# 포팅 메뉴얼

# 1) 환경 설정

## Backend

- 주요 버전
    - Springboot : 3.3.2
    - jdk : 21.0.4
    - mysql : 8.0.36
    - redis
    - AWS S3
- Dependencies
    - Spring Data JPA
    - Spring Security
    - Spring Batch
    - Spring Cloud
    - Fast API

## Frontend

- 주요 버전
    - npm 10.8.1
    - vite + react + typescript
    - tailwindcss
    - pwa
- Dependencies
    - tailwindcss
    - react query
    - axios
    - zustand

## Web Server

- nginx 1.18.0

## WAS

- tomcat 2.1.1

## Infra

- docker
- jenkins 2.424
- kafka 3.6.0

## IDE 버전

- IntelliJ IDEA 2024.1.4
- VS Code Version: 1.92.2

---

# 2) 빌드

## Frontend

- 빌드 순서

```
1. npm install
2. .env 파일 주입 (루트 경로에)
3. npm build
4. sudo cp -r dist /var/www/html/dist
5. sudo service nginx restart
```

- .env 파일

```
VITE_BASE_URL='https://j11a603.p.ssafy.io/api/'
VITE_BASE_URL='http://localhost:8080/api/'
VITE_OPENAI_API_KEY=sk-proj-qR2Jy5aoRtI-d3glV5N-x1uxbFHViWkFlZY5rlx73fvx_oUzXa5SW3s7e
VITE_PUBLIC_KEY='-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAoXexviuBKeOURtKcvHMG
WzRx8IVshA9y3ToeD2mhGnwlrZgKazhVaNgMZWEVaKu6E9LFaJqugf15cLIz2JY2
FhySQz8HJJG7uO3mvAA5Oodnf90HE7j2JfVG8ERnMQUKxWR/Njl7NNGDMaM4NVnC
VKEtMCqGmzWjtB2dm9rkd990BoK5yM/YGj9q1JvDtoNRoWMia6ZbfeJsVBSsIh0k
9i4QD2FWxo27NwfUqW8a+uQ+DQq9Br3pKi+TO8LRK2c1zgpk18d0HPnwEOWub2T2
oWy/l2d1V4NfThxa6txcI13cR2Ob5RE3Z5aKJMZedZakUW5ygFlGVeH8wZIJU1I3
0wIDAQAB
-----END PUBLIC KEY-----'
```

## Backend

▼ 유저 / 뱅크 / 퀘스트

```
1. jenkinsfile, docker-compose.yml (mysql, blue, green) 추가
2. Dockerfile 추가
```

▼ application.yml 파일

```
server:
  port: 8080

spring:
  main:
    web-application-type: reactive
  application:
    name: gateway
  cloud:
    gateway:
      default-filters:
        - DedupeResponseHeader=Access-Control-Allow-Origin Access-Control-Allow-Crede
      globalcors:
        cors-configurations:
          '[/**]':
            allowedOrigins:
              - http://localhost:5173
              - http://localhost:8080
              - http://j11a603.p.ssafy.io
              - https://j11a603.p.ssafy.io
              - http://j11a603.p.ssafy.io:8080
              - https://j11a603.p.ssafy.io:8080
```

```yaml
                allow-credentials: true
                allowedHeaders: '*'
                exposedHeaders:
                  - x-requested-with
                  - Authorization
                  - content-type
                  - credential
                  - X-AUTH-TOKEN
                  - X-CSRF-TOKEN
                allowedMethods:
                  - PUT
                  - GET
                  - POST
                  - PATCH
                  - DELETE
                  - OPTIONS
        discovery:
          locator:
            enabled: true
        routes:
          - id: user
#            uri: lb://user
            uri: http://localhost:8081/
            predicates:
              - Path=/api/user/**
            filters:
              - RewritePath=/api/user/(?<segment>.*), /${segment}
          - id: quest
            uri: lb://quest
            predicates:
              - Path=/api/quest/**
            filters:
              - RewritePath=/api/quest/(?<segment>.*), /${segment}
          - id: bank
            uri: lb://BANK
            predicates:
              - Path=/api/bank/**
            filters:
              - RewritePath=/api/bank/(?<segment>.*), /${segment}

springdoc:
  api-docs:
    enabled: true
  swagger-ui:
    use-root-path: true
    enabled: true
    urls:
      - url: /api/user/v3/api-docs
        name: user-service
      - url: /api/quest/v3/api-docs
        name: quest-service
      - url: http://j11a603.p.ssafy.io:8080/api/bank/v3/api-docs
        name: bank-service

eureka:
```

```yaml
  client:
    service-url:
      defaultZone: http://j11a603.p.ssafy.io:8761/eureka/
    fetch-registry: true
    register-with-eureka: true
  instance:
    prefer-ip-address: true

management:
  endpoint:
    gateway:
      enabled: true  # Gateway 엔드포인트 활성화
  endpoints:
    web:
      exposure:
        include: gateway, health, info  # 필요한 엔드포인트 노출

jwt:
  secret: alkjdshflekwjthaklsdjfhbzcvnaklsjghwetiuoqalhkdsfjdjsfhkaebasdlkjfhwejklrha

security:
  rsa:
    private-key: |
      -----BEGIN PRIVATE KEY-----
      MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQChd7G+K4Ep45RG
      0py8cwZbNHHwhWyED3LdOh4PaaEafCWtmAprOFVo2AxlYRVoq7oT0sVomq6B/Xlw
      sjPYljYWHJJDPwckkbu47ea8ADk6h2d/3QcTuPYl9UbwRGcxBQrFZH82OXs00YMx
      ozg1WcJUoS0wKoabNaO0HZ2b2uR333QGgrnIz9gaP2rUm8O2g1GhYyJrplt94mxU
      FKwiHST2LhAPYVbGjbs3B9Spbxr65D4NCr0GvekqL5M7wtErZzXOCmTXx3Qc+fAQ
      5a5vZPahbL+XZ3VXg19OHFrq3FwjXdxHY5vlETdnlookxl51lqRRbnKAWUZV4fzB
      kglTUjfTAgMBAAECggEAB/1vSzTF6He9f5+uZ9jdSKgYt0ky7HoDOhVFJYgZk8Du
      8F8krNeVymFeg6nBMHPG+nIkmJ+G4SHU35HB5QbR2FhDmJZG2TVo76a+EzGDo3TN
      11LtpbkyXQB9KSiJXrCcLtZhQSKpCR0Q++TSxDkT17I+FTMVhBqsZh5NDUhQo+nT
      p8ZmB6pp7VR/PjbEWv5n/En4MkDHuTpaAo3Yo1yaPZJ5ZIXHio2UBRblk1eL4fjf
      nopkz2nEi45dmmxmd/Vq8cm2rm0Kf7GfFz9LjH5UnhbvRZUSpP3WAMxWRiwmPoTD
      9nAXKBaWXp2K8SckvL8kclxrg56nwEgU9mMg0xx4aQKBgQDRNsmg0Q/ilFPtS3jc
      /1rHShfEAnH0RstcOL6JiB6K0ZVMT+dm8bsKqlvKKB0+us1dNu1bHK418fuj5QEp
      1skeay6fG49toGuLNOxRWw+P9CGmf/b3EyNgQXW+qD7MhKpqQQZN5beCLHCAj3hZ
      G0/+ngJ2dUH/tjBgNV29AyD4dQKBgQDFk3z2ElE9TJOV82NFTbbel97J/NKtoNzd
      rAsgsMn7JUAylDQ6e0vlJjcoj+n1ZBrbg8e0qGrMnMdNSxK7lu98QRl3QjNftXZB
      bQixo9rMtlPpIna2JqfpDeMSeJOU72BIzXHwmSTwvDifg0zEPIZ+baZo0YfklhOO
      TTQfnV8mJwKBgAz18+++g9lpjvl4X3hxaVJHSMmfzSdw61A/Q66JfDrlWPZtTMyX
      1RqICz+u1ID6WZaYvIOUd0ZoQ0qX+UZ6Q+NaiHza3DvSYFczZCB2H7SvTxZdfxM3
      YE2qQH0bNyi8X0JN3PIedQvfbOjJ+o6qnn4Lk+N6yCfrRXvwUmpPM0+xAoGBAK0x
      aaQnUHGks+cJkKbKxhgAFzsG1gN/9LyYuLRyZMwzzj/ILLRrKo933XhAF32Vv/lX
      XnOeAujUMd0uhwBMAijwxAsHzBME9eHKW4mksgMrfQjrvhZUcnJYSwoBCpXO9J89
      ALlA9eHVjOk/ZsbKq7ceFXbi3fXiHMhsItnn0OWXAoGBAMcgLuoClI6bff+wulnt
      OpMgIqZLfTEd9O/dNQqg9EIa+TpT/gJbXR4SicqcP01ehun6KhDBppcj9robRNcw
      Oora50AiLXKBYTozOjrRAaedZ6Rp/TRB25CY3rE9C1k+irEO3ArKxgjiS94l9pMw
      vqtdSv++wAv9BVnH53bAAzIG
      -----END PRIVATE KEY-----
```

# 4) 외부 서비스(API)

### 생성형 AI

- OpenAI

### SMS발송

- coolsms

### SSAFY API

- SSAFY 금융망 API

### 클라우드 스토리지

- S3 (AWS)