tiện lợi mà giúp họ dự đoán những số tiền chi sắp tới và từ đó đưa ra những quyết định tài chính thông minh, giúp họ quản lý tài chính cá nhân một cách hiệu quả.

1.2. Nhiệm vụ đồ án

1.2.1. Lý do chọn đề tài:

Trong thời đại ngày nay, quản lý tài chính cá nhân trở nên ngày càng phức tạp đối với mọi người. Sự đa dạng và biến động của nguồn thu nhập, cùng với thị trường tài chính liên tục thay đổi, tạo ra những thách thức lớn trong việc theo dõi và kiểm soát chi tiêu. Để đáp ứng nhu cầu ngày càng cao của người sử dụng trong việc quản lý tài chính cá nhân một cách hiệu

quả, đề tài "Website Phân Tích và Dự Đoán Chi Tiêu Cá Nhân" được chọn lựa với nhiều lý do quan trọng.

Một trong những lý do hàng đầu là khó khăn mà nhiều người phải đối mặt trong việc quản lý tài chính cá nhân như: Theo dõi và phân loại các khoản chi, đặc biệt là khi nguồn thu nhập đến từ nhiều nguồn khác nhau. Một lý do khác là nhu cầu ngày càng lớn từ phía người sử dụng cần tư vấn và dự đoán chi tiêu cá nhân của mình.

Mục tiêu chính của đề tài là tạo ra một công cụ không chỉ giúp người sử dụng theo dõi chi tiêu một cách tự động và tiện lợi mà còn hỗ trợ họ hiểu rõ hơn về mô hình chi tiêu cá nhân. Thông qua việc kết hợp sức mạnh của công nghệ thông tin và phân tích dữ liệu, hi vọng đề tài sẽ đóng góp tích cực vào việc giúp mọi người đưa ra các quyết định tài chính thông minh và đạt được mục tiêu cá nhân của họ một cách hiệu quả.

1.2.2. Ý nghĩa khoa học và thực tiễn:

Ý nghĩa khoa học: Đề tài đóng góp vào sự phát triển của công nghệ thông tin và quản lý tài chính, khai thác trí tuệ nhân tạo và phân tích dữ liệu để xây dựng các mô hình dự đoán chi tiêu cá nhân.

Ý nghĩa thực tiễn: Website giúp người sử dụng quản lý tài chính cá nhân hiệu quả, hỗ trợ dự đoán thông minh, tiết kiệm thời gian, và cung cấp người dùng hiểu biết sâu sắc về mô hình chi tiêu. Đồng thời, nó tăng cường dự đoán tài chính và mang lại lợi ích thiết thực cho cả cá nhân và cộng đồng.

1.2.3. Mục tiêu nghiên cứu:

Mục tiêu chính của nghiên cứu là xây dựng và triển khai một website phân tích và dự đoán chi tiêu cá nhân, nhằm đạt đến các mục tiêu cụ thể như: Phát triển hệ thống tự động thu thập và phân tích chi tiêu cá nhân từ các nguồn dữ liệu đã cung cấp từ người dùng. Xây dựng mô hình dự đoán chi tiêu dựa trên dữ liệu hóa đơn, giúp người sử dụng có dự đoán trước về xu hướng chi tiêu và dự đoán chi tiêu tương lai. Tối ưu hóa giao diện người dùng thân thiện, dễ sử dụng, và cung cấp thông tin chi tiết về tình hình tài chính cá nhân một cách rõ ràng và

hiệu quả. Tích hợp trí tuệ nhân tạo để dự đoán lượng chi tiêu và từ đó đưa ra các gợi ý cá nhân hóa và chiến lược quản lý tài chính tối ưu.

1.2.4. Đối tượng và phạm vi giới hạn:

Đối tượng nghiên cứu: Đối tượng của nghiên cứu này là người sử dụng cá nhân, có nhu cầu quản lý tài chính và theo dõi chi tiêu hàng ngày. Nghiên cứu tập trung vào việc phát triển một hệ thống phân tích và dự đoán chi tiêu mà người sử dụng có thể tận dụng để nâng cao kiểm soát và hiểu biết về tình hình tài chính cá nhân.

Phạm vi giới hạn: Nghiên cứu giới hạn trong phạm vi người sử dụng cá nhân, sử dụng dữ liệu từ ngân hàng và ví điện tử, và triển khai hệ thống trong một địa điểm cụ thể, đảm bảo tuân thủ bảo mật thông tin và quyền riêng tư.

1.3. Cấu trúc đồ án:

Chương 1. TỔNG QUAN: Giới thiệu ngắn gọn về đề tài, mục tiêu nghiên cứu, ý nghĩa khoa học và thực tiễn, phạm vi giới hạn và đối tượng nghiên cứu.

Chương 2. CƠ SỞ LÝ THUYẾT: Trình bày các khái niệm và phương pháp giải quyết vấn đề bao gồm mô tả các công nghệ, hệ thống, các ràng buộc hoặc các giải pháp mới, những mô hình toán, lý giải xây dựng mô hình.

Chương 3. KẾT QUẢ THỰC NGHIỆM: Trình bày quy trình làm web của nhóm từ việc lên ý tưởng đến khi hoàn thành web.

Chương 4. KẾT LUẬN VÀ KIẾN NGHỊ: Nêu những kết luận chung, khẳng định những kết quả đề ra, những đóng góp, đề xuất mới và kiến nghị cải tiến web.

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1. Frontend

2.1.1. HTML:

HTML (HyperText Markup Language) là ngôn ngữ đánh dấu siêu văn bản được sử dụng để xây dựng các trang web. Được phát triển vào những năm 1990, HTML là một phần quan trọng của Công nghệ thông tin và trở thành tiêu chuẩn chung cho việc tạo nội dung web.

Lý thuyết HTML dựa trên các khái niệm cơ bản của siêu văn bản, trong đó các thành phần của trang web được sử dụng để định dạng và tổ chức nội dung. Một tài liệu HTML được tạo thành từ các thẻ (tags) và các phần tử (elements) được chứa trong các thẻ đó.

Cấu trúc tổ chức của một tài liệu HTML thường bao gồm một thẻ "<html>" chứa các phần tử "<head>" và "<body>". Phần tử "<head>" chứa các thông tin mô tả tài liệu và thẻ "<body>" chứa nội dung hiển thị trên trình duyệt.

Một số khái niệm quan trọng khác trong lý thuyết HTML bao gồm khả năng tạo liên kết, sử dụng mẫu (templates) và các công cụ khác nhau để kiểm tra và xác thực tính hợp lệ của mã HTML.

Ưu điểm của HTML (HyperText Markup Language):

- Dễ học và sử dụng: HTML là ngôn ngữ đánh dấu cơ bản và dễ học, làm cho nó phù hợp cho người mới bắt đầu học lập trình web.
- Phổ biến và chuẩn hóa: HTML là ngôn ngữ cơ bản cho phát triển web và được hỗ trợ rộng rãi trên mọi trình duyệt, làm cho nó trở thành một tiêu chuẩn chung trong ngành.
- Hỗ trợ đa phương tiện: HTML hỗ trợ tích hợp các phương tiện đa dạng như hình ảnh, video, âm thanh, và các đối tượng multimedia khác.
- Duyệt mọi nơi: HTML là ngôn ngữ độc lập từ nền tảng, có thể chạy trên mọi thiết bị và trình duyệt.

- SEO tốt: Việc sử dụng cấu trúc đúng của HTML có thể cải thiện khả năng tìm kiếm của trang web, vì các công cụ tìm kiếm có thể hiểu và đánh giá nội dung một cách dễ dàng.
- Tích hợp với các ngôn ngữ khác: HTML có thể tích hợp dễ dàng với các ngôn ngữ lập trình và công nghệ khác như CSS (Cascading Style Sheets) và JavaScript để tạo ra trải nghiệm web đa dạng và phức tạp hơn.

Nhược điểm của HTML:

- Thiếu khả năng tương tác: HTML ban đầu được thiết kế để mô tả cấu trúc của trang web, không có khả năng tương tác phức tạp mà JavaScript thường cung cấp.
- Giao diện đơn giản: HTML chỉ mô tả cấu trúc và nội dung của trang web, không đi sâu vào thiết kế và trình bày.
- Khả năng mở rộng hạn chế: HTML có giới hạn trong việc tạo các tính năng động và phức tạp, đặc biệt là so với các ngôn ngữ lập trình web hiện đại.
- Không giữ trạng thái: HTML không giữ thông tin trạng thái giữa các yêu cầu, điều này có thể tạo ra thách thức trong việc duy trì trạng thái của ứng dụng web.
- Khó khăn khi xử lý dữ liệu người dùng: HTML không cung cấp các cơ chế tích hợp để xử lý dữ liệu người dùng và tương tác người dùng nâng cao.

2.1.2. CSS

CSS (Cascading Style Sheets) là một ngôn ngữ đánh dấu được sử dụng để mô tả cách trang web được hiển thị trên trình duyệt. Nó hoạt động cùng với HTML để tạo ra các thiết kế và định dạng hấp dẫn cho các phần tử trên trang web.

Lý thuyết CSS dựa trên các quy tắc và khái niệm về việc áp dụng các kiểu và thuộc tính cho các phần tử HTML. Với CSS, chúng ta có thể kiểm soát màu sắc, font chữ, kích thước, vị trí và nhiều thuộc tính khác của các phần tử trên trang web.

Các quy tắc CSS được áp dụng thông qua các bộ chọn (selectors) và các khối khai báo (declaration blocks). Bộ chọn xác định các phần tử HTML mà chúng ta muốn áp dụng kiểu vào, trong khi khối khai báo chứa các thuộc tính và giá trị để định dạng phần tử đó.

CSS cũng hỗ trợ các bộ chọn nâng cao như bộ chọn theo lớp (class), bộ chọn theo ID, bộ chọn theo thẻ con, và nhiều bộ chọn khác để chỉ định các phần tử cụ thể mà chúng ta muốn áp dụng kiểu.

Ưu điểm của CSS:

- Tách biệt giữa nội dung và kiểu dáng (Separation of Concerns): CSS cho phép phân chia rõ ràng giữa nội dung HTML và kiểu dáng, giúp làm cho mã nguồn trở nên dễ bảo trì hơn và linh hoạt hơn.
- Kiểm soát tốt về giao diện: CSS cung cấp nhiều thuộc tính và lựa chọn để kiểm soát giao diện của trang web, từ kích thước và màu sắc cho đến vị trí và độ đục của các phần tử.
- Tính di động (Responsive Design): CSS hỗ trợ thiết kế linh hoạt, giúp tạo ra các trang web có thể tự động điều chỉnh giao diện để phản ánh kích thước màn hình của thiết bị.
- Kế thừa và độ ưu tiên (Inheritance and Specificity): CSS sử dụng cơ chế kế thừa và độ ưu tiên để quyết định cách mà các quy tắc được áp dụng, giúp kiểm soát cách các phần tử con kế thừa kiểu dáng từ phần tử cha.
- Khả năng tùy chỉnh cao: Có thể tùy chỉnh nhiều khía cạnh của giao diện người dùng, từ font chữ đến biên, đổ bóng, và chuyển động.

Nhược điểm của CSS:

- Quản lý layout khó khăn: Trong một số trường hợp, quản lý layout với CSS có thể trở nên phức tạp và khó khăn, đặc biệt là trong các dự án lớn và phức tạp.

- Trình duyệt tương thích: CSS không luôn hoạt động đồng đều trên tất cả các trình duyệt. Một số thuộc tính có thể được hiểu và thực thi khác nhau trên các trình duyệt khác nhau.
- Khả năng hiệu suất: Một số hiệu suất có thể bị ảnh hưởng khi sử dụng CSS phức tạp hoặc nhiều quy tắc CSS, đặc biệt trên các thiết bị có cấu hình yếu.
- Cảm nhận khó khăn: Đối với người mới học, cú pháp và cách hoạt động của CSS có thể khá khó hiểu và làm cho việc làm quen với nó mất thời gian.
- Chống hiệu ứng Flash (Flash of Unstyled Content - FOUC): Đôi khi, trình duyệt có thể hiển thị nội dung mà không có kiểu dáng (FOUC) trong khi tải trang, điều này có thể làm giảm trải nghiệm người dùng.

2.1.3. BOOTSTRAP:

Bootstrap là một framework CSS phổ biến và mạnh mẽ được sử dụng để xây dựng giao diện người dùng cho các trang web và ứng dụng web. Nó cung cấp một bộ công cụ và các lớp CSS được chuẩn hóa để giúp tạo ra các trang web thân thiện với người dùng và có thiết kế đáp ứng (responsive design).

Lý thuyết Bootstrap dựa trên cơ sở của CSS và bổ sung thêm các lớp CSS và các thành phần tương tác để tạo ra các trang web chuyên nghiệp một cách nhanh chóng và dễ dàng. Với Bootstrap, chúng ta có thể sử dụng các lớp CSS đã được định nghĩa sẵn để tạo ra các thành phần như hệ thống lưới (grid system), nút (buttons), biểu mẫu (forms), thanh điều hướng (navigation bar) và nhiều thành phần khác.

Bootstrap cũng cung cấp một thư viện các thành phần JavaScript như hộp thoại (modal), thanh trượt (carousel), tab, và nhiều hiệu ứng tương tác khác. Điều này giúp tạo ra các trang web động và tương tác mà không cần viết mã JavaScript từ đầu.

Với việc sử dụng Bootstrap, chúng ta có thể nhanh chóng tạo ra các trang web đáp ứng với thiết kế chuyên nghiệp, điều chỉnh các kiểu dáng và định dạng một cách dễ dàng thông

qua việc sử dụng các lớp CSS và các thành phần đã được chuẩn bị sẵn. Bootstrap cũng giúp đảm bảo tính nhất quán giữa các trang web và đơn giản hóa quá trình phát triển.

Cách sử dụng Bootstrap bao gồm liên kết tệp CSS và JavaScript của Bootstrap vào tài liệu HTML của ta. Sau đó, có thể sử dụng các lớp CSS và các thành phần Bootstrap trong mã HTML của ta để tạo ra giao diện và các thành phần trên trang web.

Ưu điểm của Bootstrap:

- Dễ sử dụng: Bootstrap được thiết kế để đơn giản hóa quá trình phát triển web, giúp những người mới học và phát triển nhanh chóng.
- Responsive Design: Bootstrap hỗ trợ thiết kế đáp ứng (responsive design), giúp trang web hiển thị đẹp trên nhiều thiết bị và kích thước màn hình khác nhau.
- Grid System mạnh mẽ: Bootstrap cung cấp một hệ thống grid linh hoạt, giúp dễ dàng xây dựng giao diện theo ý muốn và đảm bảo cấu trúc đồng nhất.
- Thư viện CSS và JavaScript sẵn có: Bootstrap đi kèm với nhiều lớp CSS và các thành phần JavaScript được xây dựng trước, giúp giảm độ phức tạp trong việc tạo giao diện và thêm các tính năng tương tác.
- Hỗ trợ Cross-browser: Bootstrap đã được kiểm thử và tối ưu hóa để hoạt động đồng đều trên nhiều trình duyệt khác nhau.
- Cộng đồng lớn: Do Bootstrap được sử dụng rộng rãi, có một cộng đồng lớn của những người phát triển và hỗ trợ trực tuyến.
- Tích hợp dễ dàng: Bootstrap có thể tích hợp dễ dàng với nhiều công cụ và framework khác.

Nhược điểm của Bootstrap:

- Quá phổ biến: Do Bootstrap được sử dụng rộng rãi, có thể dẫn đến việc các trang web sử dụng Bootstrap có thể trở nên giống nhau và thiếu tính cá nhân hóa.
- Dư thừa mã nguồn: Vì Bootstrap cung cấp nhiều tính năng, nếu chỉ sử dụng một phần nhỏ, có thể xuất hiện dư thừa mã nguồn không cần thiết.

- Khả năng mở rộng hạn chế: Trong một số trường hợp, Bootstrap có thể hạn chế khả năng mở rộng và điều chỉnh theo yêu cầu cụ thể của dự án.
- Tải trang chậm: Nếu không tối ưu hóa đúng cách, việc sử dụng toàn bộ Bootstrap có thể làm tăng thời gian tải trang.
- Tùy chỉnh khó khăn: Trong một số trường hợp, việc tùy chỉnh giao diện để đạt được thiết kế độc đáo có thể khó khăn hơn so với việc sử dụng mã nguồn tùy chỉnh.

2.1.4. JAVASRIPT (JS):

JavaScript (JS) là một ngôn ngữ lập trình dựa trên đối tượng, phát triển ban đầu để thực hiện các chức năng tương tác trên trình duyệt web. Tuy nhiên, ngày nay, JavaScript đã trở thành một ngôn ngữ phổ biến và mạnh mẽ, được sử dụng trong nhiều lĩnh vực như phát triển web, ứng dụng di động và phát triển ứng dụng máy tính.

Lý thuyết JavaScript dựa trên cú pháp và quy tắc của ngôn ngữ lập trình. Nó cung cấp các cấu trúc điều khiển như câu lệnh rẽ nhánh (if/else), vòng lặp (for, while), và hàm (function) để điều khiển luồng thực thi của chương trình.

JavaScript cũng hỗ trợ các kiểu dữ liệu cơ bản như số, chuỗi, mảng và đối tượng. Chúng ta có thể sử dụng các toán tử và phép tính để thực hiện các phép toán và xử lý dữ liệu trong JavaScript.

Một trong những khía cạnh quan trọng của JavaScript là khả năng tương tác với HTML và CSS. Có thể sử dụng JavaScript để thay đổi, thêm hoặc xóa các phần tử HTML, điều khiển các sự kiện tương tác như nhấp chuột, di chuyển và gửi dữ liệu đến máy chủ thông qua các yêu cầu HTTP.

Ngoài ra, JavaScript cung cấp một loạt các đối tượng và phương thức tích hợp sẵn (built-in) để thực hiện các chức năng phổ biến như xử lý chuỗi, thao tác với mảng, tạo đối tượng, và xử lý sự kiện. Cũng có thể sử dụng các thư viện và framework JavaScript như jQuery, React và Vue.js để mở rộng khả năng và hiệu suất của mã JavaScript của mình.

Ưu điểm của JavaScript:

- Khả năng tương tác trực tiếp với trình duyệt: JavaScript chủ yếu được sử dụng để tương tác với các yếu tố trên trang web và thực hiện các thay đổi trực tiếp mà không cần tải lại trang.
- Dễ học và triển khai: JavaScript là một ngôn ngữ dễ học, đặc biệt là đối với những người mới học lập trình. Nó cũng rất linh hoạt và có thể triển khai một cách nhanh chóng.
- Thư viện và Framework phong phú: JavaScript có nhiều thư viện và framework mạnh mẽ như React, Angular, Vue.js, jQuery, giúp giảm độ phức tạp của việc phát triển ứng dụng web.
- Chạy trên nhiều nền tảng: JavaScript có thể chạy trên nhiều nền tảng khác nhau, bao gồm cả trình duyệt web phổ biến như Chrome, Firefox, và Safari.
- Hỗ trợ đa dạng: JavaScript được hỗ trợ rộng rãi trên nhiều loại thiết bị và trình duyệt khác nhau, giúp đảm bảo tính tương thích.
- Cộng đồng lớn: Cộng đồng phát triển JavaScript rất lớn mạnh, điều này có nghĩa là có nhiều tài nguyên, hỗ trợ và thông tin trực tuyến.

Nhược điểm của JavaScript:

- Bảo mật: JavaScript chạy ở phía người dùng, điều này có nghĩa là mã nguồn có thể bị truy cập và sửa đổi bởi người dùng, đặt ra các vấn đề về bảo mật.
- Chậm trên trình duyệt cũ: Đối với những trình duyệt cũ hoặc thiết bị có tài nguyên hạn chế, JavaScript có thể chạy chậm, làm giảm trải nghiệm người dùng.
- Không an toàn (untyped): JavaScript là một ngôn ngữ không kiểu, điều này có thể dẫn đến các lỗi kiểu trong quá trình thực thi.
- Quản lý bộ nhớ: Quản lý bộ nhớ của JavaScript tự động, điều này có thể dẫn đến việc xuất hiện các vấn đề liên quan đến hiệu suất khi không được quản lý đúng.
- Khả năng xử lý đồng thời (concurrency): JavaScript thường sử dụng mô hình xử lý sự kiện và làm việc với callback, điều này có thể làm tăng độ phức tạp của mã nguồn trong các ứng dụng lớn.

- Không hỗ trợ đa luồng: JavaScript không có hỗ trợ cho đa luồng trực tiếp, điều này có thể ảnh hưởng đến hiệu suất trong một số trường hợp.

2.2. Backend

2.2.1. Python

Python là một ngôn ngữ lập trình thông dịch, đa mục đích và dễ đọc. Nó được thiết kế để tối ưu cho sự đơn giản và tăng cường hiệu suất lập trình. Python hỗ trợ nhiều paradigms lập trình, bao gồm lập trình hàm, lập trình hướng đối tượng và lập trình thủ tục, làm cho nó trở thành công cụ linh hoạt cho nhiều ứng dụng khác nhau.

Ngôn ngữ này có cú pháp đơn giản và rõ ràng, giúp người lập trình dễ dàng đọc, viết, và duy trì mã nguồn. Python cũng được cộng đồng lập trình đánh giá cao với nhiều thư viện và frameworks mạnh mẽ, như NumPy, Pandas, và Flask, giúp tăng cường khả năng xử lý dữ liệu, phân tích, và phát triển ứng dụng web.

Những lợi ích của Python bao gồm:

- Các nhà phát triển có thể dễ dàng đọc và hiểu một chương trình Python vì ngôn ngữ này có cú pháp cơ bản giống tiếng Anh.
- Python giúp cải thiện năng suất làm việc của các nhà phát triển vì so với những ngôn ngữ khác, họ có thể sử dụng ít dòng mã hơn để viết một chương trình Python.
- Python có một thư viện tiêu chuẩn lớn, chứa nhiều dòng mã có thể tái sử dụng cho hầu hết mọi tác vụ. Nhờ đó, các nhà phát triển sẽ không cần phải viết mã từ đầu.
- Các nhà phát triển có thể dễ dàng sử dụng Python với các ngôn ngữ lập trình phổ biến khác như Java, C và C++.
- Cộng đồng Python tích cực hoạt động bao gồm hàng triệu nhà phát triển nhiệt tình hỗ trợ trên toàn thế giới. Nếu gặp phải vấn đề, bạn sẽ có thể nhận được sự hỗ trợ nhanh chóng từ cộng đồng.

- Trên Internet có rất nhiều tài nguyên hữu ích nếu bạn muốn học Python. Ví dụ: bạn có thể dễ dàng tìm thấy video, chỉ dẫn, tài liệu và hướng dẫn dành cho nhà phát triển.
- Python có thể được sử dụng trên nhiều hệ điều hành máy tính khác nhau, chẳng hạn như Windows, macOS, Linux và Unix.

Các đặc điểm sau tạo nên sự độc đáo của ngôn ngữ lập trình Python:

- Python là một ngôn ngữ thông dịch: Python là một ngôn ngữ thông dịch, điều này nghĩa là ngôn ngữ này trực tiếp chạy từng dòng mã. Nếu có lỗi trong mã chương trình, nó sẽ ngừng chạy. Do đó, lập trình viên có thể nhanh chóng tìm ra lỗi trong đoạn mã.
- Python là một ngôn ngữ dễ sử dụng: Python sử dụng từ ngữ giống trong tiếng Anh. Không giống như các ngôn ngữ lập trình khác, Python không sử dụng dấu ngoặc ôm. Thay vào đó, ngôn ngữ này sử dụng thụt đầu dòng.
- Python là một ngôn ngữ linh hoạt: Các lập trình viên không cần phải khai báo loại biến khi viết mã bởi vì Python sẽ xác định chúng vào thời điểm chạy. Vì vậy, bạn có thể viết các chương trình Python một cách nhanh chóng hơn.
- Python là một ngôn ngữ cấp cao: Python gần gũi với ngôn ngữ con người hơn các ngôn ngữ lập trình khác. Do đó, các lập trình viên không cần phải lo lắng về những chức năng cơ bản của nó như kiến trúc và quản lý bộ nhớ.
- Python là một ngôn ngữ lập trình hướng đối tượng: Python coi mọi thứ đều là đối tượng, nhưng ngôn ngữ này cũng hỗ trợ các phương thức lập trình khác như lập trình hàm và lập trình cấu trúc.

2.2.2. Cơ sở dữ liệu MySQL

Cơ sở dữ liệu MySQL được sử dụng trong dự án để lưu trữ và quản lý dữ liệu. Sử dụng ngôn ngữ truy vấn SQL thông qua mysql.connector để thực hiện các thao tác như truy vấn, thêm, sửa, xóa dữ liệu. MySQL cung cấp tính năng an toàn và bảo mật, hỗ trợ mã hóa để bảo

vệ thông tin cơ sở dữ liệu. Điều này giúp tối ưu hóa hiệu suất bằng cách cấu hình và sử dụng chỉ mục cùng các kỹ thuật tối ưu hóa truy vấn. MySQL là một hệ quản lý cơ sở dữ liệu phổ biến, đáng tin cậy và phù hợp cho các ứng dụng đa dạng với nhu cầu lưu trữ và truy xuất dữ liệu.

Ưu điểm của MySQL:

- Dễ sử dụng: MySQL là cơ sở dữ liệu tốc độ cao, ổn định, dễ sử dụng và hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh.
- Độ bảo mật cao: MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên Internet khi sở hữu nhiều tính năng bảo mật thậm chí là ở cấp cao.
- Đa tính năng: MySQL hỗ trợ rất nhiều chức năng SQL được mong chờ từ một hệ quản trị cơ sở dữ liệu quan hệ cả trực tiếp lẫn gián tiếp.
- Khả năng mở rộng và mạnh mẽ: MySQL có thể xử lý rất nhiều dữ liệu và hơn thế nữa nó có thể được mở rộng nếu cần thiết.
- Nhanh chóng: Việc đưa ra một số tiêu chuẩn cho phép MySQL để làm việc rất hiệu quả và tiết kiệm chi phí, do đó nó làm tăng tốc độ thực thi.

Nhược điểm của MySQL:

- Giới hạn: Theo thiết kế, MySQL không có ý định làm tất cả và nó đi kèm với các hạn chế về chức năng mà một vào ứng dụng có thể cần.
- Độ tin cậy: Cách các chức năng cụ thể được xử lý với MySQL (ví dụ tài liệu tham khảo, các giao dịch, kiểm toán,...) làm cho nó kém tin cậy hơn so với một số hệ quản trị cơ sở dữ liệu quan hệ khác.
- Dung lượng hạn chế: Nếu số bản ghi của bạn lớn dần lên thì việc truy xuất dữ liệu của bạn là khá khó khăn, khi đó chúng ta sẽ phải áp dụng nhiều biện pháp để tăng tốc độ truy xuất dữ liệu như là chia tải database này ra nhiều server, hoặc tạo cache MySQL

Cách vận hành chính trong môi trường MySQL như sau:

- MySQL tạo ra bảng để lưu trữ dữ liệu, định nghĩa sự liên quan giữa các bảng đó.
- Client sẽ gửi yêu cầu SQL bằng một lệnh đặc biệt trên MySQL.
- Ứng dụng trên server sẽ phản hồi thông tin và trả về kết quả trên máy client.

2.2.3. *FastAPI*

FastAPI là một framework web nhanh chóng, dựa trên Python, được thiết kế để xây dựng API hiệu suất cao và dễ sử dụng. FastAPI sử dụng kiểu chú thích (type annotations) để tạo ra API mà không cần nhiều mã nguồn, đồng thời tự động tạo ra tài liệu API (OpenAPI và JSON Schema) và hỗ trợ xử lý bất đồng bộ với ASGI (Asynchronous Server Gateway Interface).

Nó tích hợp Pydantic để kiểm tra và serialize dữ liệu đầu vào và đầu ra của API, sử dụng kiểu chú thích Python để xác định kiểu dữ liệu. FastAPI cung cấp hỗ trợ xác thực OAuth2, giúp bảo vệ API và quản lý quyền truy cập, và có khả năng xử lý Websockets và GraphQL, mở ra khả năng xây dựng ứng dụng thời gian thực và giao diện tương tác động. Được xây dựng trên nền tảng Starlette và hỗ trợ ASGI, FastAPI cho phép tích hợp dễ dàng với nhiều middleware và tiện ích từ cộng đồng, làm cho việc phát triển ứng dụng web API trở nên nhanh chóng và hiệu quả.

Các tính năng chính:

- Nhanh (Fast): Từ chính document của framework có thể tự tin so sánh tốc độ ngang bằng với NodeJs và Go (mình sẽ làm benchmark để kiểm chứng điều này trong những phần sau - phải thấy mới tin được).
- Triển khai nhanh (Fast to code): Tăng tốc độ triển khai các tính năng từ 200% đến 300% (cá nhân mình cũng cảm thấy điều đó, các bạn có thể trải nghiệm để cảm nhận).
- Ít lỗi hơn (Fewer bugs): Giảm khoảng 40% lỗi do lập trình viên gây ra (mình đoán có thể do một số tính đã được bao đóng và triển khai sẵn nên làm giảm lỗi phát sinh trong quá trình lập trình, và có thể có thêm một số nguyên nhân khác, bla bla...).

- Trực quan và tiện dụng (Intuitive): Được nhiều trình chỉnh sửa hỗ trợ, có thể chạy trên nhiều nền tảng khác nhau. Tốn ít thời gian gỡ lỗi hơn (hưởng lợi từ Python là một ngôn ngữ thông dịch)
- Dễ dàng tiếp cận (Easy): Thiết kế dễ sử dụng và hỏi tập nhanh thông qua docs.
- Code ngắn gọn (Short): Code ngắn gọn, bao gồm nhiều tính năng đã được triển khai sẵn và giúp giảm thiểu bugs.
- Mạnh mẽ (Robust): Code sẵn sàng cho môi trường production và API docs được sinh tự động.

2.2.4. KNN (*K-Nearest Neighbors*)

K-nearest neighbor là một trong những thuật toán supervised-learning đơn giản nhất (mà hiệu quả trong một vài trường hợp) trong Machine Learning. Khi training, thuật toán này không học một điều gì từ dữ liệu training (đây cũng là lý do thuật toán này được xếp vào loại lazy learning), mọi tính toán được thực hiện khi nó cần dự đoán kết quả của dữ liệu mới. K-nearest neighbor có thể áp dụng được vào cả hai loại của bài toán Supervised learning là Classification và Regression. KNN còn được gọi là một thuật toán Instance-based hay Memory-based learning.

Với KNN, trong bài toán Classification, label của một điểm dữ liệu mới (hay kết quả của câu hỏi trong bài thi) được suy ra trực tiếp từ K điểm dữ liệu gần nhất trong training set. Label của một test data có thể được quyết định bằng major voting (bầu chọn theo số phiếu) giữa các điểm gần nhất, hoặc nó có thể được suy ra bằng cách đánh trọng số khác nhau cho mỗi trong các điểm gần nhất đó rồi suy ra label. Chi tiết sẽ được nêu trong phần tiếp theo.

Trong bài toán Regresssion, đầu ra của một điểm dữ liệu sẽ bằng chính đầu ra của điểm dữ liệu đã biết gần nhất (trong trường hợp $K=1$), hoặc là trung bình có trọng số của đầu ra của những điểm gần nhất, hoặc bằng một mối quan hệ dựa trên khoảng cách tới các điểm gần nhất đó.

Một cách ngắn gọn, KNN là thuật toán đi tìm đầu ra của một điểm dữ liệu mới bằng cách chỉ dựa trên thông tin của K điểm dữ liệu trong training set gần nó nhất (K-lân cận), không quan tâm đến việc có một vài điểm dữ liệu trong những điểm gần nhất này là nhiều.

Ưu điểm của KNN

- Độ phức tạp tính toán của quá trình training là bằng 0.
- Việc dự đoán kết quả của dữ liệu mới rất đơn giản.
- Không cần giả sử gì về phân phối của các class.

Nhược điểm của KNN

- KNN rất nhạy cảm với nhiễu khi K nhỏ.

Tổng quan, KNN là một thuật toán mà mọi tính toán đều nằm ở khâu test. Trong đó việc tính khoảng cách tới từng điểm dữ liệu trong training set sẽ tốn rất nhiều thời gian, đặc biệt là với các cơ sở dữ liệu có số chiều lớn và có nhiều điểm dữ liệu. Với K càng lớn thì độ phức tạp cũng sẽ tăng lên. Ngoài ra, việc lưu toàn bộ dữ liệu trong bộ nhớ cũng ảnh hưởng tới hiệu năng của KNN.

2.2.5. Boto3

Boto3 là một thư viện lập trình Python được phát triển bởi Amazon Web Services (AWS) để hỗ trợ việc tương tác với và quản lý các dịch vụ điện toán đám mây của AWS thông qua API. Được xây dựng trên cơ sở của Botocore, thư viện cơ sở cho tất cả các thư viện SDK (Software Development Kit) của AWS, Boto3 cung cấp một giao diện thuận tiện cho các nhà phát triển để thực hiện các thao tác như tạo, đọc, cập nhật, và xóa tài nguyên trên AWS

Boto3 hỗ trợ nhiều dịch vụ của AWS, bao gồm nhưng không giới hạn vào lưu trữ đối tượng với Amazon S3, tính toán đám mây với Amazon EC2, quản lý cơ sở dữ liệu với Amazon DynamoDB và Amazon RDS, cũng như nhiều dịch vụ khác. Điều này cho phép

nhà phát triển xây dựng và triển khai ứng dụng trên AWS và thực hiện các tác vụ quản lý tài nguyên một cách hiệu quả.

Các khái niệm chính trong Boto3 bao gồm "Session" để quản lý thông tin xác thực và cấu hình, "Resource" và "Client" để tương tác với tài nguyên và dịch vụ AWS, "Operation" để thực hiện các hoạt động cụ thể trên dịch vụ, và "Paginator" để xử lý kết quả trả về qua nhiều trang.

Boto3 không chỉ giúp tự động hóa các nhiệm vụ quản lý tài nguyên, mà còn mang lại tính linh hoạt cho các nhà phát triển khi xây dựng ứng dụng đám mây, bằng cách cung cấp các phương thức và tiện ích đa dạng để tương tác với nền tảng AWS.

Ưu điểm của Boto3:

- Dễ sử dụng và Linh hoạt: Boto3 cung cấp một giao diện lập trình Python thuận tiện, giúp nhà phát triển tương tác với các dịch vụ AWS một cách dễ dàng và linh hoạt.
- Hỗ trợ Rộng Rãi: Boto3 hỗ trợ nhiều dịch vụ khác nhau của AWS, bao gồm lưu trữ, tính toán, cơ sở dữ liệu, trí tuệ nhân tạo và nhiều hơn nữa, giúp phục vụ nhu cầu đa dạng của ứng dụng.
- Tích Hợp Mạnh Mẽ: Boto3 được tích hợp chặt chẽ với các dịch vụ AWS, điều này giúp đảm bảo rằng thư viện luôn cập nhật với các tính năng mới và thay đổi trong các dịch vụ.
- Cộng Đồng Hỗ Trợ Lớn: Do được phát triển và duy trì bởi AWS, Boto3 có sự hỗ trợ lớn từ cộng đồng và nhận được sự chăm sóc chính thức từ Amazon, giúp đảm bảo tính ổn định và bảo mật.
- Dữ Liệu Tương Thích (Data-Driven): Boto3 hỗ trợ các loại dữ liệu phổ biến như JSON và các đối tượng Python, tạo điều kiện thuận lợi cho việc làm việc với dữ liệu trong các ứng dụng Python.

Nhược điểm của Boto3:

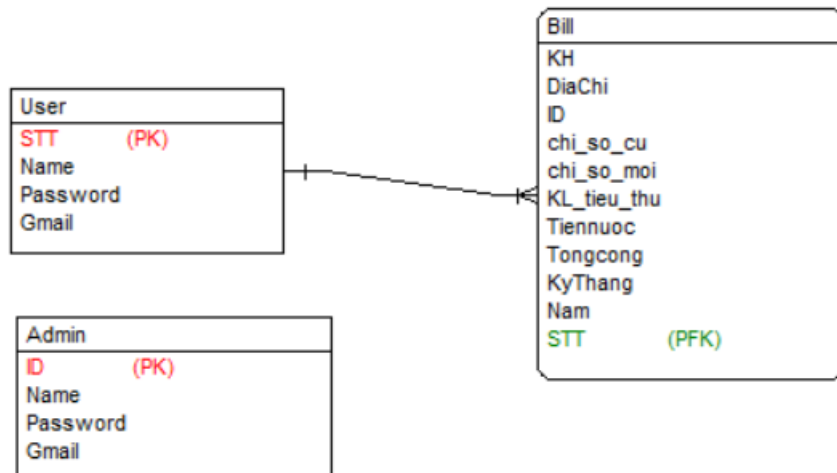
- Tài Nguyên Nhẹ và Chậm Chạp: Mặc dù Boto3 cung cấp tính năng mạnh mẽ, nhưng có thể gặp phải hiệu suất không đạt yêu cầu trong một số tình huống, đặc biệt là khi cần xử lý lượng lớn các tài nguyên.
- Tài Liệu Khó Hiểu: Mặc dù có nhiều tài liệu và ví dụ, nhưng một số người mới sử dụng Boto3 có thể gặp khó khăn trong việc hiểu và sử dụng các tính năng phức tạp của thư viện.
- Phụ Thuộc vào Internet: Một số tính năng của Boto3 có thể đòi hỏi kết nối internet để tải về các dữ liệu mô tả dịch vụ từ các nguồn khác nhau, điều này có thể tạo ra các vấn đề khi triển khai trong môi trường không có kết nối internet.
- Khả Năng Mở Rộng Hạn Chế: Trong một số trường hợp, khả năng mở rộng của Boto3 có thể trở nên hạn chế, đặc biệt là khi ứng dụng cần xử lý hàng nghìn yêu cầu mỗi giây.
- Quản Lý Phiên Khó Khăn: Quản lý phiên và xác thực có thể trở nên phức tạp đối với các ứng dụng lớn và phức tạp, đặc biệt là khi sử dụng nhiều tài khoản và quyền truy cập khác nhau.

Ngoài ra, sử dụng các thư viện trong Python như: PIL, bcrypt, pandas, sqlite3,...

CHƯƠNG 3. KẾT QUẢ THỰC NGHIỆM

3.1. Phân tích thiết kế hệ thống

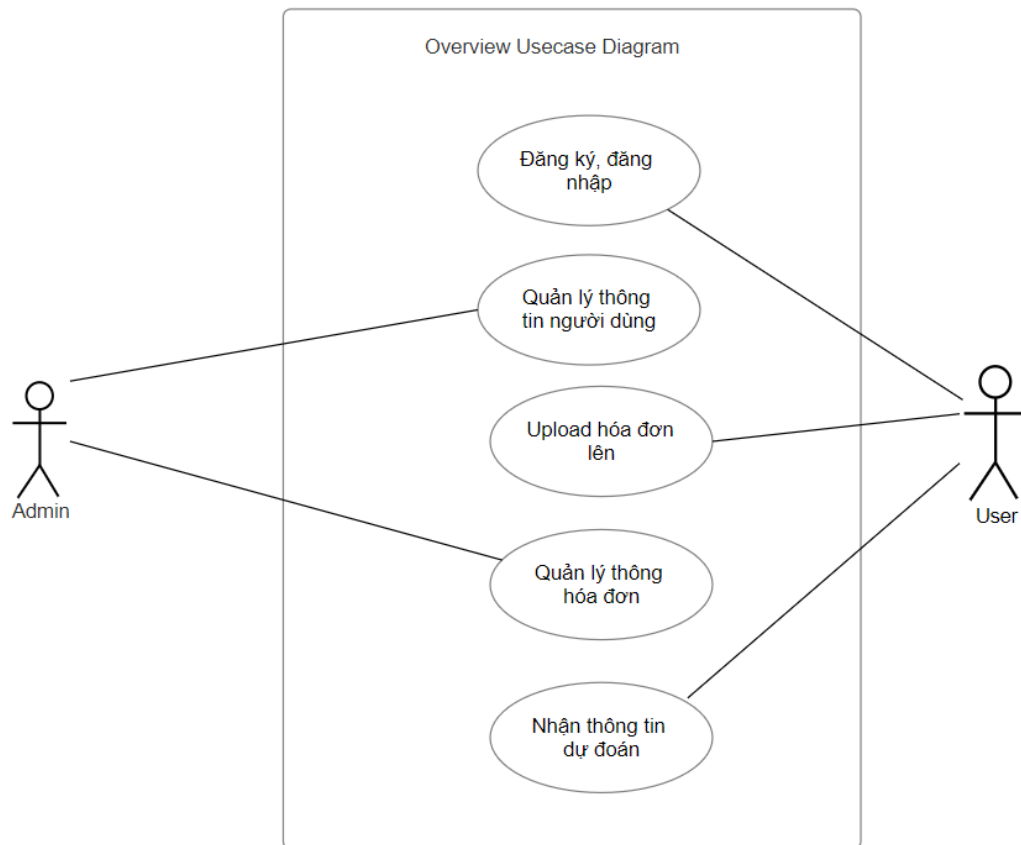
3.1.1. Mô hình thực thể quan hệ (ERD):



Hình 1: Mô hình thực thể quan hệ ERD

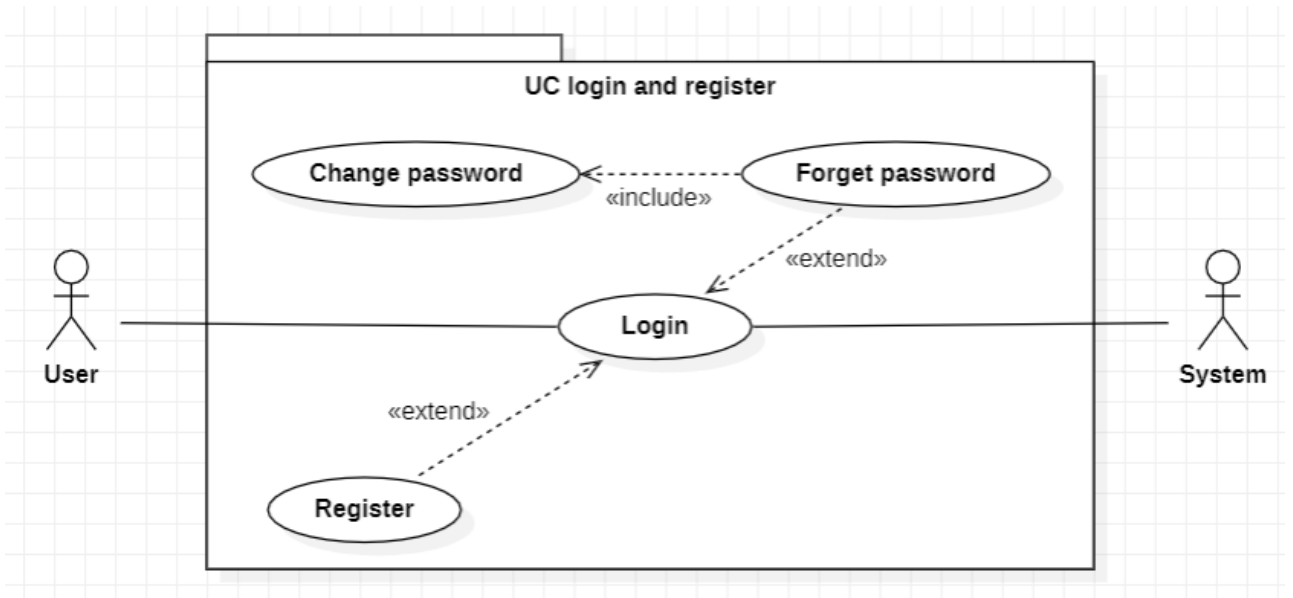
3.1.2. Sơ đồ Usecase:

3.1.2.1. Use case tổng quát:



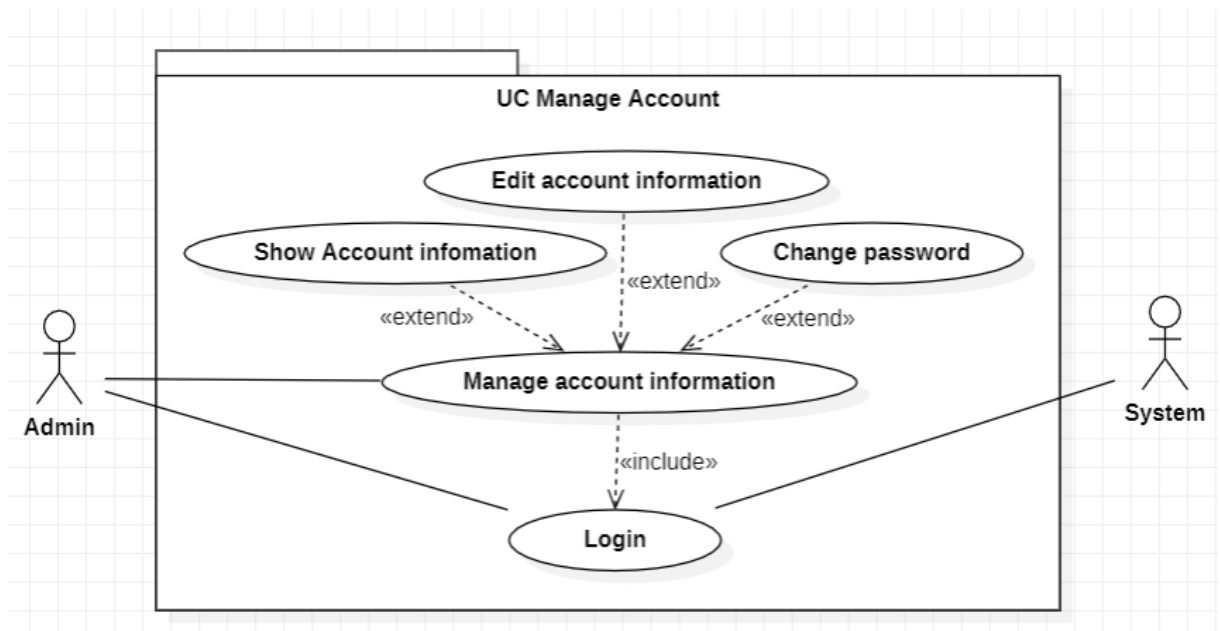
Hình 2: Usecase Tổng quát

3.1.2.2. Usecase đăng nhập:



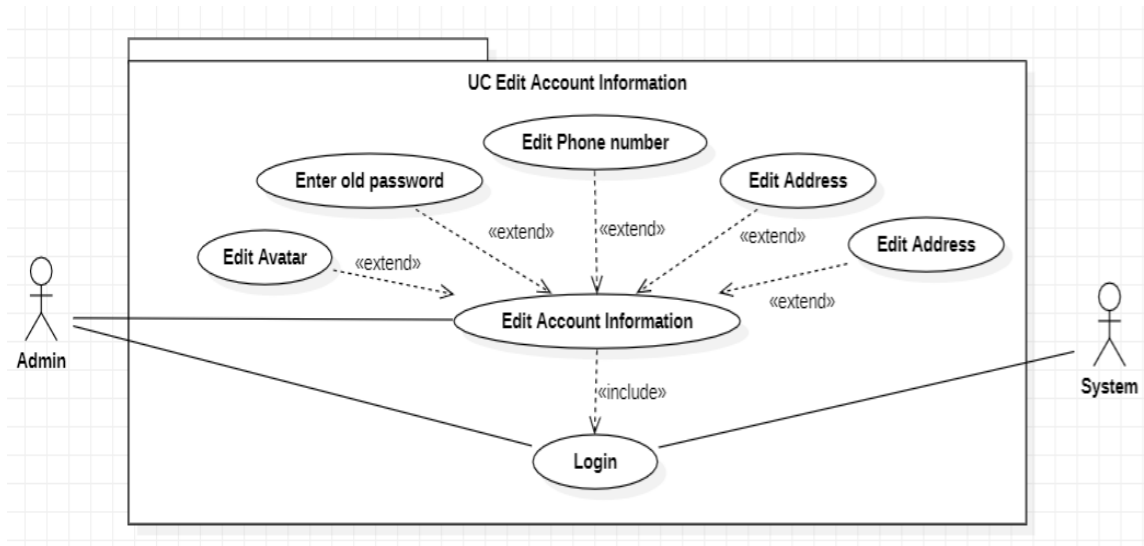
Hình 3: Usecase đăng nhập

3.1.2.3. Use case quản lý thông tin tài khoản:



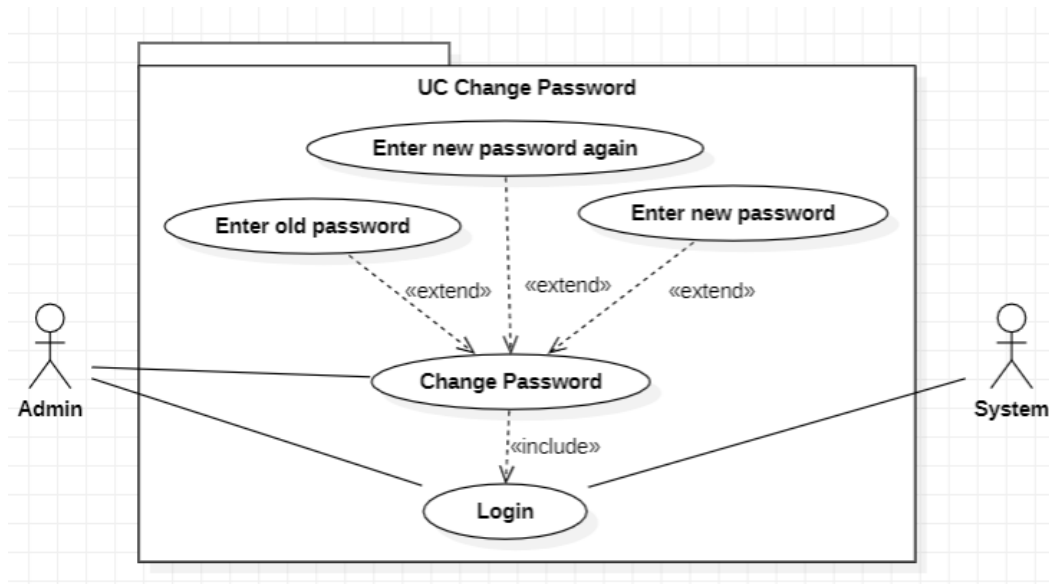
Hình 4: Usecase quản lý thông tin tài khoản

3.1.2.4. Use case chỉnh sửa thông tin tài khoản:



Hình 5: Usecase chỉnh sửa thông tin tài khoản

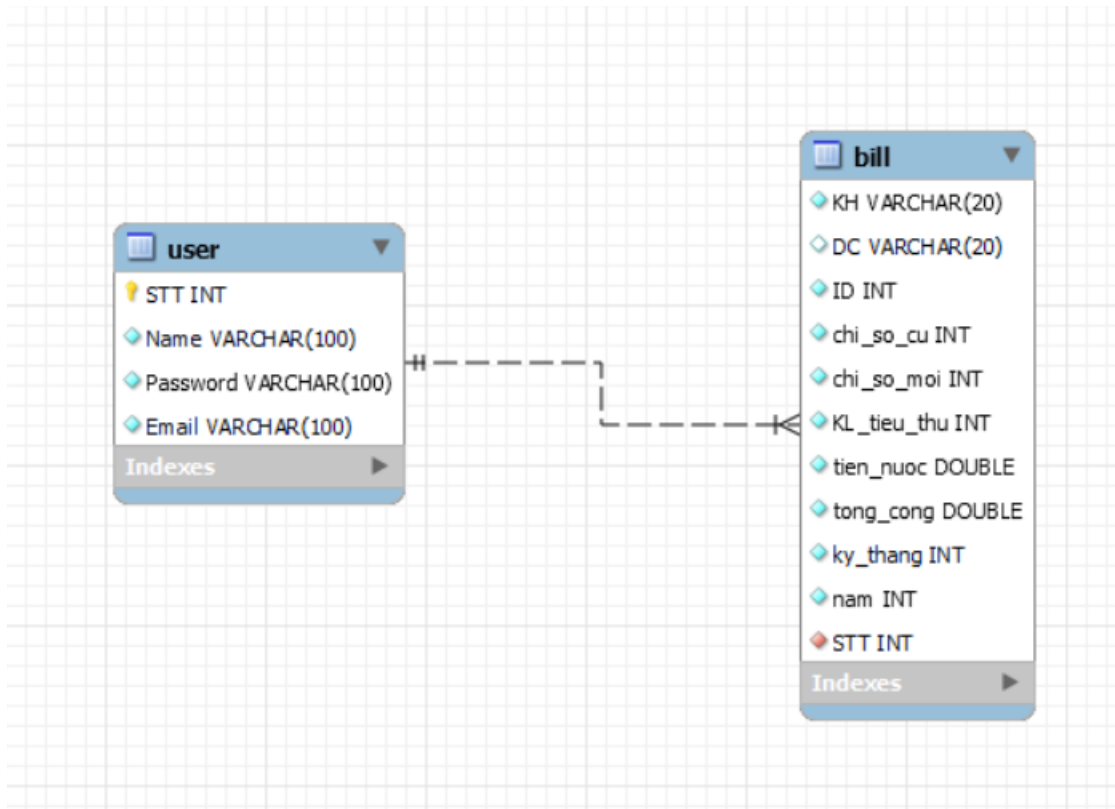
3.1.2.5. Usecase đổi mật khẩu:



Hình 6: Usecase đổi mật khẩu

3.1.3. Mô hình cơ sở dữ liệu:

3.1.3.1. Sơ đồ trong MSSQL:



Hình 7: Sơ đồ Diagram trong MSSQL

3.1.3.2. Giải thích thuộc tính trong cơ sở dữ liệu:

- Bảng Bill

STT	Tên Thuộc tính	Kiểu dữ liệu	Độ dài	Khóa	Mô tả
1	KH	Nvarchar	20		Tên người dùng
2	DC	Varchar	20		Địa chỉ người dùng
3	ID	Int			ID người dùng
4	Chi_so_cu	Int			Chỉ số cũ tiền nước

5	Chi_so_moi	Int			Chỉ số mới tiền nước
6	KL_tieu_thu	Int			Khối lượng tiêu thụ
7	Tien_nuoc	double			Tiền nước tiêu thụ
8	Tong_cong	double			Tổng cộng
9	Ky_thang	Int			Số tháng
10	Nam	Int			Năm
11	STT	Int			STT người dùng

Bảng 1: Bảng Bill

- Bảng User

STT	Tên Thuộc tính	Kiểu dữ liệu	Độ dài	Khóa	Mô tả
1	STT	Int		PK	ID Người dùng
2	Name	Nvarchar	100		Tên người dùng
3	Password	Varchar	100		Mật khẩu
4	Email	Nvarchar	100		Email người dùng

Bảng 2: User

3.2. Giao diện Website

3.2.1. Các giai đoạn hoàn thành website:

Giai đoạn 1. Tìm hiểu đề tài, thu thập và phân tích yêu cầu của khách hàng. Điều này bao gồm việc xác định chức năng, giao diện, tính năng đặc biệt và yêu cầu kỹ thuật của website phân tích và dự đoán chi tiêu cá nhân.

Giai đoạn 2. Thiết kế: Tạo ra bản thiết kế giao diện cho website. Điều này bao gồm thiết kế trang chủ, các trang đăng ký, đăng nhập, trang upload, trang dự đoán, và các trang khác liên quan. Thiết kế cũng bao gồm việc xác định cấu trúc thông tin, sơ đồ dữ liệu và quy trình upload hóa đơn.

Giai đoạn 3. Phát triển frontend: Xây dựng giao diện người dùng (frontend) của website. Sử dụng các ngôn ngữ và công cụ như HTML, CSS và JavaScript để tạo ra các trang web tương tác và hấp dẫn.

Giai đoạn 4. Phát triển backend: Xây dựng hệ thống backend của website phân tích và dự đoán chi tiêu cá nhân. Điều này bao gồm việc lập trình các chức năng đăng nhập, đăng ký, quản lý người dùng, quản lý hóa đơn, xử lý và tích hợp với các hệ thống dự đoán.

Giai đoạn 5. Xây dựng cơ sở dữ liệu: Một website phân tích và dự đoán chi tiêu cá nhân thường cần một cơ sở dữ liệu để lưu trữ thông tin hóa đơn, thông tin người dùng và các giao dịch. Giai đoạn này liên quan đến thiết kế và triển khai cơ sở dữ liệu, bao gồm việc tạo bảng, quan hệ, chỉ mục và các truy vấn cần thiết.

Giai đoạn 6. Kiểm thử và gỡ lỗi: Sau khi hoàn thành phát triển, website phân tích và dự đoán chi tiêu cá nhân cần được kiểm thử và gỡ lỗi để đảm bảo tính ổn định và chất lượng. Các kiểm thử bao gồm kiểm thử chức năng, kiểm thử tương thích và kiểm thử hiệu năng.

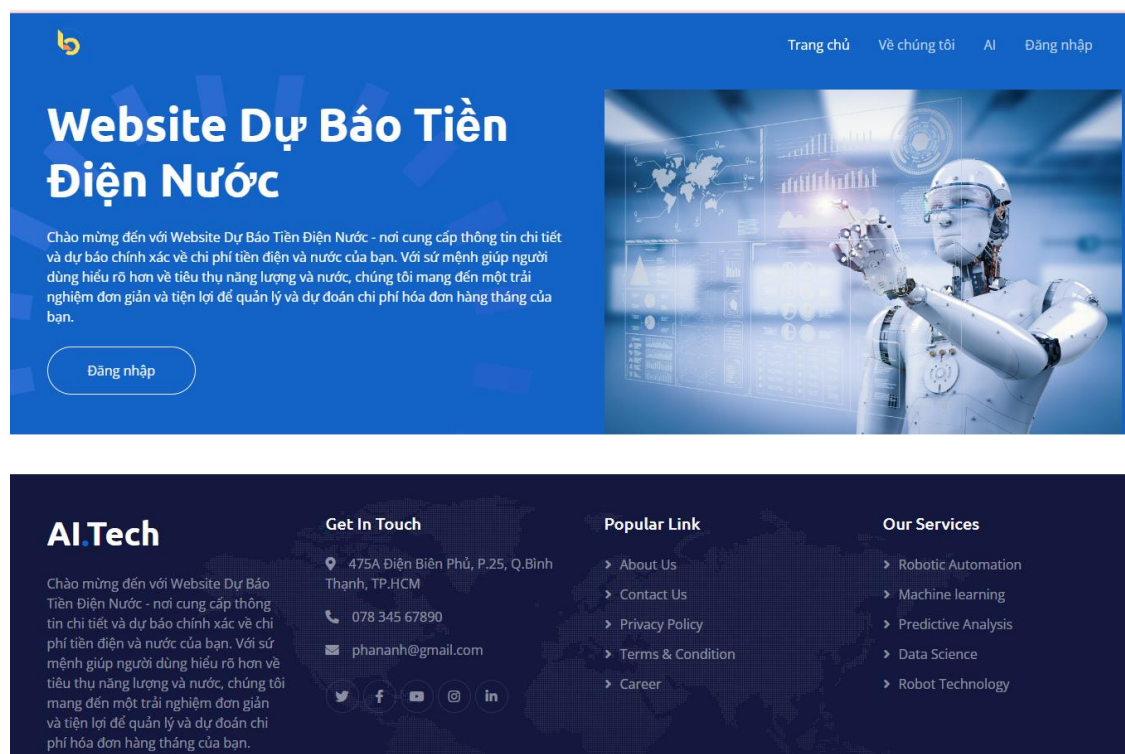
3.2.2. Giao diện người dùng:

3.2.2.1. Trang chủ:

Khi người dùng truy cập vào trang web, họ sẽ thấy trang chủ với một thanh điều hướng rõ ràng, bao gồm các chức năng như đăng nhập, đăng ký, AI, và trang về chúng tôi.

Người dùng có thể dễ dàng đăng nhập, đăng ký tài khoản để truy cập vào trang chính.

Cuối cùng, ở chân trang, người dùng có thể tìm thấy các thông tin cơ bản về Website TechAI.



Hình 8: Giao diện Trang chủ

3.2.2.2. Đăng ký, đăng nhập:

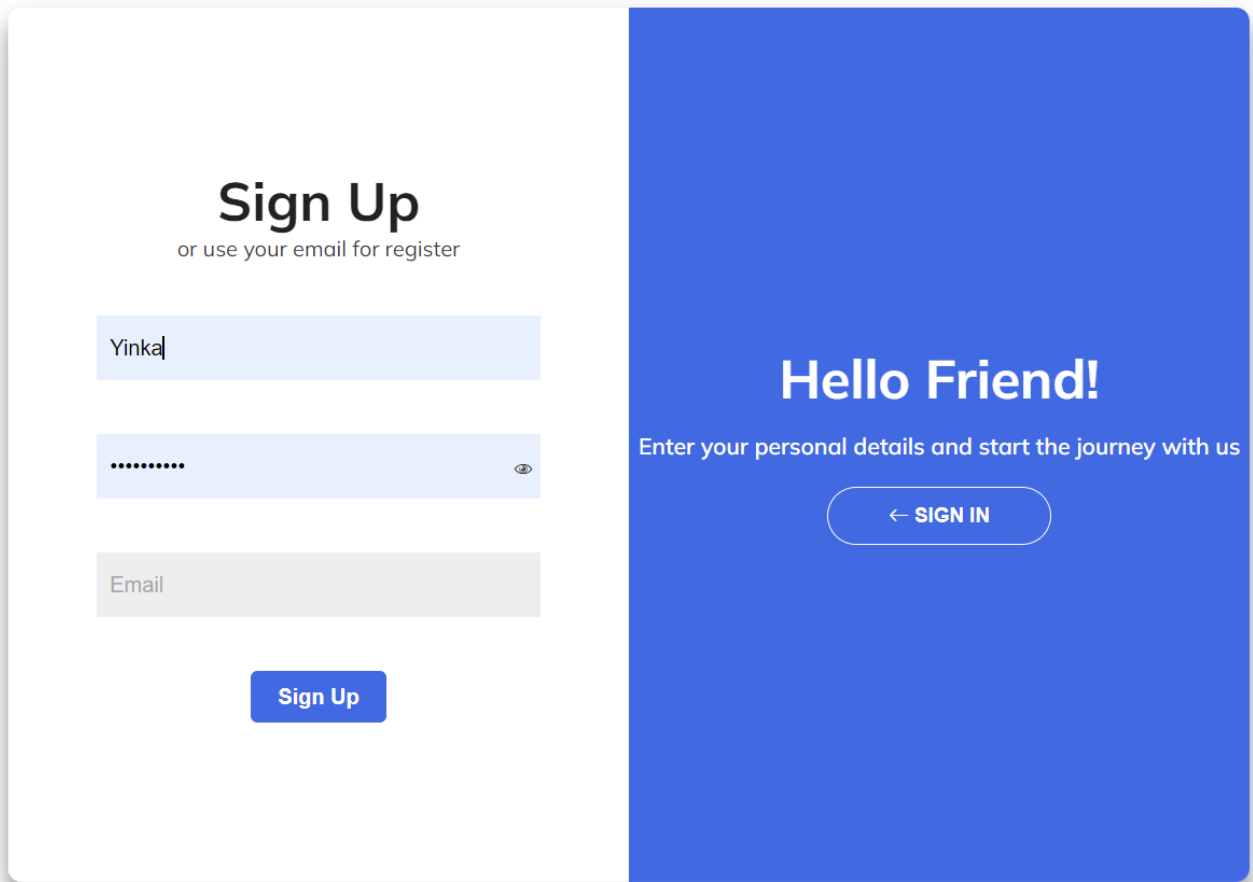
Khi người dùng nhấp vào nút "Đăng nhập" trên trang menu, họ sẽ được chuyển hướng đến trang đăng nhập vào hệ thống. Tại đây, người dùng sẽ cần điền thông tin tài khoản của mình vào các trường tương ứng. Nếu người dùng nhập sai hoặc bỏ trống thông tin, hệ thống sẽ từ chối đăng nhập và thông báo cho người dùng rằng tên đăng nhập hoặc mật khẩu không chính xác.

Nếu người dùng nhập đúng thông tin và nhấp vào nút "Đăng nhập", hệ thống sẽ kiểm tra thông tin và chuyển hướng người dùng đến trang chủ của trang web. Tại đây, khách hàng có thể sử dụng đầy đủ các chức năng của hệ thống.

Nếu người dùng chưa có tài khoản, họ có thể nhấp vào nút "Đăng ký" để tạo tài khoản mới. Người dùng sẽ điền đầy đủ các thông tin yêu cầu. Nếu người dùng nhập không đúng định dạng hoặc bỏ trống thông tin, hệ thống sẽ từ chối việc đăng ký và thông báo lỗi tương ứng. Sau khi nhập đầy đủ thông tin và nhấp vào nút "Đăng ký", hệ thống sẽ kiểm tra xem người dùng đã tồn tại trong cơ sở dữ liệu chưa. Nếu chưa tồn tại, hệ thống cho phép người dùng đăng ký. Trong trường hợp đã tồn tại, hệ thống sẽ thông báo rằng tài khoản đã có trong cơ sở dữ liệu.

The image shows a login interface. On the left, a white panel contains the 'Sign In' section. It has the text 'Sign In' in bold, followed by 'or use your account' in smaller text. Below this are two light blue input fields: the first contains the text 'Yinka', and the second contains a series of dots for a password, with an eye icon to its right. A blue 'Sign In' button is positioned below the password field, with a blue link 'Quên mật khẩu?' underneath it. On the right, a solid blue panel contains the text 'Welcome Back!' in large white font, followed by 'To keep with us please login with your personal info' in smaller white font. At the bottom of this panel is a white rounded button with the text 'SIGN UP →' in blue.

Hình 9: Giao diện trang Đăng Nhập

The image shows a user registration and login interface. On the left, a white panel contains a 'Sign Up' form. The form has three input fields: a first name field with 'Yinka' entered, a password field with masked characters and a toggle icon, and an email field with the placeholder 'Email'. Below these fields is a blue 'Sign Up' button. Above the first name field is the text 'or use your email for register'. On the right, a blue panel contains a 'Hello Friend!' greeting, a prompt to 'Enter your personal details and start the journey with us', and a white button with a left arrow and the text 'SIGN IN'.


Hình 10: Giao diện trang Đăng ký

3.2.2.3. Trang upload hóa đơn:

Tại trang này, người dùng sẽ upload các hóa đơn tiền nước hàng tháng của mình vào. Sau khi người dùng bấm tải hóa đơn lên, sẽ tự động đọc nội dung từ hóa đơn.

Dữ liệu sau khi tải lên website sẽ được lưu vào database nằm trong bảng ‘bill’.

Sau khi người dùng đã tải hóa đơn của mình theo định kỳ từng tháng. Và muốn xem dự đoán tiền điện nước hóa đơn tháng tiếp theo, thì họ sẽ bấm nút “Xem số liệu phân tích của tôi”.

Trang chủVề chúng tôiAIWelcome, PhanThiAnh! | Đăng xuất

OCR Hóa đơn

Choose File

12345(8).jpg

Tải lên

Kết quả:

Key	Value
KH	PhanThiAnh
DC	HCM
ID	123456
chi_so_cu	1859
chi_so_moi	1882
KL_tieu_thu	23
tien_nuoc	198
tong_cong	285165
ky_thang	8
nam	2023

Xem số liệu phân tích của tôi

Hình 11: Giao diện trang upload hóa đơn

3.2.2.4. Trang phân tích dữ liệu:

Đây là trang hiển thị toàn bộ thông tin dữ liệu hóa đơn theo từng tháng mà người dùng đã upload lên. Bên dưới sẽ là chi phí dự đoán cho 12 tháng tiếp theo trong tương lai.



Dự đoán tiền nước của bạn

Bảng thống kê hóa đơn của bạn

PhanThiAnh	HCM	123456	1699	1721	22	187.0	274.0	1	2023	xóa
PhanThiAnh	HCM	123456	1721	1745	24	210.0	287.0	2	2023	xóa
PhanThiAnh	HCM	123456	1745	1764	19	154.0	241.0	3	2023	xóa
PhanThiAnh	HCM	123456	1764	1785	21	175.0	263.0	4	2023	xóa
PhanThiAnh	HCM	123456	1785	1809	24	210.0	297.0	5	2023	xóa
PhanThiAnh	HCM	123456	1809	1834	25	221.0	309.0	6	2023	xóa
PhanThiAnh	HCM	123456	1859	1882	23	198.0	285165.0	8	2023	xóa
PhanThiAnh	HCM	123456	1620	1640	20	164.0	251.0	9	2022	xóa
PhanThiAnh	HCM	123456	1640	1659	19	154.0	241.0	10	2022	xóa
PhanThiAnh	HCM	123456	1659	1679	20	164.0	251.0	11	2022	xóa
PhanThiAnh	HCM	123456	1679	1699	20	164.0	251.0	12	2022	xóa
PhanThiAnh	HCM	123456	1834	1859	25	221.0	309.0	7	2023	xóa

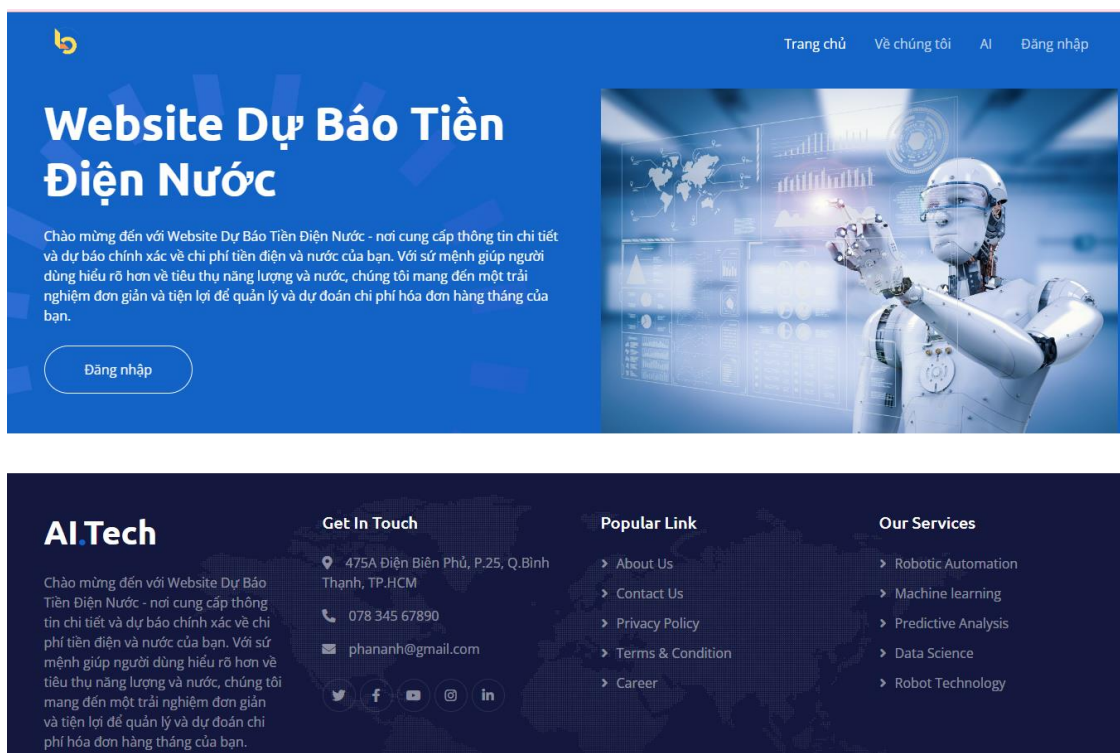
Kết quả dự đoán cho tháng tiếp theo:

Chi phí: [202.]

Hình 12: Giao diện trang phân tích dữ liệu

3.2.2.5. Trang về chúng tôi:

Giới thiệu thông tin nhà cung cấp như email hỗ trợ, số điện thoại, địa chỉ để người dùng hoặc người có nhu cầu có thể liên hệ.

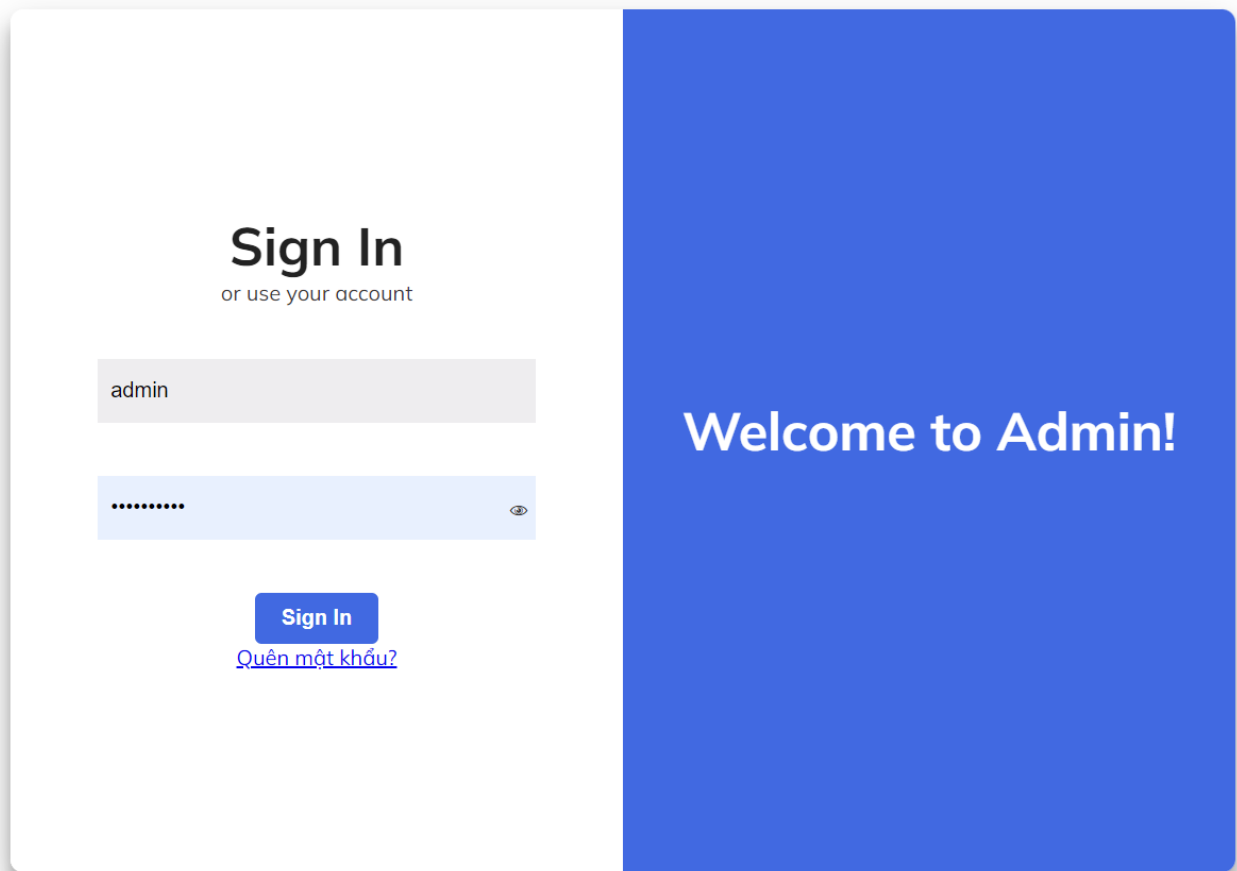


Hình 13: Giao diện về chúng tôi

3.2.3. Giao diện admin:

3.2.3.1. Trang đăng nhập admin:

Trang đăng nhập Admin cung cấp một biểu mẫu đơn giản để người dùng nhập tài khoản và mật khẩu của mình. Sau khi người dùng điền đúng thông tin tài khoản, nếu tài khoản đó được cấp quyền quản trị, họ sẽ được chuyển hướng vào trang quản trị Admin. Trong trường hợp người dùng không nhập đúng tài khoản hoặc mật khẩu, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng nhập lại thông tin.

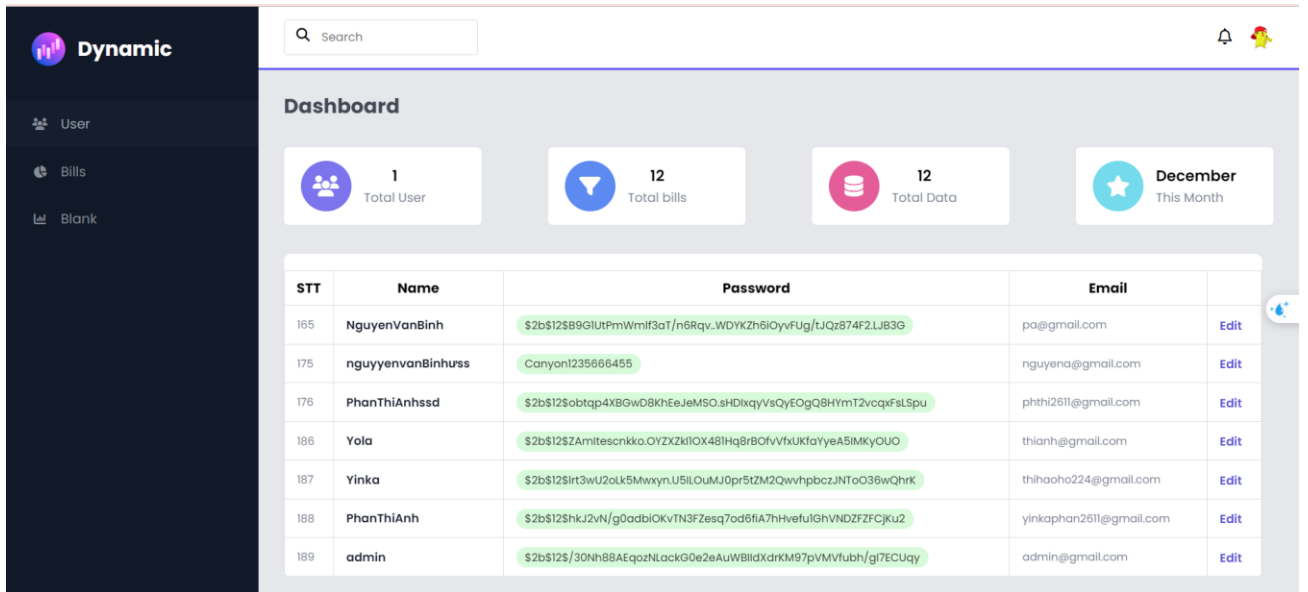


Hình 14: Giao diện trang Đăng nhập tài khoản Admin

3.2.3.2. Trang chủ admin:

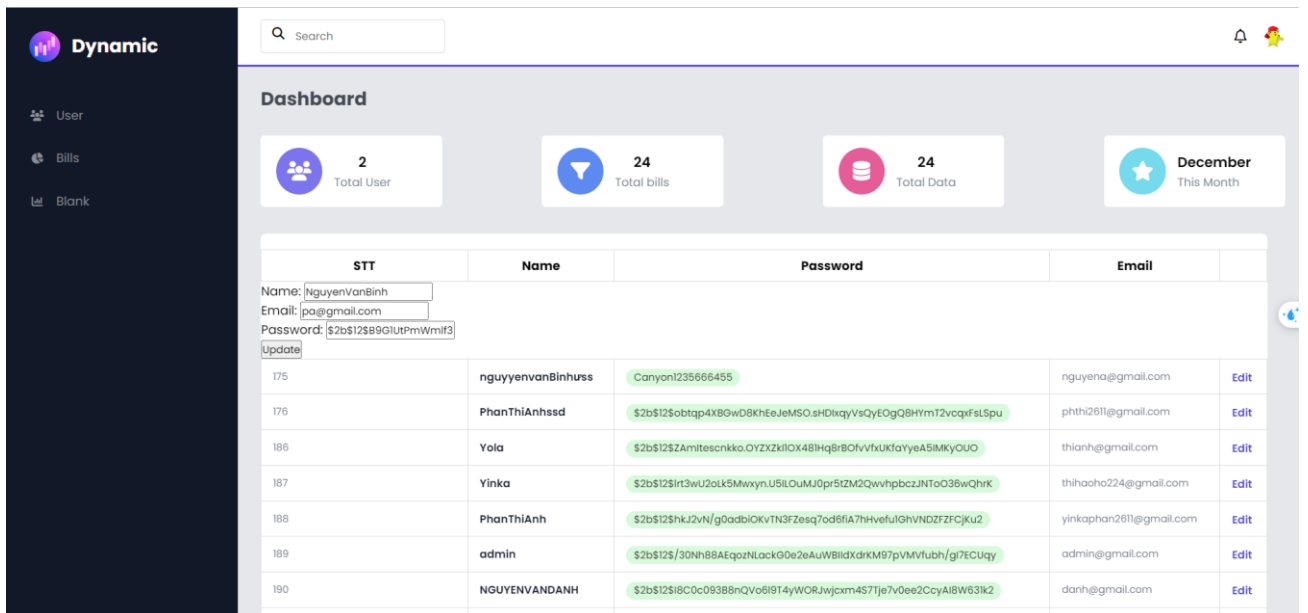
Trang tổng quan Admin cung cấp một bức tranh toàn cảnh về hoạt động của trang web, hiển thị các chỉ số quan trọng như số lượng người dùng, số liệu hóa đơn tải lên, thông tin user, thông tin hóa đơn của mỗi user,....

Trang tổng quan này giúp người quản trị dễ dàng theo dõi và đánh giá hiệu suất kinh doanh của trang web. Thông qua việc hiển thị số liệu và biểu đồ, người quản trị có thể nhanh chóng nhìn thấy xu hướng và biến động của các chỉ số quan trọng. Điều này giúp họ đưa ra các quyết định và chiến lược phù hợp để tăng cường hiệu quả kinh doanh và đáp ứng nhu cầu của khách hàng.



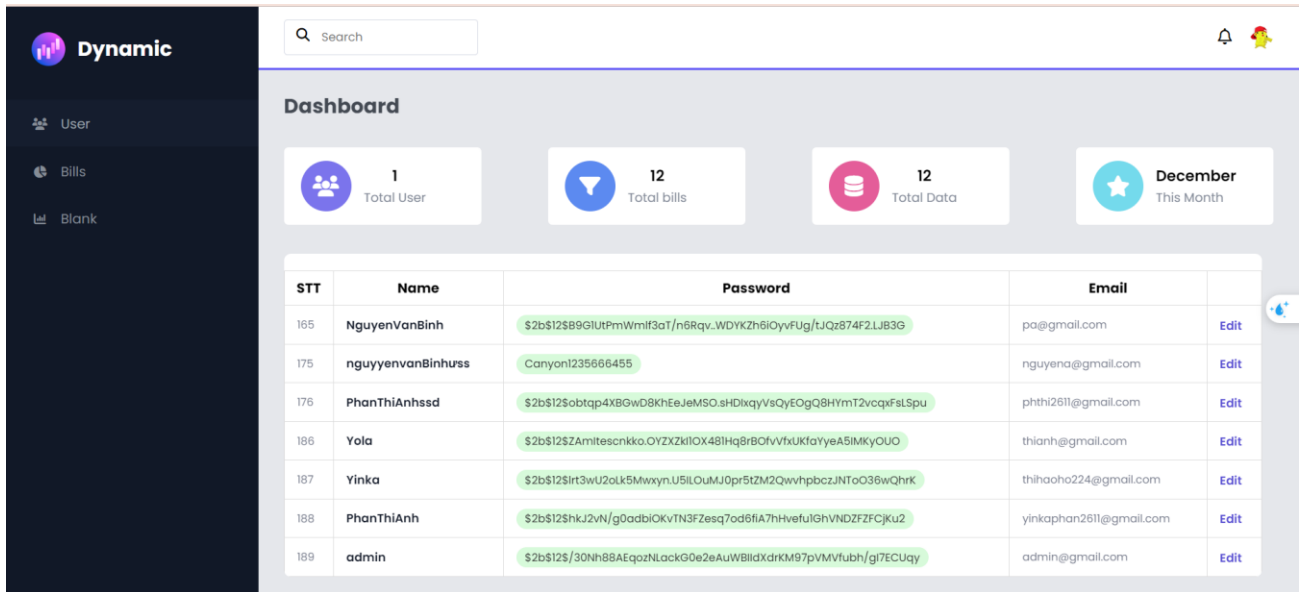
Hình 15: Giao diện trang chủ Admin

3.2.3.3. Trang chỉnh sửa thông tin người dùng



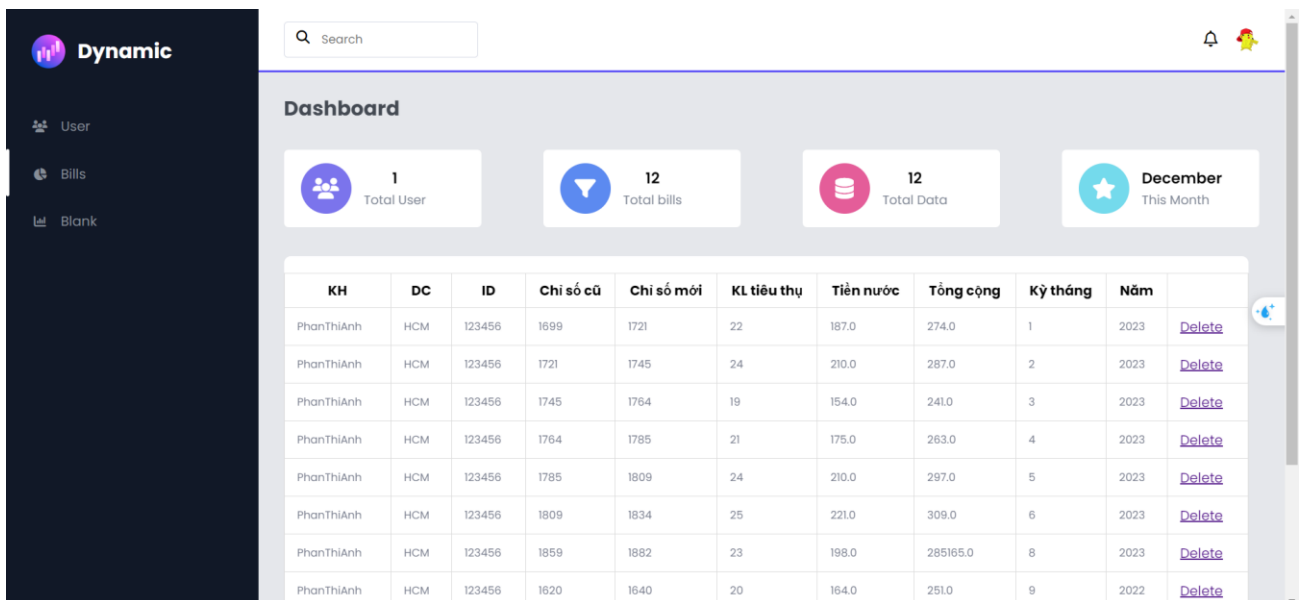
Hình 16: Trang chỉnh sửa thông tin người dùng

3.2.3.4. Trang quản lý người dùng:



Hình 17: Giao diện trang quản lý người dùng

3.2.3.5. Trang quản lý hóa đơn:



Hình 18: Giao diện trang quản lý hóa đơn

3.2.3.6. Trang quản lý số liệu phân tích của từng user:

Hiển thị thông tin user đã được lưu trong cơ sở dữ liệu.

Admin có thể thêm người dùng mới, xóa hoặc sửa người đã có

Admin có thể thay đổi số liệu trong hóa đơn từng tháng của user

The screenshot displays a web dashboard for 'Dynamic'. On the left is a dark sidebar with navigation links: 'User', 'Bills', and 'Blank'. The main area features a 'Dashboard' header with a search bar and notification icons. Below the header are four summary cards: 'Total User' (1), 'Total bills' (12), 'Total Data' (12), and 'December This Month'. The central part of the dashboard contains a table with user details.

STT	Name	Password	Email	
165	NguyenVanBinh	\$2b\$12\$B9GIutPmWmif3aT/n6Rqv..WDYkZh6iOyvfUg/tJQz874f2LJB3G	pa@gmail.com	Phân tích
175	nguyenvanBinhuus	CanyonI235666455	nguyena@gmail.com	Phân tích
176	PhanThiAnhssd	\$2b\$12\$obtap4XBGwD8KhEeJeMSO.sHDixqYVsQyEOgQ8HYmT2vcqxfSL\$pu	phthi261@gmail.com	Phân tích
186	Yola	\$2b\$12\$ZAmitecnkko.OYZXkI0X48IHq8rBOfvfXkfaYyeA5IMkyOUO	thianh@gmail.com	Phân tích
187	Yinka	\$2b\$12\$irt3wU2oLk5Mwxyn.U5iLOuMJ0pr5tZM2QwvhpbczJNTao36wQhrK	thihaoho224@gmail.com	Phân tích
188	PhanThiAnh	\$2b\$12\$hkj2vN/g0adbiOKvTN3fZesq7od6fiA7hHefuIGHVNDZFZFCJKuZ	yinkaphan261@gmail.com	Phân tích
189	admin	\$2b\$12\$/30Nh88AEqozNLackG0e2eAuWBliDXdrKM97pVMVfubh/g7ECUqy	admin@gmail.com	Phân tích

Hình 19: Giao diện trang quản lý số liệu phân tích của từng user

3.2.3.7. Trang phân tích hóa đơn người dùng trong quyền admin



Dự đoán tiền nước của bạn

Bảng thống kê hóa đơn của bạn

KH	DC	ID	chi_so_cu	chi_so_moi	KL_tieu_thu	tien_nuoc	tong_cong	ky_thang	nam	
PhanThiAnh	HCM	123456	1699	1721	22	187.0	274.0	1	2023	xóa
PhanThiAnh	HCM	123456	1721	1745	24	210.0	287.0	2	2023	xóa
PhanThiAnh	HCM	123456	1745	1764	19	154.0	241.0	3	2023	xóa
PhanThiAnh	HCM	123456	1764	1785	21	175.0	263.0	4	2023	xóa
PhanThiAnh	HCM	123456	1785	1809	24	210.0	297.0	5	2023	xóa
PhanThiAnh	HCM	123456	1809	1834	25	221.0	309.0	6	2023	xóa
PhanThiAnh	HCM	123456	1859	1882	23	198.0	285165.0	8	2023	xóa
PhanThiAnh	HCM	123456	1620	1640	20	164.0	251.0	9	2022	xóa
PhanThiAnh	HCM	123456	1640	1659	19	154.0	241.0	10	2022	xóa
PhanThiAnh	HCM	123456	1659	1679	20	164.0	251.0	11	2022	xóa
PhanThiAnh	HCM	123456	1679	1699	20	164.0	251.0	12	2022	xóa
PhanThiAnh	HCM	123456	1834	1859	25	221.0	309.0	7	2023	xóa
NGUYEN	HCM	123457	2107	2130	23	198.0	286.0	12	2022	xóa
NGUYEN	HCM	123457	2085	2107	22	187.0	274.0	11	2022	xóa
NGUYEN	HCM	123457	2130	2151	21	175.0	263.0	1	2023	xóa
NGUYEN	HCM	123457	2151	2178	27	244.0	332115.0	2	2023	xóa
NGUYEN	HCM	123457	2178	2199	21	175.0	263.0	3	2023	xóa
NGUYEN	HCM	123457	2199	2225	26	233.0	320.0	4	2023	xóa

Hình 20: Trang phân tích hóa đơn người dùng quyền admin

CHƯƠNG 4. KẾT LUẬN VÀ KIẾN NGHỊ

4.1. Kết luận

Qua việc xây dựng trang web dự đoán phân tích chi phí hóa đơn tiền điện nước, nhóm đã nắm được cách thiết kế một trang web sử dụng Python và làm việc với trang web, từ việc triển khai lên môi trường Internet cho đến đáp ứng yêu cầu của người dùng. Các thành viên trong nhóm đã thay phiên nhau hoàn thiện giao diện của trang web và đến nay, người dùng có thể dễ dàng sử dụng trang web thông qua giao diện người dùng thân thiện và hấp dẫn. Công việc backend cũng được nhóm cùng nhau học hỏi và xây dựng, cho phép người dùng truy cập các chức năng cơ bản của một trang web.

Một trong những nhiệm vụ quan trọng để hoàn thiện trang web là thiết kế cơ sở dữ liệu cho hệ thống. Sau nhiều cuộc thảo luận và trao đổi, nhóm đã đồng ý và hoàn thành một cơ sở dữ liệu phù hợp cho trang web. Điều này đã giúp đẩy nhanh quá trình phát triển của trang web.

Cuối cùng sẽ là những phần mà nhóm đã hoàn thành trong đồ án này:

Các chức năng dành cho người dùng: Đăng nhập, đăng ký thành viên trên hệ thống, tải hóa đơn lên theo từng kỳ tháng, xem số liệu phân tích trong tương lai, chỉnh sửa thông tin cá nhân của mình,...

Các chức năng quản trị: Quản lý user, quản lý hóa đơn, thống kê người dùng, thống kê hóa đơn, phân tích dữ liệu.

Qua đó, người dùng có thể tương tác với hệ thống và quản lý thông tin người dùng một cách an toàn và bảo mật.

4.2. Hướng phát triển:

Đề tài có thể phát triển và mở rộng theo nhiều hướng cụ thể như sau:

- Nâng cấp giao diện chuyên nghiệp và đẹp mắt hơn.
- Thiết kế giao diện phù hợp với nhiều loại thiết bị hơn.

- Thống kê số lượng người ghé trang web.
- Đăng nhập bằng Google, Facebook.
- Thêm các trang tin tức, diễn đàn cho người dùng thảo luận.
- Thêm quảng cáo để người dùng biết nhiều hơn
- Tải hóa đơn theo URL.

Trong tương lai, hệ thống sẽ bổ sung thêm nhiều chức năng hơn nữa để tăng tính hiệu quả và giúp cho người dùng sử dụng an toàn bảo mật hơn.

Bên cạnh các chức năng cần thiết đó, còn một số lỗi trong hệ thống sẽ được khắc phục tốt hơn, cho website sử dụng an toàn và cải tiến hơn.

TÀI LIỆU THAM KHẢO

- [1] Hướng dẫn tạo FastAPI: <https://viblo.asia/p/fastapi-phan-1-gioi-thieu-va-setup-moi-truong-WR5JRxjQVGv>
- [2] Hồi quy tuyến tính là gì? <https://aws.amazon.com/vi/what-is/linear-regression/>
- [3] Phân tích hồi quy tuyến tính đơn giản [https://bvag.com.vn/wp-content/uploads/2013/01/k2_attachments PHAN-TICH-HOI-QUY-TUYEN-TINH-DON-GIAN.pdf](https://bvag.com.vn/wp-content/uploads/2013/01/k2_attachments_PHAN-TICH-HOI-QUY-TUYEN-TINH-DON-GIAN.pdf)
- [4] Đề tài Phân tích thiết kế hệ thống Quản lý tiền điện của một xã <https://luanvan.net.vn/luan-van/de-tai-phan-tich-thiet-ke-he-thong-quan-ly-tien-dien-cua-mot-xa-52824/>
- [5] Đồ án phân tích thiết kế hệ thống quản lý tiền điện <https://www.thuvientailieu.vn/tai-lieu/do-an-phan-tich-thiet-ke-he-thong-quan-li-tien-dien-3919/>