  
**THỰC TẬP CHUYÊN ĐỀ VÀ ĐỒ ÁN CHUYÊN MÔN**

***ĐỀ TÀI***

**Lập Trình Ứng Dụng Web**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | **Th.S Trịnh Văn Chung** |
| **Sinh viên thực hiện:** | **Lê Hoàng Long** |
| **Mã sinh viên:** | **2210900037** |
| **Lớp:** | **CNTT3** |
| **Khóa:** | **22** |

**@FITNTU-PJ 10/2024**

**MỤC LỤC**

**[DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ](#_heading=h.kgcv8k)** [5](#_heading=h.kgcv8k)

**[DANH MỤC BẢNG BIỂU](#_heading=h.34g0dwd)** [8](#_heading=h.34g0dwd)

**[DANH MỤC HÌNH ẢNH](#_heading=h.1jlao46)** [10](#_heading=h.1jlao46)

**[CHƯƠNG 1: NGHIÊN CỨU CƠ SỞ LÝ THUYẾT](#_heading=h.43ky6rz)** [11](#_heading=h.43ky6rz)

[1.1.](#_heading=h.2iq8gzs) Tổng quan về lập trình thiết kế giao diện website 11

[1.1.1.](#_heading=h.xvir7l) Giới thiệu về HTML 11

[1.1.2.](#_heading=h.3hv69ve) Giới thiệu về CSS3 11

[1.1.3.](#_heading=h.1x0gk37) Tổng quan về ngôn ngữ lập trình Javascript 11

[1.1.4.](#_heading=h.4h042r0) Thư viện Jquery 11

[1.1.5.](#_heading=h.2w5ecyt) Framework Bootstrap 11

[1.2.](#_heading=h.1baon6m) Tổng quan về hệ quản trị cơ sở dữ liệu SQL Server 11

[1.2.1.](#_heading=h.3vac5uf) Giới thiệu về SQL Server 11

[1.2.2.](#_heading=h.2afmg28) Ưu điểm của SQL Server 11

[1.2.3.](#_heading=h.pkwqa1) Nhược điểm của SQL Server 11

[1.3.](#_heading=h.39kk8xu) Tổng quan về ngôn ngữ lập trình phía máy chủ 11

[1.3.1.](#_heading=h.1opuj5n) Giới thiệu về nền tảng công nghệ .NET 11

[1.3.2.](#_heading=h.48pi1tg) Giới thiệu về ngôn ngữ lập trình C# 11

[1.3.3.](#_heading=h.2nusc19) Tổng quan về Framework ASP.NET MVC 5 12

**[CHƯƠNG 2: TỔNG QUAN VỀ ĐỀ TÀI](#_heading=h.1302m92)** [13](#_heading=h.1302m92)

[2.1.](#_heading=h.3mzq4wv) Đặt vấn đề 13

[2.2.](#_heading=h.2250f4o) Hệ thống hiện tại 13

[2.3.](#_heading=h.haapch) Hệ thống đề xuất 13

[2.4.](#_heading=h.319y80a) Giới hạn của hệ thống 13

[2.5.](#_heading=h.1gf8i83) Yêu cầu về phần cứng và phần mềm 13

[2.5.1.](#_heading=h.40ew0vw) Yêu cầu tối thiểu 13

[2.5.2.](#_heading=h.2fk6b3p) Yêu cầu đề nghị 13

**[CHƯƠNG 3:](#_heading=h.upglbi)** [15](#_heading=h.upglbi)

**[PHÂN TÍCH YÊU CẦU KHÁCH HÀNG VÀ THIẾT KẾ HỆ THỐNG](#_heading=h.3ep43zb)** [15](#_heading=h.3ep43zb)

[3.1.](#_heading=h.1tuee74) Người dùng của hệ thống 15

[3.2.](#_heading=h.4du1wux) Chức năng của hệ thống 15

[3.3.](#_heading=h.2szc72q) Systeme Designs (Thiết kế hệ thống) 18

[3.3.1.](#_heading=h.184mhaj) Entity Relationship Diagram (Mô hình quan hệ thực thể) 18

[3.4.](#_heading=h.3s49zyc) Database Design (Thiết kế cơ sở dữ liệu) 18

[3.5.](#_heading=h.279ka65) Table RelationShip Diagram (Mô hình quan hệ bảng) 19

[3.6.](#_heading=h.meukdy) SiteMap (Sơ đồ cấu trúc site) 19

[3.6.1.](#_heading=h.36ei31r) Site map (Cấu trúc trang): Dành cho khách vãng lai 20

[3.6.2.](#_heading=h.1ljsd9k) Sitemap (Cấu trúc trang): Dành cho thành viên 20

[3.6.3.](#_heading=h.45jfvxd) Sitemap (Cấu trúc trang): Dành cho quản trị viên 20

[3.7.](#_heading=h.2koq656) Algorithms (Giải thuật) 21

[3.7.1.](#_heading=h.zu0gcz) Đăng ký 21

[3.7.2.](#_heading=h.3jtnz0s) Đăng nhập 22

[3.7.3.](#_heading=h.1yyy98l) Đăng xuất 22

[3.7.4.](#_heading=h.4iylrwe) Đổi mật khẩu 23

[3.7.5.](#_heading=h.2y3w247) Thay đổi thông tin cá nhân 24

[3.7.6.](#_heading=h.1d96cc0) Thêm sản phẩm 25

[3.7.7.](#_heading=h.3x8tuzt) Xóa sản phẩm 26

[3.7.8.](#_heading=h.2ce457m) Đặt mua 27

[3.7.9.](#_heading=h.rjefff) Tìm kiếm 28

**[CHƯƠNG 4: XÂY DỰNG HỆ THỐNG](#_heading=h.3bj1y38)** [29](#_heading=h.3bj1y38)

[4.1.](#_heading=h.1qoc8b1) Xây dựng phần mềm 29

[4.1.1.](#_heading=h.4anzqyu) Một số mã nguồn chương trình 29

[4.1.2.](#_heading=h.2pta16n) Một số giao diện người dùng và chức năng cụ thể 29

[1.](#_heading=h.14ykbeg) Giao diện trang chủ 29

[2.](#_heading=h.3oy7u29) Giao diện trang quản trị 29

[3.](#_heading=h.243i4a2) ….. 29

**[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN](#_heading=h.j8sehv)** [30](#_heading=h.j8sehv)

[5.1.](#_heading=h.338fx5o) Kết luận 30

[5.2.](#_heading=h.1idq7dh) TaskSheet (bảng kế hoạch công việc) 31

[5.3.](#_heading=h.42ddq1a) Checklist (bảng kiểm tra chức năng) 32

[5.3.1.](#_heading=h.2hio093) Kiểm tra dữ liệu nhập 32

[5.3.2.](#_heading=h.wnyagw) Kiểm tra các liên kết 33

**[TÀI LIỆU THAM KHẢO](#_heading=h.3gnlt4p)** [34](#_heading=h.3gnlt4p)

# **DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ**

# **DANH MỤC BẢNG BIỂU**

# **DANH MỤC HÌNH ẢNH**

# **CHƯƠNG 1: NGHIÊN CỨU CƠ SỞ LÝ THUYẾT**

## Tổng quan về lập trình thiết kế giao diện website

## Giới thiệu về HTML

* Về cơ bản, HTML 5 là một phiên bản mới sửa đổi thứ 5 của ngôn ngữ World Wide Web: the Hypertext Markup Language (HTML).HTML 5 sẽ cho phép một lớp ứng dụng web mới ra đời, hỗ trợ nội dung đa phương tiện và các chức năng offline mà không cần đến những công nghệ độc quyền đi kèm.

## Giới thiệu về CSS3

* CSS (Cascading Style Sheets) được sử dụng để định dạng các phần tử tạo bởi các ngôn ngữ đánh dấu như HTML. Nói cách khác, HTML định dạng các phần tử trên trang web như tiêu đề, bảng, tạo đoạn văn,… Còn CSS giúp các phần tử HTML trở nên phong phú và nổi bật hơn bằng cách cung cấp các phần tử trang trí, hỗ trợ thay đổi màu chữ, tùy chỉnh màu cho trang hoặc thay đổi cấu trúc của website.
* CSS hoạt động bằng cách tìm các lựa chọn như thẻ HTML, lớp, tên ID,… sau đó áp dụng các thuộc tính được thay đổi cho vùng được chọn.  CSS3 là phiên bản thứ 3 của CSS, cũng là phiên bản mới nhất của ngôn ngữ này.
* CSS3 không chỉ kế thừa tất cả những ưu điểm của các phiên bản trước mà còn có nhiều tính năng cải tiến hơn. Vì vậy, CSS3 mang lại rất nhiều tiện ích cho người dùng, đồng thời được các lập trình viên ưa chuộng, sử dụng nhiều trong quá trình thiết kế website.

## Tổng quan về ngôn ngữ lập trình Javascript

* ****Javascript**** chính là một ngôn ngữ lập trình web rất phổ biến ngày nay. Javascript được tích hợp đồng thời nhúng vào HTML để hỗ trợ cho website trở nên sống động hơn. Chúng cũng đóng vai trò tương tự như một phần của website, cho phép Client-side Script từ người dùng tương tự máy chủ (Nodejs) để tạo ra những website động.
* Thông thường, ****JavaScript****sẽ được nhúng trực tiếp vào một website hoặc chúng được tham chiếu qua file .js hoặc .JavaScript.
* Đây là một ngôn ngữ đến từ phía Client nên Script sẽ được download về máy client khi truy cập.
* Tại đây, chúng sẽ được hệ thống xử ý. Vì vậy, bạn không cần phải tải về máy server rồi chờ cho chúng xử lý xong mới phản hồi được kết quả đến client.
* Bên cạnh việc tìm hiểu javascript là ngôn ngữ gì thì chúng được sử dụng để làm gì cũng rất quan trọng. Việc nắm bắt được mục đích của ngôn ngữ đặc biệt này sẽ giúp bạn dễ dàng sử dụng chúng hơn trong công việc. Cụ thể như sau:
* ****Thay đổi nội dung HTML:**** Một trong số nhiều phương thức HTML JavaScript chính là getElementById (). Chúng được sử dụng để tìm một phần tử của HTML với id =”demo" và dùng để thay đổi nội dung của phần từ (Internal HTML) sang thành “Hello JavaScript”
* ****Thay đổi giá trị thuộc tính HTML:**** Tổng quan về javascript còn có thể sử dụng để thay đổi các giá trị của thuộc tính. Ví dụ: thay đổi thuộc tính src (source) của tag<img>.
* ****Thay đổi kiểu HTML:****Đây chính là một hoạt động biến thể của việc thay đổi thuộc tính của HTML ở trên. Ví dụ: document.getElementById(‘demo’).style.fontSize = ’35px;
* ****Ẩn các phần tử HTML:****Một hoạt động tiếp theo là Javascript có thể ẩn được các phần tử HTML. Chúng có thể được thực hiện thông qua hoạt động thay đổi kiểu hiển thị các phần tử HTML.
* ****Hiển thị các phần tử HTML:****Một điểm đặc biệt là JavaScript có thể hiển thị được các yếu tố HTML ẩn. Đồng thời, cũng có thể thực hiện được thông qua cách thay đổi kiểu hiển thị phần tử.
* **Ưu Điểm :**
* Chương trình rất dễ học.
* Những lỗi Javascript rất dễ để phát hiện, từ đó giúp bạn sửa lỗi một cách nhanh chóng hơn.
* Những trình duyệt web có thể dịch thông qua HTML mà không cần sử dụng đến một compiler.
* JS có thể hoạt động ở trên nhiều nền tảng và các trình duyệt web khác nhau.
* Được các chuyên gia đánh giá là một loại ngôn ngữ lập trình nhẹ và nhanh hơn nhiều so với các ngôn ngữ lập trình khác.
* JS còn có thể được gắn trên một số các element hoặc những events của các trang web.
* **Nhược Điểm :**
* JS Code Snippet khá lớn.
* JS dễ bị các hacker và scammer khai thác hơn.
* JS cũng không có khả năng đa luồng hoặc đa dạng xử lý.
* Có thể được dùng để thực thi những mã độc ở trên máy tính của người sử dụng.
* Những thiết bị khác nhau có thể sẽ thực hiện JS khác nhau, từ đó dẫn đến sự không đồng nhất.
* Vì tính bảo mật và an toàn nên các Client-Side **Javascript** sẽ không cho phép đọc hoặc ghi các file.
* JS không được hỗ trợ khi bạn sử dụng ở trong tình trạng thiết bị được kết nối mạng.

## Thư viện Jquery

* ****jQuery****là một thư viện được viết bằng [JavaScript](https://vietnix.vn/javascript-la-gi/" \t "https://vietnix.vn/jquery-la-gi/_blank) đa tính năng, nhanh và nhỏ gọn. jQuery hoạt động theo phương châm****Write less – Do more**** (viết ít hơn, làm nhiều hơn). Mục đích của jQuery là làm cho việc sử dụng JavaScript trên trang web trở nên dễ dàng hơn.
* jQuery thực hiện rất nhiều tác vụ phổ biến đòi hỏi nhiều dòng mã JavaScript để hoàn thành và gói chúng thành các phương thức mà có thể gọi bằng một dòng mã. jQuery cũng đơn giản hóa rất nhiều thứ phức tạp từ JavaScript, AJAX call và thao tác [DOM](https://vietnix.vn/dom-la-gi/" \t "https://vietnix.vn/jquery-la-gi/_blank).
* jQuery được tích hợp từ nhiều module khác nhau. Các module phổ biến của jQuery bao gồm:
* ****Ajax:**** Xử lý Ajax.
* ****Event:**** Xử lý sự kiện.
* ****Atributes:**** Xử lý các thuộc tính của đối tượng HTML.
* ****DOM:**** Xử lý Data Object Model.
* ****Effect:**** Xử lý hiệu ứng.
* ****Form:**** Xử lý sự kiện liên quan tới form.
* ****Selector:**** Xử lý luồng lách giữa các đối tượng HTML.

## Framework Bootstrap

* **Bootstrap** là một framework **CSS** mã nguồn mở, được thiết kế để giúp các nhà phát triển xây dựng các trang web và ứng dụng web dễ dàng hơn. Bootstrap cung cấp một tập hợp các công cụ và thành phần giao diện người dùng (UI) sẵn có như lưới (grid), các biểu mẫu, các nút, các thanh điều hướng, và nhiều thành phần khác.
* Dưới đây là một số đặc điểm nổi bật của Bootstrap:
* **Responsive**: Bootstrap được thiết kế với tính năng "responsive", giúp các trang web hiển thị tốt trên nhiều loại thiết bị từ máy tính để bàn, máy tính bảng đến điện thoại di động. Hệ thống lưới của Bootstrap tự động điều chỉnh kích thước và bố cục trang web để phù hợp với màn hình.
* **Dễ sử dụng**: Với các lớp CSS và thành phần sẵn có, nhà phát triển có thể nhanh chóng tạo ra giao diện đẹp mắt mà không cần viết nhiều mã từ đầu.
* **Cross-browser compatibility**: Bootstrap tương thích với hầu hết các trình duyệt hiện nay như Chrome, Firefox, Safari, và IE.
* **JavaScript Plugins**: Bootstrap cũng cung cấp nhiều plugin JavaScript sẵn có, chẳng hạn như carousel, modals, dropdowns, và tooltips, giúp tăng cường tính tương tác của trang web.
* **Tùy chỉnh dễ dàng**: Người dùng có thể dễ dàng tùy chỉnh Bootstrap để phù hợp với nhu cầu dự án của mình bằng cách thay đổi các biến trong Bootstrap hoặc sử dụng phiên bản tùy chỉnh.

## Tổng quan về hệ quản trị cơ sở đữ liệu SQL Server

## Giới thiệu về SQL Server

* **SQL Server** là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) được phát triển bởi **Microsoft**. Nó được thiết kế để quản lý và lưu trữ lượng dữ liệu lớn và hỗ trợ nhiều thao tác trên dữ liệu, từ việc lưu trữ, truy vấn, phân tích đến bảo mật. SQL Server thường được sử dụng cho các ứng dụng doanh nghiệp, từ quản lý cơ sở dữ liệu cho các hệ thống web đến phân tích dữ liệu.

## Ưu điểm của SQL Server

* **Tốc độ cao :**

Bằng cách sử dụng các truy vấn SQL, người dùng có thể truy xuất nhanh chóng một lượng lớn hồ sơ từ cơ sở dữ liệu.

* **Không cần code :**
* Rất dễ để quản lý các hệ thống cơ sở dữ liệu bằng việc sử dụng SQL chuẩn mà không cần phải viết code.
* **Tiêu chuẩn được xác định rõ** **:**
* SQL đã được thiết lập từ lâu và được công bố chuẩn đầu tiên bởi ISO và ANSI.
* **Tính linh hoạt :**
* SQL có thể sử dụng trên PC, server và thậm chí là trên smart phone.
* **Ngôn ngữ tương tác :**
* Ngôn ngữ truy vấn cấu trúc dữ liệu có thể được sử dụng để giao tiếp với cơ sở dữ liệu và nhận câu trả lời cho các câu hỏi phức tạp trong vài giây.
* **Multiple data views :**
* Với sự trợ giúp của ngôn ngữ SQL, người dùng có thể tạo các hiển thị khác nhau về cấu trúc cơ sở dữ liệu và cơ sở dữ liệu cho những người dùng khác.

## Nhược điểm của SQL Server

* **Chi phí cao :**
* Chi phí vận hành của một số phiên bản SQL khá cao. Đó là lý do tại sao một số lập trình viên không thể sử dụng SQL được.
* **Giao diện phức tạp :**
* Một bất lợi lớn khác là giao diện của SQL khá là phức tạp, điều này đã gây ra cản trở cho những người dùng SQL trong việc truy cập và quản lý nó.
* **Không được toàn quyền kiểm soát :**
* Các lập trình viên sử dụng SQL không có toàn quyền kiểm soát cơ sở dữ liệu do các quy tắc nghiệp vụ bị ẩn.

## Tổng quan về ngôn ngữ lập trình phía máy chủ

## Giới thiệu về nền tảng công nghệ .NET

* **.NET** là một nền tảng nguồn mở để xây dựng các ứng dụng trên máy tính để bàn, web và thiết bị di động có thể chạy nguyên bản trên bất kỳ hệ điều hành nào. Hệ thống **.NET** bao gồm các công cụ, thư viện và ngôn ngữ hỗ trợ phát triển phần mềm hiện đại, có quy mô linh hoạt và hiệu năng cao. Một cộng đồng nhà phát triển hoạt động tích cực trong việc duy trì và hỗ trợ nền tảng **.NET**.
* Nói một cách dễ hiểu, nền tảng **.NET** là phần mềm có thể thực hiện những tác vụ sau:
* Dịch mã ngôn ngữ lập trình **.NET** thành hướng dẫn mà thiết bị máy tính có thể xử lý.
* Cung cấp các tiện ích để phát triển phần mềm hiệu quả. Ví dụ: nó có thể tìm thời gian hiện tại hoặc in văn bản trên màn hình.
* Xác định một tập gồm các loại dữ liệu để lưu trữ thông tin như văn bản, số và ngày tháng trên máy tính.

## Giới thiệu về ngôn ngữ lập trình C#

* **C#** (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000. **C#** là ngôn ngữ lập trình hiện đại, hướng đối tượng và được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là **C++** và **Java**.

## Tổng quan về Framework ASP.NET MVC 5

* **ASP.NET MVC 5** là một framework phát triển web mạnh mẽ được Microsoft phát triển, sử dụng kiến trúc **MVC (Model-View-Controller)**. Framework này giúp xây dựng các ứng dụng web dễ duy trì, mở rộng và có tính linh hoạt cao bằng cách tách riêng các phần của ứng dụng thành ba thành phần chính: **Model**, **View**, và **Controller**. Dưới đây là tổng quan về ASP.NET MVC 5:

### 1. ****Kiến trúc MVC****

* Kiến trúc **MVC** là một mô hình phân chia các thành phần của ứng dụng web thành ba phần tách biệt:
* **Model**: Đại diện cho dữ liệu và các logic nghiệp vụ của ứng dụng. Model quản lý dữ liệu, xử lý logic nghiệp vụ, và tương tác với cơ sở dữ liệu.
* **View**: Hiển thị giao diện người dùng, có nhiệm vụ trình bày thông tin từ Model đến người dùng. Các View thường được viết bằng cú pháp **Razor**, một engine kết hợp giữa HTML và C#.
* **Controller**: Nhận yêu cầu từ người dùng, xử lý chúng và tương tác với Model để lấy dữ liệu. Sau đó, Controller gửi dữ liệu đến View để hiển thị cho người dùng.

### 2. ****Tính năng nổi bật của ASP.NET MVC 5****

#### **a. **Routing (Hệ thống định tuyến)****

* ASP.NET MVC 5 sử dụng hệ thống định tuyến (routing) để ánh xạ URL từ trình duyệt đến các phương thức của Controller. Điều này giúp tạo ra các URL thân thiện với người dùng và công cụ tìm kiếm (SEO-friendly URLs).

#### **b. **Tối ưu hóa SEO****

* Với khả năng tạo các URL thân thiện và kiểm soát tốt về cách URL được tạo, ASP.NET MVC 5 giúp cải thiện SEO cho các ứng dụng web, giúp các trang web dễ dàng được công cụ tìm kiếm lập chỉ mục.

#### **c. **Hỗ trợ Dependency Injection (DI)****

* ASP.NET MVC 5 hỗ trợ tích hợp **Dependency Injection (DI)**, giúp dễ dàng quản lý các phụ thuộc giữa các thành phần trong ứng dụng và tăng tính dễ bảo trì cũng như khả năng kiểm thử (unit testing).

#### **d. **Lập trình bất đồng bộ (Asynchronous programming)****

* ASP.NET MVC 5 hỗ trợ **async/await**, giúp các ứng dụng web xử lý các yêu cầu không đồng bộ, từ đó cải thiện hiệu suất và khả năng phản hồi của ứng dụng, đặc biệt khi xử lý số lượng lớn yêu cầu đồng thời.

#### **e. **Bảo mật mạnh mẽ****

* ASP.NET MVC 5 tích hợp sẵn các tính năng bảo mật như:
* **Xác thực (Authentication)**: Hỗ trợ các phương pháp xác thực như **OAuth**, **Forms Authentication**, và **Windows Authentication**.
* **Quản lý quyền truy cập (Authorization)**: Dễ dàng quản lý quyền người dùng thông qua **Role-based Authorization**.

#### **f. **Scaffolding****

* Scaffolding là tính năng giúp tự động tạo mã cho các thao tác CRUD (Create, Read, Update, Delete) cơ bản. Điều này giúp tiết kiệm thời gian và công sức khi phát triển các ứng dụng web, đặc biệt là khi làm việc với dữ liệu.

#### **g. **Bootstrap tích hợp****

* ASP.NET MVC 5 tích hợp sẵn **Bootstrap**, giúp các nhà phát triển dễ dàng tạo ra các trang web với thiết kế responsive mà không cần phải cấu hình thêm từ đầu. Điều này cải thiện trải nghiệm người dùng trên nhiều loại thiết bị khác nhau.

# **CHƯƠNG 2: TỔNG QUAN VỀ ĐỀ TÀI**

## Đặt vấn đề

* Trong bối cảnh thị trường game nhập vai đang phát triển mạnh mẽ, Wuthering Waves đã tạo ra nhiều kỳ vọng về một tựa game đột phá với lối chơi hấp dẫn, đồ họa xuất sắc và hệ thống chiến đấu phức tạp. Tuy nhiên, với các yêu cầu về phần cứng ngày càng cao từ những trò chơi thế hệ mới, người chơi cần có một hệ thống phù hợp để trải nghiệm game một cách mượt mà nhất.

## Hệ thống hiện tại

* Hiện tại, trò chơi Wuthering Waves đang trong giai đoạn thử nghiệm beta. Các hệ thống hiện tại của trò chơi bao gồm:
* **Gameplay**: Lối chơi tập trung vào chiến đấu nhịp độ nhanh, với hệ thống chiến đấu yêu cầu phản xạ nhanh và chiến lược. Người chơi cũng có thể điều khiển nhiều nhân vật khác nhau với kỹ năng đặc biệt.
* **Đồ họa**: Trò chơi có đồ họa 3D chất lượng cao, đòi hỏi phần cứng mạnh để có thể chạy mượt mà ở cài đặt cao nhất.
* **Tương tác thế giới**: Người chơi có thể tương tác với môi trường, khám phá các khu vực ẩn, và tham gia các nhiệm vụ phụ để phát triển câu chuyện và nhân vật.

## Hệ thống đề xuất

* Dựa trên các đặc điểm của Wuthering Waves, hệ thống đề xuất sẽ cần phải đáp ứng được yêu cầu về xử lý đồ họa cao, đồng thời cung cấp khả năng xử lý mượt mà trong các cảnh chiến đấu phức tạp.

## Giới hạn của hệ thống

* **Hiệu năng**: Trò chơi yêu cầu hệ thống phần cứng cao, nên các máy tính cũ hoặc yếu có thể gặp phải tình trạng giật, lag, hoặc thậm chí không chạy được trò chơi.
* **Yêu cầu về mạng**: Là một trò chơi có yếu tố trực tuyến, kết nối mạng không ổn định có thể ảnh hưởng đến trải nghiệm chơi game.
* **Tương thích**: Một số cấu hình hoặc hệ điều hành có thể không tương thích với trò chơi, dẫn đến lỗi khi khởi chạy hoặc trong quá trình chơi.

## Yêu cầu về phần cứng và phần mềm

## Yêu cầu tối thiểu

|  |  |  |
| --- | --- | --- |
|  | **Server** | **Client** |
| Hardware | \* Ram:8GB  \* HDD >=20GB free  \* CPU: Intel Core i5-4460 hoặc AMD Ryzen 3 1200  \* Connect the Internet. | \* Ram:4GB  \* CPU : Intel Core i5-4460 hoặc AMD Ryzen 3 1200  \* Connect the Internet. |
| Software | \* Hệ điều hành: DirectX 11  \* NetFramework: 4.7 trở lên | \* Hệ điều hành: DirectX 12  \* IE 8.0, Firefox 4.0, Chrome 15 |

## Yêu cầu đề nghị

|  |  |  |
| --- | --- | --- |
|  | **Server** | **Client** |
| Hardware | \* Ram: >= 16Gb  \* HDD: >= 30GB free  \* CPU: Intel Core i7-6700K hoặc AMD Ryzen 5 3600  \* Connect the Internet. | \* Ram:32GB  \* CPU: Intel Core i7-6700K hoặc AMD Ryzen 5 3600  \*Connect the Internet. |
| Software | \* Hệ điều hành: Window 11  \* Net Framework: 4.8 trở lên | \* Từ Windows XP / Vista / 7 / 8 / 8.1… trở lên.  \* IE 11.0 hoặc Firefox 16.0 trở lên, Chrome 19.0 trở lên. |

# **CHƯƠNG 3:**

# **PHÂN TÍCH YÊU CẦU KHÁCH HÀNG VÀ THIẾT KẾ HỆ THỐNG**

## Người dùng của hệ thống

***Đối tượng người dùng của hệ thống bao gồm***

* **Guest (Khách vãng lai)**: Người dùng chưa đăng ký tài khoản, chỉ có quyền xem thông tin sự kiện.
* **Member (Thành viên)**: Người dùng đã đăng ký tài khoản, có thể đăng nhập và xem các sự kiện.
* **Admin (Quản trị hệ thống)**: Người quản trị hệ thống, có quyền quản lý sự kiện và bài viết.

## Chức năng của hệ thống

* + 1. ***Guest*** *(Chức năng dành cho khách vãng lai)*
* **Xem sự kiện**

|  |  |
| --- | --- |
| **Mô tả** | Khách vãng lai có thể xem thông tin chi tiết từng sự kiện, bao gồm tên sự kiện, thời gian, địa điểm, và mô tả chi tiết. |
| **Nhập** | Lựa chọn sự kiện cần xem. |
| **Xử lý** | Tìm kiếm thông tin sự kiện trong cơ sở dữ liệu. |
| **Xuất** | Hiển thị thông tin sự kiện bao gồm tên, mô tả, thời gian và địa điểm. |

* **Đăng ký thành viên**

|  |  |
| --- | --- |
| **Mô tả** | Khách vãng lai có thể đăng ký tài khoản để trở thành thành viên. |
| **Nhập** | Cung cấp thông tin như tên đăng nhập, email, mật khẩu. |
| **Xử lý** | Hệ thống kiểm tra thông tin và lưu tài khoản mới vào cơ sở dữ liệu.Mật khẩu phải gồm ít nhất 6 ký tự và không có ký tự |
| **Xuất** | Thông báo đăng ký thành công hoặc thông báo lỗi nếu có. |

* + 1. ***Member functions*** *(Chức năng dành cho thành viên)*

Thành viên chính thức của website sẽ có các chức năng giống như khách vãng lai như:

* + Xem sản phẩm.
  + Tìm kiếm sản phẩm.
  + Lựa chọn sản phẩm vào giỏ hàng.

***Ngoài ra thành viên chính thức còn có các chức năng sau***:

* **Xem Thông Tin Sự Kiện**

|  |  |
| --- | --- |
| **Mô tả** | Thành viên có thể xem thông tin sự kiện chi tiết giống như khách vãng lai. |
| **Nhập** | Lựa chọn sự kiện cần xem. |
| **Xử lý** | Hệ thống tìm kiếm thông tin sự kiện trong cơ sở dữ liệu. |
| **Xuất** | Hiển thị thông tin sự kiện bao gồm tên, mô tả, thời gian và địa điểm. |

* **Đăng nhập**

|  |  |
| --- | --- |
| **Mô tả** | Thành viên đã đăng ký có thể đăng nhập vào hệ thống. |
| **Nhập** | Cung cấp tên đăng nhập và mật khẩu. |
| **Xử lý** | Hệ thống xác thực thông tin người dùng. |
| **Xuất** | Truy cập vào tài khoản cá nhân nếu thông tin chính xác, hoặc thông báo lỗi nếu không hợp lệ. |

* + 1. ***Admin functions*** *(Chức năng dành cho quản trị viên)*

***Quản trị viên có các chức năng chính như sau:***

* **Quản Lý Sự Kiện**

|  |  |
| --- | --- |
| **Mô tả** | Quản trị viên có thể thêm, sửa, hoặc xóa thông tin sự kiện. |
| **Nhập** | Nhập hoặc chỉnh sửa thông tin sự kiện bao gồm tên, mô tả, thời gian, và địa điểm. |
| **Xử lý** | Hệ thống lưu trữ hoặc cập nhật thông tin sự kiện trong cơ sở dữ liệu. |
| **Xuất** | Hiển thị thông báo về trạng thái thêm, sửa, hoặc xóa sự kiện thành công hoặc thất bại. |

* **Quản lý bài viết**

|  |  |
| --- | --- |
| **Mô tả** | Quản trị viên có thể thêm, sửa, hoặc xóa bài viết tin tức liên quan đến sự kiện. |
| **Nhập** | Nhập hoặc chỉnh sửa nội dung bài viết bao gồm tiêu đề, nội dung, hình ảnh (nếu có). |
| **Xử lý** | Lưu trữ hoặc cập nhật thông tin bài viết trong cơ sở dữ liệu. |
| **Xuất** | Hiển thị thông báo về trạng thái thêm, sửa, hoặc xóa bài viết thành công hoặc thất bại. |

## Systeme Designs (Thiết kế hệ thống)

## Entity Relationship Diagram (Mô hình quan hệ thực thể)

1. **Người dùng (User)**

* **user\_id** (PK): Khóa chính, định danh duy nhất cho người dùng.
* **username**: Tên đăng nhập của người dùng.
* **email**: Địa chỉ email của người dùng.
* **password**: Mật khẩu của người dùng (đã mã hóa).
* **role**: Vai trò của người dùng (quản trị viên hoặc người dùng thường).
* **created\_at**: Ngày tạo tài khoản.

1. **Sự kiện (Event)**

* **event\_id** (PK): Khóa chính, định danh duy nhất cho mỗi sự kiện.
* **title**: Tiêu đề của sự kiện.
* **description**: Mô tả chi tiết sự kiện.
* **event\_date**: Ngày diễn ra sự kiện.
* **created\_at**: Ngày tạo sự kiện.
* **author\_id** (FK): Khóa ngoại, liên kết với bảng **User** để xác định người quản trị đã tạo sự kiện.

1. **Bài viết (Article)**

* **article\_id** (PK): Khóa chính, định danh duy nhất cho mỗi bài viết.
* **title**: Tiêu đề của bài viết.
* **content**: Nội dung bài viết.
* **created\_at**: Ngày tạo bài viết.
* **author\_id** (FK): Khóa ngoại, liên kết với bảng **User** để xác định quản trị viên đã tạo bài viết.

***Mô tả chi tiết các thuộc tính của các thực thể***:

1. Thực thể quản trị (**QUAN\_TRI**):   
   Lưu trữ thông tin người quản trị viên hệ thống.

**QUAN\_TRI**

**Tai\_khoan**

Mat\_khau

Trang\_thai

· **quan\_tri\_id** (PK): Khóa chính, định danh duy nhất cho quản trị viên.

· **username**: Tên đăng nhập của quản trị viên.

· **email**: Địa chỉ email của quản trị viên.

· **password**: Mật khẩu đã mã hóa.

· **created\_at**: Ngày tạo tài khoản quản trị viên.

· **status**: Trạng thái tài khoản (hoạt động, bị khóa).

· **role**: Vai trò, luôn là quản trị viên trong hệ thống (admin).

1. Thực thể khách hàng (**KHACH\_HANG**): Lưu trữ thông tin khách hàng khi đăng ký, mua hàng tại hệ thống.

· **khach\_hang\_id** (PK): Khóa chính, định danh duy nhất cho khách hàng.

· **username**: Tên đăng nhập của khách hàng.

· **email**: Địa chỉ email của khách hàng.

· **password**: Mật khẩu đã mã hóa.

· **created\_at**: Ngày tạo tài khoản khách hàng.

· **status**: Trạng thái tài khoản (hoạt động, bị khóa).

· **role**: Vai trò, luôn là thành viên thường (member) trong hệ thống.

1. Thực Thể Sự Kiện (SU\_KIEN)

Lưu trữ thông tin các sự kiện được quản trị viên đăng trên hệ thống.

* **su\_kien\_id** (PK): Khóa chính, định danh duy nhất cho mỗi sự kiện.
* **title**: Tiêu đề của sự kiện.
* **description**: Mô tả chi tiết về sự kiện.
* **event\_date**: Ngày diễn ra sự kiện.
* **created\_at**: Ngày tạo sự kiện.
* **author\_id** (FK): Khóa ngoại, liên kết với bảng **QUAN\_TRI** để xác định quản trị viên đã tạo sự kiện.
* **location**: Địa điểm tổ chức sự kiện.

1. **Thực thể bài viết (BAI\_VIET)**

Lưu trữ các bài viết thông báo hoặc tin tức liên quan đến sự kiện được đăng bởi quản trị viên.

* **bai\_viet\_id** (PK): Khóa chính, định danh duy nhất cho mỗi bài viết.
* **title**: Tiêu đề của bài viết.
* **content**: Nội dung bài viết.
* **created\_at**: Ngày tạo bài viết.
* **author\_id** (FK): Khóa ngoại, liên kết với bảng **QUAN\_TRI** để xác định quản trị viên đã tạo bài viết.

## Database Design (Thiết kế cơ sở dữ liệu)

Thiết kế có sở dữ liệu trên hệ quản trị dữ liệu MS SQL (SQL Server)

1. Bảng **USERS ( người dùng )**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu DataType** | **Null** | **Ràng buộc**  **Constraint** | **Mô tả Content** |
| user\_id | INT | No | Primary Key,; idnetity(1,1) | Mã người dùng (khóa chính) |
| taikhoan | Varchar(50) | No | Unique | Tên đăng nhập |
| matkhau | VARCHAR(100) | No |  | Mật khẩu đã mã hóa |
| email | VARCHAR(100) | No |  | Email |

1. Bảng **EVENT ( sự kiện )**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu DataType** | **Null** | **Ràng buộc**  **Constraint** | **Mô tả Content** |
| ev\_id | Int | No | Primary Key; identity(1,1) | Mã sự kiện (khóa chính) |
| tieude | VARCHAR(255) | No |  | Tiêu đề sự kiện |
| noidung | TEXT | YES |  | Mô tả chi tiết sự kiện |

1. Bảng Tham gia sự kiện

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| tg\_id | INT | No | Primary Key; identity(1,1) | Mã bài viết (khóa chính) |
| user\_id | INT | No | Foreign Key (users.user\_id) | Người dùng |
| ev\_id | INT | Yes | Foreign Key (Sukien.ev\_id) | Mã sự kiện |
| ngaythamgia | DATE | No |  | Ngày tham gia |

1. Bảng Thông tin Users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| tt\_id | INT | No | Primary Key; Auto\_increment | Mã đăng ký (khóa chính) |
| user\_id | INT | No | Foreign Key (users.user\_id) | Người dùng đã đăng ký sự kiện |
| Fullname | Varchar(100) | No |  | Tên đầy đủ |
| Avatar | Varchar(100) | No |  | Ảnh người dùng |

1. Bảng RATING ( ĐÁNH GIÁ)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Null | Ràng buộc | Mô tả |
| dg\_id | INT | No | Primary Key; Auto\_increment | Mã bản ghi (khóa chính) |
| user\_id | INT | No | Foreign Key (users.user\_id) | Người dùng đã thực hiện |
| dg\_value | INT CHECK | No |  | Đánh giá 1-5 sao |
| ngaydg | DATE | No |  | Ngày đánh giá |

## Table RelationShip Diagram (Mô hình quan hệ bảng)

1. **USERS**:
   1. **user\_id** là khóa chính (PK), xác định duy nhất từng người dùng trong hệ thống.
   2. **role** xác định quyền hạn của người dùng: **guest**, **member**, hoặc **admin**.
2. **EVENTS**:
   1. **event\_id** là khóa chính (PK) của bảng sự kiện.
   2. **author\_id** là khóa ngoại (FK) liên kết với **user\_id** của bảng **USERS** để xác định quản trị viên tạo sự kiện.
3. **Tham gia sự kiện**:
   1. **tg\_id** là khóa chính (PK) của bảng bài viết.
   2. **user\_id** là khóa ngoại (FK) liên kết với **user\_id** của bảng **USERS** để xác định quản trị viên tạo bài viết.
   3. ev\_id là khoá ngoại (FK) liên kết với ev\_id của bảng Sukien để xác định sự kện tham gia
4. Thông tin Users:
   1. **tt\_id** là khóa chính (PK).
   2. **user\_id** là khóa ngoại (FK) liên kết với **user\_id** của bảng **USERS** để xác định thông tin đầy đủ của người dùng
5. RATING ( Đánh giá) :
   1. **dg\_id** là khóa chính (PK).
   2. **user\_id** là khóa ngoại (FK) liên kết với **user\_id** của bảng **USERS** để xác định người dùng thực hiện thao tác.

### Sơ Đồ Quan Hệ (Table Relationship Diagram):

* **USERS và EVENT**: Một quản trị viên (user\_id) có thể tạo nhiều sự kiện (ev\_id), nhưng mỗi sự kiện chỉ do một quản trị viên tạo.
* **USERS và REGISTRATIONS**: Một người dùng (user\_id) có thể đăng ký nhiều sự kiện, nhưng mỗi lần đăng ký chỉ thuộc về một sự kiện duy nhất (ev\_id).
* **EVENT và REGISTRATIONS**: Một sự kiện (ev\_id) có thể có nhiều người dùng đăng ký, nhưng mỗi lần đăng ký chỉ thuộc về một người dùng duy nhất (user\_id).
* **USERS và RATING**: Người dùng (user\_id) có thể đánh giá nhiều sự kiện hoặc bài viết, với mỗi đánh giá được lưu trong bảng **RATING**.

## SiteMap (Sơ đồ cấu trúc site)

## Site map (Cấu trúc trang): Dành cho khách vãng lai

1. **Trang chủ**

* ·Hiển thị danh sách sự kiện
* Nút xem chi tiết sự kiện

1. **Trang sự kiện chi tiết**

* Tên sự kiện
* Mô tả sự kiện
* Thời gian diễn ra
* Địa điểm

1. **Đăng ký tài khoản**

* · Form đăng ký (Tên đăng nhập, Email, Mật khẩu)
* Thông báo đăng ký thành công hoặc lỗi

1. **Đăng nhập**

* · Form đăng nhập (Tên đăng nhập, Mật khẩu)
* Thông báo đăng nhập thành công hoặc lỗi

## Sitemap (Cấu trúc trang): Dành cho thành viên

1. **Trang chủ**

* Hiển thị danh sách sự kiện
* Nút xem chi tiết sự kiện
* Nút đăng xuất

1. **Trang sự kiện chi tiết**

* Tên sự kiện
* Mô tả sự kiện
* Thời gian diễn ra
* Địa điểm
* Nút đăng ký sự kiện

1. **Tài khoản của tôi**

* Thông tin cá nhân (Tên đăng nhập, Email)
* Lịch sử các sự kiện đã đăng ký
* Nút cập nhật thông tin tài khoản (nếu cần)

1. **Đăng xuất**

* Đăng xuất khỏi hệ thống và trở về giao diện khách vãng lai

## Sitemap (Cấu trúc trang): Dành cho quản trị viên

1. **Trang quản trị**

* Dashboard (Tổng quan sự kiện và bài viết)
* Nút quản lý sự kiện
* Nút quản lý bài viết
* Nút quản lý người dùng

1. **Quản lý sự kiện**

* Danh sách sự kiện
* Thêm sự kiện mới
* Sửa/xóa sự kiện hiện có

1. **Quản lý bài viết**

* Danh sách bài viết
* Thêm bài viết mới
* Sửa/xóa bài viết hiện có

1. **Quản lý người dùng**

* Danh sách người dùng
* Quản lý vai trò (Khách vãng lai, Thành viên, Quản trị viên)
* Khóa/Mở khóa tài khoản người dùng

1. **Tài khoản của tôi**

* Thông tin cá nhân quản trị viên
* Đổi mật khẩu

1. **Đăng xuất**

* · Đăng xuất khỏi giao diện quản trị

## Algorithms (Giải thuật)

## Đăng ký



|  |  |
| --- | --- |
| * **Kiểm tra thông tin nhập**:   **Mô tả:** Kiểm tra tính hợp lệ của các thông tin người dùng nhập vào dựa trên các quy định của hệ thống:   * **Tài khoản (Username):** Bắt buộc nhập, độ dài từ 6 đến 50 ký tự. * **Mật khẩu (Password):** Bắt buộc nhập, phải từ 6 ký tự trở lên. * **Email:** Bắt buộc nhập, đúng định dạng email. * **Số điện thoại (Phone):** Kiểm tra tính hợp lệ của số điện thoại (10 chữ số).   **Giải thuật:**   1. Kiểm tra từng trường thông tin nhập (username, email, mật khẩu, số điện thoại) có đúng định dạng và đủ điều kiện không. 2. Nếu có bất kỳ trường nào không hợp lệ, hiển thị thông báo lỗi. 3. Nếu tất cả trường hợp lệ, chuyển sang bước kiểm tra sự tồn tại của tài khoản. | * **Kiểm tra sự tồn tại của tài khoản:**   **Mô tả:** Đảm bảo rằng tên đăng nhập (username) là duy nhất trong hệ thống.  **Giải thuật:**   1. Truy vấn cơ sở dữ liệu để kiểm tra xem **username** hoặc **email** đã tồn tại hay chưa. 2. Nếu **username** hoặc **email** đã tồn tại, hiển thị thông báo lỗi cho người dùng và yêu cầu nhập lại. 3. Nếu không, lưu thông tin người dùng mới vào cơ sở dữ liệu, hiển thị thông báo đăng ký thành công. |

## Đăng nhập



|  |
| --- |
| **Mô tả:** Xác thực người dùng bằng cách so sánh thông tin đăng nhập với dữ liệu lưu trữ trong hệ thống.  **Giải thuật:**   1. Người dùng nhập **username** và **password**. 2. Truy vấn cơ sở dữ liệu theo **username**. 3. Nếu **username** không tồn tại, hiển thị thông báo lỗi. 4. Nếu tồn tại, so sánh **password** đã mã hóa từ cơ sở dữ liệu với mật khẩu mà người dùng đã nhập. 5. Nếu khớp, cho phép đăng nhập và chuyển người dùng đến trang cá nhân. 6. Nếu không khớp, hiển thị thông báo sai mật khẩu. |

## Đăng xuất



|  |
| --- |
| **Mô tả:** Xóa phiên đăng nhập của người dùng và đưa họ trở về giao diện khách vãng lai.  **Giải thuật:**   1. Khi người dùng bấm nút đăng xuất, hệ thống xóa các dữ liệu phiên (session) của người dùng. 2. Chuyển người dùng về trang chủ với tư cách **guest**. |

## Đổi mật khẩu



|  |
| --- |
| **Mô tả:** Cho phép người dùng đã đăng nhập thay đổi mật khẩu hiện tại bằng mật khẩu mới.  **Giải thuật:**   1. Người dùng nhập mật khẩu hiện tại, mật khẩu mới, và xác nhận mật khẩu mới. 2. Kiểm tra xem mật khẩu hiện tại có khớp với dữ liệu trong cơ sở dữ liệu không. 3. Nếu không khớp, hiển thị thông báo lỗi. 4. Nếu khớp, kiểm tra mật khẩu mới và xác nhận mật khẩu mới có trùng khớp và hợp lệ không. 5. Nếu đúng, cập nhật mật khẩu mới đã mã hóa vào cơ sở dữ liệu và thông báo thành công. |

## Thay đổi thông tin cá nhân



|  |
| --- |
| **Mô tả:** Cho phép người dùng đã đăng nhập chỉnh sửa các thông tin như email, số điện thoại, địa chỉ.  **Giải thuật:**   1. Người dùng nhập các thông tin cần chỉnh sửa. 2. Kiểm tra tính hợp lệ của các trường (email, số điện thoại). 3. Nếu hợp lệ, cập nhật cơ sở dữ liệu với thông tin mới. 4. Thông báo cho người dùng việc cập nhật thành công. |

## Thêm sự kiện ( dành cho ADMIN)

|  |
| --- |
| **Mô tả:** Quản trị viên có thể thêm các sự kiện mới vào hệ thống.  **Giải thuật:**   1. Quản trị viên nhập thông tin sự kiện bao gồm: **tên sự kiện**, **mô tả**, **thời gian**, **địa điểm**. 2. Kiểm tra tính hợp lệ của các trường. 3. Nếu hợp lệ, lưu thông tin sự kiện vào cơ sở dữ liệu. 4. Thông báo cho quản trị viên về trạng thái thành công hoặc thất bại. |

## Xóa sự kiện ( dành cho ADMIN )

|  |
| --- |
| **Mô tả:** Quản trị viên có thể xóa một sự kiện khỏi hệ thống.  **Giải thuật:**   1. Quản trị viên chọn sự kiện cần xóa. 2. Kiểm tra xem sự kiện có tồn tại trong cơ sở dữ liệu không. 3. Nếu tồn tại, xóa sự kiện và cập nhật cơ sở dữ liệu. 4. Thông báo cho quản trị viên về trạng thái thành công hoặc thất bại. |

## Đăng ký tham gia sự kiện ( dành cho thành viên )

|  |
| --- |
| **Mô tả:** Thành viên có thể đăng ký tham gia các sự kiện trong hệ thống.  **Giải thuật:**   1. Thành viên chọn sự kiện muốn đăng ký. 2. Truy vấn cơ sở dữ liệu xem thành viên đã đăng ký sự kiện này chưa. 3. Nếu chưa, thêm thông tin đăng ký vào cơ sở dữ liệu. 4. Thông báo thành công cho thành viên. 5. Nếu đã đăng ký, hiển thị thông báo lỗi. |

## Tìm kiếm sự kiện



|  |
| --- |
| **Mô tả:** Cho phép người dùng tìm kiếm các sự kiện dựa trên từ khóa.  **Giải thuật:**   1. Người dùng nhập từ khóa tìm kiếm (tên sự kiện, ngày, hoặc địa điểm). 2. Truy vấn cơ sở dữ liệu để tìm các sự kiện có chứa từ khóa trong tiêu đề hoặc mô tả. 3. Hiển thị danh sách các sự kiện khớp với từ khóa cho người dùng. 4. Nếu không có sự kiện nào, hiển thị thông báo "Không tìm thấy sự kiện." |

# **CHƯƠNG 4: XÂY DỰNG HỆ THỐNG**

## Xây dựng phần mềm

## Một số mã nguồn chương trình

**- Login :  
HTML :**   
<div class="form-box login">

<h2>Login</h2>

<form action="#">

<div class="input-box">

<span class="icon"><ion-icon name="person"></ion-icon></span>

<input type="Users" required>

<label>Tài Khoản</label>

</div>

<div class="input-box">

<span class="icon"><ion-icon name="lock-closed"></ion-icon></span>

<input type="Password" required>

<label>Mật Khẩu</label>

</div>

<div class="remember-forgot">

<label><input type="checkbox">

Remember Me</label>

<a href="#">Forgot Password?</a>

</div>

<button type="submit" class="btn">Login</button>

<div class="login-register">

<p>Don't Have an account?<a href="#" class="register-link">Register</a></p>

</div>

</form>

</div>  
  
**- Register :   
HTML:**   
<div class="form-box register">

<h2>Registration</h2>

<form action="#">

<div class="input-box">

<span class="icon"><ion-icon name="mail"></ion-icon></span>

<input type="Email" required>

<label>Email</label>

</div>

<div class="input-box">

<span class="icon"><ion-icon name="person"></ion-icon></span>

<input type="Users" required>

<label>Tài Khoản</label>

</div>

<div class="input-box">

<span class="icon"><ion-icon name="lock-closed"></ion-icon></span>

<input type="Password" required>

<label>Mật Khẩu</label>

</div>

<div class="remember-forgot">

<label><input type="checkbox">

Tôi đồng ý với câc thoả thuận và điều khoản.</label>

</div>

<button type="submit" class="btn">Login</button>

<div class="login-register">

<p>Already Have an account?<a href="login.html" class="login-link">Login</a></p>

</div>

</form>

</div>

## Một số giao diện người dùng và chức năng cụ thể

**Trang chủ (Home Page)**:

* 1. Hiển thị các sự kiện nổi bật và bài viết mới nhất.
  2. Cung cấp các thông tin chính về **Wuthering Wave** và điều hướng đến các trang chính như sự kiện, bài viết, và đánh giá.

**Giao diện sự kiện (Event Interface)**:

* 1. Hiển thị danh sách sự kiện sắp tới và chi tiết từng sự kiện.
  2. Chức năng cho người dùng đăng ký, đánh giá và nhận thông báo về sự kiện.

**Quản lý tài khoản người dùng (User Management Interface)**:

* 1. Cung cấp giao diện cho người dùng cập nhật thông tin cá nhân, bao gồm ảnh đại diện, tên đầy đủ, và email.
  2. Hiển thị danh sách sự kiện mà người dùng đã đăng ký.

**Giao diện bài viết (Article Interface)**:

* 1. Cho phép quản trị viên tạo và chỉnh sửa các bài viết trên trang.
  2. Hiển thị danh sách các bài viết để người dùng đọc và đưa ra phản hồi.

**Giao diện đánh giá (Rating Interface)**:

* 1. Hiển thị đánh giá tổng quan cho mỗi sự kiện hoặc bài viết.
  2. Chức năng cho người dùng đánh giá theo hệ thống 5 sao và xem đánh giá trung bình.

**Ghi nhận thao tác quản trị (Admin Action Log Interface)**:

* 1. Cho phép quản trị viên xem lại lịch sử thao tác, bao gồm các hoạt động quản lý sự kiện và bài viết.

# **KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

## Kết luận

· **Mô tả dự án:** Trong phần này, bạn có thể tổng hợp các bước mà bạn đã thực hiện trong quá trình xây dựng hệ thống. Bạn có thể đề cập đến cách hệ thống đã giúp giải quyết vấn đề hoặc cung cấp các giải pháp cụ thể cho mục đích đề tài, như việc quản lý sự kiện và người dùng trong web Wuthering Waves.

- **Ưu điểm:**

- Đề tài mang lại cái nhìn tổng quan về quy trình phân tích và thiết kế hệ thống web.

- Áp dụng các phương pháp và công nghệ lập trình hiện đại như **HTML, CSS, PHP** để xây dựng giao diện và tính năng của web.

- Hệ thống thân thiện với người dùng, dễ sử dụng.

- **Nhược điểm:**

- Do thời gian hạn chế, hệ thống chưa mở rộng và chưa có các tính năng tương tác mạnh mẽ.

- Hệ thống chưa có tính năng đăng bài, bình luận, do đó cần mở rộng trong tương lai.

## Task Sheet (bảng kế hoạch công việc)

|  |  |  |  |
| --- | --- | --- | --- |
| **Giai đoạn** | **Từ ngày … đến….** | **Công việc** | **Tỉ lệ hoàn thành** |
| Chương 1:  (100%) |  | - Xây dựng cơ sở dữ liệu , giao diện đăng nhập và đăng ký | 100% |
| Chương 2:  (100%) |  | - Phát triển chức năng xem sự kiện | 100% |
| Chương 3:  (100%) |  | - Phát triển trang quản trị | 100% |
| Chương 4:  (100%) |  | - Tối ưu hóa hệ thống, kiểm tra và sửa lỗi | 100% |
| Chương 5:  (100%) |  | * Hoàn thiện báo cáo, kiểm tra tổng quát | 100% |

### Ghi chú chi tiết:

* **Chương 1:** Bạn đã hoàn thành việc xây dựng cơ sở dữ liệu cho hệ thống web, thiết kế giao diện đăng nhập và đăng ký cho người dùng và quản trị viên.
* **Chương 2:** Phát triển chức năng xem sự kiện cho tất cả người dùng (khách, thành viên).
* **Chương 3:** Phát triển chức năng quản trị sự kiện, bài viết (dành riêng cho admin).
* **Chương 4:** Thực hiện kiểm tra, tìm và sửa các lỗi phát sinh, đảm bảo hệ thống chạy ổn định.
* **Chương 5:** Hoàn thiện báo cáo, kiểm tra tổng thể để chuẩn bị cho việc nghiệm thu.

## Checklist (bảng kiểm tra chức năng)

## Kiểm tra dữ liệu nhập

|  |  |  |
| --- | --- | --- |
| **TT** | **Các thành phần** | **Tình trạng** |
| 1 | Các trường trên form không được trống theo yêu cầu | OK |
| 2 | Dữ liệu nhập phù hợp (Chuỗi, số, ngày, …) | OK |
| 3 | Tài khoản đăng nhập, email là duy nhất | OK |
| 4 | Các ràng buộc (thêm, sửa, xóa) | OK |
| 5 | Kiểm tra dữ liệu nhập trên form đầy đủ | OK |
| 6 | Xử lý dữ liệu trên giỏ hàng đầy đủ | OK |

## Kiểm tra các liên kết

|  |  |  |
| --- | --- | --- |
| **TT** | **Các liên kết** | **Tình trạng** |
| 1 | Tất cả các trang đều hoạt động | OK |
| 2 | Tất cả các liên kết đều hoạt động đúng | OK |
| 3 | Các trang đều không có lỗi chính tả | OK |
| 4 | Các mẫu sử dụng trên mỗi trang đều phù hợp | OK |
| 5 | Các thành phần trong trang đều hoạt động | OK |
| 6 | Ứng dụng tương thích với nhiều trình duyệt phổ biến | OK |
| 7 | Dữ liệu sau đều có nhắc nhở, thông báo | OK |
| 8 | Các trang đều có liên kết đến trang khác | OK |

# **TÀI LIỆU THAM KHẢO**

#### **Tài liệu:**

#### **Website:**

1. <https://www.w3schools.com/>
2. <https://support.microsoft.com/vi-vn/topic/31846479-c656-f2a4-bc24-c9803a97e62c>
3. <https://topdev.vn/blog/sql-server-la-gi/>
4. https://wutheringwaves.kurogames.com/en/main/