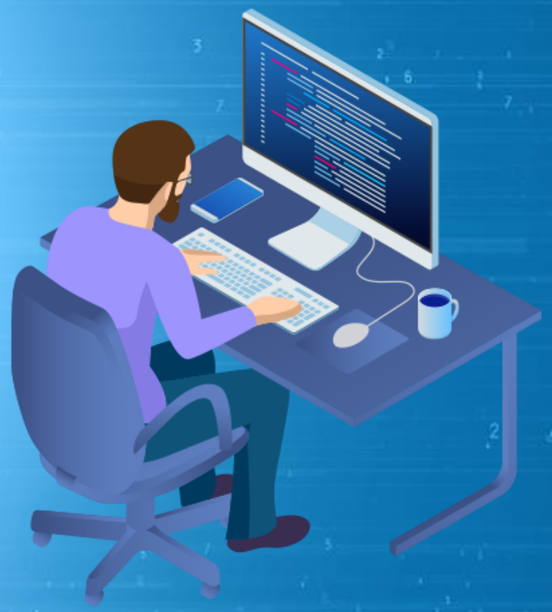
TECHNOLOGY



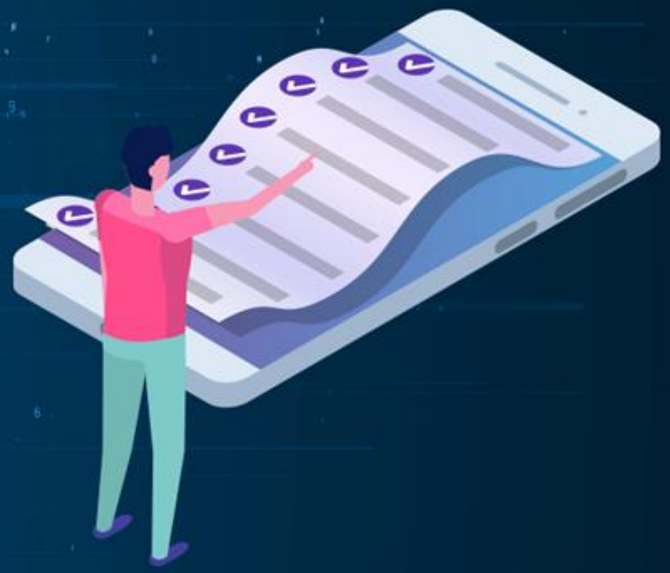
Caltech Coding Bootcamp

Capstone Project

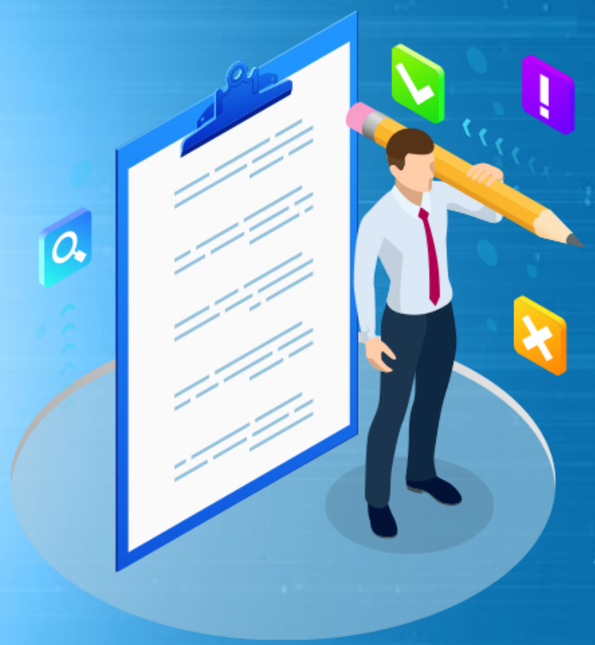
Domain: Healthcare

Objectives

To create a dynamic and responsive Java e-healthcare web application for ordering medicines of different categories



Problem Statement and Motivation



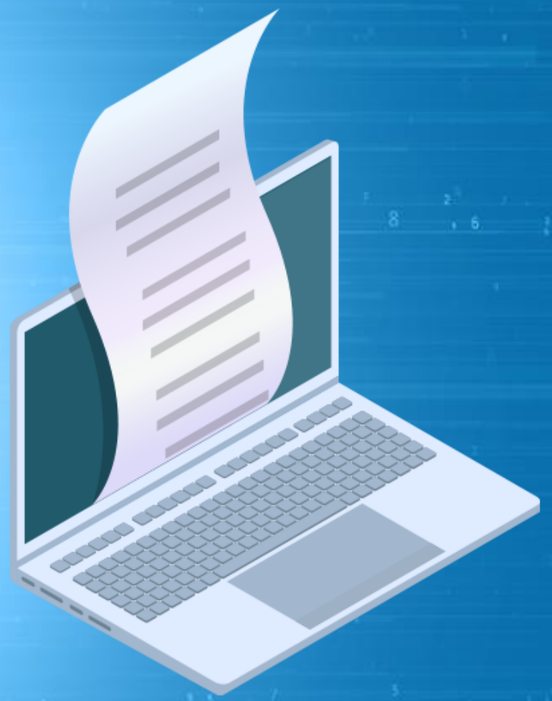
Problem Statement:

In this project, you should be able to develop a front-end web app using Angular and a Java backend using Spring Boot, configure the applications using Docker containers, deploy the project on AWS using CI CD Pipeline

Real-World Scenario:

Medicare is a company that supplies medicines and a couple of other healthcare essentials at an affordable price. They found out that online ordering of medicines with companies, such as 100mg and mfine are gaining more profits by eliminating middlemen from the equation. As a result, the team decided to hire a Full Stack developer to develop a healthcare web application with a rich and user-friendly interface.

Industry Relevance



The following are the DevOps tools and their corresponding AWS services used in this project. These skills are widely used in the industry:

- **Angular** - Platform for building mobile and desktop web applications
- **Java and SpringBoot** - Leading technology to develop backend
- **Docker** - Deliver Software as Containers
- **AWS** - Host and Deploy your Apps in World's leading Cloud Platform
- **Jenkins** - Use Jenkins to build CI CD Pipelines

Tasks

The following tasks outline the CI/CD pipeline creation process:

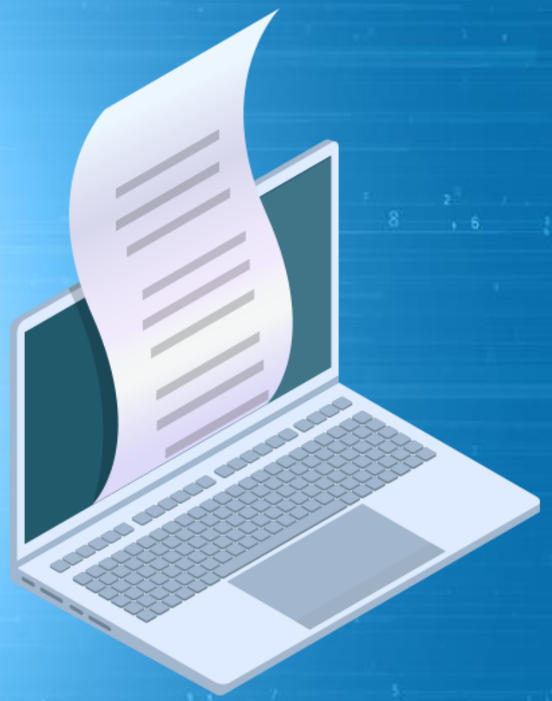
1. Develop the Front End with Angular for the Admin to add Medicines from the Interface with authentication for the Admin User.
2. Develop an End User Web Application listing the medicines using Angular.
3. Define the structure of a database and create necessary tables using SQL in MySQL Database or MongoDB as per your preference.
4. Develop a Java Backend using Spring Boot containing various microservices.

Tasks

The following tasks outline the CI/CD pipeline creation process:

5. Perform the Front-End and Back-End communication using HTTP Client
6. Define Jenkinsfile for both Angular and Java Projects for the automated builds.
7. Define Dockerfile for both Angular and Java Projects to develop images and run them as containers.
8. Develop a CI CD Pipeline in Jenkins for both Angular and Java Backend Projects.
9. Using AWS Launch EC2 Instances and configure other required ServicesDeploy the Projects on EC2 Instance

Project Reference



- **Task 1, 2:** Angular Components, Routing, Services and AuthGuard, Forms
- **Task 3:** SQL CRUD Commands, Primary and Foreign Key Relationship
- **Task 4:** Spring Boot Web Dependency, RestController, RequestMapping, Post and Get Requests
- **Task 5:** Angular HTTP Client Library, HTTP Request Response, JSON
- **Task 6:** Jenkinsfile stages and step declarations
- **Task 7:** Dockerfile and commands to assemble an image
- **Task 8:** Jenkins Pipeline Project Creation with git SCM
- **Task 9:** AWS, EC2, SSH/CloudShell Connection, Tool Configuration

TECHNOLOGY

Thank You

simplilearn