**Đề bài:** Ngân hàng ABC hiện có 1 danh sách gồm 10k tài khoản. Hãy xây dựng 1 trang web portal đơn giản để quản lý danh sách tài khoản ngân hàng đó. Mô hình triển khai hệ thống như sau:

Flask

OpenAPI v2 (aka Swageer)

**REST API**

MongoDB

Angular

**Single Page Application**

Hình 1: Mô hình triển khai hệ thống

**Yêu cầu:**

* ***Database***: dữ liệu được lưu trên database sử dụng MongoDB. Có thể sử dụng file account.json đính kèm để làm dữ liệu mẫu hoặc tự xây dựng dữ liệu chứa 10k bank account cho riêng mình.
* ***Rest API***: xây dựng Restful API sử dụng Python Flask đáp ứng các tính năng mà Portal cần, thiết kế API sử dụng Open API v2 (aka Swagger) là một điểm cộng.
* ***Single Page Application (SPA)***: xây dựng Portal quản lý danh sách tài khoản ngân hàng trên sử dụng Angular 7 (stable) với các chức năng: Tìm kiếm, Xem chi tiết, Thêm, Sửa, Xóa tài khoản.

**Chú ý:**

* Portal và API chạy trên 2 ứng dụng khác nhau, không sử dụng chung 1 ứng dụng chạy cả frontend và backend.
* Các chức năng trên cần được phân quyền theo user đăng nhập cả trên Web Portal (SPA) và Web Service (Rest API).
  + Tạo 1 trang đăng nhập đơn giản, tài khoản đăng nhập lưu trong DB với 2 role là **normal** và **admin**. Role **normal** thì được quyền tìm kiếm và xem chi tiết một bank account. Role **admin** thì có thêm quyền thêm, sửa, xóa.
  + User với role nào thì chỉ xem được các chức năng tương ứng với role đó trên portal và chỉ được gọi vào các API tương ứng. Ví dụ user với role normal thì không được phép gọi vào API sửa bank account.
* Dữ liệu bank account tìm kiếm được hiển thị theo dạng bảng, có phân trang.
* Có thể tìm kiếm được tất cả các trường trong bảng bank account.
* Sử dụng JWT

**Các tài liệu cần gửi:**

* Tài liệu mô tả và thiết kế hệ thống.
* Source code SPA, REST API.
* Dữ liệu mẫu bank account được export từ Mongo DB.
* Hướng dẫn sử dụng (bao gồm ảnh chụp màn hình các tính năng đã yêu cầu ở trên).