

**B210 : Dipping**

삼성SW청년아카데미 대전캠퍼스 7기

공통프로젝트 6주, 2022/07/11 ~ 2022/08/19

**포팅 매뉴얼**

**담당 컨설턴트 : 이승윤**

**이길상(팀장), 김주은, 김형주, 이성재, 박상협, 허유민**

**<프로젝트 기술 스택>**

1. 이슈관리 : Jira
2. 형상관리 : Git
3. 커뮤니케이션 : Mattermost, Notion
4. 개발환경
   1. OS : Window 10
   2. IDE :
      1. Intellij : 2019.03.5, 2022.
      2. Visual Studio Code
      3. Figma
   3. Database : MySql
   4. Server : AWS EC2 (MobaXterm)
      1. Ubuntu 20.04.2 LTS
      2. NGINX 1.18.0
   5. 상세 사용
      1. Backend
         1. Java 11
         2. Spring Boot Gradle 6.8.3
         3. lombok , swagger3
      2. Frontend
         1. React
         2. ChakraUI
         3. TypeScript

**<빌드 상세내용>**

1. 서버 접속
   1. MobaXterm 접속
   2. Session => SSH
      1. RemoteHost : <http://i7b210.p.ssafy.io/>
      2. Advanced SSH settings : User private key에 .pem파일 추가
   3. login as : ubuntu
   4. 서버 접속 완료
2. 백엔드 빌드 방법
   1. cd S07P12B210/back/dipping
   2. chmod +x gradlew or ./gradlew
   3. ./gradlew build -debug
3. 프론트엔드 빌드 방법
   1. cd S07P12B210/front
   2. npm run build
4. 배포 명령어
   1. cd S07P12B210/back/dipping/build/libs
   2. java -jar dipping-0.0.1-SNAPSHOT.jar

**<배포 특이사항>**

1. 백엔드 스프링 application-prod.yml 추가

dipping/src/main/resource에 application-prod.yml 파일 추가한다.

(api키, jwt토큰 키, 계정 데이터 등의 정보를 포함하므로 git ignore 되어있습니다.)

\*\*\* 아래부터는 yml 파일 내용입니다(탭의 위치가 중요하므로 유의) \*\*\*

# 서버 포트번호

server:

port: 3001

# jwt 헤더와 시크릿키

jwt:

header: Authorization

secret-key: 236979CB6F1AD6B6A6184A31E6BE37DB3818CC36871E26235DD67DCFE4041492

# 비밀번호 찾기시 보내는 메일 계정

spring:

mvc:

pathmatch:

matching-strategy: ANT\_PATH\_MATCHER

mail:

host: smtp.naver.com

port: 465

username: /\*네이버 이메일\*/

password: /\*네이버 계정 비밀번호\*/

properties:

mail.smtp.auth: true

mail.smtp.ssl.enable: true

# 데이터베이스 연결

datasource:

driver-class-name: com.mysql.cj.jdbc.Driver

url: jdbc:mysql://[디비 주소]:3306/dipping?serverTimezone=Asia/Seoul&characterEncoding=utf8

username: #디비 유저 아이디

password: #디비 유저 비밀번호

security:

oauth2:

client:

registration:

kakao:

client-id: 6284a45a782edf525bc50afe13191872

redirect-uri: <http://i7b210.p.ssafy.io/login/oauth2/code/kakao>

client-authentication-method: POST

client-secret: 13MKoSLs8rRtKpy5Dc4EEbN7XcB7vIWR

authorization-grant-type: authorization\_code

scope: account\_email

client\_name: kakao

google:

client-id: 103305653148-2b6ib34sgmaee95s80omkjpl8nl83g3q.apps.googleusercontent.com

client-secret: GOCSPX-7\_4ocukFSkeDCM4Dpw-HIdcUbQMb

scope: email

provider:

kakao:

authorization-uri: https://kauth.kakao.com/oauth/authorize

token-uri: https://kauth.kakao.com/oauth/token

user-info-uri: https://kapi.kakao.com/v2/user/me

user-name-attribute: id

redis:

lettuce:

pool:

max-active: -1

max-idle: -1

min-idle: 2

port: 6379

host: 127.0.0.1

jpa:

generate-ddl: true

hibernate:

# format\_sql: true

ddl-auto: update

logging:

root: info

org.springframework.web: debug

org.apache.tomcat: debug

app:

firebase-configuration-file: ./serviceAccountKey.json

firebase-bucket: dipping-f118c.appspot.com

1. Nginx default 파일 수정

server {

listen 80;

location ~ ^/(api|login/oauth2|oauth2/authorization)/ {

proxy\_pass http://localhost:3001;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header Host $http\_host;

}

location / {

root /home/ubuntu/S07P12B210/front/build;

index index.html index.htm;

try\_files $uri /index.html;

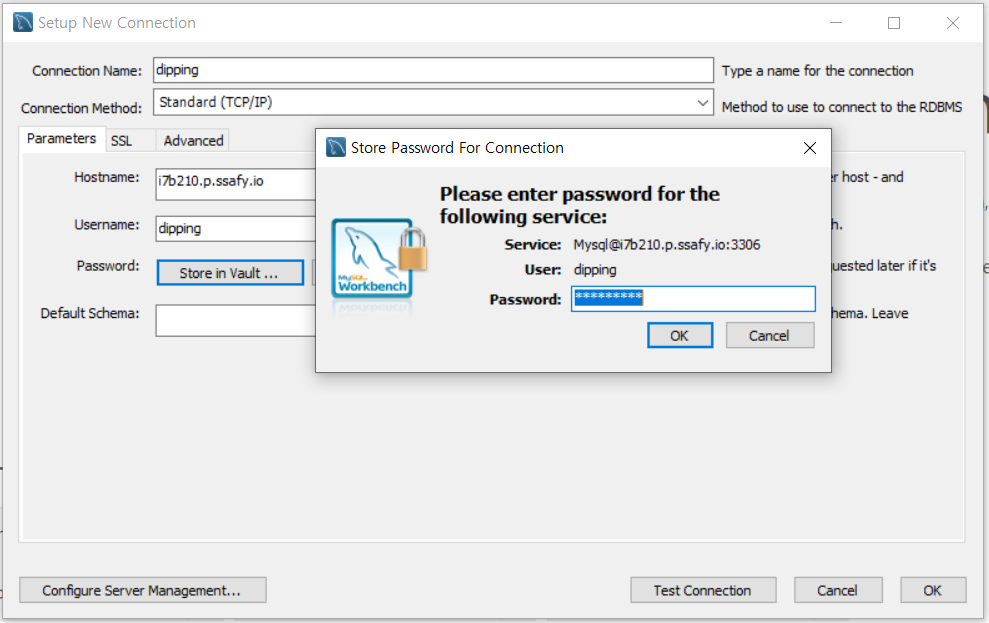
}

level:

**<DB 계정>**

#MySQL 워크벤치 사용방법

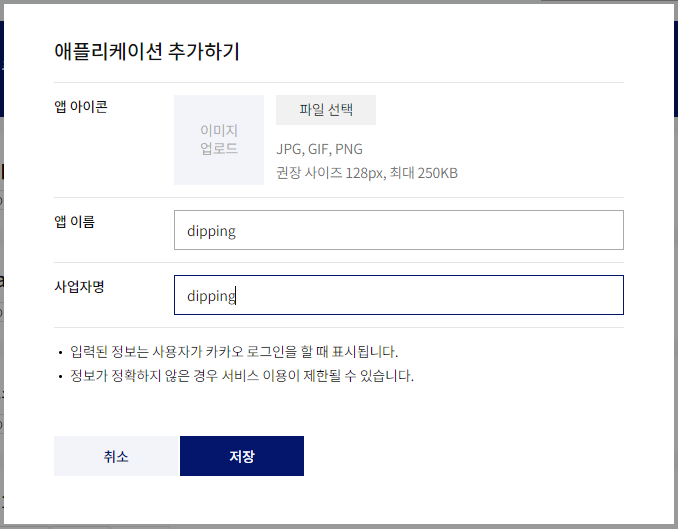
1. MySql workbench 다운로드
2. root가 아닌 새로운 연결 생성



username: dipping, password: elvldthtm

**<외부서비스>**

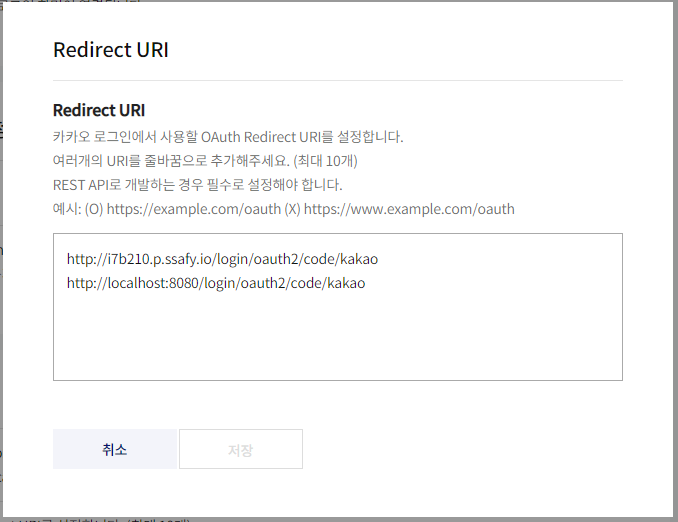
1. 소셜 로그인 : 디핑 서비스에서 회원가입/로그인을 카카오 및 구글 에서도 진행할 수 있습니다. 서비스 기능에 집중할 수 있으며, 사용자는 개인정보를 기억할 필요없이 사용자 취향을 저장하는 단계로 넘어가 편의성을 제공합니다.
   1. 카카오
      1. 어플리케이션 추가



* + 1. 도메인 등록



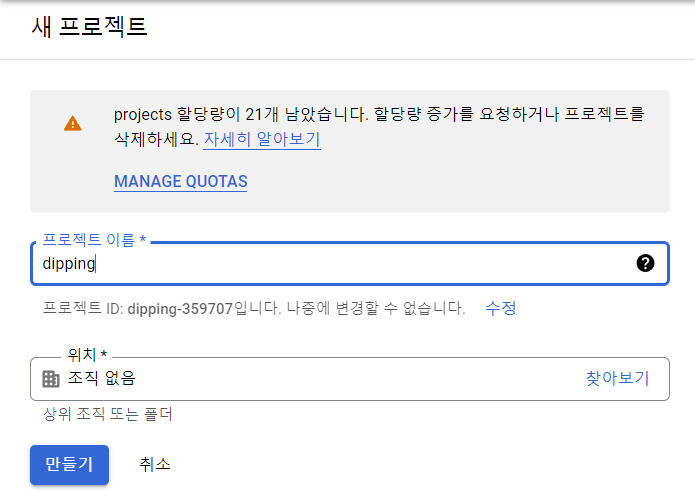
* + 1. redirect URI 설정



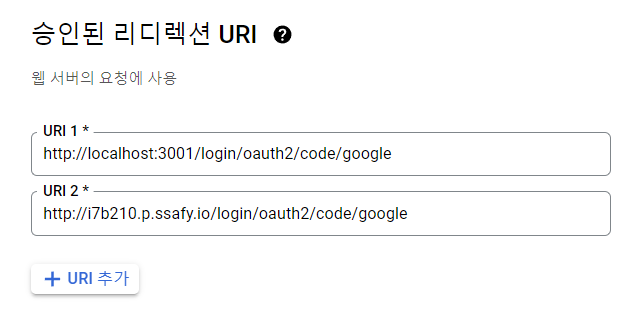
* + 1. 로그인 활성화



* 1. 구글
     1. 어플리케이션 추가



* + 1. redirect URI 설정



1. 비밀번호 찾기 할 때 인증 코드 메일 전송 기능 : 사용자가 비밀번호를 잊은 경우 작성한 이메일(사용자 ID)를 입력하여 해당 이메일로 인증 코드를 전송. 추후 비밀번호를 재설정할 수 있는 페이지로의 리다이렉트 링크를 제공하여 사용자의 편의성을 제공



1. Firebase Storage : 간편한 사용자 프로필 저장을 위해 구글 파이어베이스 스토리지를 사용했습니다.
   1. Firebase 프로젝트 생성 : 프로젝트 명 dipping
   2. 빌드 - 스토리지 에서 생성된 버킷 확인 : ex) gs://dipping-f118c.appspot.com
      1. application-prod.yml에서 app:firebase-bucket:에 gs:// 뒷부분 추가
   3. 프로젝트 설정 - 서비스 계정 - Admin SDK 구성 스니펫 : Java로 새 비공개 키 생성
      1. 다운로드한 비공개 키 json파일을 serviceAccountKey.json 으로 이름 변경 후 dipping/src/main/resource에 저장