**ĐẶC TẢ YÊU CẦU PHẦN MỀM (SRS)**

**HỆ THỐNG QUẢN LÝ ĐIỂM DANH BẰNG NHẬN DIỆN KHUÔN MẶT – TRƯỜNG ĐẠI HỌC THỦY LỢI**

# 1. Mục đích tài liệu

Tài liệu này mô tả các yêu cầu chức năng, phi chức năng, và yêu cầu chất lượng cho hệ thống quản lý điểm danh sinh viên sử dụng công nghệ nhận diện khuôn mặt (Face Recognition). Hệ thống nhằm tự động hóa quy trình điểm danh, giảm thiểu gian lận, tăng độ chính xác, và hỗ trợ công tác quản lý học tập tại Trường Đại học Thủy Lợi.

# 2. Mô tả tổng quan

Hiện nay, phương thức điểm danh thủ công gây tốn thời gian, dễ sai sót và gian lận (điểm danh hộ). Hệ thống được xây dựng nhằm hỗ trợ giảng viên điểm danh tự động bằng camera, cho phép sinh viên điểm danh bằng nhận diện khuôn mặt qua thiết bị di động, và cung cấp giao diện web đa vai trò: Admin – Giảng viên – Sinh viên.

# 3. Đối tượng sử dụng hệ thống

|  |  |  |
| --- | --- | --- |
| **Mã đối tượng** | **Tên** | **Mô tả** |
| **US-01** | **Sinh viên (Student)** | - Là người dùng cuối của hệ thống, truy cập ứng dụng Flutter bằng tài khoản được cấp. - Thực hiện điểm danh bằng camera hoặc mã QR. - Có thể xem lịch sử điểm danh, trạng thái buổi học và thông tin lớp học. - Không có quyền chỉnh sửa dữ liệu hoặc cấu hình. |
| **US-02** | **Giảng viên (Lecturer)** | - Là người trực tiếp quản lý lớp học và thực hiện điểm danh. - Có quyền tạo, chỉnh sửa, xóa lớp học, môn học và buổi học. - Tạo mã QR cho sinh viên điểm danh và giám sát quá trình nhận diện khuôn mặt theo thời gian thực. - Có thể xuất báo cáo chuyên cần, thống kê sinh viên và chỉnh sửa điểm danh khi cần. |
| **US-03** | **Quản trị viên hệ thống (System Admin)** | - Là đối tượng có toàn quyền trong hệ thống. - Quản lý tài khoản giảng viên và sinh viên, thiết lập tham số nhận diện (ngưỡng Euclidean distance, thời gian hợp lệ). - Quản lý sao lưu dữ liệu, bảo trì, cập nhật hệ thống và giám sát log hoạt động. - Không thể bị xóa khỏi hệ thống. |

# 4. Phân nhóm chức năng

|  |  |  |
| --- | --- | --- |
| **Mã nhóm chức năng** | **Tên nhóm chức năng** | **Mô tả chi tiết** |
| **FC-01** | **User Module** | Chịu trách nhiệm quản lý toàn bộ thông tin người dùng của hệ thống bao gồm sinh viên, giảng viên và quản trị viên. Module này cung cấp các chức năng như đăng ký, đăng nhập, đổi mật khẩu, phân quyền truy cập và cập nhật hồ sơ cá nhân. Đồng thời, đảm bảo tính bảo mật và tính toàn vẹn của dữ liệu người dùng trong suốt vòng đời sử dụng. |
| **FC-02** | **Class Module** | Quản lý toàn bộ quy trình liên quan đến lớp học, môn học và buổi học. Giảng viên có thể tạo mới, chỉnh sửa, xóa lớp học, gán sinh viên vào lớp, và tạo buổi học tương ứng. Module này cũng chịu trách nhiệm đồng bộ dữ liệu lớp học với danh sách sinh viên từ tệp Excel hoặc API nội bộ. |
| **FC-03** | **Attendance Module** | Là module cốt lõi của hệ thống, đảm nhận xử lý toàn bộ quy trình điểm danh tự động. Hỗ trợ hai cơ chế chính: quét mã QR để mở phiên điểm danh và nhận diện khuôn mặt thông qua camera hoặc ảnh tải lên. Module này sử dụng TensorFlow.js / face-api.js để trích xuất đặc trưng khuôn mặt và so khớp dữ liệu sinh viên trong cơ sở dữ liệu MySQL. |
| **FC-04** | **Report Module** | Phụ trách tổng hợp và hiển thị dữ liệu thống kê chuyên cần của sinh viên. Giảng viên và quản trị viên có thể xem tỷ lệ điểm danh theo buổi, theo môn học hoặc toàn kỳ. Module này cũng hỗ trợ xuất báo cáo ra các định dạng phổ biến như Excel và PDF, đồng thời cung cấp biểu đồ trực quan để đánh giá hiệu quả học tập. |
| **FC-05** | **Admin Module** | Cung cấp cho quản trị viên hệ thống khả năng giám sát, cấu hình và điều hành toàn bộ nền tảng. Bao gồm quản lý tài khoản người dùng, cấu hình ngưỡng nhận diện khuôn mặt, giám sát nhật ký hoạt động (log) và xử lý các sự cố kỹ thuật. Đây là module có quyền cao nhất, đảm bảo hệ thống vận hành ổn định và an toàn. |
| **FC-06** | **Security Module** | Đảm bảo an toàn và bảo mật dữ liệu người dùng, đặc biệt là dữ liệu khuôn mặt. Module này chịu trách nhiệm mã hóa, xác thực JWT, phân quyền chi tiết cho từng nhóm người dùng, và phát hiện truy cập bất thường. Ngoài ra, hệ thống có cơ chế sao lưu, khôi phục và giám sát nhật ký bảo mật nhằm giảm thiểu rủi ro rò rỉ thông tin. |

# 5. Chi tiết chức năng

Chi tiết chức năng theo từng module:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Mã nhóm chức năng** | **Mã chức năng** | **Mã đối tượng sử dụng** | **Tên chức năng** | **Mô tả chi tiết** |
| FC-01 | FC-01-01 | US-01, US-02, US-03 | Đăng ký tài khoản | Tạo tài khoản mới: họ tên, mã số, email, mật khẩu; xác thực email/OTP; lưu MySQL. |
| FC-01-02 | US-01, US-02, US-03 | Đăng nhập | Đăng nhập bằng email + mật khẩu; phát hành JWT; kiểm soát phiên an toàn. |
| FC-01-03 | US-01, US-02, US-03 | Đăng xuất | Kết thúc phiên; vô hiệu hóa token tạm; ghi log hoạt động. |
| FC-01-04 | US-01, US-02, US-03 | Cập nhật thông tin cá nhân | Cập nhật avatar, số điện thoại, email, lớp/bộ môn; kiểm tra hợp lệ dữ liệu. |
| FC-01-05 | US-01, US-02, US-03 | Quên / Đổi mật khẩu | Đổi mật khẩu trong cài đặt; gửi OTP khôi phục qua email khi quên mật khẩu. |
| FC-01-06 | US-03, US-04 | Phân quyền người dùng | Gán/chỉnh quyền cho Giảng viên, Sinh viên, Nhân viên hỗ trợ theo vai trò. |
| FC-02 | FC-02-01 | US-02 | Tạo lớp học / môn học | Tạo lớp/môn, sinh mã định danh, gán giảng viên và danh sách sinh viên. |
| FC-02-02 | US-02 | Cập nhật lớp học / môn học | Sửa tên, mô tả, lịch học, sĩ số; đồng bộ theo học kỳ. |
| FC-02-03 | US-02 | Tạo buổi học | Khai báo ngày/giờ/phòng; sinh mã QR; thời hạn hiệu lực điểm danh. |
| FC-02-04 | US-02 | Quản lý sinh viên | Thêm/xóa sinh viên; import Excel; cập nhật thông tin và trạng thái. |
| FC-02-05 | US-01, US-02 | Xem lịch & danh sách buổi | Xem thời khóa biểu, danh sách buổi theo tuần/tháng; trạng thái điểm danh. |
| FC-03 | FC-03-01 | US-01 | Điểm danh bằng QR code | Quét QR mở phiên; gửi thông tin lớp/buổi kèm mã xác thực tới API. |
| FC-03-02 | US-01 | Điểm danh bằng nhận diện khuôn mặt | Bật camera hoặc tải ảnh; trích xuất vector (face-api.js); so khớp mẫu; ghi nhận kết quả. |
| FC-03-03 | US-02 | Giám sát điểm danh thời gian thực | Bảng real-time: danh sách sinh viên đã/đang điểm danh, ảnh xác minh, trạng thái hợp lệ. |
| FC-03-04 | US-02 | Chấm công thủ công / xác nhận | Sửa trạng thái điểm danh đối với trường hợp ngoại lệ hoặc lỗi nhận diện. |
| FC-03-05 | US-02 | Quản lý kết quả điểm danh | Lọc/tìm kiếm/xuất kết quả theo buổi/lớp; khóa dữ liệu sau khi xác nhận. |
| FC-04 | FC-04-01 | US-02, US-03 | Xem thống kê chuyên cần | Biểu đồ/tỷ lệ có mặt-vắng-muộn theo lớp, buổi, sinh viên; lọc theo thời gian. |
| FC-04-02 | US-02 | Xuất báo cáo điểm danh | Xuất Excel/PDF; tùy chọn cột; đóng dấu lớp/khoa; chữ ký số (tùy chọn). |
| FC-04-03 | US-01 | Xem lịch sử điểm danh cá nhân | Danh sách buổi; trạng thái; thời gian xác nhận; ghi chú giảng viên (nếu có). |
| FC-05 | FC-05-01 | US-03 | Quản lý người dùng | Thêm/sửa/xóa; khóa/mở tài khoản; đặt lại mật khẩu; gán vai trò. |
| FC-05-02 | US-03 | Quản lý cấu hình hệ thống | Thiết lập ngưỡng so khớp, thời hạn QR, giới hạn thời gian điểm danh, sao lưu. |
| FC-05-3 | US-03 | Quản lý nhật ký hoạt động | Xem log đăng nhập, thay đổi dữ liệu, lỗi API; xuất log phục vụ kiểm tra. |
| FC-05-04 | US-03 | Sao lưu & khôi phục dữ liệu | Lịch sao lưu MySQL; tải bản sao; khôi phục theo ngày/giờ. |
| FC-06 | FC-06-01 | US-03, US-04 | Quản lý bảo mật | Bật HTTPS; cấu hình JWT; chính sách mật khẩu; giới hạn tốc độ; CORS. |
| FC-06-02 | US-03, US-05 | Quản lý dữ liệu khuôn mặt | Quản trị dữ liệu mẫu: cập nhật/xóa theo yêu cầu; mã hóa lưu trữ; theo dõi truy cập. |
| FC-06-03 | US-03 | Phân tích rủi ro & cảnh báo | Phát hiện bất thường (đăng nhập thất bại, truy cập trái phép); gửi cảnh báo cho admin. |

**6. YÊU CẦU PHI CHỨC NĂNG**

**6.1. Hiệu năng**

**•** Hệ thống phải xử lý tối thiểu 500 yêu cầu đồng thời (concurrent requests) mà không suy giảm đáng kể tốc độ phản hồi.

**•** Thời gian phản hồi API cốt lõi (đăng nhập, mở phiên điểm danh bằng QR, xác thực khuôn mặt) trung bình ≤ 1.0 giây; P95 ≤ 1.5 giây.

**•** Thời gian tải trang/màn hình Flutter: ≤ 1.5 giây với nội dung cơ bản; ≤ 2.0 giây với màn hình dữ liệu lớn (danh sách lớp/buổi học nhiều bản ghi).

**•** Độ trễ điểm danh end‑to‑end (từ lúc gửi ảnh tới khi ghi nhận) trung bình ≤ 1.2 giây với điều kiện mạng ổn định.

**6.2. Khả năng mở rộng**

**•** MySQL hỗ trợ mở rộng đọc (read replica) và phân vùng dữ liệu theo học kỳ/khoa nếu cần.

**•** Module nhận diện khuôn mặt tách lớp (service riêng hoặc thư viện) để thay thế/nâng cấp mà không ảnh hưởng các module khác.

**•** Cho phép tích hợp module mới (ví dụ: GPS, Liveness Detection) thông qua API chuẩn và cơ chế versioning.

**6.3. Bảo mật**

**•** Áp dụng RBAC phân quyền theo vai trò (Sinh viên, Giảng viên, Local Admin, System Admin).

**•** Xác thực bằng JWT; phiên làm việc có thời hạn và cơ chế làm mới (refresh token).

**•** Mật khẩu phải được băm bằng bcrypt (tối thiểu 10 rounds); không lưu trữ dạng thuần (plaintext).

**6.4. Khả năng bảo trì**

**•** Mã nguồn tổ chức theo kiến trúc phân lớp rõ ràng (controller/service/repository) và tiêu chuẩn code style.

**•** API được version hóa (v1, v2…) để nâng cấp không phá vỡ tương thích ngược.

**•** Tài liệu kiến trúc, hướng dẫn triển khai, và nhật ký thay đổi (CHANGELOG) phải được cập nhật thường xuyên.

**•** Thiết lập test tự động (unit/integration) bao phủ tối thiểu 60% các đường logic cốt lõi.

**6.5. Tính chính xác và nhất quán**

**•** Ghi nhận dữ liệu điểm danh phải là giao dịch nguyên tử: tạo bản ghi, lưu thời gian, trạng thái và ràng buộc lớp/buổi.

**•** Các API ghi dữ liệu phải idempotent để tránh trùng lặp khi retry (ví dụ: theo session\_id/attendance\_id).

**•** Đồng bộ thời gian hệ thống (NTP) để đảm bảo dấu thời gian (timestamp) nhất quán giữa client và server.

**•** Tỷ lệ nhận diện đúng mục tiêu ≥ 95% trong điều kiện ánh sáng/phạm vi khuyến nghị; ngưỡng so khớp (Euclidean/cosine) có thể cấu hình.

**6.6. Khả năng phục hồi**

**•** Sao lưu MySQL định kỳ (tối thiểu hằng ngày); lưu trữ bản sao tại vị trí độc lập.

**•** RTO (Recovery Time Objective) ≤ 2 giờ; RPO (Recovery Point Objective) ≤ 24 giờ.

**•** Cơ chế retry/bù trừ khi lỗi tạm thời (mạng, dịch vụ nhận diện) và cảnh báo khi thất bại nhiều lần.

**•** Giám sát sức khỏe dịch vụ (health checks) và tự động khởi động lại khi gặp lỗi nghiêm trọng.

**6.7. Giao diện người dùng**

**•** Giao diện Flutter đơn giản, đồng nhất, hỗ trợ tiếng Việt và responsive (mobile, tablet, web).

**•** Thao tác điểm danh ≤ 3 bước: Quét QR → Bật camera → Xác thực thành công/thất bại.

**•** Đảm bảo khả năng truy cập (accessibility) cơ bản: kích thước chữ tối thiểu, độ tương phản, thông báo lỗi rõ ràng.

**•** Có trạng thái dự phòng (fallback UI) khi camera bị từ chối quyền hoặc kết nối kém.

**7. KIẾN TRÚC HỆ THỐNG (Dự án Flutter + Laravel + MySQL)**

**7.1. Mô hình tổng thể**

Hệ thống áp dụng mô hình Client–Server 3 lớp, tối ưu cho ứng dụng Flutter (Android /Web):  
• Lớp trình bày (Client): Ứng dụng Flutter (mobile & web) – giao diện người dùng, quét QR, chụp ảnh/camera, gửi yêu cầu điểm danh.  
• Lớp dịch vụ (API): Laravel – xác thực JWT, xử lý nghiệp vụ, điều phối nhận diện khuôn mặt, sinh mã QR, xuất báo cáo.  
• Lớp dữ liệu (DB & Storage): MySQL lưu dữ liệu nghiệp vụ lưu ảnh mẫu khuôn mặt & minh chứng điểm danh.

**7.2. Thành phần Frontend (Flutter)**

• Nền tảng: Flutter (Android/iOS/Web). State management gợi ý: Riverpod/Bloc.  
• Camera: gói camera hoặc mobile\_scanner (Web dùng getUserMedia do Flutter web cung cấp).  
• Quét QR: qr\_code\_scanner hoặc mobile\_scanner để đọc mã QR mở phiên điểm danh.  
• Kết nối API: http/dio; lưu token JWT an toàn (secure storage trên mobile, cookies/LocalStorage trên web).  
• Xử lý UI: màn hình đăng nhập, quét QR, bật camera, xem kết quả, lịch sử; hỗ trợ tiếng Việt & responsive.

**7.3. Xử lý nhận diện khuôn mặt (Face Recognition)**

Phương án mặc định (đề xuất): xử lý phía server để đồng bộ chất lượng và kiểm soát bảo mật.  
• Server-side: Node.js tích hợp TensorFlow.js/face-api.js để trích xuất vector (embedding) và so khớp (cosine/Euclidean) với dữ liệu mẫu trong DB/Storage.  
• Client-side (tùy chọn nâng cao): dùng tflite\_flutter/mediapipe để trích xuất embedding ngay trên thiết bị; chỉ gửi embedding/metadatas lên API (tiết kiệm băng thông).

**7.4. Thành phần Backend (Laravel)**

* **Framework:** Laravel 11 (PHP ≥ 8.2), API ở routes/api.php, prefix /api/v1.
* **Bảo mật (JWT):** tymon/jwt-auth (đăng nhập, refresh, blacklist); ttl=15 phút, refresh\_ttl=7 ngày.
* **Module nghiệp vụ:** User, Class, Session (buổi học), Attendance.
* **Upload ảnh:** Upload multipart bằng Request::file() + Storage (disk local/public); tùy chọn xử lý ảnh intervention/image.
* **QR Code:** simplesoftwareio/simple-qrcode sinh QR; ký **HMAC-SHA256** với hash\_hmac() (payload: {sid, exp, sig}).
* **Import/Export Excel:** maatwebsite/excel (import danh sách SV, export điểm danh .xlsx).
* **Export PDF:** barryvdh/laravel-dompdf (render Blade → PDF báo cáo).

**7.5. Cơ sở dữ liệu (MySQL)**

Bảng chính (gợi ý): users, roles, classes, class\_students, sessions, attendance, face\_samples.  
• attendance gồm: attendance\_id, session\_id, user\_id, status (present/absent/late), verified\_at, method (qr/face), confidence.  
• face\_samples gồm: sample\_id, user\_id, embedding (mã hóa), image\_url (tùy chọn), created\_at.

**7.6. QR & Camera**

• Nội dung QR: session\_id, class\_id, exp, signature (HMAC). App Flutter quét và gửi lên API để mở phiên điểm danh.  
• Camera: Flutter camera/mobile\_scanner; trên web dùng getUserMedia qua Flutter web; cho phép chụp ảnh/stream khung hình để server xử lý.

**7.7. Triển khai & Vận hành**

• Docker Compose: tách service (api, db, storage, reverse proxy).  
• Mở rộng ngang: scale replica API; MySQL có read-replica khi cần; tách service Face Recognition nếu cần GPU.

**7.8. Luồng dữ liệu điểm danh (rút gọn)**

1) Giảng viên tạo buổi học → API sinh session + QR (exp, signature).  
2) Sinh viên quét QR trên Flutter → app gửi token + session\_id + ảnh/embedding lên API.  
3) API xác thực JWT → chuyển ảnh tới module nhận diện → so khớp embedding với mẫu.  
4) Ghi bản ghi attendance (status, confidence, timestamp) vào MySQL → trả kết quả cho app.  
5) Giảng viên theo dõi thời gian thực và xuất báo cáo khi kết thúc buổi.

# 8. Yêu cầu chất lượng

|  |  |  |  |
| --- | --- | --- | --- |
| **Yêu cầu chất lượng** | **Chuẩn tham chiếu** | **Mục tiêu định lượng** | **Phương thức đánh giá / kiểm thử** |
| Độ tin cậy (Reliability) | ISO/IEC 25010 | Uptime ≥ 99.9%; MTBF ≥ 1,000 giờ; P95 lỗi ứng dụng = 0 trong ca thi/điểm danh. | Stress/soak test; chaos test lỗi dịch vụ phụ; giám sát uptime qua health-check định kỳ 30s. |
| Tính chính xác & Nhất quán dữ liệu | ISO/IEC 25012 | Độ chính xác nhận diện ≥ 95% (ROC-AUC ≥ 0.97); dữ liệu điểm danh idempotent; mất bản ghi = 0. | Kiểm thử tích hợp end‑to‑end; so khớp đối chiếu log API ↔ DB; kiểm thử idempotency theo attendance\_id. |
| Khả năng sử dụng (Usability) | ISO 9241‑11 | SUS ≥ 80; Tỉ lệ hoàn thành nhiệm vụ ≥ 90% trong ≤ 3 thao tác (Quét QR → Bật camera → Xác thực). | UAT với ≥ 20 người dùng (SV/GV); đo thời gian thao tác; khảo sát SUS/CSAT sau thử nghiệm. |
| Hiệu năng (Performance) | ISO/IEC 25010 | ≥ 500 concurrent requests; P95 API core ≤ 1.5s; Latency điểm danh end‑to‑end ≤ 1.2s. | Load test (k6/JMeter) cho login/QR/nhận diện; profiling nút nghẽn; tối ưu truy vấn MySQL. |
| Bảo mật (Security) | ISO/IEC 27001; OWASP Top 10 | RBAC + JWT (token exp ≤ 2h, refresh ≤ 7d); bcrypt ≥ 10 rounds; TLS 1.2+; AES‑256 at‑rest. | Pentest định kỳ; VA hàng quý theo OWASP; kiểm tra cấu hình TLS/HSTS; review mã hóa & log bảo mật. |
| Khả năng phục hồi (Resilience) | ISO/IEC 27040 | RTO ≤ 2h; RPO ≤ 24h; phục hồi bản sao lưu toàn vẹn 100% bảng nghiệp vụ. | Diễn tập DR mỗi học kỳ; khôi phục thử từ backup; kiểm tra tính toàn vẹn (checksum) sau khôi phục. |
| Khả năng bảo trì (Maintainability) | ISO/IEC 25010 | Coverage unit/integration ≥ 60%; API versioning (v1+); PR review 100%; CC per fn ≤ 10. | CI chạy test tự động; static analysis; quy trình code review; đo cyclomatic complexity. |
| Khả năng quan sát (Observability) | SRE Practice (SLO/SLA) | SLO: P95 login/QR/nhận diện ≤ 1.5s; Error budget ≤ 0.1%/tuần; 100% sự cố có post‑mortem. | Thu thập log/metrics/traces; alert theo ngưỡng SLO; RCA & hành động khắc phục được theo dõi. |
| Khả năng truy cập & giao diện (Accessibility/UI) | WCAG 2.1 (Áp dụng cơ bản) | Độ tương phản ≥ 4.5:1; font ≥ 14px; hỗ trợ tiếng Việt; responsive mobile/web. | Heuristic evaluation; test trên thiết bị thật; kiểm tra tương phản & kích thước mục tiêu cảm ứng. |

# 9. Tiêu chí chấp nhận

• Hoàn thành ≥ 95% yêu cầu chức năng.  
• Không quá nhiều lỗi nghiêm trọng sau kiểm thử.  
• Giao diện đồng nhất, đầy đủ 3 vai trò (Admin / Giảng viên / Sinh viên).  
• Thời gian nhận diện khuôn mặt ≤ 5 giây.  
• Bàn giao đúng tiến độ, có hướng dẫn triển khai.