**Phân Tích Yêu Cầu Và Đặc Tả (SRS)**

**1. Giới thiệu (Introduction)**

**1.1 Mục đích (Purpose)**

Tài liệu này nhằm mục đích mô tả chi tiết các yêu cầu về chức năng, giao diện người dùng và các yếu tố phi chức năng cho dự án xây dựng hệ thống quản lý điểm danh học sinh (Students Attendance Tracking System).

**1.2 Phạm vi (Scope)**

* Hệ thống quản lý điểm danh học sinh sẽ bao gồm các chức năng chính sau:
* Hệ thống chon phép CRUD (Create, Read, Update, Delete) thông tin chi tiết của học sinh, bao gồm tên và lớp học.
* Hệ thống cho phép CRUD thông tin lớp học, bao gồm tên lớp.
* Hệ thống hỗ trợ gán học sinh vào lớp học.
* Thực hiện điểm danh và theo dõi thống kê tỷ lệ chuyên cần.
* Hệ thống hỗ trợ điểm danh học sinh mỗi ngày theo các trạng thái: Có mặt (present), vắng mặt (absent), ốm (sick), hoặc khác (other).
* Cung cấp một cửa sổ modal để nhập lý do cụ thể khi chọn trạng thái “other”.
* Hệ thống cho phép lọc dữ liệu điểm danh theo tháng và theo lớp.
* Dữ liệu điểm danh sẽ được hiển thị dưới dạng bảng để dễ theo dõi.
* Cung cấp bảng điều khiển (Dashboard) và chức năng phân tích.
* Dashboard sẽ hiển thị biểu đồ trực quan hoá dữ liệu điểm danh.
* Hệ thống phục vụ hai đối tượng người dùng chính: Admin và giáo viên.

**1.3 Định nghĩa, viết tắt (Definitions, Acronyms)**

* **CRUD**: Create, Read, Update, Delete.
* **SRS**: Software Requirements Specification.
* **UI**: User Interface.
* **Modal**: Một hộp thoại hoặc cửa sổ pop-up yêu cầu người dùng tương tác.
* **Responsive**: Khả năng giao diện tự động điều chỉnh để hiển thị tốt trên các kích thước màn hình khác nhau.
* **Dashboard**: Bảng điều khiển tổng quan, hiển thị các thông tin tóm tắt và phân tích.
* **Component**: Một phần độc lập, có thể tái sử dụng của giao diện người dùng

**1.4 Tài liệu tham khảo (References)**

* Laravel 10+ documentation
* Laravel Livewire documentation
* MySQL Documentation

**1.5 Tổng quan tài liệu (Document Overview)**

Tài liệu gồm 5 phần chính: Giới thiệu, Mô tả tổng quan, Yêu cầu cụ thể, Giao diện người dùng và Các yêu cầu khác.

* **Sơ đồ tổng thể dự án (High-level architecture)**

**A diagram of a data flow

AI-generated content may be incorrect.**

**2. Mô tả tổng quan (Overall Description)**

**2.1 Bối cảnh dự án (Project Context)**

Hệ thống được phát triển trên nền tảng Laravel Framework, dùng MySQL làm hệ quản trị cơ sở dữ liệu. Giao diện frontend sẽ được xây dựng bằng Laravel Livewire để tạo sự tương tác.

**2.2 Đặc điểm người dùng (User Characteristics)**

* **Sơ đồ use case**
* **Admin**:
* Có quyền quản lý người dùng (thêm, sửa, xoá tài khoản, gán vai trò).
* Có quyền quản lý lớp học (CRUD).
* Có quyền quản lý học sinh (CRUD).
* Có quyền xem tất cả các thống kê điểm danh trong hệ thống.
* Admin là người dùng có kinh nghiệm sử dụng các hệ thống quản trị.
* **Teacher**:
* Có quyền điểm danh học sinh cho các lớp mà giáo viên đó phụ trách.
* Có quyền quản lý (CRUD) học sinh thuộc các lớp mà giáo viên đó phụ trách.
* Có quyền xem thống kê điểm danh cho các lớp mà giáo viên đó phụ trách.
* Giáo viên là người dùng có kỹ năng sử dụng máy tính cơ bản và trình duyệt web.

**2.3 Ràng buộc và giới hạn (Constraints)**

* Laravel framework 10+
* Database sử dụng MySQL

**2.4 Giả định và sự phụ thuộc (Assumptions & Dependencies)**

* **Giả định:**
* Người dùng cuối (Admin, Teacher) có kiến thức cơ bản về sử dụng trình duyệt web.
* Hệ thống sẽ chạy tốt trên các trình duyệt web hiện đại như Chrome, Edge và Firefox
* **Phụ Thuộc:**
* Hệ thống phụ thuộc vào môi trường máy chủ (server) có cài đặt PHP và MySQL

**3. Yêu cầu cụ thể (Specific Requirements)**

**3.1 Chức năng (Functional Requirements)**

**FR1: User Roles và Authentication**

* **FR1.1: Đăng ký và Đăng nhập người dùng**
* **Mô tả:** Hệ thống cho phép người dùng đăng ký tài khoản mới và đăng nhập vào hệ thống
* **Actor:** Người dùng mới, Admin, Teacher.
* **Luồng chính:**

1. **Đăng ký:** Người dùng cung cấp Email, mật khẩu và các thông tin cần thiết khác (ví dụ: Tên). Hệ thống kiểm tra tích hợp của dữ liệu, hash mật khẩu và lưu tài khoản. (Admin có thể tạo tài khoản cho teacher).
2. **Đăng nhập:** Người dùng nhập Email và mật khẩu. Hệ thống xác thực thông tin đăng nhập.
3. **Nếu xác thực** thành công, hệ thống sẽ đièu hướng người dùng đến giao diện Dashboard phù hợp với vai trò của họ.

* **Dữ liệu đầu vào:** Email, Password.
* **Dữ liệu đầu ra:** Thông báo đăng nhập thành công hoặc thất bại.
* **Luồng sự kiện (ví dụ Đăng nhập):**

1. Người dùng vào trang đăng nhập.
2. Người dùng nhập mật khẩu email, password và submit.
3. Hệ thống kiểm tra thông tin, redirect nếu hợp lệ, thông báo lỗi nếu sai.

* **Trường hợp ngoại lệ:**
  + - **FR1.1.1:** Sai thông tin đăng nhập: Hệ thống thông báo lỗi.
    - **FR1.1.2:** Không nhập đủ thông tin: Hệ thống yêu cầu nhập đầy đủ thông tin.
* **FR1.2: Phân quyền Admin và Teacher**
* **Mô tả:** Hệ thống phân biệt và áp dụng quyền hạn khác nhau cho Admin và Teacher.
* **Actor:** Hệ thống.
* **Quy tắc nghiệp vụ:**
  + - Admin có toàn quyền truy cập và quản lý tất cả các chức năng và dữ liệu của hệ thống.
    - Teacher chỉ có quyền truy cập vào các chức năng liên quan đến điểm danh và quản lý học sinh/lớp học mà họ được phân công.

**FR2: Quản lý học sinh và lớp học**

* **FR2.1: CRUD thông tin học sinh**
* **Mô tả:** Cho phép tạo, đọc, cập nhật và xoá thông tin của học sinh.
* **Actor:** Admin, Teacher (trong phạm vi lớp được phân công).
* **Đầu vào:** Tên học sinh, tuổi, lớp học.
* **Quy tắc nghiệp vụ:** Mã học sinh phải là duy nhất.
* **FR2.2: CRUD thông tinh lớp học**
* **Mô tả:** Cho phép tạo, đọc, cập nhật và xoá thông tin của các lớp học.
* **Actor:** Admin.
* **Đầu vào:** Tên lớp học.
* **Quy tắc nghiệp vụ:** Tên lớp học phải là duy nhất.
* **FR2.3: Gán học sinh vào lớp học**
* **Mô tả:** Cho phép liên kết một học sinh với một lớp học cụ thể.
* **Actor:** Admin, Teacher (trong phạm vi lớp được phân công).
* **Quy tắc nghiệp vụ:** Một hoc sinh chỉ thuộc một lớp tại một thời điểm. Việc gán học sinh vào một lớp sẽ cập nhật thông tin lớp hiện tại của học sinh đó.

**FR3: Điểm danh học sinh**

* **FR3.1: Điểm danh hang ngày**
* **Mô tả:** Giáo viên có thể điểm danh học sinh của một lớp vào mỗi ngày.
* **Actor:** Teacher.
* **Input:** Lớp học, ngày điểm danh, trạng thái điểm danh cho từng học sinh.
* **Output:** Bản ghi điểm danh được lưu vào cơ sở dữ liệu.
* **Các trạng thái điểm danh:** Có mặt (present), vắng mặt (absent), ốm (sick), khác (other).
* **FR3.2: Nhập lý do cho trạng thái “other”**
* **Mô tả:** Khi giáo viên chọn trạng thái “other”, một cửa sổ modal sẽ hiện lên để nhập lý do cụ thể.
* **Actor:** Teacher.
* **Input:** Chuỗi văn bản lý do.

**FR4: Lọc và thống kê điểm danh**

* **FR4.1: Lọc điểm danh**
* **Mô tả:** Cho phép người dùng lọc các bản ghi điểm danh.
* **Actor:** Admin, Teacher.
* **Input:** Lựa chọn tháng , lựa chọn lớp học.
* **Output:** Bảng dữ liệu điểm danh được cập nhật theo tiêu chi lọc.
* **FR4.2: Hiển thị dữ liệu dưới dạng bảng**
* **Mô tả:** Dữ liệu điểm danh được hiển thị dưới dạng bảng để dễ dàng theo dõi.
* **Actor:** Admin, Teacher.
* **Output:** Bảng hiển thị thông tin chi tiết của các bản ghi điểm danh (ví dụ: tên học sinh, lớp, ngày , trạng thái, lý do).
* **Tính năng bổ sung:** Bảng có thể bao gồm các tuỳ chọn hành động (action dropdown) để chỉnh sửa hoặc xem chi tiết bàn ghi điểm danh.

**FR5: Dashboard và Analytics**

* **FR5.1: Biểu đồ trực quan hoá dữ liệu điểm danh**
* **Mô tả:** Dashboard sẽ hiển thị các biểu đồ trực quan hoá dữ liệu điểm danh, cung cấp cái nhìn tổng quan nhanh chóng.
* **Actor:** Admin, Teacher.
* **Output:** Các biểu đồ hiển thị điểm danh theo trạng thái, theo lớp, hoặc theo tổng thể.
* **Quy tắc nghiệp vụ:**
* Admin có thể xem biểu đồ của tất cả các lớp.
* Teacher chỉ xem được biểu đồ của các lớp mà họ phụ trách.

**3.2 Phi chức năng (Non-functional Requirements)**

| **Yêu cầu** | **Mô tả** |
| --- | --- |
| Bảo mật (Security) | - Hash mật khẩu bằng **bcrypt** để bảo vệ thông tin đăng nhập.  - Quản lý phiên đăng nhập an toàn để đảm bảo chỉ người dùng được xác thực mới có quyền truy cập. - Thực thi phân quyền dựa trên vai trò để ngăn chặn truy cập trái phép vào các chức năng và dữ liệu. |
| Hiệu suất (Performance) | - Tải trang nhanh, thời gian tải không vượt quá 3 giây.  - Các thao tác xử lý dữ liệu (CRUD, lọc) phải có phản hồi nhanh chóng, dưới 2 giây. |
| Khả dụng  (Usability) | - Giao diện phải trực quan và dễ sử dụng.  - Giao diện phải **responsive**, đảm bảo hiển thị tốt và trải nghiệm người dùng nhất quán trên các kích thước màn hình khác nhau (máy tính, máy tính bảng, điện thoại).  - Cung cấp phản hồi rõ ràng cho người dùng (thông báo lỗi, thành công). |
| Tính tương thích (Compatibility) | - Hệ thống phải hỗ trợ và hoạt động tốt trên các trình duyệt web hiện đại như Chrome, Firefox, và Edge. |
| Khả năng bảo trì (Maintainability) | - Mã nguồn phải được viết theo chuẩn của Laravel, đảm bảo dễ đọc và dễ hiểu.  - Hệ thống phải dễ dàng mở rộng, cho phép thêm các chức năng mới mà không làm ảnh hưởng lớn đến cấu trúc hiện có. |

**4. Giao diện người dùng (User Interface Requirements)**

* Giao diện responsive, thân thiện với người dùng
* Sử dụng các framework CSS như TailwindCSS và Bootstrap để xây dựng giao diện.
* Các components giao diện chính sẽ bao gồm: Forms (biểu mẫu), Tables (bảng), Modals (cửa sổ pop-up), Dropdowns (menu thả xuống).
* **Các màn hình chính:**
* Trang đăng nhập: Màn hình đầu tiên người dùng tương tác, bao gồm các trường nhập liệu Email, Password, nút Đăng nhập.
* Dashboard: Cung cấp cái nhìn tổng quan về hệ thống và dữ liệu điểm danh.
* Màn hình quản lý học sinh và lớp học: Các màn hình riêng biệt để quản lý danh sách và thông tin chi tiết của từng học sinh và lớp học.
* Màn hình điểm danh: Giao diện cho phép giáo viên thực hiện điểm danh học sinh.
* Màn hình thống kê và lọc dữ liệu điểm danh: Hiển thị dữ liệu điểm danh dưới dạng bảng cùng với các tuỳ chọn lọc.