

# Bài 1 – Tổng quan

Web full stack

Frontend: Reactjs

Backend: Springboot

Database: MySQL, Mongo

## 1. Kiến trúc web

\* truyền thống:

-Frontend + Backend+ database gộp chung

-HTML render trực tiếp từ server

ví dụ: JSP, PHP thuần, ASP.NET webforms...

Nhược điểm: Chậm + khó mở rộng + khó bảo trì,...

\* Client-server hiện đại

Client (Browser/Mobile App) => http (JSON) =>

RESTful API (Backend) => Database

- Frontend, backend tách biệt

-Giao tiếp qua RESTful API

-Dữ liệu trao đổi ở dạng JSON

=====

Kiến trúc sử dụng trong môn SBA301

\* **Frontend: React JS**

-Xây dựng SPA (Single Page Application)

-Quản lý UI theo component

-Giao tiếp với server thông qua: fetch/axios

+**Vai trò của ReactJS:**

-Hiển thị giao diện

-Gửi request http (GET, POST, PUT, DELETE)

-Nhận và xử lý JSON từ API

### **\*Backend: Spring Boot**

-Xây dựng API

-Tích hợp JPA, Validation, Security,...

-Tự động cấu hình

#### **+Vai trò:**

-Xử lý nghiệp vụ

-Quản lý DB

-Xác thực và phân quyền

-Trả JSON về cho React JS

=====

REST???

(Representaion State Transfer)

sử dụng trong webservice

Nguyên tắc:

Stateless

http Method

client-server

----

GET => /api/users => lấy danh sách

GET => /api/users/1 => lấy chi tiết

POST => insert => /api/user => tạo mới

PUT => update => /api/user/1 => cập nhật

DELETE => delete => /api/users/1 => xóa

=====

Http status code:

404: not found

500: Internal Server Error

403 Forbidden

400 Bad Request

401 Unauthorized

200 OK

201 Created

=====

Lệnh tạo dự án React JS:

2 công cụ:

create-react-app

vite

=====

`npx create-react-app my-web //tao`

`cd my-web //di chuyen vao thu muc web`

`npm start //chay`

=====