

Task 3: Architecture design

Kiến trúc hệ thống

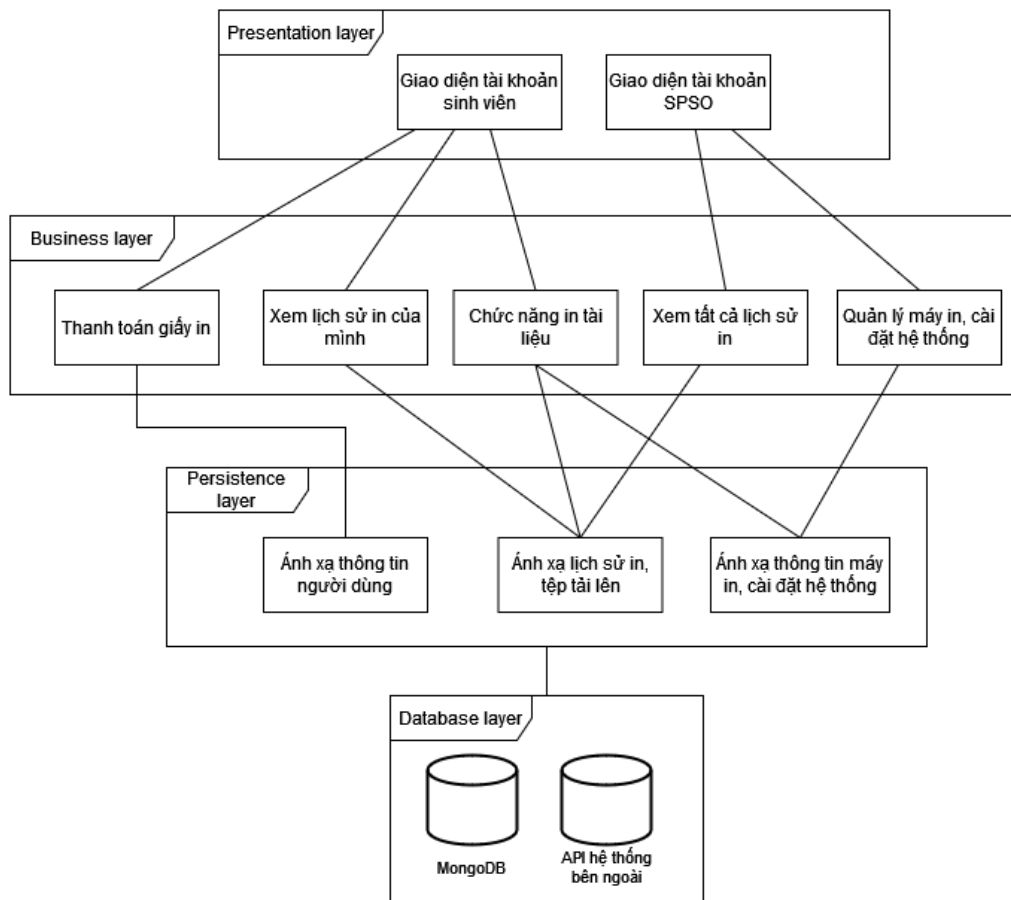


Figure 1: Layered Architecture của hệ thống

Mô tả: Là sơ đồ kiến trúc phân lớp mô tả các tầng chính trong hệ thống SSPS. Hệ thống được chia thành các lớp theo chức năng, từ giao diện người dùng đến cơ sở dữ liệu. Các tầng sẽ giao tiếp với nhau thông qua API fetch hoặc axios, và mỗi tầng chỉ giao tiếp với tầng kế nó.

- Tầng Presentation Layer (Tầng trình bày):
 - Giao diện tài khoản sinh viên.
 - Giao diện tài khoản SPSO (người quản trị hoặc nhân viên hỗ trợ).

=> Đây là nơi người dùng tương tác trực tiếp với hệ thống.

- Tầng Business Layer (Tầng nghiệp vụ):
 - Chức năng in tài liệu: Xử lý các tác vụ liên quan đến việc in ấn.
 - Thanh toán giấy in: Hỗ trợ mua giấy.
 - Xem lịch sử in của mình: Cho phép người dùng tra cứu lịch sử in trước đó.
 - Xem tất cả lịch sử in: Cho phép người dùng tra cứu tất cả lịch sử in.
 - Quản lý máy in, cài đặt hệ thống: Chức năng dành cho người quản trị để quản lý thiết bị.

=> Đây là tầng xử lý logic nghiệp vụ của hệ thống.

- Tầng Persistence Layer (Tầng lưu trữ):
 - Ảnh xạ lịch sử in, tệp tài liệu tải lên.
 - Ảnh xạ thông tin máy in, cài đặt hệ thống.
 - Ảnh xạ thông tin người dùng.

=> Tầng này chịu trách nhiệm yêu cầu lấy và lưu trữ dữ liệu cần thiết để hỗ trợ các hoạt động nghiệp vụ.

Tầng Database Layer (Tầng cơ sở dữ liệu): MongoDB: Cơ sở dữ liệu nội bộ. API hệ thống bên ngoài: Cung cấp tích hợp với các hệ thống bên ngoài.

=> Tầng này lưu trữ dữ liệu lâu dài và hỗ trợ giao tiếp với các cơ sở dữ liệu hoặc dịch vụ bên ngoài.

Ý nghĩa: Sơ đồ này phân chia rõ ràng các trách nhiệm trong hệ thống, giúp tăng tính bảo trì, khả năng mở rộng, và quản lý dễ dàng hơn. Mỗi tầng tập trung vào một nhiệm vụ cụ thể và giao tiếp với tầng bên trên hoặc bên dưới.

Kế hoạch trình bày giao diện (Presentation strategy)

Hệ thống HCMUT-SSPS được xây dựng theo kiến trúc phân lớp với các tầng: giao diện người dùng (Presentation layer), lớp nghiệp vụ (Business layer), và lớp dữ liệu (Persistence layer và Database layer). Trong đó, tầng giao diện người dùng được thiết kế một cách đơn giản và trực quan, dễ dàng truy cập và thao tác với các chức năng chính của hệ thống như đăng tải tệp, chọn máy in, thanh toán, tra cứu lịch sử in,... Giao diện sử dụng ReactJS để tối ưu hóa khả năng tái sử dụng các thành phần giao diện và nâng cao hiệu suất nhờ cơ chế Virtual DOM. Kết hợp với Tailwind CSS, UI được thiết kế linh hoạt, dễ tùy chỉnh và tương thích với nhiều kích thước màn hình. Kiến trúc phân lớp giúp hệ thống dễ dàng bảo trì và mở rộng, với mọi thao tác của người dùng được xử lý qua API, truyền tải đến các lớp dưới, đảm bảo tính phân tách và an toàn dữ liệu.

Phương pháp lưu trữ dữ liệu (Data storage approach)

Hệ thống sẽ sử dụng MongoDB làm cơ sở lưu trữ dữ liệu chính vì tính linh hoạt về cấu trúc dữ liệu và khả năng mở rộng cao. Ngoài ra, hệ thống không có các mối quan hệ phức tạp giữa các bảng như trong cơ sở dữ liệu quan hệ SQL, và MongoDB cho phép lưu trữ các đối tượng phức tạp trực tiếp trong một document mà không cần “join” các bảng, giúp tăng hiệu năng truy vấn. Do vậy, nhóm chúng em quyết định chọn MongoDB làm cơ sở dữ liệu.

Các bảng dữ liệu chính sẽ bao gồm:

- Users (thông tin về tài khoản người dùng như là lượng giấy trong tài khoản)
- Printers (thông tin máy in như là vị trí, trạng thái)
- Printing Logs (lịch sử in ấn)
- System Configurations (cấu hình hệ thống).

Đối với tài liệu cần in, hệ thống sẽ sử dụng lưu trữ đám mây Google Cloud Storage để lưu trữ các file tạm thời.

Quản lý API (API management)

Hệ thống sẽ xây dựng một API Gateway làm điểm vào chung cho ứng dụng, đồng thời tích hợp với các dịch vụ bên ngoài thông qua RESTful API. API của hệ thống sẽ chia làm 2 loại:

- API bên ngoài: Việc xác thực người dùng, thanh toán hay giao tiếp với Database sẽ được thực hiện thông qua HCMUT_SSO, BKPay thông qua các API được cung cấp. Dự tính là sẽ sử dụng API Key được cấp và gọi HTTP Request tương ứng, API Key sẽ được lưu vào biến môi trường (tệp .env) để bảo mật.
- API hệ thống: Các API trên hệ thống SSPS như là GET, POST, DELETE, phục vụ cho các chức năng của người dùng, như là chức năng in ấn, chức năng quản lý của SPSO, chức năng thông báo của máy in hay chức năng xem lịch sử in ấn,... Dự tính sẽ sử dụng fetch hoặc axios của RestfulAPI để giao tiếp giữa frontend và backend.

Component Diagram

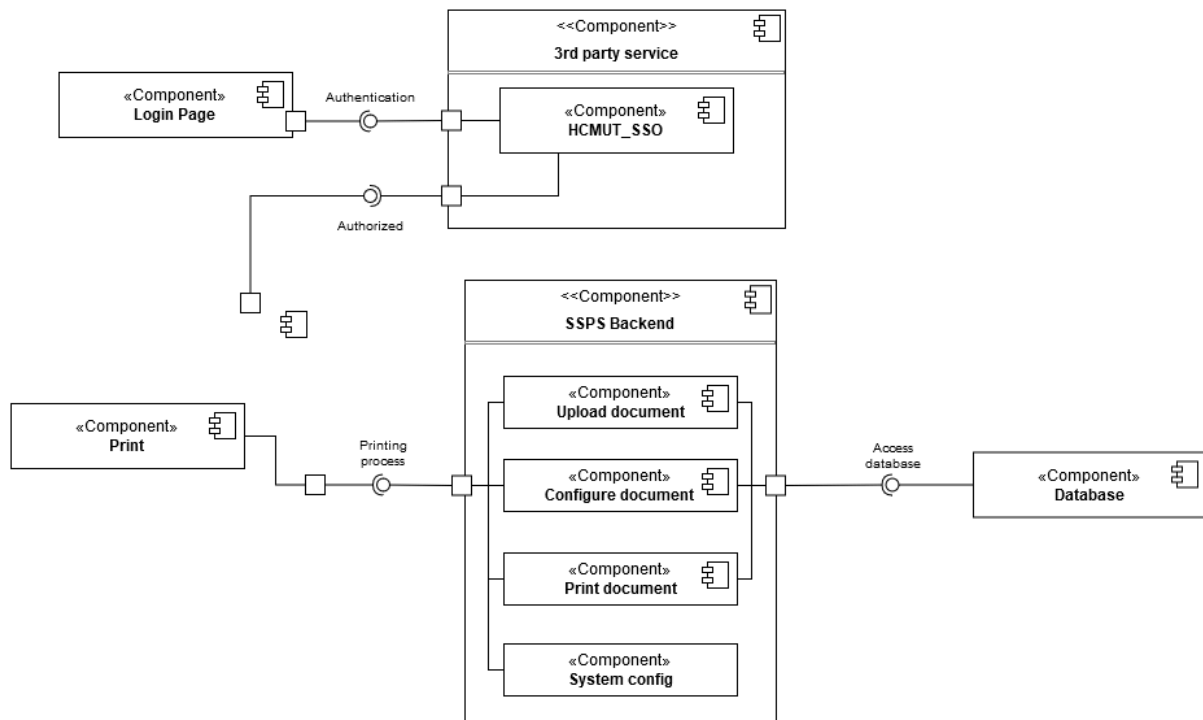


Figure 2: Component Diagram của hệ thống

Mô tả sơ đồ component của Print Process Module:

- Module bao gồm các thành phần chính như: Login, Student Page, SSPS (Student Printing Service System), 3rd Party Service và Database. Và các thành phần sẽ giao tiếp với nhau thông qua các API tương ứng.
- Login: Đây là nơi người dùng sẽ thực hiện chức năng đăng nhập, giao tiếp với dịch vụ HCMUT_SSO, một thành phần từ bên thứ ba, để thực hiện việc xác thực thông tin đăng nhập.
- Student Page: Sau khi sinh viên được xác thực đăng nhập, họ sẽ được dẫn đến Student Page, là trang chủ, nơi cung cấp các chức năng liên quan đến việc in ấn.
- Hệ thống SSPS là nơi xử lý các quy trình in ấn, bao gồm các thành phần: Đăng tải tài liệu (Upload document), Chỉnh sửa tùy chọn in (Configure document), In tài liệu (Print document), và Cài đặt hệ thống (System config) giúp kiểm tra định dạng tệp và các tùy chọn in có được cho phép không. Mỗi thành phần này đóng vai trò trong việc xử lý và quản lý tài liệu trước khi in.
- Database: được liên kết với SSPS để lưu trữ và truy xuất thông tin cần thiết trong quá trình đăng tải, chỉnh sửa và in tài liệu như là số trang còn trong tài khoản người dùng và lưu tệp, lịch sử in ấn của người dùng.