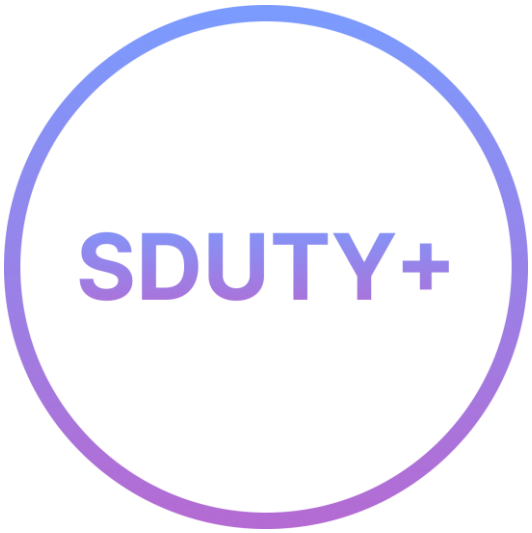


D205 : SDUTY+

삼성SW청년아카데미 구미 캠퍼스 7기

자율 프로젝트

2022/10/11 ~ 2022/11/21

포팅 매뉴얼

담당 컨설턴트 : 박종철

김남희(팀장), 서재형(부팀장), 김정윤, 배시현, 배한용, 편예린

목차

1. 프로젝트 기술 스택

- 형상관리 : Gitlab
- 이슈관리 : Jira atlassian
- 프로젝트관리 : Notion
- 커뮤니케이션 : Mattermost, Webex
- 디자인 : Figma
- OS : Windows 10, 11
- Database : MariaDB 3.0.7
- Language : Kotlin, Java11
- Front-End (Android)
    - Android Studio Dolphin | 2021.3.1
    - Kotlin 1.7.20
    - JDK 11.0.13
    - Gradle 7.5
    - SDK
        - min 21
        - target 33
        - compile 33
    - Retrofit: 2.9.0
- Back-End
    - JDK 11.0.16
    - Spring Boot 2.7.5
    - Gradle 7.5.1
    - Lombok, JPA, Swagger Ui 3.0.0, Spring Security 5.7.3
    - Firebase 8.1.0
- Server
    - AWS EC2
    - ubuntu 20.04 LTS
    - Docker
    - Jenkins
    - CertBot
- IDE
    - HeidiSQL 12.1.0

- **Android Studio Dolphin | 2021.3.1**
- **IntelliJ IDEA | 2022.1.4, 2022.2.3**

2. 빌드 상세 내용

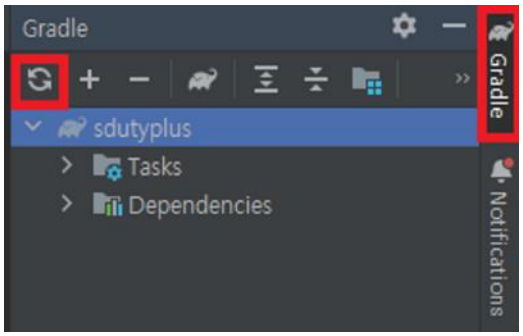
2-1. Back-end build

1. Git Bash Terminal open
2. project repository 로 이동 (Backend/sdutyplus)

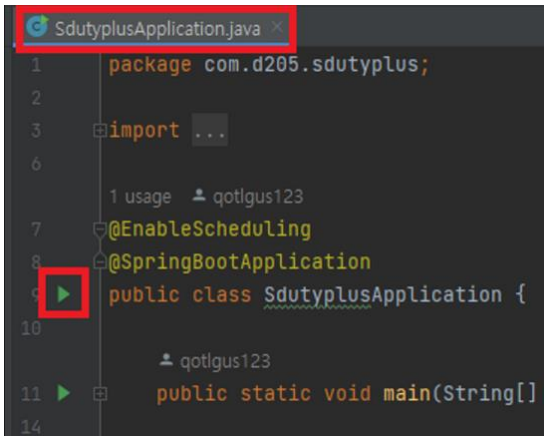
3. ./gradlew bootjar 입력

4. 빌드완료

- IntelliJ Build
  - Gradle 재빌드



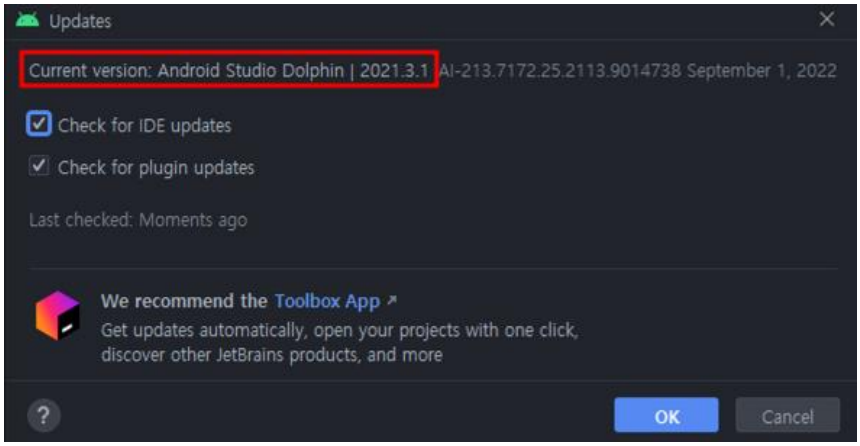
  - SdutyplusApplication.java 빌드



2-2 Front-End Build

1. Android Studio 설치 및 버전 확인 - Dolphin (2021.3.1)

   Help - Check for Updates - 팝업의 오른쪽 아래에 있는 Configure Updates 클릭하면 버전 확인이 가능하다.

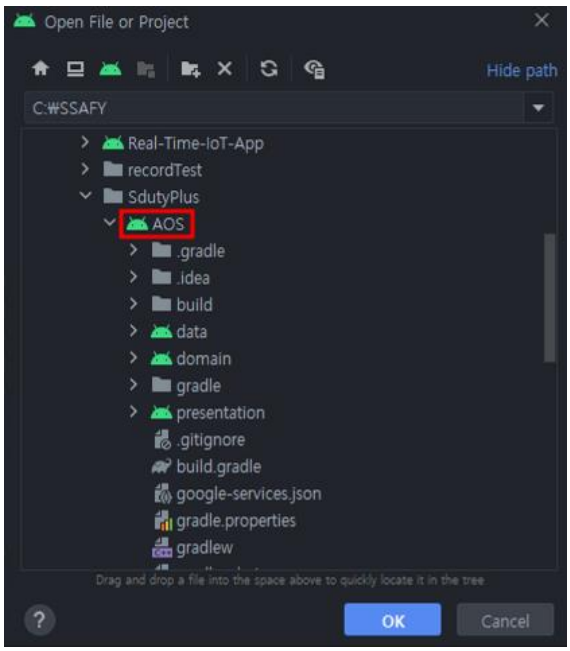


2. Kotlin 버전 확인 (1.7.20 인지 확인)

   File - Settings - Languages & Frameworks - Kotlin 클릭 - Current Kotlin plugin version 확인

3. 프로젝트 열기

   File - Open 클릭 - AOS 폴더(안드로이드 아이콘) 클릭 후 OK 버튼 클릭



4. 프로젝트의 Gradle Version 확인

   File - Project Structure 탭 클릭

   Project 탭에서 Gradle Version 이 맞는지 확인 (Gradle Version : 7.5)

5. Gradle JDK 확인

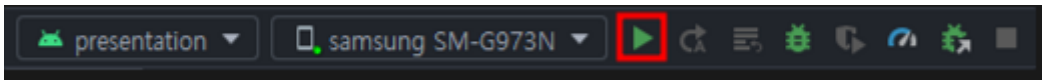
   File - Settings - Build, Execution, Deployment - Build Tools - Gradle 클릭 - Gradle JDK 가 version 11.0.13 으로 되어있는지 확인

6. 휴대폰과 Android Studio 연결을 위해, 휴대폰 개발자 모드를 켠 후 USB 디버깅 허용 시키기

   휴대폰마다 방법이 다르므로, '휴대폰 기종 + 개발자 모드'라고 키워드를 검색하여 개발자 모드를 활성화하고 USB 디버깅을 허용한다.

   예시 링크

7. 아래와 같이 휴대폰 연결을 확인하고, 초록색 Run 버튼 클릭



8. Build 성공 시, 휴대폰에 앱 첫 번째 화면이 나오면서 설치된 것을 확인할 수 있다.

3. 배포 특이사항

3-1 HTTP setting

1. 포트 열기

   sudo ufw allow 22

   sudo ufw enable

2. 사용자에게 Docker 의 sudo 권한 부여하기

   sudo usermod -aG docker ubuntu

3. Nginx 설치
   - 이미지 설치 : docker pull nginx
   - 이미지 확인 : docker images
4. Nginx 실행
   - docker container run —name nginx -d -p 80:80 -p 443;443 -v /etc/letsencrypt:/etc/letsencrypt nginx
5. Dokcer 내부 접속
   - docker exec -t containerID /bin/bash
6. 명령어 설치
   - apt-get update
   - apt-get install vim
7. Certbot 설치
   - apt install python3-certbot-nginx
8. nginx.conf 수정
   - include /etc/nginx/site-available/*;
   - mkdir sites-available;
   - vi d205.kro.kr;

```
server{
    server_name d205.kro.kr;
}
```

9. 인증서 생성
    - certbot --authenticator webroot -w /usr/share/nginx/html --installer nginx -d [도메인주소]
    - 이후 1 번 선택

3-2 Jenkins container setting

1. Docker 설치
    1. 패키지 설치
        - apt update
        - apt install apt-transport-https
        - apt install ca-certificates
        - apt install curl
        - apt install software-properties-common
    2. 도커 설치
        - curl -fsSL <https://download.docker.com/linux/ubuntu/gpg> | sudo apt-key add -
        - add-apt-repository "deb[arch=amd64]<https://download.docker.com/linux/ubuntu> bionic stable"
    3. 최신버전 업데이트 및 설치
        - apt update
        - apt install docker-ce
2. Jenkins 설치

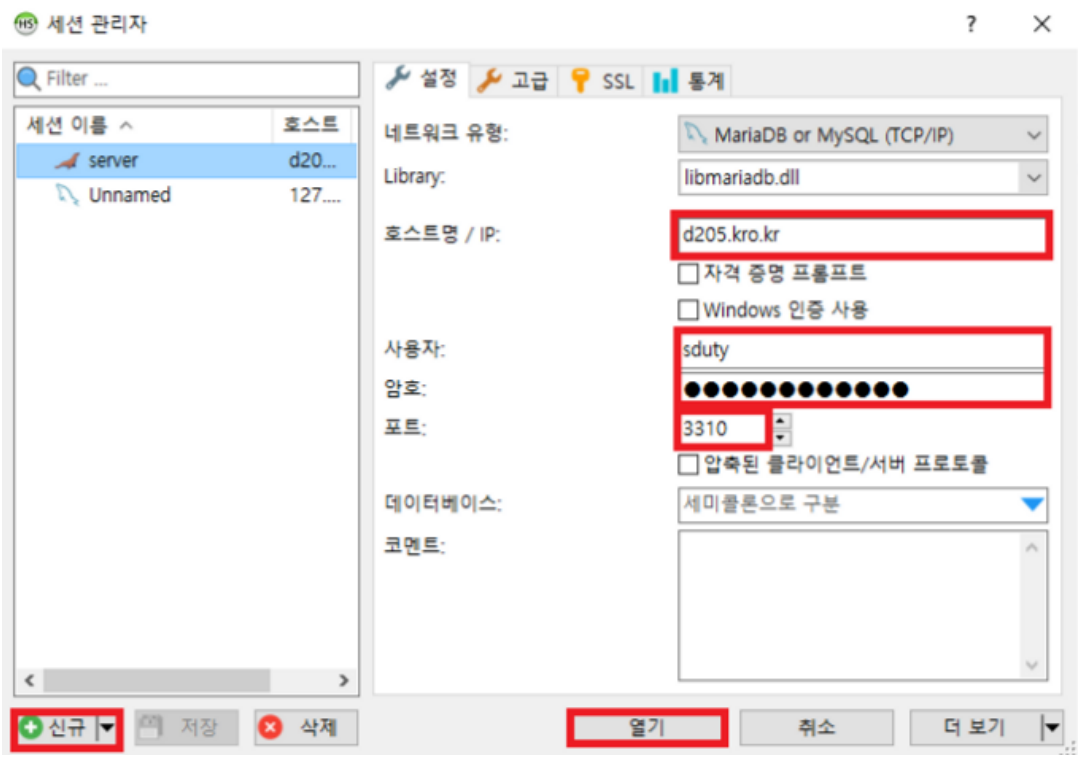
```
sudo docker run -d --name jenkins -u root --privileged \
-p '9090:8080' \
-v '/home/ubuntu/docker-volume/jenkins:/var/jenkins_home' \
-v '/var/run/docker.sock:/var/run/docker.sock' \
-v '/usr/bin/docker:/usr/bin/docker' \
jenkins/jenkins
```

3. GitLab token 생성
    1. GitLab Settings → accessToken → token 생성
4. Jenkins 접속 및 setting
    1. jenkins 비밀번호 / sudo docker logs jenkins
    2. Install suggested plugins 클릭
    3. Jenkins 관리
        - 플러그인 관리

        - GitLab 플러그인 설치
    - 시스템 설정 - GitLab
        - Gitlab host URL : host URL
        - Credentials add 후 API token 에 GitLab token 입력
4. 새로운 Item - Pipeline
    - Pipeline script from SCM
        - SCM = Git
        - URL = git project url
        - Credentials add 후 Username with password 입력
        - Branchesto build : */자동 배포할 브랜치 이름
    - 빌드유발
        - GitLab webhook - 고급
            - Secret Token → GitLab Webhook token 추가
5. Server log 확인 방법
    1. Container ID 확인 : docker ps -a
    2. Log 확인 : docker logs -f [Container Id]

4. DB 계정

- HeidiSQL 실행



- 신규 생성
- 호스트명, 사용자, 암호, 포트 설정
  - Hostname : d205.kro.kr
  - Username : sduty
  - Password : kkbbsp885520
  - Port : 3310
- 열기 클릭

5. Firebase 사용 방법

5-1 Back-End

1. 디펜던시 등록 (build.gradle)
    - implementation group: 'com.google.firebase', name: 'firebase-admin', version: '8.1.0'
2. Firebase 연동
    - Project 생성 및 Firestore 생성
    - 프로젝트 설정 → 서비스계정메뉴 → 새 비공개 키 생성 → json 파일 다운로드
    - json 파일명을 serviceAccountKey.json 으로 변경 후 resources 폴더로 복사
    - **application.properties**

```
app.firebase-configuration-file=./serviceAccountKey.j
son
app.firebase-bucket=sdutyplus.appspot.com
```

5-2 Front-End

https://firebase.google.com/

- 시작하기 버튼 클릭 후 프로젝트 추가하여 프로젝트 생성
- 좌측 메뉴에 Storage 클릭 후 시작하기 클릭
- 프로덕션 모드 선택 후 Cloud Storage 위치를 설정
- 규칙 탭을 눌러 allow read, write: if false; 를 allow read, write: if true;로 바꾼다.
- 안드로이드 아이콘을 눌러 앱을 추가한다.
- 안드로이드 프로젝트의 패키지 명을 입력 후 'google-services.json' 파일을 다운로드 하여 프로젝트 내 'app' 폴더 안에 넣어준다.
- Firebase SDK 를 프로젝트 수준의 build.gradle 파일에 추가한 후 완료한다.
- Android 상단 탭에서 Tools 를 클릭하고 Firebase 를 선택한다.
- 우측 메뉴에서 Cloud Storage for Firebase 를 클릭 한 후 Get started with Cloud Storage [KOTLIN]을 클릭한다.
- Connect your app to Firebase 와 Add Cloud Storage to your app 을 클릭하여 파이어베이스 연동을 완료한다.

6. 외부 서비스

6-1. Kakao Login 설정

https://developers.kakao.com/console/app

- 애플리케이션 추가 후 네이티브 앱 키를 AndroidManifest.xml 에 추가
- 키 해시 발급 후 Android 플랫폼에 package 명과 함께 추가

```
keytool -exportcert -alias androiddebugkey -keystore %USERPROFILE%\.android\debug.keystore
-storepass android -keypass android | openssl sha1 -binary | openssl base64
```

- Release 용 (구글 플레이스토어 등록 시 키 해시 추가해야 함)

```
keytool -exportcert -alias <RELEASE_KEY_ALIAS> -keystore <RELEASE_KEY_PATH> | openssl
sha1 -binary | openssl base64
```

- 팀 관리에서 카카오 로그인을 이용할 이메일 추가

*플레이 스토어 등록 시 Google Play Console 앱 서명의 SHA-1 값을 입력

```
echo <SHA-1> | xxd -r -p | openssl base64
```

6-2. Naver Login 설정

https://developers.naver.com/apps/#/list

- 애플리케이션 등록
- 안드로이드 ApplicationClass NaverIdLoginSDK.initailize 에 ClientID, Client Secret 추가
- 멤버 관리에서 네이버 로그인을 이용할 아이디 추가