

**MÔN HỌC: ĐỒ ÁN CNPM**

# **ARCHITECTURE**

**HỆ THỐNG KÝ KẾT VĂN BẢN ONLINE - SIGNATEXT**



**NHÓM 07**

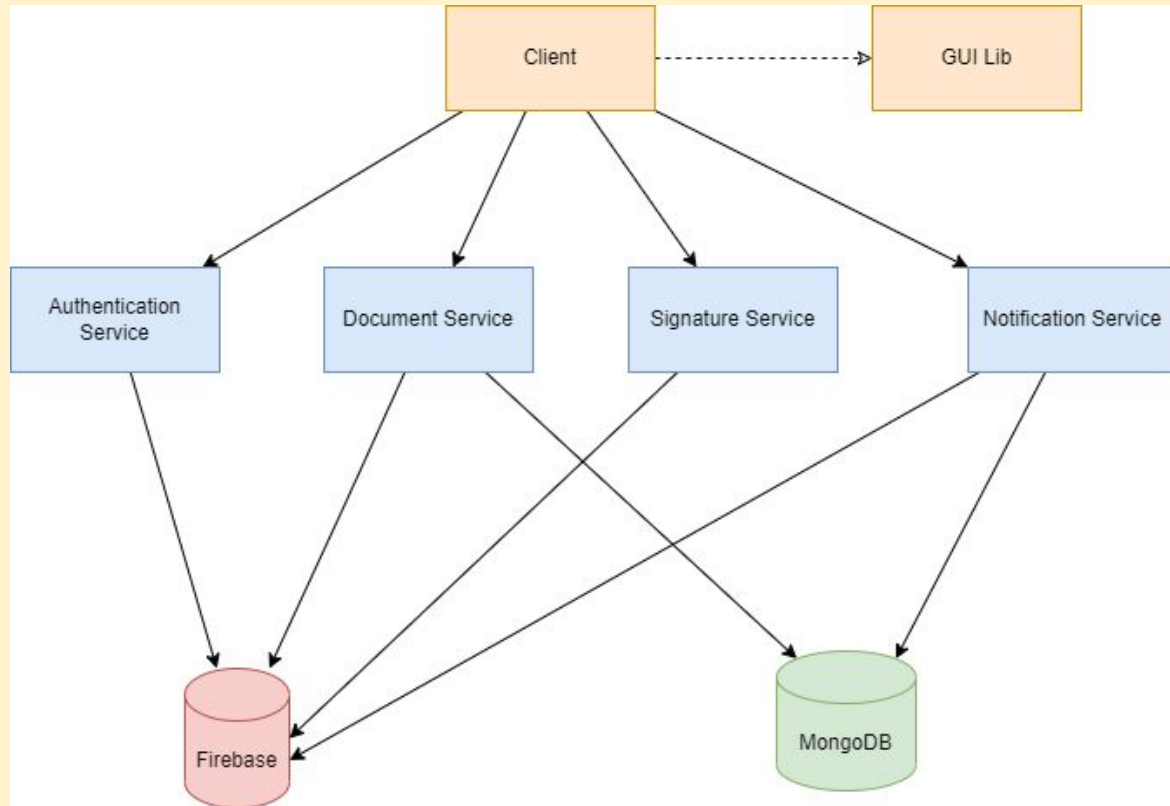
## **THÔNG TIN THÀNH VIÊN:**

1. Nguyễn Sĩ Nhân - 1553021.
2. Nguyễn Phúc Tấn - 18127205.
3. Nguyễn Ngọc Thanh Tâm - 19127268.
4. Đặng Duy Khang - 19127431.
5. Nguyễn Vũ Duy Khương- 19127455.
6. Lê Nhật Minh - 19127470.
7. Phạm Nguyễn Tường Vy - 19127636.

# **NỘI DUNG CHÍNH**

- 01. LOGICAL VIEW:**
- 02. DEPLOYMENT VIEW:**
- 03. PROCESS VIEW:**
- 04. DEVELOPMENT VIEW:**

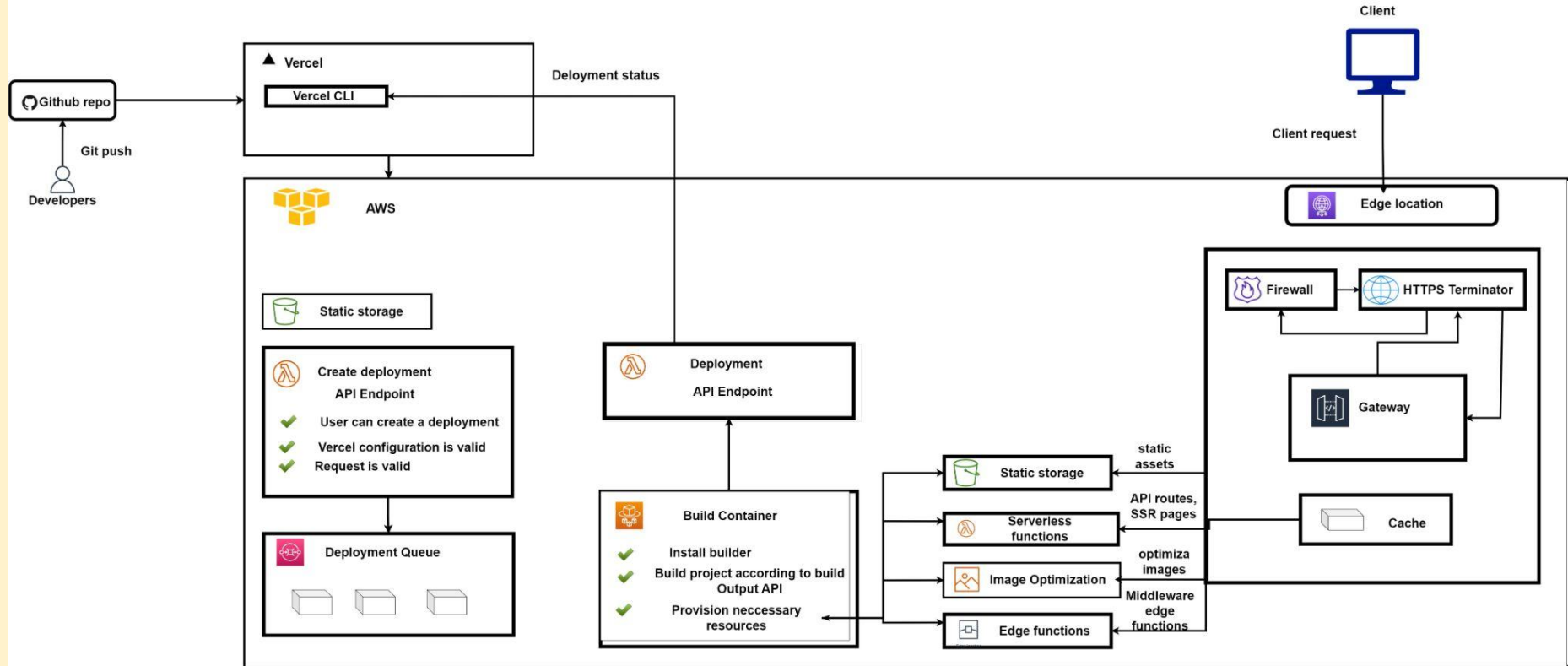
# LOGICAL VIEW



# LOGICAL VIEW

- Gồm 3 phần chính: Client → Service → DB
- Trong đó:
  - Client: có sử dụng GUI Lib (Material UI) để xây dựng giao diện
  - Service:
    - Authentication Service: kết nối với Firebase để quản lý tài khoản và đăng nhập, đăng ký
    - Document Service: quản lý tài liệu cần ký được lưu trữ trên Firebase và MongoDB
    - Signature Service: quản lý chữ ký điện tử được lưu trữ trên Firebase
    - Notification Service: quản lý thông báo được lưu trữ trên Firebase và MongoDB
  - DB: sử dụng Firebase và MongoDB

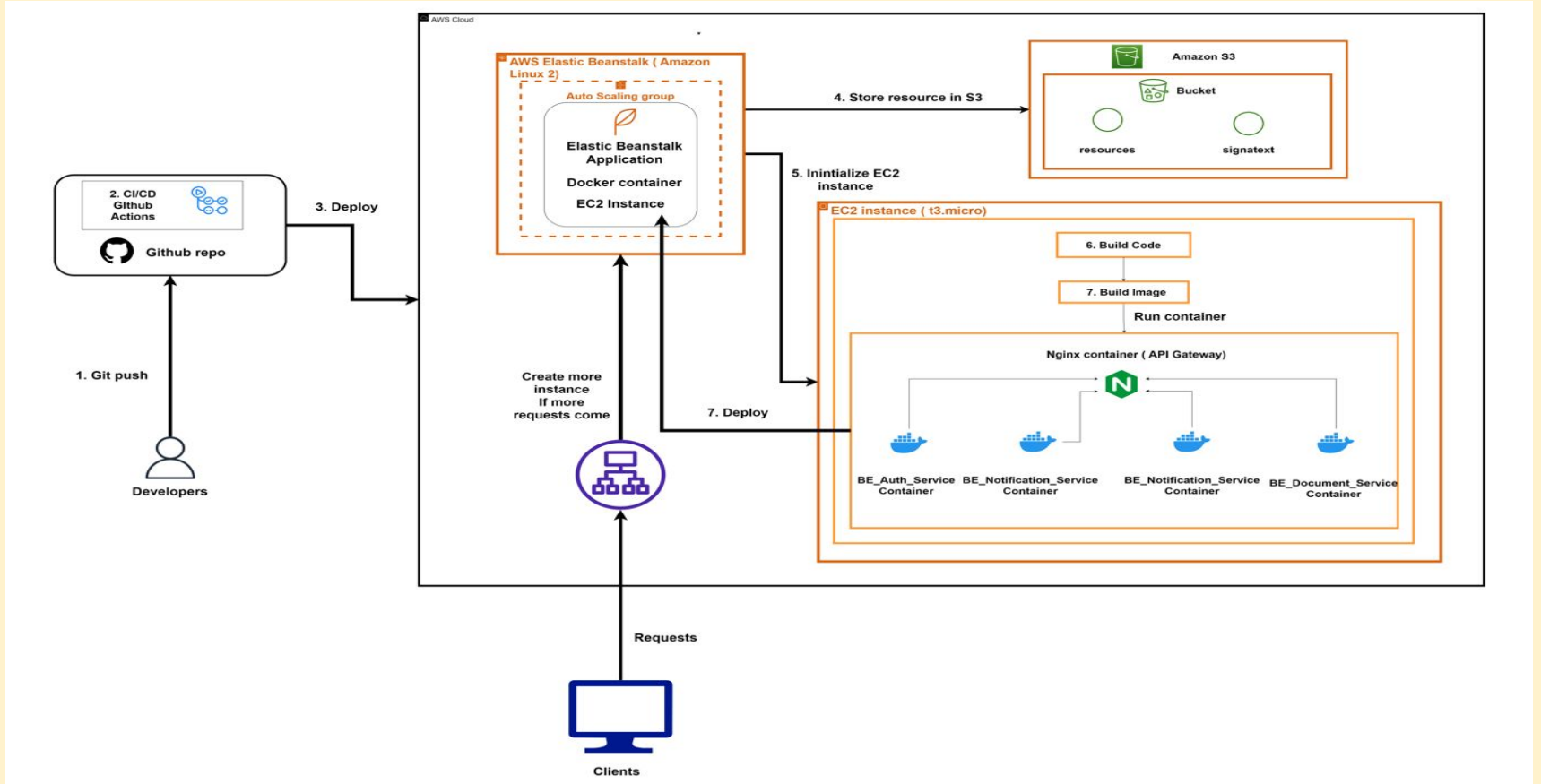
# DEPLOYMENT VIEW - FRONTEND



# DEPLOYMENT VIEW - FRONTEND

- Gồm 3 phần chính: Github Repo → Vercel → AWS
- Trong đó:
  - Github Repo: Nơi chứa project cần deploy
  - Vercel:
    - Nơi kết nối với Github Repo, chịu trách nhiệm deploy project lên môi trường internet.
    - Tuy nhiên đằng sau Vercel chính là AWS chịu trách nhiệm chạy và khởi tạo các kết nối cũng như là nơi deploy ứng dụng.
    - Người dùng sẽ không thấy tình trạng deploy của AWS mà sẽ thông qua Vercel CLI. Vercel CLI đóng vai trò là một lớp trừu tượng, thể hiện tình trạng deploy của ứng dụng và người dùng tương tác với Vercel CLI. Vercel sẽ gọi đến AWS để xử lý.
  - AWS: sử dụng S3, Docker container cùng với các functions để chạy ứng dụng trên môi trường internet. Cho phép người dùng truy cập theo vùng thông qua Edge Location.

# DEPLOYMENT VIEW - BACKEND

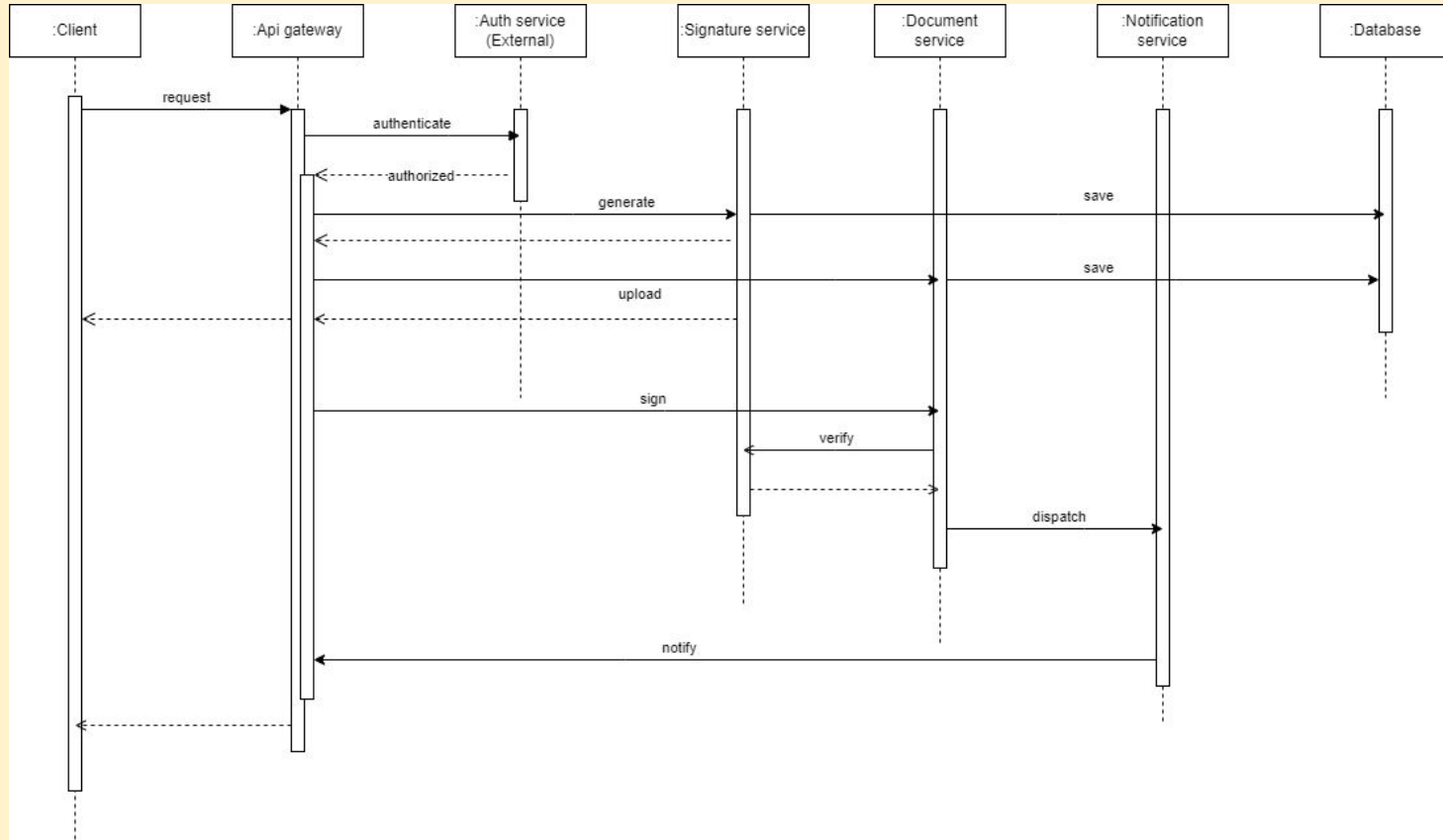




# DEPLOYMENT VIEW - BACKEND

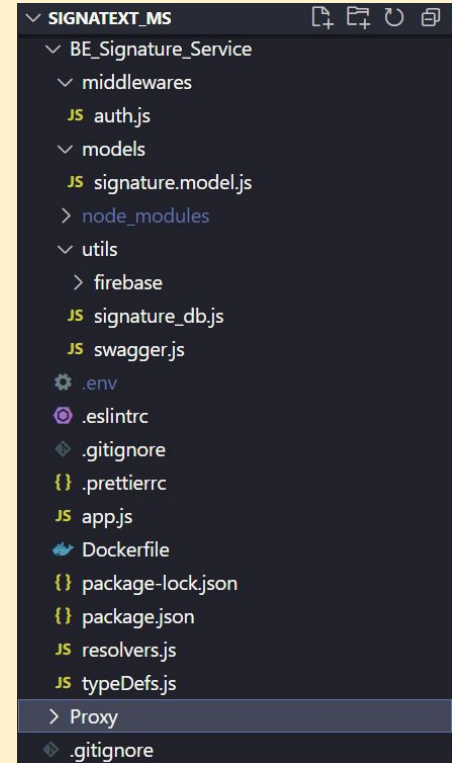
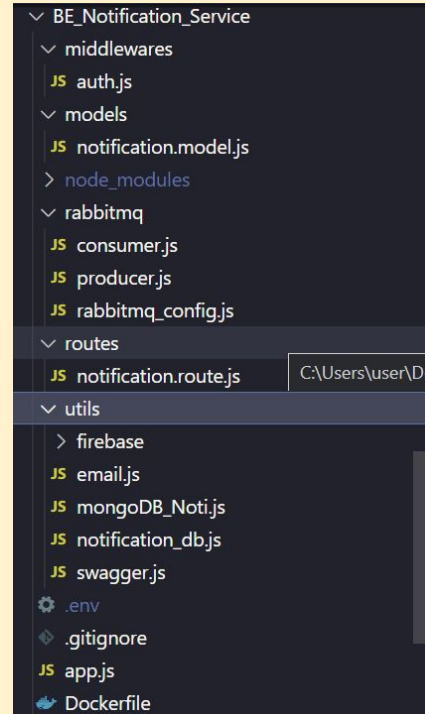
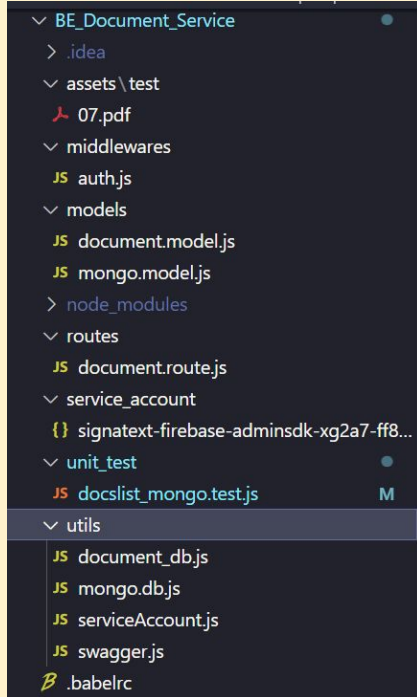
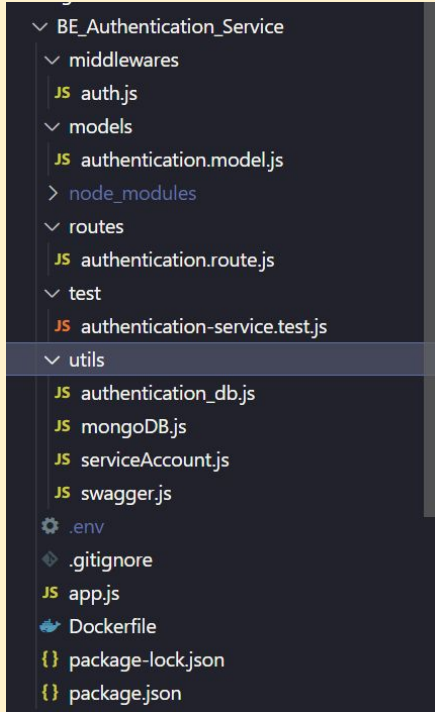
- Gồm 2 phần chính: Github Actions → AWS Elastic Beanstalk (EB)
- Trong đó:
  - Github Actions: thực hiện CI/CD, kết nối với AWS Elastic Beanstalk và deploy ứng dụng lên AWS.
  - AWS Elastic Beanstalk: môi trường chạy ứng dụng trên Cloud, bao gồm các thành phần quan trọng sau:
    - S3: được khởi tạo ngay khi code được deploy lên AWS, EB sẽ tạo S3 thông qua các quyền được cấp và S3 sẽ là nơi lưu giữ những mã nguồn được deploy từ Github.
    - EC2 Instance: được tạo sau S3, là một máy chủ ảo chạy ứng dụng thông qua Docker, nhận file từ S3, thực hiện các câu lệnh build image, chạy container và sẽ là nơi ứng dụng được chạy.
    - Auto Scaling: tính năng tự động mở rộng thêm EC2 Instance để có thể xử lý nhiều lượt truy cập đồng thời khi có quá nhiều truy cập đến một EC2 Instance hoặc quá tải CPU của một máy chủ.
    - Application Load Balancer: thực hiện việc cân bằng tải của EB. Khi có lưu lượng truy cập vượt quá định mức, Application Load Balancer chuyển hướng các request đến các instance phù hợp được tạo thêm nhờ Auto Scaling. Đảm bảo ứng dụng sẽ luôn chạy ổn định và không bị sập.

# PROCESS VIEW



# DEVELOPMENT VIEW

## Project BackEnd



# DEVELOPMENT VIEW

## Project FrontEnd

