



# [사무국 제출] 프로젝트 문서

## 1. Gitlab 소스 클론 이후 빌드 및 배포할 수 있도록 정리한 문서

### 각종 버전 및 명령어 정보

#### (1) 버전 관리 툴 정보

#### (2) 빌드 관련 정보

1) [프론트] : Install & Build 정상 여부 확인

2) [백엔드] : Build 정상 여부 확인

#### (3) 배포 관련 정보

1) [프론트] : Dockerfile, 이미지 빌드, 컨테이너 실행

2) [백엔드] : Dockerfile, 이미지 빌드, 컨테이너 실행

3) [데이터베이스] MySQL(Docker) / Redis

4) [Nginx] 설정 : 포트포워딩, SSL 설정

## 2. 프로젝트에서 사용하는 외부 서비스 정보를 정리한 문서

## 3. DB 덤프 파일 최신본

## 4. 시연 시나리오

# 1. Gitlab 소스 클론 이후 빌드 및 배포할 수 있도록 정리한 문서

## 각종 버전 및 명령어 정보

### (1) 버전 관리 툴 정보

- git 2.30.2
- 외부 서비스
  - Gitlab (SSAFY제공)
  - Webhook

### (2) 빌드 관련 정보

- Open JDK 17
  - Gradle 9.0
- Node.js 14.17.0
  - npm 9.6.7
- Jenkins latest (Docker)

## 1) [프론트] : Install & Build 정상 여부 확인

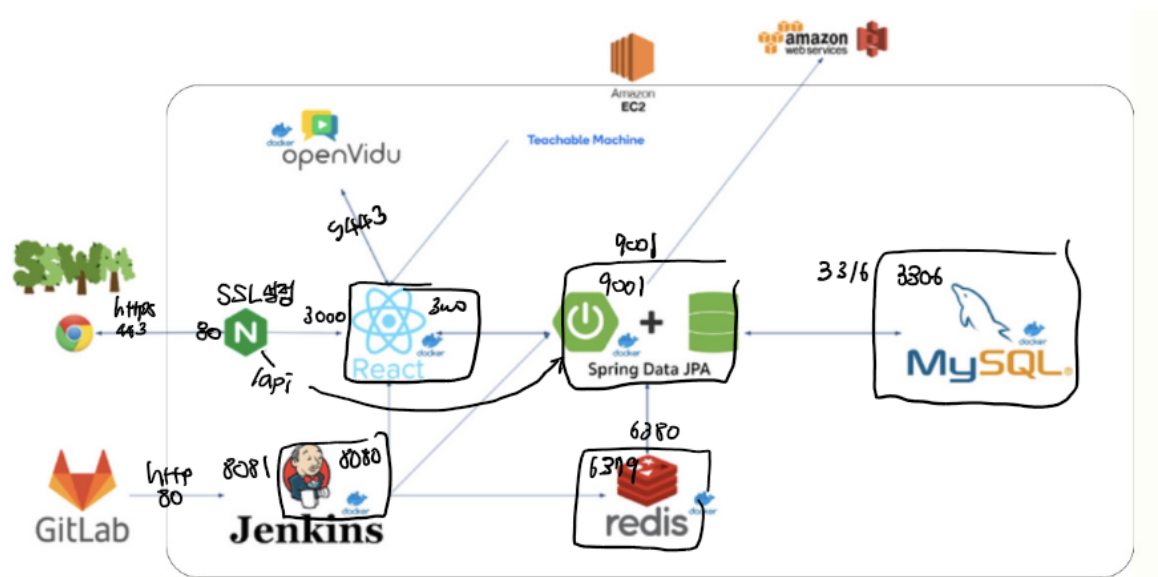
```
cd frontend/sswm-front
pwd
npm i
npm run build
```

## 2) [백엔드] : Build 정상 여부 확인

```
cd backend/sswm
chmod +x gradlew
./gradlew build
```

## (3) 배포 관련 정보

- [공통] Docker 20.10.21
- 외부 서비스
  - AWS EC2 (SSAFY 제공)
  - AWS S3
- [프론트] serve 9.8.1



## 1) [프론트] : Dockerfile, 이미지 빌드, 컨테이너 실행

```
FROM node:18 AS build
WORKDIR /app
COPY package*.json ./
```

```
COPY build/ ./build
RUN npm i -g serve
EXPOSE 3000
ENTRYPOINT ["serve", "-s", "build"]
```

```
sudo rm -rf /home/ubuntu/front-server/build
```

```
cd front-server
docker build -t react-app-image .
docker rm -f sswm-front-container
docker run --name sswm-front-container -d -p 3000:3000 react-app-image
```

## 2) [백엔드] : Dockerfile, 이미지 빌드, 컨테이너 실행

```
FROM openjdk:17
ARG IDLE_PROFILE
ARG JAR_FILE=/home/ubuntu/app-server/*.jar
ENV ENV_IDLE_PROFILE=$IDLE_PROFILE
COPY ./sswm-0.0.1-SNAPSHOT.jar app.jar
RUN echo $ENV_IDLE_PROFILE
ENTRYPOINT ["java", "-Dspring.profiles.active=${ENV_IDLE_PROFILE}", "-jar", "/app.jar"]
```

```
if docker ps -a | grep sswm-springboot-container; then docker rm -f sswm-springboot-container; fi
if docker images -a | grep app-server_application; then docker rmi -f app-server_application:latest; fi
cd app-server
sudo docker-compose up -d
```

```
# Use postgres/example user/password credentials
version: '3.1'

services:
  application:
    build: .
    container_name: sswm-springboot-container
    environment:
      SPRING_DATASOURCE_URL: jdbc:mysql://i9a206.p.ssafy.io:3316/sswm
      SPRING_DATASOURCE_USERNAME: root
      SPRING_DATASOURCE_PASSWORD: ssjmt206
    ports:
      - 9001:9001
    restart: always
```

## 3) [데이터베이스] MySQL(Docker) / Redis

```
version: '3.1'

services:
  database:
    image: mysql:8.0
    container_name: sswm-mysql-container
    restart: always
```

```

environment:
  MYSQL_ROOT_PASSWORD : ssjjmt206
  MYSQL_DATABASE : sswm
  TZ: Asia/Seoul
ports:
  - 3316:3306
redis:
  image: redis:latest
  container_name: sswm-redis-container
  restart: always
  ports:
    - 6380:6379

```

#### 4) [Nginx] 설정 : 포트포워딩, SSL설정

```

server {
    location /{
        proxy_connect_timeout 90;
        proxy_send_timeout 90;
        proxy_read_timeout 90;
        proxy_pass http://localhost:3000;
    }
    location /api {
        proxy_connect_timeout 90;
        proxy_send_timeout 90;
        proxy_read_timeout 90;
        proxy_pass http://localhost:9001/api;
    }
    listen 443 ssl;
    ssl_certificate /etc/letsencrypt/live/i9a206.p.ssafy.io/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/i9a206.p.ssafy.io/privkey.pem;
}
server {
    if ($host = i9a206.p.ssafy.io/) {
        return 301 https://$host$request_uri;
    }
    listen 80;
    server_name i9a206.p.ssafy.io/;
    return 404;
}

```

## 2. 프로젝트에서 사용하는 외부 서비스 정보를 정리한 문서

- 소셜 로그인
  - Google OAuth 2.0
  - Kakao Social Login
- AWS
  - AWS S3
- OpenVidu Server 2.20.0
- teachable machine image 0.8.5

- teachable machine pose 0.8.3

### 3. DB 덤프 파일 최신본

별첨

### 4. 시연 시나리오

시연 순서에 따른 Site 화면별, 실행별 상세 설명

- 회원가입 (구글)
  - 닉네임 -> 시작하기
  - 로그인 (카카오톡)
- 마이페이지
  - 나무 생성
  - 좌측 네비게이션 클릭을 통해 전체적으로 빈 부분 보여줌
- 메인 페이지
  - 태그로 스터디방 검색(앞에 3개)
  - 스터디 검색
  - 스터디룸입장(공개방)
  - 스터디룸입장(비공개방)
- 스터디룸 페이지(회원)
  - 공지사항 확인
  - top3 확인
  - 라이브 입장
- 라이브 페이지
  - 휴식
  - 자동 휴식
  - 마이크 끄기
  - 알림
  - 스트레칭 시작
- 스트레칭 페이지
  - 끝난 후 마이페이지에서 점수 증가 확인

- 방생성 페이지
  - 방 생성
- 스터디룸 설정 페이지(방장)
  - 방장 권한 넘기기
  - 회원 차단
- 마이 페이지
  - 나무 성장 확인
  - 공부 로그 확인
- 회원 정보 수정(선택)
  - 수정 후 저장