****

**E204 : Royale**

삼성청년SW아카데미 부울경캠퍼스 7기

자율프로젝트 [7주 2022.10.10 ~ 2022.11.25]

**포팅 매뉴얼**

담당 컨설턴트 : 김성재

조찬홍[팀장], 민경대, 안세영, 이태엽

**목차**

**1. 프로젝트 개요 ----------------------------------------------------------------- 2**

**2. 프로젝트 기술 스택 ------------------------------------------------------------- 2**

**3. 빌드 상세 내용 ---------------------------------------------------------------- 3**

**4. 배포 특이사항 ----------------------------------------------------------------- 4**

**5. 외부 서비스 ------------------------------------------------------------------- 5**

**6. 시연 시나리오 ----------------------------------------------------------------- 6**

**1. 프로젝트 개요**

“League Royale”은 주짓수 대회로, 기존의 프로세스는 모든 데이터를 수기로 관리하며, 이에 따른 다양한 문제점이 발생합니다. 따라서 이번 프로젝트를 통해 기존의 프로세스를 디지털 트랜스포메이션하여 수기로 관리되는 파편화된 데이터들의 연동으로 스태프에게는 업무 효율성을 선수들에게는 편리함을 제공하는 것을 목표로 합니다.

**2. 프로젝트 기술 스택**

1) 이슈관리 : Jira

2) 형상관리 : Gitlab

3) 커뮤니케이션 : Mattermost, Notion

4) 개발 환경

(1) OS : Windows 10, Mac

(2) IDE

① IntelliJ 2022.1.3 (Ultimate Edition)

② Visual Studio Code 1.70.0v

(3) DataBase : MySQL Workbench 8.0CE

(4) Server : AWS EC2 (MobaXterm)

① Ubuntu 20.04 LTS

② Docker 20.10.17v (server/client 동일)

(5) 상세 사용

① BackEnd

- Java (OpenJDK Zulu 8.33.0.1-win64)

- Spring Boot Gradle 6.7

- Node.js v16.16.0

② FrontEnd

- HTML5, CSS, JavaScript(ES6)

- React.js 17.0.2

③ AWS S3

**3. 빌드 상세 내용**

1) 백엔드 빌드 방법

(1) git clone <https://lab.ssafy.com/s07-webmobile1-sub2/S07P12E101.git>

(2) cmd 창에서 클론한 폴더 \backend로 이동합니다.

(3) ./gradlew clean build 명령어를 통해 빌드를 짂행합니다.

(4) build\libs 폴더에서 생성된 jar 파일을 확인 할 수 있습니다.

2) 프론트엔드 빌드 방법

(1) frontend 폴더에서 npm install 명령어를 통해 Node Package Manager를 설치합니다.

(2) npm run build 명령어를 사용하여 프로젝트를 빌드합니다.

(3) build 폴더 내에서 빌드된 파일들을 확인할 수 있습니다.

**4. 배포 특이사항**

1) AWS EC2에 만들어진 백엔드 빌드 파일(jar파일), 프론트 엔드 빌드 파일(build폴더)를 업로드

후 자동배포를 구현합니다.

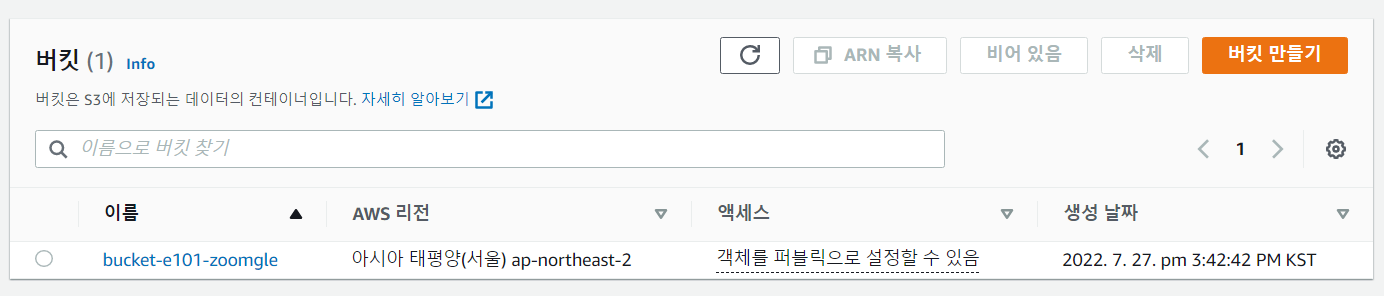
2) NGINX 설정 파일

|  |
| --- |
| version: "3.7"  services:  frontend:  image: frontend-react  build:  context: frontend/  dockerfile: Dockerfile  ports:  - "80:80"  - "443:443"  volumes:  - /home/ubuntu/docker-volume/ssl:/var/www/html  container\_name: "frontend"    db:  image: mysql:8.0.29  environment:  MYSQL\_USER: "${MYSQL\_USER}"  MYSQL\_DATABASE: "${MYSQL\_DATABASE}"  MYSQL\_PASSWORD: "${MYSQL\_PASSWORD}"  MYSQL\_ROOT\_PASSWORD: "${MYSQL\_ROOT\_PASSWORD}"  TZ: Asia/Seoul  volumes:  - "/home/ubuntu/db\_data:/var/lib/mysql"  ports:  - "3306:3306"  container\_name: mysql  backend:  build:  context: ./backend/royale  dockerfile: Dockerfile  image: backend-spring  volumes:  - /home/ubuntu/docker-volume/jenkins/workspace/royale/backend/royale/src/main/resources/img:/img  ports:  - "8080:8080"  container\_name: backend  restart: always  environment:  - TZ=Asia/Seoul  depends\_on:  - db |

**5. 외부 서비스**

1) AWS S3

(1) 버킷 생성



(2) 버킷 엑세스 설정



**6. 시연 시나리오**

1) 최종 발표 (22.11.21)

(1) 인사말

(2) UCC

(3) 서비스 소개

(4) 라이브 시연

(5) 마무리

2) 라이브 시연

(1) 메인 페이지

(2) 리그 페이지

(3) 리그 참가 신청 (선수)

(4) 대회 마감 및 대진표 생성 (스태프)

(5) 대진표 조회

(6) 계체 통과 상태 변경 (스태프)

(7) 스코어 보드 (스태프)

(8) 경기 결과 반영

(9) 경기 현황 조회