

A307: Devtree 포팅 메뉴얼

1. 프로젝트 기술 스택

2. 빌드 관련 주요 내용

Backend

Frontend

3. 배포 시 특이사항

사전작업

NGINX conf 파일 설정

DB 설정

Opnvidu 설정

배포

4. DB 접속 정보 및 ERD

DB 접속 정보 (로컬에서 접근)

ERD

1. 프로젝트 기술 스택

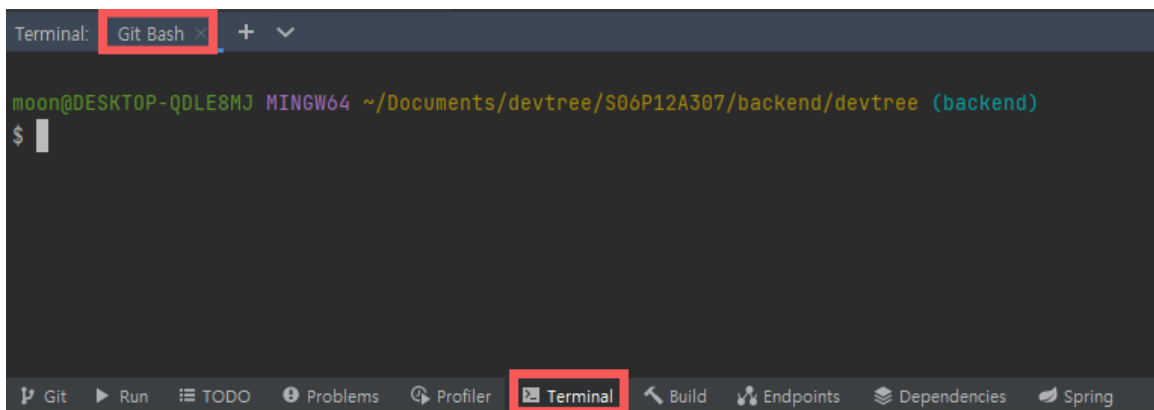
- 이슈관리 : Jira
- 형상관리 : Gitlab
- 팀 커뮤니케이션 : Discord, Notion, Mattermost
- 개발 환경
 - OS : Windows 10
 - IDE
 - IntelliJ IDEA 2021.3.1
 - Visual Studio Code 1.63.2
 - Adobe Illustrator , Figma
 - Server
 - AWS EC2 - Ubuntu 20.04 LTS
 - Nginx 1.18.0
- 상세 사용
 - Backend
 - Java (Open-JDK Zulu 8.33.0.1 / 11.0.X)

- Gradle 7.3.3
- lombok
- Openvidu 2.18.0
- Hibernate, Spring Data JPA, Spring Security
- Frontend
 - HTML5, CSS3, JavaScript(ES6)
 - React 17.0.2, Redux
 - Node.js 16.13.0
 - Openvidu 2.18.0
 - Material UI 5.0.0

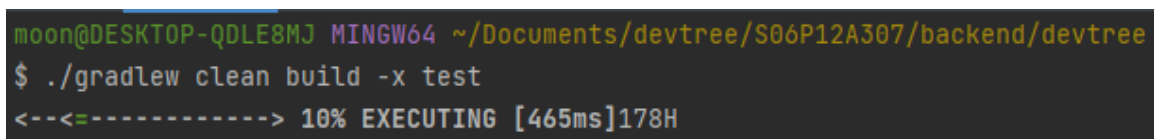
2. 빌드 관련 주요 내용

Backend

1. IntelliJ 하단에 Terminal을 클릭하여 Git Bash를 추가합니다. (Powershell이나 CMD도 무관합니다.)



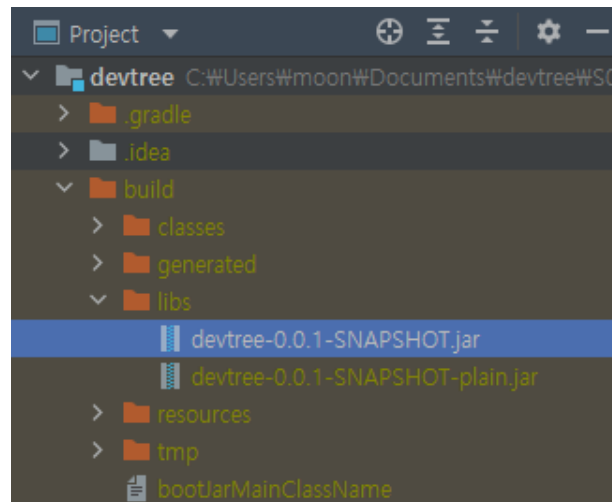
2. 현재 디렉토리 위치가 `~/backend/devtree` 라면 테스트 코드를 제외하고 빌드하기 위해 `./gradlew clean build -x test` 명령어를 입력합니다.



3. 빌드 성공 메시지를 확인 후, IntelliJ 우측 패키지 관리자에서 build/libs 하위에 2개의 jar 파일이 생성되며 devtree-0.0.1-SNAPSHOT.jar 파일이 추후 EC2에 수동 배포시 사용할 파일입

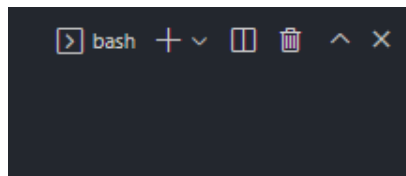
니다.

```
BUILD SUCCESSFUL in 8s
6 actionable tasks: 6 executed
```



Frontend

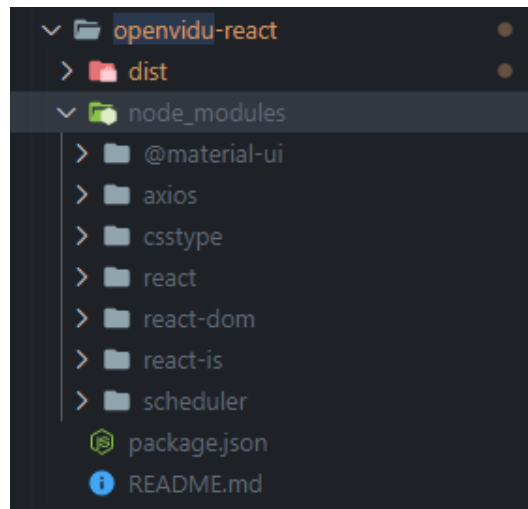
1. VS Code를 실행하여 `ctrl + shift + `` 를 눌러 터미널 창을 생성 후 동일하게 git bash를 실행시켜줍니다.



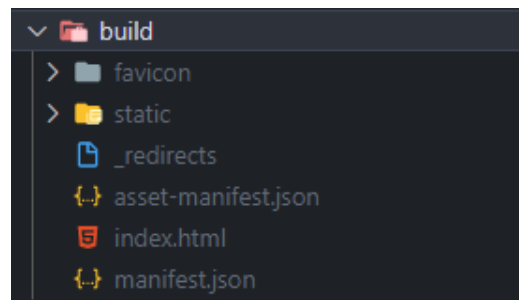
2. 프로젝트에 사용되는 라이브러리를 설치하기 위해 `npm install` 명령어를 실행합니다.

```
moon@DESKTOP-QDLE8MJ MINGW64 ~/Documents/devtree/S06P12A307/client (pre-develop)
$ npm install
```

3. 설치가 완료되면 `node_modules` 폴더가 생성되며, 하위에 `openvidu-react/node_modules` 를 찾아서 `react` 와 `react-dom` 폴더를 삭제합니다.



4. 삭제 후 `npm run build` 명령어를 입력하면 build 폴더가 생기며 추후 EC2에 수동 배포시 사용할 폴더입니다.



3. 배포 시 특이사항

사전작업

NGINX conf 파일 설정

```
server {

    server_name i6a307.p.ssafy.io;

    location / {
        root    /home/ubuntu/devtree/build;
        index  index.html index.htm;
        try_files $uri $uri/ /index.html;
    }

    location /v1 {
        proxy_pass https://localhost:8080;
    }

    listen 443 ssl; # managed by Certbot
```

```

ssl_certificate /etc/letsencrypt/live/i7a307.p.ssafy.io/fullchain.pem; # managed by Certbot
ssl_certificate_key /etc/letsencrypt/live/i6a307.p.ssafy.io/privkey.pem; # managed by Certbot
include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot
ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot
}

server {
    if ($host = i6a307.p.ssafy.io) {
        return 301 https://$host$request_uri;
    } # managed by Certbot

    listen 80;

    server_name i6a307.p.ssafy.io;
    return 404; # managed by Certbot
}

```

DB 설정

Mysql을 기준으로 진행합니다.

1. EC2에 Mysql을 설치합니다.

```
sudo apt-get install mysql-server
```

2. 별도의 계정을 만들어 권한을 부여합니다.

```

sudo mysql -u root -p
CREATE USER '새로운 계정'@'%' IDENTIFIED BY '비밀번호';
GRANT ALL PRIVILEGES ON *.* TO 'root'@'%' WITH GRANT OPTION;
FLUSH PRIVILEGES;

```

Openvidu 설정

Openvidu 공식문서 기준 On premises 방식으로 배포를 진행합니다.

1. opt 폴더 아래에 Openvidu 서버를 설치합니다.

```
curl https://s3-eu-west-1.amazonaws.com/aws.openvidu.io/install_openvidu_latest.sh | bash
```

2. opt/openvidu 로 디렉토리를 이동 후 env 파일을 엽니다.

```
cd /opt/openvidu
```

```
vi .env
```

3. .env 파일을 다음과 같이 설정합니다. 단, 처음 SSL 인증서를 제공 받기 위해서는 처음에는

```
HTTP_PORT=80 HTTPS_PORT=443
```

설정을 유지하고 인증서 발급 이후에 변경해줍니다.

```
# Domain name. If you do not have one, the public IP of the machine.
# For example: 198.51.100.1, or openvidu.example.com
DOMAIN_OR_PUBLIC_IP=i6a307.p.ssafy.io

# OpenVidu SECRET used for apps to connect to OpenVidu server and user
OPENVIDU_SECRET=MY_SECRET

# Certificate type:
# - selfsigned: Self signed certificate. Not recommended for production.
#               Users will see an ERROR when connected to web page.
# - owncert:    Valid certificate purchased in a Internet services company.
#               Please put the certificates files inside folder ./owncert
#               with names certificate.key and certificate.cert
# - letsencrypt: Generate a new certificate using letsencrypt. Please
#               required contact email for Let's Encrypt in LETSENCRYPT_EMAIL
#               variable.
CERTIFICATE_TYPE=letsencrypt

# If CERTIFICATE_TYPE=letsencrypt, you need to configure a valid email
LETSENCRYPT_EMAIL=anwngh@naver.com

# Proxy configuration
# If you want to change the ports on which openvidu listens, uncomment

# Allows any request to http://DOMAIN_OR_PUBLIC_IP:HTTP_PORT/ to be automatically
# redirected to https://DOMAIN_OR_PUBLIC_IP:HTTPS_PORT/.
# WARNING: the default port 80 cannot be changed during the first boot
# if you have chosen to deploy with the option CERTIFICATE_TYPE=letsencrypt
HTTP_PORT=8442

# Changes the port of all services exposed by OpenVidu.
# SDKs, REST clients and browsers will have to connect to this port
HTTPS_PORT=8443
```

4. 설정을 마치고 다음 명령어를 통해 openviud 서버를 시작합니다.

```
./openvidu start
```

배포

빌드 후 생성된 jar 파일과 build 폴더를 EC2에 업로드 후 배포를 진행합니다.

1. `sudo systemctl start nginx` 를 입력하여 Nginx를 실행합니다.
2. `sudo nohup java -jar devtree-0.0.1-SNAPSHOT.jar &` 를 입력하여 터미널 세션이 끊겨도 프로그램이 돌아가고 백그라운드에서 실행이 되게끔 합니다.

이후 도메인을 입력하여 확인을 하거나 `sudo netstat -ntlp` 명령어를 통해 정상적으로 실행되는지 확인합니다.

tcp	0	0 0.0.0.0:443	0.0.0.0:*	LISTEN	422601/nginx: master
tcp	0	0 0.0.0.0:8443	0.0.0.0:*	LISTEN	297122/nginx: master
tcp	0	0 0.0.0.0:4443	0.0.0.0:*	LISTEN	262605/docker-proxy
tcp	0	0 127.0.0.1:33060	0.0.0.0:*	LISTEN	145794/mysqld
tcp6	0	0 :::1:6379	:::*	LISTEN	297076/redis-server
tcp6	0	0 127.0.0.1:8080	:::*	LISTEN	457217/java

4. DB 접속 정보 및 ERD

DB 접속 정보 (로컬에서 접근)

- Connection Method : Standard TCP/IP over SSH
- SSH Hostname : i6a307.p.ssafy.io
- Username : rootnode
- Password : rootnodedevtree

ERD

