Required Software:-

node v20.18.0

npm 11.2.0

axios

java 17

maven 3.9.9

Docker version 28.0.1,

Assuming all the required things in place to run frontend go to inside bitcoin directory for local run Frontend app:-

npm install, npm install axios, npm install -g serve, serve -s build

For Backend local run

mvn clean install

mvn spring-boot:run

**NFR**

• Demonstrate SOLID, 12 Factor and HATEOAS principles, Design Patterns in the design and implementation

**>> Demonstrated possible implementation of the above guidelines.**

• Demonstrate Performance, Optimization & Security aspects

**>>Demonstrated performance and optimization of the functional flow and output**

• Demonstrate Production readiness of the code

**>> Frontend and Backend code is containerized and production ready to deploy through deployment script Azure-deployment.yml**

• Demonstrate TDD & BDD & Quality aspects

**>> TDD and BDD implemented for the written code**

Documentation

Summary of Design Patterns Used

|  |  |
| --- | --- |
| * **Service Pattern** | Business logic encapsulation (@Service) |

|  |  |
| --- | --- |
| * **Dependency Injection (DI)** | Constructor-based dependency injection |

|  |  |
| --- | --- |
| * **Repository Pattern** | PriceRepository, InMemoryPriceRepository |

|  |  |
| --- | --- |
| * **HATEOAS** | EntityModel with links |

|  |  |
| --- | --- |
| * **Strategy Pattern** | CurrencyConverter (for exchange rate logic) |

|  |  |
| --- | --- |
| * **Singleton Pattern** | Spring-managed singleton (@Service annotation) |
| **Flow Diagram** |  |
|  |  |
|  | C:\Users\USER\Downloads\BitcoinDiagram.drawio.png |

Build & Deploy

Frontend and Backend application are containerized; all artefact are available into the Repository.

**Frontend**

cd history

mvn clean install

docker build -t history:v5 .

docker tag history:v5 terraformonkarsharma9876/history:v5

docker push terraformonkarsharma9876/history:v5

docker run -p 8080:8080 -t terraformonkarsharma9876/history/history:v5

**Backend**

cd bitcoin

docker build -t bitcoin:v5 .

docker tag bitcoin:v5 terraformonkarsharma9876/bitcoin:v5

docker push terraformonkarsharma9876/bitcoin:v5

docker run -p 3000:80 -t terraformonkarsharma9876/bitcoin:v5

Docker Repo : [Docker Hub](https://hub.docker.com/repositories/terraformonkarsharma9876)

We can test the application after running above images as container

<http://localhost:3000>

We can access swagger documentation: -

<http://localhost:8080/swagger-ui/index.html>

<http://localhost:8080/v3/api-docs>

I have preapared the **Azure-Deployment.yml** this script is written to use Azure DevOps and GCP GKE to deploy the service into the created cluster and generate the ssl certificate by given domain name and IP Address.

Terraform deployment script located inside **history/deploy**

We have fill the variable written into the Azure-Deployment.yml to deploy it into the cloud.