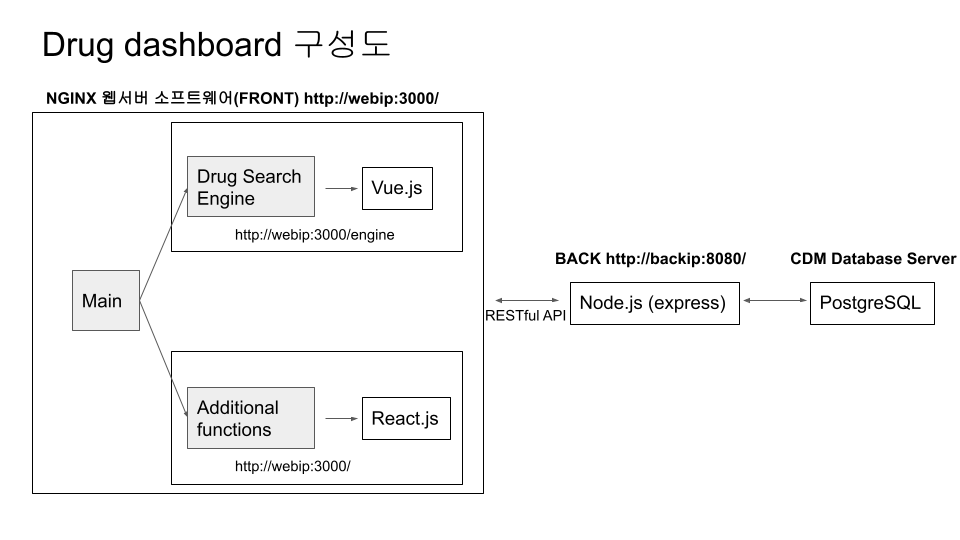
**Drug dashboard 설치 매뉴얼**

1. 설치 및 개발 환경, 필요 권한

* Front: React.js + Vue.js
* Back: Node.js (express)
* DB: PostgreSQL 14 (Create table이 가능한 스키마 및 계정 필요)
* Linux 기반의 CDM database에 접근 가능한 서버 (웹 및 api process 호스팅용, 2개의 port 필요)
* Docker image build가 가능한 서버 계정

**Network:**

**Docker build시 사용되는 이미지(node:latest, nginx:stable)가 이미 서버 내의 로컬 repository에 pull되어 있는 경우 인터넷 연결이 필요하지 않고, 없는 경우 docker hub에서 접속하여 불러와야 하기 때문에 외부 네트워크와의 연결이 필요합니다.**

****

1. https://github.com/SNUH-BMI/DrugDashboard에 접속해 프로젝트 clone을 해줍니다.
2. 프로젝트 폴더로 이동해 back/app 폴더 안으로 들어가 해당 경로에서 VScode와 같은 코드에디터를 열어줍니다. 그 후 해당 경로의 터미널에서 npm install 을 입력해 패키지 업데이트를 해줍니다.
3. 2번을 front/drugdashboard\_react, frontVue/app, DDmakejson 폴더들 안에서 똑같이 진행해줍니다.
4. 캐시 테이블 만들기 – DD cache table 만들기 문서에 있는 쿼리들을 모두 실행해 프로젝트에서 사용할 cache table들을 모두 생성해줍니다.
5. 4번에서 만들어진 drug\_count.json 파일을 front/drugdashboard\_react/src/cached\_data/ 폴더 안에 넣어줍니다.
6. 4번에서 만들어진 atc.json 파일을 복사해 back/app/data/, front/drugdashboard\_react/src/cached\_data/ 폴더 안에 넣어줍니다.
7. 4번에서 만들어진 drug\_route\_type.json 파일을 복사해 back/app/data 폴더 안에 넣어줍니다.
8. 이제 각각 프로젝트들의 config.json 및 db.json을 작성해줄 차례입니다.

back – back/app/config/config.json

위의 경로로 들어가 config.json 파일을 열어주고, 해당 값들을 소속 기관에 맞게 변경해주시면 됩니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

main\_exposure\_cache부터 dates\_cache 까지는 4에서 생성한 캐시테이블들의 schema.tablename 입니다. CREATE TABLE을 하실 때 테이블 이름을 변경하지 않으셨다면, 캐시스키마명 이라고 되어 있는 부분을 모두 찾아 실제 캐시 테이블들이 만들어진 스키마 이름을 입력해 주시면 됩니다. (find, replace all을 하셔도 됩니다.)

main\_schema는 CDM스키마명 부분에 실제 cdm data가 들어있는 스키마의 이름을 적어주시면 됩니다.

institution\_sourcename\_column은 source\_to\_concept\_map 테이블의 기관 id가 담겨있는 컬럼명을 입력해주시면 됩니다.

institution\_name은 웹에서 보여질 기관명 영어 약자를 기관명 부분에 적어주시면 됩니다.

ALT\_code부터 TBil\_range까지는 검사 수치 이상 횟수 기능에서 쓰이는 각 measurement들의 omop\_concept\_id와 정상 수치 range입니다. OMOP CDM을 쓰고 있고, range를 바꾸고자 하시는 경우가 아니라면 변경하지 않으셔도 됩니다.

Web\_address의 경우에는 web이 구동될 서버의 주소를 써주면 됩니다. Port는 기본으로 3000번 포트로 빌드가 되어, 중간에 webip 부분만 서버의 ipv4 주소로 바꿔주시면 됩니다. (127.0.0.1 형식)

front – front/drugdashboard\_react/src/config.json

위의 경로로 들어가 config.json 파일을 열어주고, 해당 값들을 소속 기관에 맞게 변경해주시면 됩니다.

텍스트, 화면, 디스플레이, 스크린샷이(가) 표시된 사진

자동 생성된 설명

homepage의 경우에는 web이 구동될 서버의 주소를 써주면 됩니다. Port는 기본으로 3000번 포트로 빌드가 되어, 중간 localhost 부분만 서버의 ipv4 주소로 바꿔주시면 됩니다. (127.0.0.1 형식)

server의 경우에는 backend api process가 실행될 서버의 주소를 써주면 됩니다. Port는 기본으로 8080번 포트로 빌드가 되어, 중간 127.0.0.1 부분만 서버의 ipv4 주소로 바꿔주시면 됩니다. (매뉴얼을 그대로 따라 하시는 경우 front web과 back이 동일한 서버에서 구동이 되니 hompage의 ip 주소에 뒤 포트번호만 바꿔서 입력해주시면 됩니다.)

institution\_name은 웹에서 보여질 기관명 영어 약자를 기관명 부분에 적어주시면 됩니다.

institution\_sourcename\_column은 source\_to\_concept\_map 테이블의 기관 id가 담겨있는 컬럼명을 입력해주시면 됩니다.

ALT\_code부터 TBil\_code까지는 검사 수치 이상 횟수 기능에서 쓰이는 각 measurement들의 omop\_concept\_id입니다. OMOP CDM을 쓰고 있으시다면 변경하지 않으셔도 됩니다.

frontVue – frontVue/app/src/config.json

위의 경로로 들어가 config.json 파일을 열어주고, 해당 값들을 소속 기관에 맞게 변경해주시면 됩니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

homepage의 경우에는 web이 구동될 서버의 주소를 써주면 됩니다. Port는 기본으로 3000번 포트로 빌드가 되어, 중간 localhost 부분만 서버의 ipv4 주소로 바꿔주시면 됩니다. (127.0.0.1 형식)

server의 경우에는 backend api process가 실행될 서버의 주소를 써주면 됩니다. Port는 기본으로 8080번 포트로 빌드가 되어, 중간 127.0.0.1 부분만 서버의 ipv4 주소로 바꿔주시면 됩니다. (매뉴얼을 그대로 따라 하시는 경우 front web과 back이 동일한 서버에서 구동이 되니 hompage의 ip 주소에 뒤 포트번호만 바꿔서 입력해주시면 됩니다.)

institution\_name은 웹에서 보여질 기관명 영어 약자를 기관명 부분에 적어주시면 됩니다.

institution\_sourcename\_column은 source\_to\_concept\_map 테이블의 기관 id가 담겨있는 컬럼명을 입력해주시면 됩니다.

ALT\_code부터 TBil\_code까지는 검사 수치 이상 횟수 기능에서 쓰이는 각 measurement들의 omop\_concept\_id입니다. OMOP CDM을 쓰고 있으시다면 변경하지 않으셔도 됩니다.

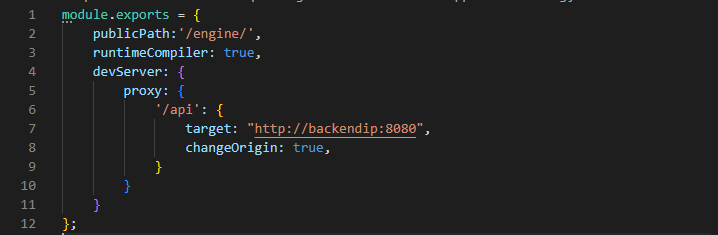
back – back/app/config/db.json

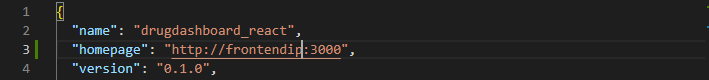
위의 경로로 들어가 db.json 파일을 열어주고, 해당 값들을 해당 기관의 CDM 데이터베이스 서버 정보로 변경해주시면 됩니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

1. IP 주소 설정을 해주기 위해, frontVue/app/vue.config.js 파일을 열고, target 부분에 backend api process가 구동되는 ip 주소를 넣어주세요. 또한, front/drugdashboard\_react/package.json 파일을 열고 homepage 부분에 frontend web이 구동되는 ip 주소를 넣어주세요.





1. 프로젝트들을 모두 build 해주기 위해 2.번과 같이 back/app/ 에서 코드 에디터를 열고, 터미널을 연 후, npm run build 를 입력해주세요. front/drugdashboard\_react/ , frontVue/app/ 에서도 똑같이 터미널을 열고, npm run build를 해주세요. 그 후 웹 이름 설정을 위해 빌드된 frontVue/app/dist/index.html 파일에 들어가 <title>Vue app</title> 안의 Vue app 부분을 Drug Dashboard로 바꿔주세요.



1. 마지막으로 해당 프로젝트들의 docker image 생성 후 container 구동을 해줍니다. 터미널에 접속해 프로젝트 폴더로 이동해주신 후 아래의 명령어를 입력해주세요.

docker build -t dd\_final\_front:0 .

그 후, cd 명령어를 통해 back 폴더에 들어가 아래 명령어를 입력해주세요.

docker build –t dd\_final\_back:0

docker images 명령을 통해 2개의 docker image가 잘 추가되었는지 확인해주세요.

잘 생성이 되었다면 해당 이미지들을 통해 컨테이너를 구동해주기 위해 아래의 명령어들을 입력해주세요.

docker run -itd --name drug\_dashboard\_front -p 3000:80 --restart=always dd\_final\_front:0

docker run -itd --name drug\_dashboard\_back -p 8080:8080 --restart=always dd\_final\_back:0

docker ps 명령을 통해 2개의 container가 추가가 되었는지 확인해주세요.

설정해주신 homepage 주소를 입력해 웹이 잘 구동이 되는지 확인해주세요.