

Web Report 개선

프로젝트 계획서

01 Contents

프로젝트 개요

프로젝트 구조

프로젝트 UI



01

프로젝트 개요

01 프로젝트 개요

Project Name : "Mighty WebReport"

Project Period : 2022.05.09(월) ~

2022.06.17(금) (약 5주간)

Project Purpose

기존의 Web Report 보다 사용자 친화적, 속도향상, 보안강화를 목적에 두고 있습니다.

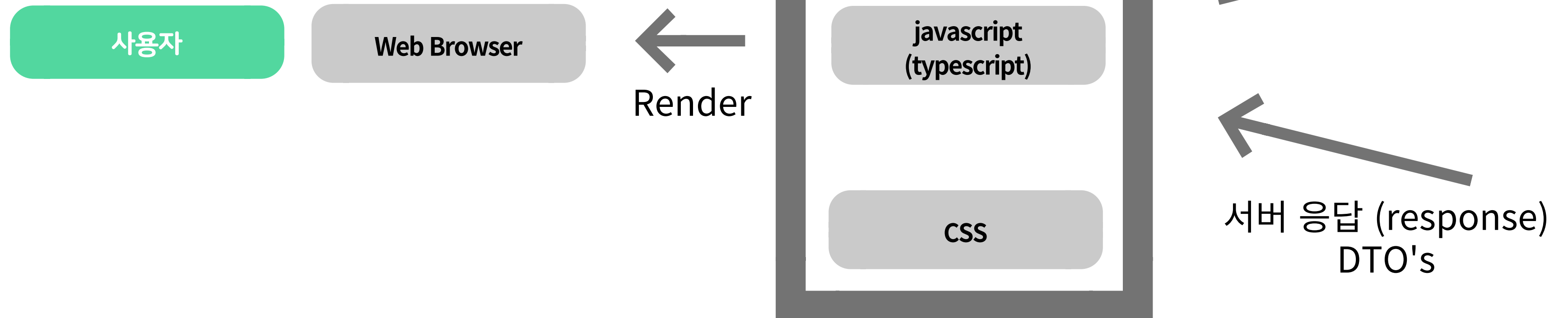
1. 보안강화 : 기존의 사이트는 비밀번호를 사용자(client)단에서 encode 하여 서버에 보내줍니다. 이 과정에서 xss가 일어날 가능성이 있고, 또한 web secure가 되어있지 않아 csrf 요청에도 취약합니다. 이러한 점을 csrf에는 완전 안전, xss에는 보다 안전한 방식으로 업그레이드 합니다.

2. 트래픽 감소 : session 방식에서 token 인증방식으로 바꾸면서 요청횟수를 줄이고, 뿐만 아니라 데이터를 React Store기능을 활용하여 서버가 아닌 사용자(client)단에 보관하게 하여 서버의 부담을 줄이고, 속도 향상하는 방식입니다.

3. UI/UX 업그레이드 : SPA(Single Page Application) 방식으로 핸드폰 어플리케이션과 같이 깜박임이 존재하지 않고, 페이지가 좀더 부드러워 집니다.
거기에 더해 페이지 전반적으로 사용하기 편하게 업그레이드 합니다.

01 프로젝트 개요

Software Configuration - Front End (사용자)



01 프로젝트 개요

Software Configuration - Front End (사용자)

React 는 Single Page Application 으로 DOM을 동적으로 생성, 수정, 제거를 부드럽게 해줌으로서, 사용자들이 Web을 더 부드럽고 편하게 사용 가능하게 해준다.

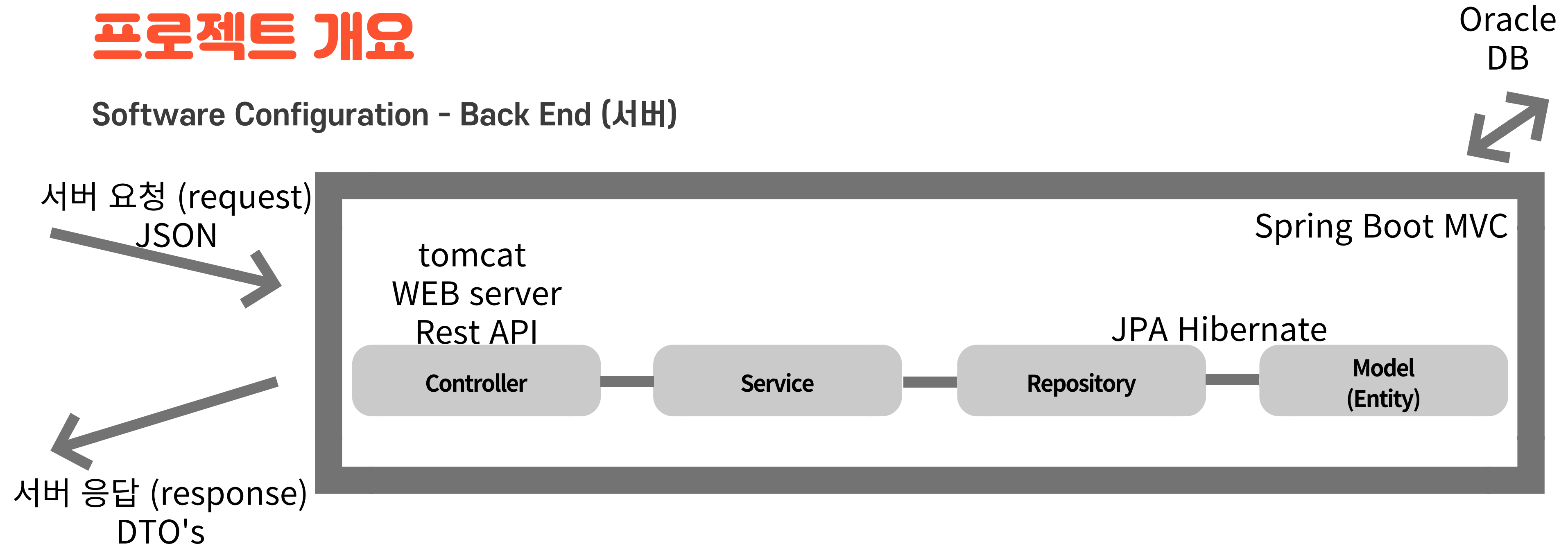
redux를 활용해 데이터를 최대한 Client가 부담하게 함으로서, 백엔드 요청 횟수와 쿼리 발생 횟수를 최소화 시킴으로서 서버 관리를 더 용이하게 해준다.

즉 기존 프로젝트에 비교해서 클라이언트는 더 무거워 지되, 서버를 더 가볍게 하여서 프로젝트의 부담을 많이 덜어준다.

기술명	버전	설명
React	18.1.0	함수형 Component 기반 View template
React-router-dom	6.3.0	사용자가 요청한 URL에 맞는 화면 Routing
React-redux	8.0.1	상태관리 라이브러리
redux-saga	1.1.3	Generator 기반 redux 미들웨어 상태관리 비동기처리 담당
styled-component	5.3.5	scss 문법으로 js를 활용해 css를 동적으로 활용
typescript	4.6.4	javascript 컴파일러 (런타임 에러 방지)
axios	0.27.2	Promiss 기반 request 요청 생성
pretendard	1.2.2	폰트

01 프로젝트 개요

Software Configuration - Back End (서버)



01 프로젝트 개요

Software Configuration - Back End (서버)

Front-End를 Back-End와 완전 분리, 기존의 ASP.NET과는 다르게 view에 html 데이터를 보내는 것이 아닌, (Model And View 방식) DTO를 JSON형식으로 보내줌으로서 (RestAPI 방식) 백엔드 서버는 데이터를 검색, 가공을 해주어 보내주기만 한다.

보안은 Spring Interceptor를 이용하여 요청을 뺏어와서 권한을 검사한다.

기술명	버전	설명
Spring boot	2.6.7	IOC와 DI를 활용한 객체 관리 아키텍처인 Spring 을 좀더 용이하게 사용
Spirng web	2.6.7	Spring 내장인 WAS Tomcat 서버를 구축하고, RESTful 서비스 개발하는데 사용
Spring security	2.6.7	CORS, CSRF를 방지하고, Intercept 를 이용해 URL 별로 권한을 설정
jjwt	0.11.5	JSON Web Token 생성, 해독
JPA	2.6.7	ORM중 하나로, DB 테이블을 객체 단위로 매핑하여서 객체지향적으로 데이터 관리, Parameter 방법으로 Query Injection 안전
QueryDSL	5.0.0	JPA JPQL을 객체지향으로 관리 가능하게 해주는 query 라이브러리
Oracle	11g	DataBase

01 프로젝트 개요

Hardware Configuration

제품	설명
Web Server	tomcat server
DataBase Server	Oracle 11g Server
PC	IE 11 미만 미지원 (윈도우 xp 이하 미지원)

01

프로젝트 업무 범위

업무	업무 범위
문서화	프로젝트 계획서 작성
시스템 구축	<div>분석</div> <div>- 기존 WebRepost 사이트 분석</div> <div>설계</div> <div>- UI 업그레이드 설계</div> <div>- 추가 기능 설계</div> <div>구현</div> <div>- Daily Moment</div> <div>- Operation Movement Monitoring</div> <div>- Daily Shipping Status</div> <div>- Device 출하기준</div> <div>- Inventory 현황</div> <div>- LOT 정보 조회</div> <div>- Hold 현황</div> <div>- AVI Yield Report</div> <div>- PTEST Yield Report</div> <div>- 최종 PTEST Yield Report</div> <div>- 당월 판매실적 현황</div>
테스트	테스트

01 형상관리

GitHub Repository

The screenshot shows a GitHub repository page for 'mighty-webreport' by user 'show5116'. The repository is on the 'master' branch, has 1 branch, and 0 tags. It contains 47 commits. The file list includes 'frontend', 'src/main', '.gitignore', 'README.md', 'build.gradle', 'gradlew', 'gradlew.bat', and 'settings.gradle'. The README.md file is open, showing the project title 'Mighty-WebReport Project' and instructions to use Spring security & jjwt and run the Command Line. The right sidebar shows repository statistics: 0 stars, 1 watching, 1 fork, and a language distribution chart for Java (38.2%), JavaScript (32.9%), TypeScript (28.0%), and HTML (0.9%).

File	Commit Message	Time Ago
frontend	FEAT : add request interceptor	1 hour ago
src/main	FEAT : add request interceptor	1 hour ago
.gitignore	FEAT : add getPlantList	12 days ago
README.md	FEAT : set login state, login persist	10 days ago
build.gradle	CHORE : React Auto Build	6 days ago
gradlew	init	25 days ago
gradlew.bat	init	25 days ago
settings.gradle	init	25 days ago

<https://github.com/show5116/mighty-webreport>

github repository를 통해 형상관리중,
repository는 private으로 공개를 하였으며,
반드시 초대를 통해서만 소스코드에 접근이 가능하다.

COMMIT 규칙을 간단히 설정하여 버전관리를 하여서
이슈가 발생하게 되면 문제점을 찾기 쉽게 하였다.

commit 규칙
FEAT - 기능 추가
STYLE - 스타일 변경
CHORE - 빌드, 설정 변경
REFACTOR - 리팩토링
FIX - 에러 수정

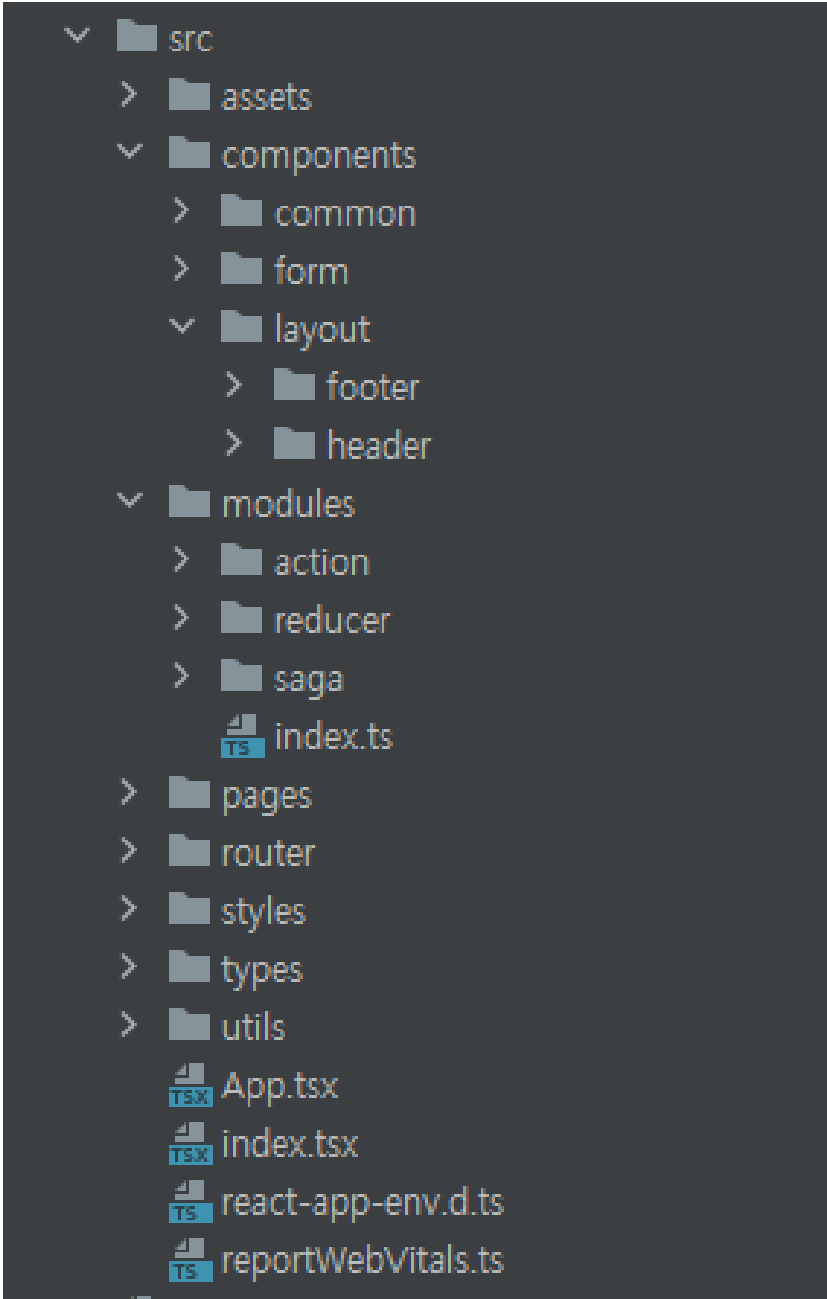
이슈 관리
이슈는 git issue를 통해 관리

02

프로젝트 구조

04

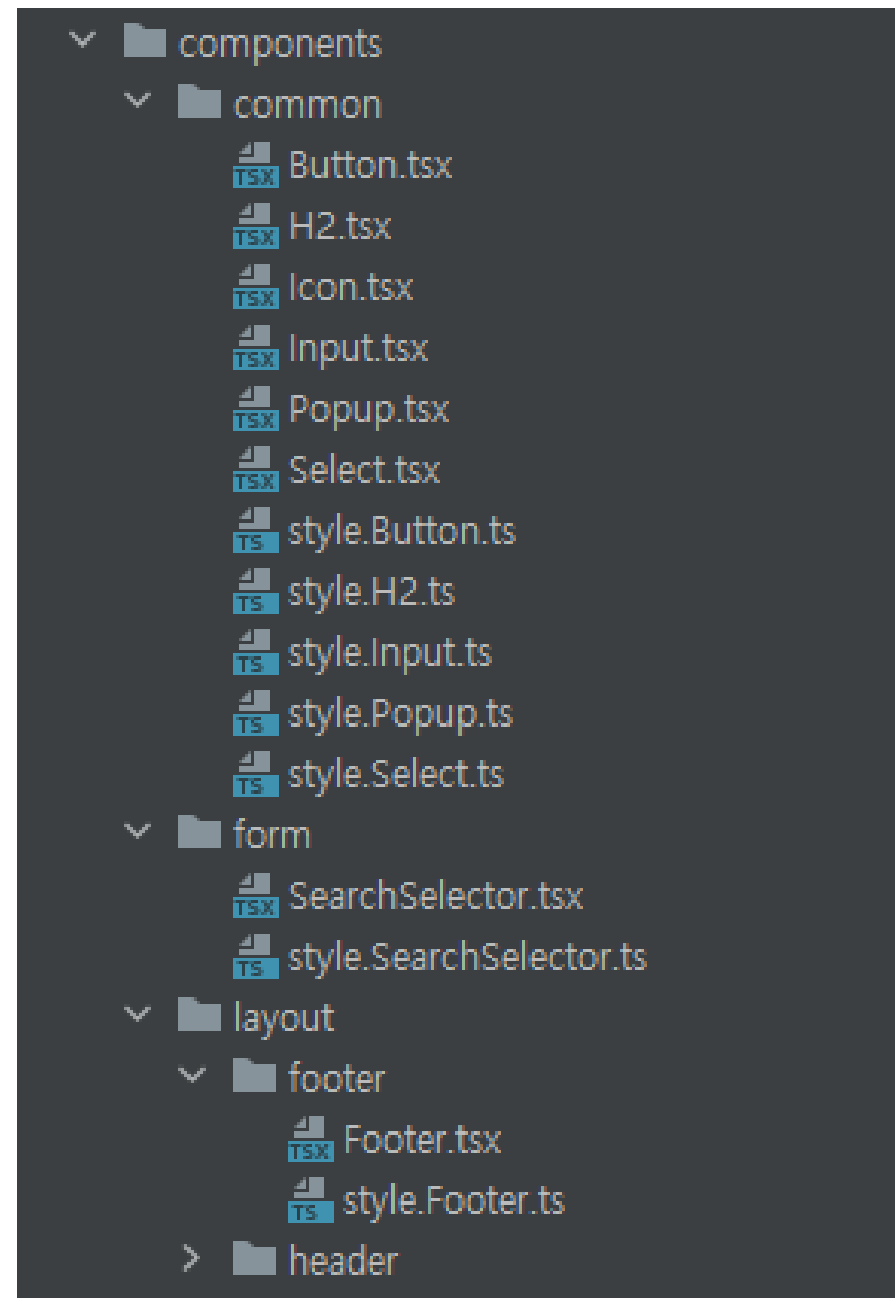
패키지 구조 (front)



폴더명	설명
assets	img, png, svg 등의 정적 파일 관리
components	page 내부에 사용될 Component 관리 common - Input, Button 등 공통 컴포넌트 form - searchbox 등 form 컴포넌트 layout - 헤더, 메뉴바, 푸터 등 레이아웃 컴포넌트
modules	redux를 통한 front end 상태관리 action - 상태관리 선언 reducer - 상태초기화 및 관리 saga - 비동기 상태 관리
pages	사용자들에게 보여질 page 관리
router	page와 URL을 매핑하며, 외부 입력으로부터 intercept 해와 화면을 구성
styles	글로벌 style (font, background 등), color 관리
types	server에서 넘어올 DTO의 타입 정의, 및 프로젝트 공통 타입 정의
utils	request interceptor로 header 세팅을 하며, 직접 만든 useHook을 사용

04

네이밍 규칙 (front)



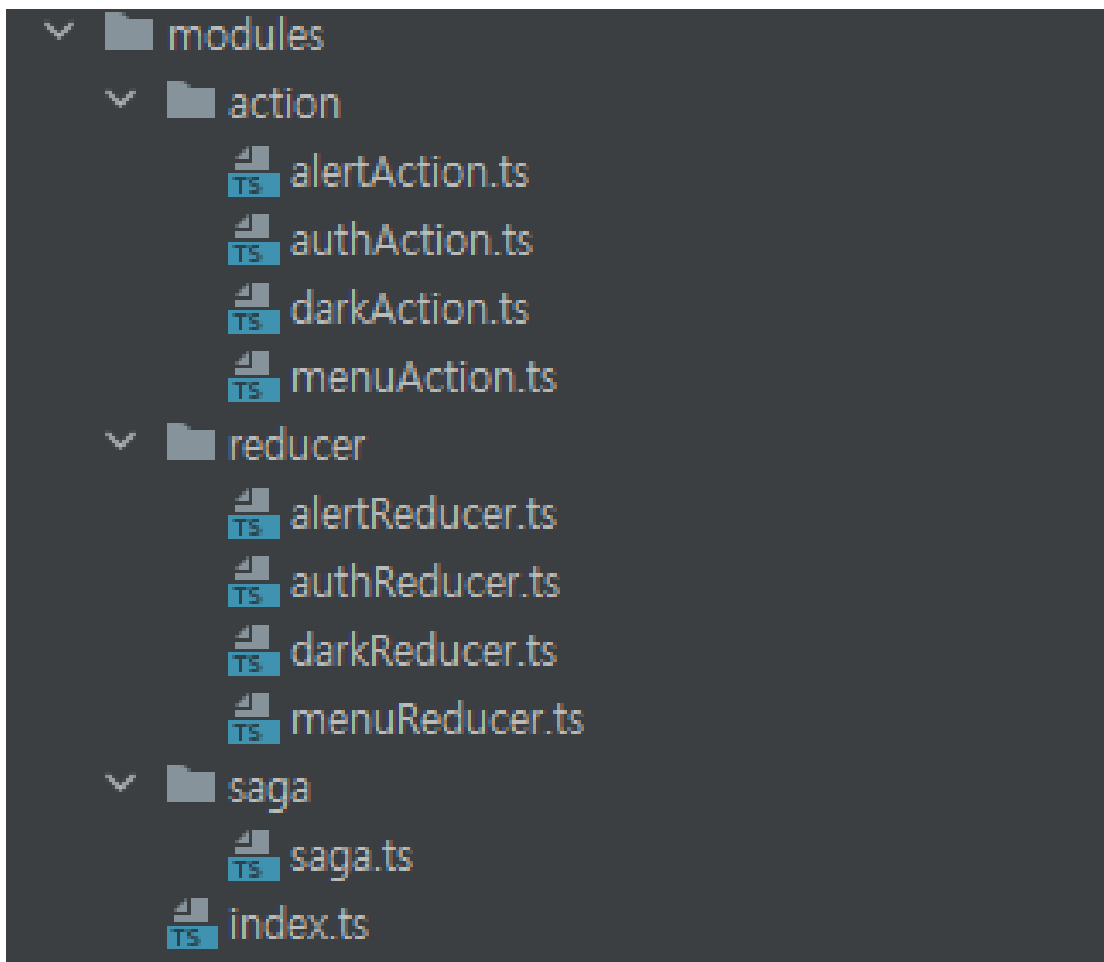
컴포넌트들은 항상 첫자를 대문자로 하는 PascalCase로 한다.

css는 style.(컴포넌트명).ts 로 파일을 생성해서 따로 관리

공통 컴포넌트는 태그와 같은 이름을 사용 (input -> Input)

태그에 존재하지 않는 컴포넌트의 경우, 기능에 맞게 이름을 사용 (Popup.tsx, SearchSelector.tsx)

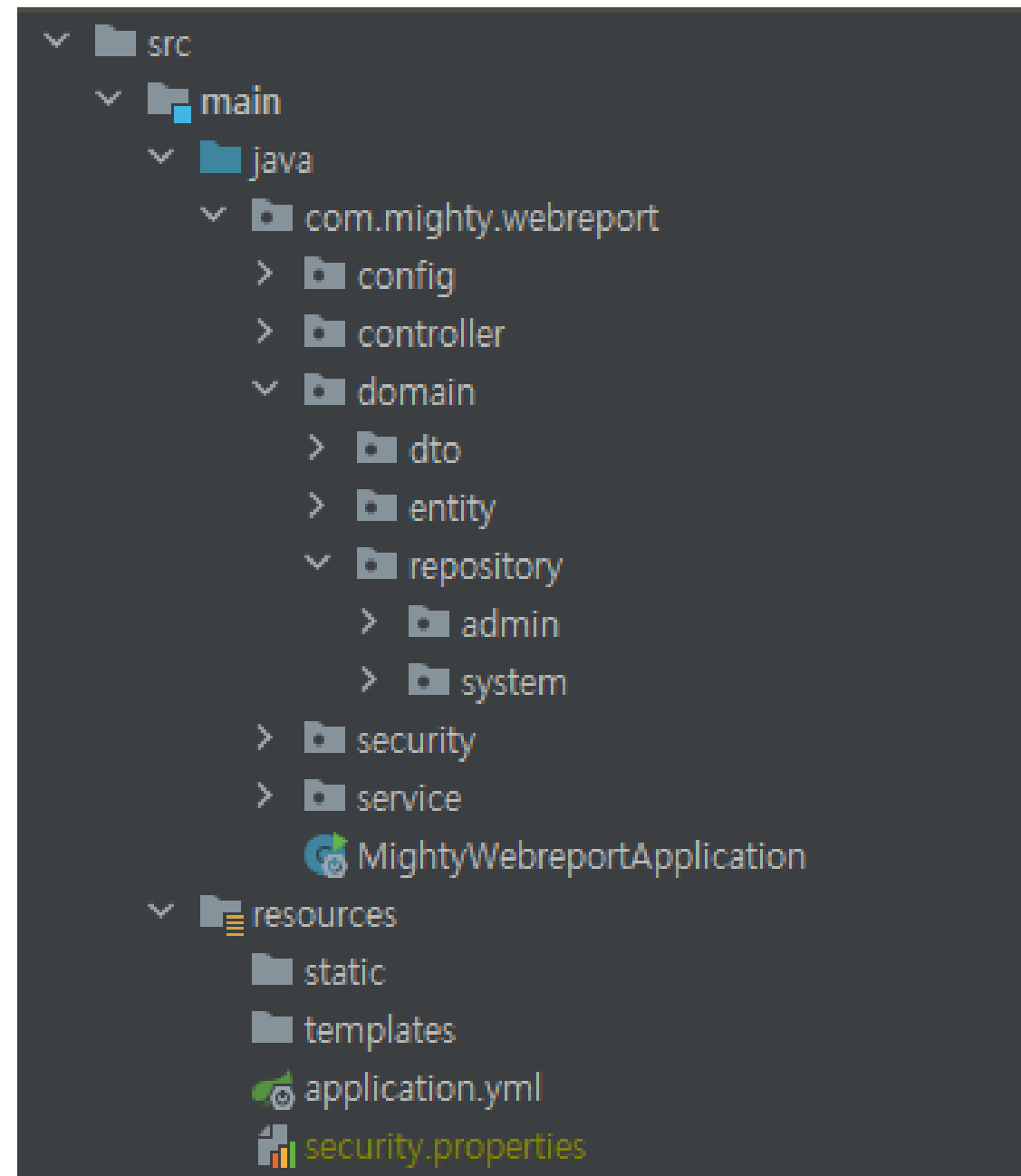
04 네이밍 규칙 (front)



컴포넌트가 아닌 ts 파일들은 첫자를 소문자로 하는 camelCase로 한다.

상태관리 파일은 상태관리를 선언해주는 action파일과 상태를 관리해주는 reducer 파일로 나누어서 관리하고,
index.ts에서 reducer 파일을 매핑해준다.

04 패키지 구조 (back)



폴더명	설명
config	Web을 포함해서 back-end의 configuration 관리
controller	client의 요청을 get, post, put, delete로 받아서 비즈니스 로직(service)에 이어주고, 결과물을 response에 담아 전송
domain	data를 관리 dto - response, request에 사용할 객체 entity - DB의 테이블에 매핑되어있는 객체 repository - query 관리
security	프로젝트의 보안 및, 에러 관리
service	비즈니스 로직, 한 메소드가 한 transaction이 되어서, 혹시 모를 에러에 rollback 단위를 관리
resources	application.yml - 프로젝트 세팅 security.properties - DB 정보나, jwt 키와 같은 보안키 관리 (프로젝트 외부도 가능)

* dto와 entity를 나눈이유

둘을 나누지 않고 entity만으로 같은 기능을 할 순 있으나 MVC 패턴에 맞게 controller는 데이터 이동, service는 비즈니스 로직 으로 분리를 확실하게 해줌.
(혹시 모를 DB 변화를 방지)

04

보안 (back)

▼ config

DatabaseConfig

GlobalPropertySource

QueryDslConfig

SecurityConfig

WebMvcConfig

> controller

> domain

▼ security

AccountContext

AuthCode

AuthToken

@CurrentUser

CustomUserDetailsService

JwtAuthenticationEntryPoint

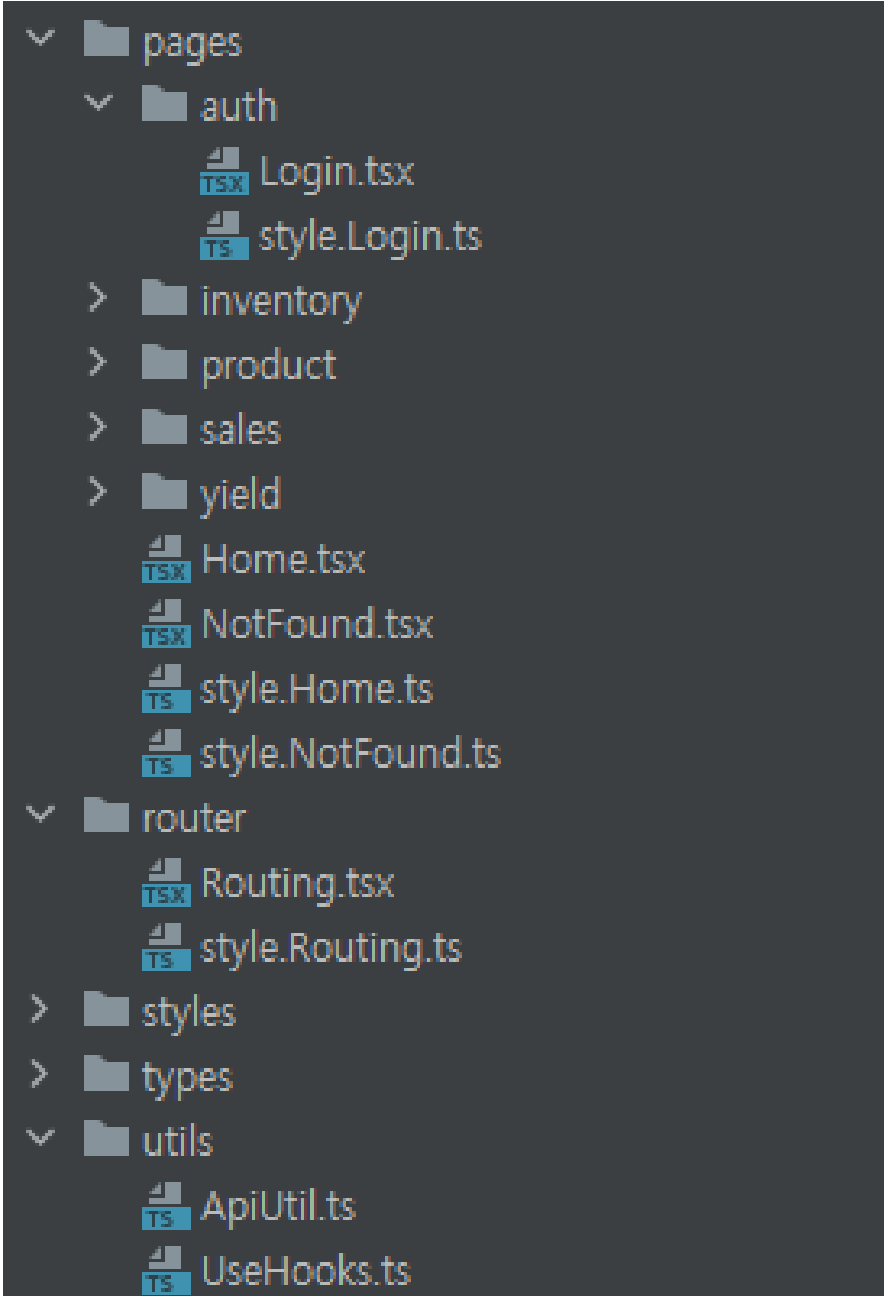
JwtAuthFilter

JwtAuthProvider

파일명	설명
SecurityConfig	CORS, CSRF차단, URL마다 권한 설정, 권한이 걸린 URL요청을 Filter와 연결, 정적파일(img,png...) 외의 파일 접근을 차단, Password Encoder 설정(MD5에 header를 더한 방식)
AccountContext	프로젝트에서 사용할 계정 단위 id, 이름, 계정의 권한
AuthCode	Exception 코드 (enum)
CurrentUser	request 요청의 로그인 되어있는 계정을 발견
CustomUser DetailsService	로그인 서비스, UserNotFoundException 이나 CreditionException과 같은 로그인 에러를 출력
JwtAuthentication EntryPoint	Exception 핸들러, 에러 발생시 response에 에러코드를 주입
JwtAuthFilter	권한이 있는 request요청에 권한을 확인하고 맞지않은경우 Exception(AuthCode) 발생
JwtAuthProvider	로그인 성공시, 권한 확인시 마다 새로운 권한 token을 발급

04

보안 (front)

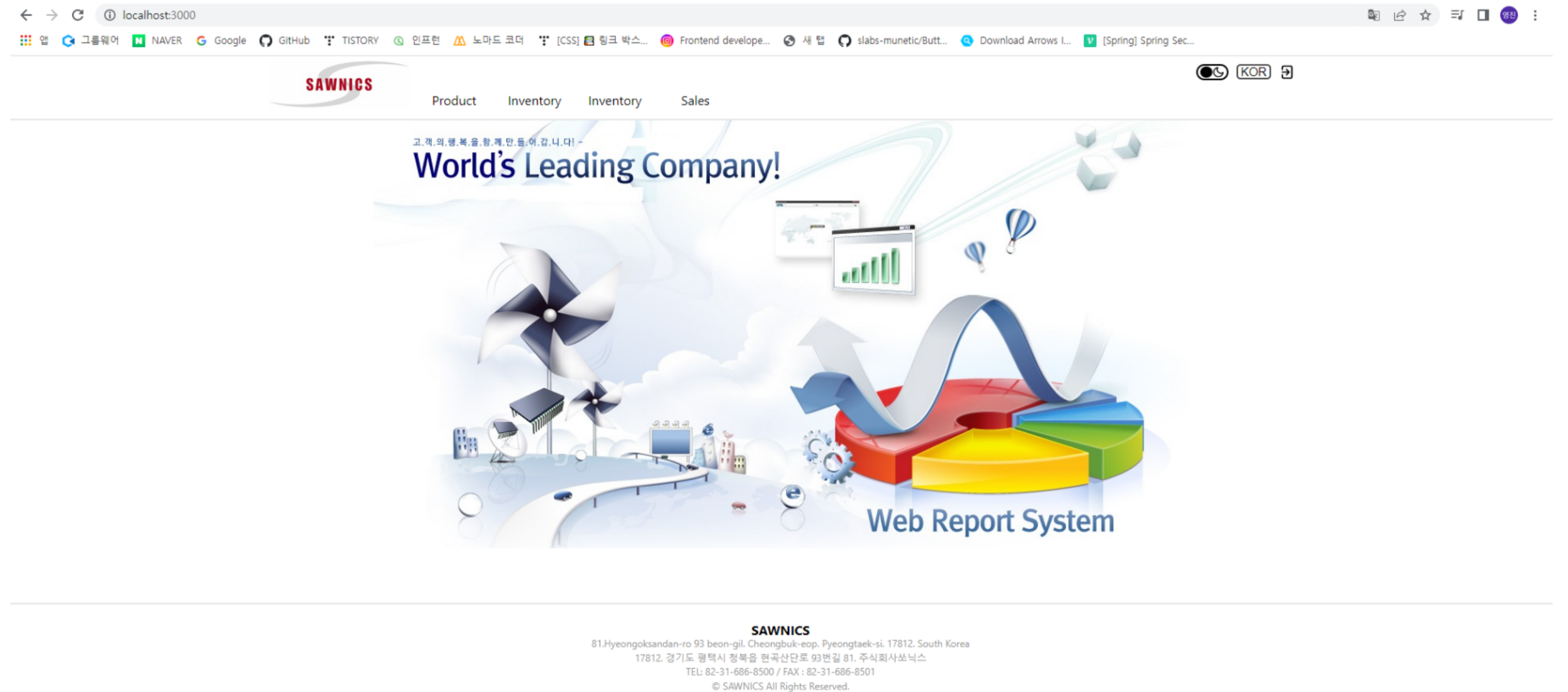


파일명	설명
Login.tsx	로그인 페이지 axios를 이용해 아이디와 비밀번호 plant로 로그인을 요청하고, 맞는 정보일 경우 token을 발급
NotFound.tsx	404 (Page Not Found 페이지)
Routing.tsx	인증토큰이 만료되거나, 로그인이 되어있지 않을경우 강제로 /login url로 이동시킨다. 사용자가 url 직접접근을 할 경우 허용되지 않은 url은 강제로 NoutFound.tsx로 이동시킨다.
ApiUtil.tsx	axios 요청을 Intercept해와서 request header에 jwt 인증토큰을 담아준다.

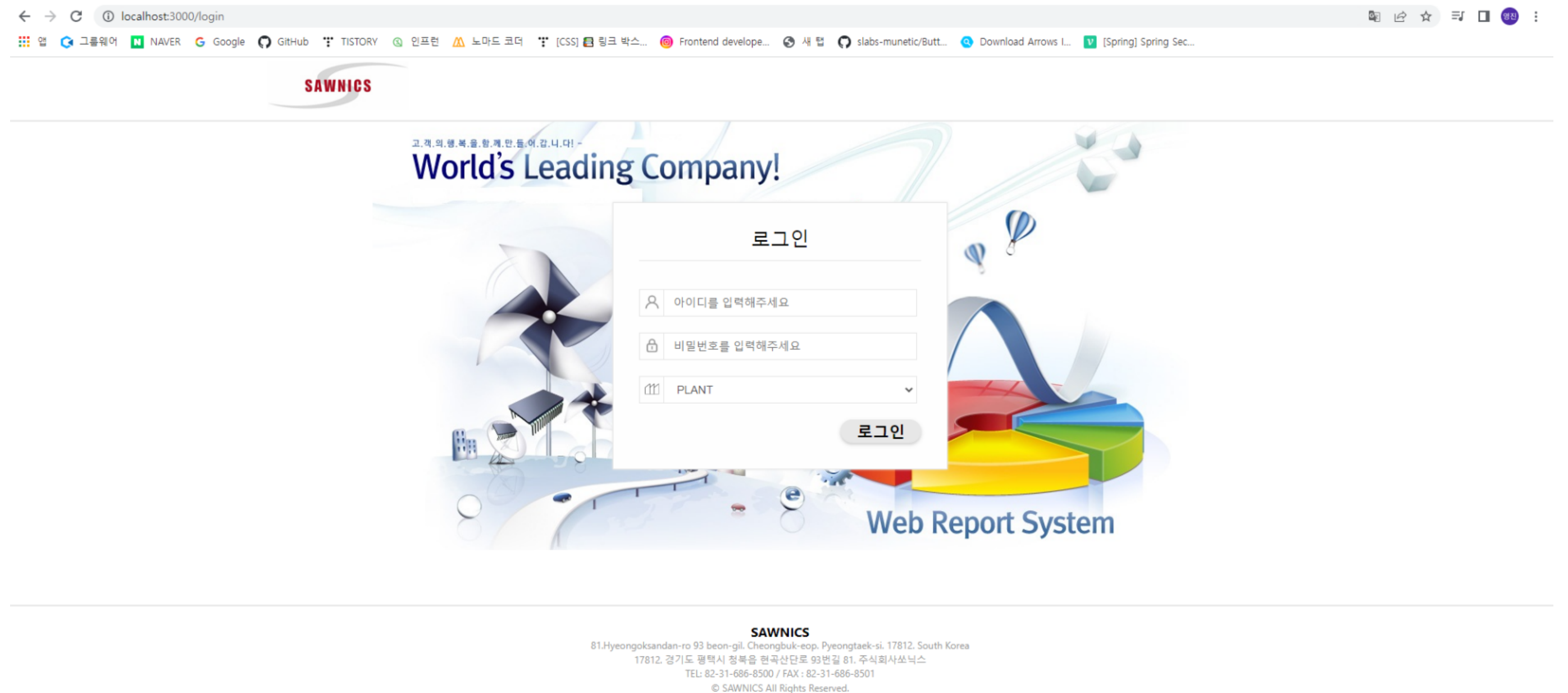
03

프로젝트 UI

04 UI (메인 페이지)



04 UI (로그인 페이지)



04 UI (조회 페이지)

디자인 초안이라 개발하면서 변경될 예정

조회 결과 테이블 미포함

← → ↺ localhost:3000/inventory/lot-status

앱 그룹웨어 NAVER Google GitHub TISTORY 인프런 노마드 코더 [CSS] 링크 박스... Frontend develop... 새 탭 slabs-munetic/Butt... Download Arrows L... [Spring] Spring Sec...

SAWNICS Product Inventory Inventory Sales

LOT 정보 조회

Customer
Q Search...

BROADCOM
LGE

공정
Q Search...

C0100
C0101

제품
Q Search...

FCSDU001

조회 시작
금일
2021.12.14 (화)

조회 끝
금일
2021.12.14 (화)
12월 2021
일 월 화 수 목 금 토
28 29 30 1 2 3 4
5 6 7 8 9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31 1

SAWNICS
81.Hyeongoksandan-ro 93 beon-gil, Cheongbuk-eop, Pyeongtaek-si, 17812, South Korea
17812, 경기도 평택시 청북읍 현곡산단로 93번길 81, 주식회사쏘닉스
TEL: 82-31-686-8500 / FAX : 82-31-686-8501
© SAWNICS All Rights Reserved.

감사합니다!