프론트엔드 (로그인 기능)

* 로그인 기능 사용

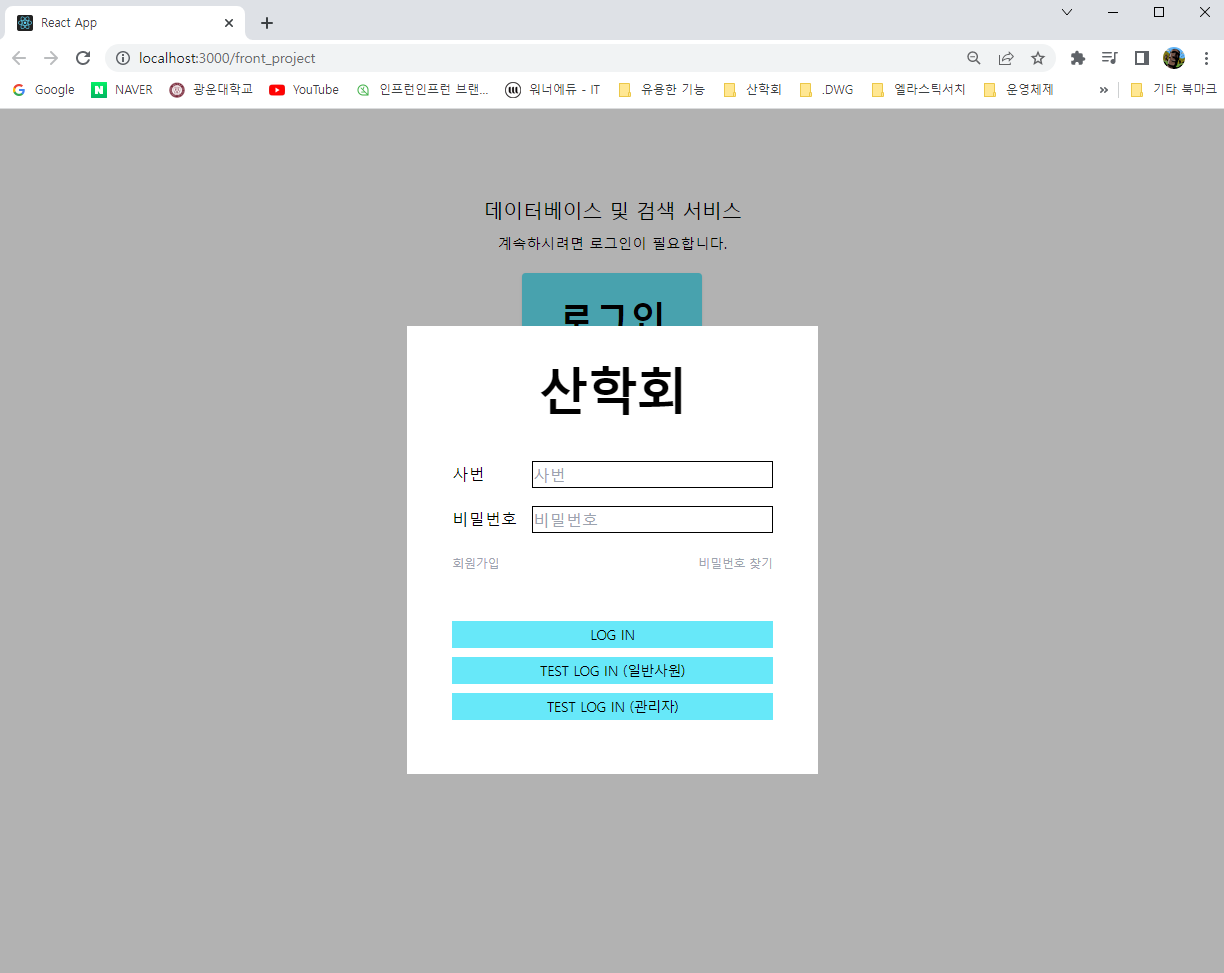
1. 임시 로그인 백서버 오픈(백서버 경로에서) 및 실행



텍스트이(가) 표시된 사진

자동 생성된 설명

1. 초기 화면



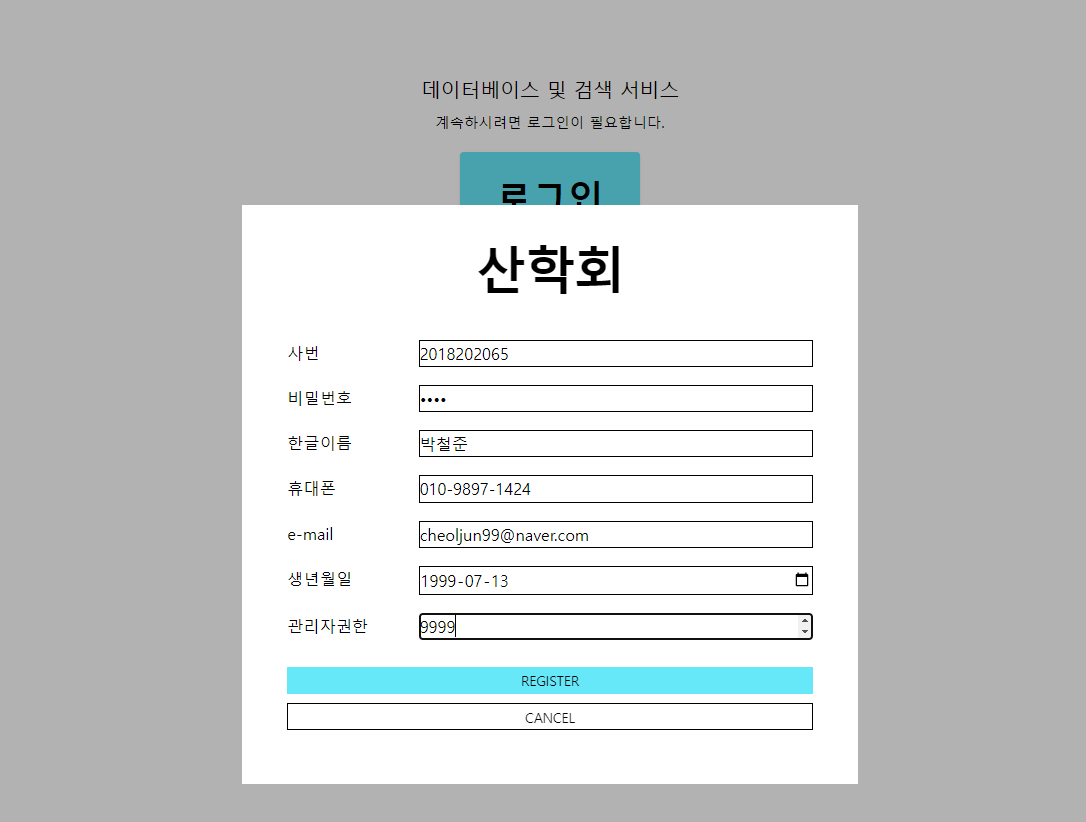
1. 회원가입

초기화면은 아래와 같다.

테이블이(가) 표시된 사진

자동 생성된 설명

이후 아래와 같이 정보를 입력한다. 비밀번호(1234)



이후 REGISTER버튼을 통해 회원가입을 진행하게 되는 데 이때 사용하는 관리자 권한 인증키의 경우 9999의 경우가 관리자 권한을 받는 인증 키이며 해당 키를 제외한 모든 값을 입력은 일반회원의 권한을 받는다.

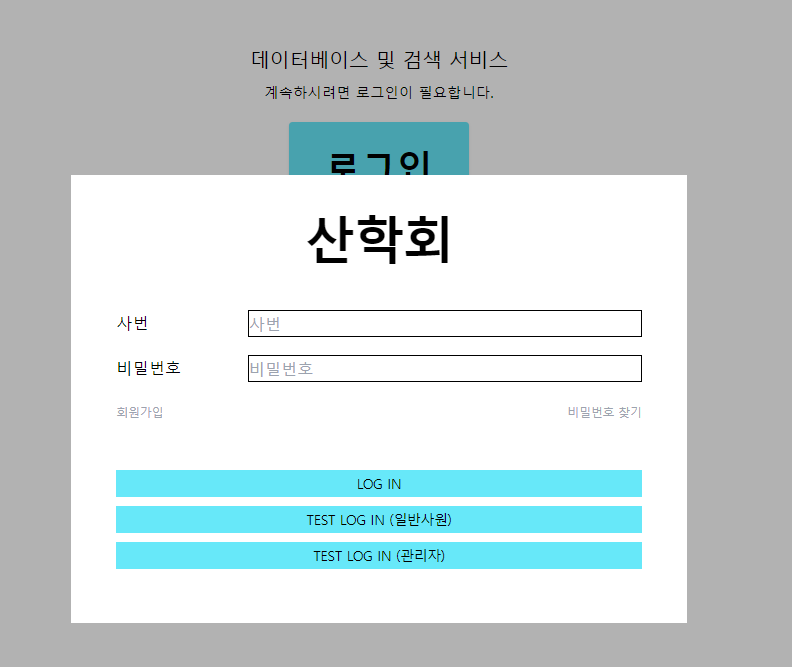
아래는 잘못된 값이 들어왔을 경우(로그인 서버에 사번이 이미 존재하는 경우 or 정보 입력을 누락하였을 경우)출력하는 화면이다.

테이블이(가) 표시된 사진

자동 생성된 설명

1. 로그인

초기화면은 아래와 같다.



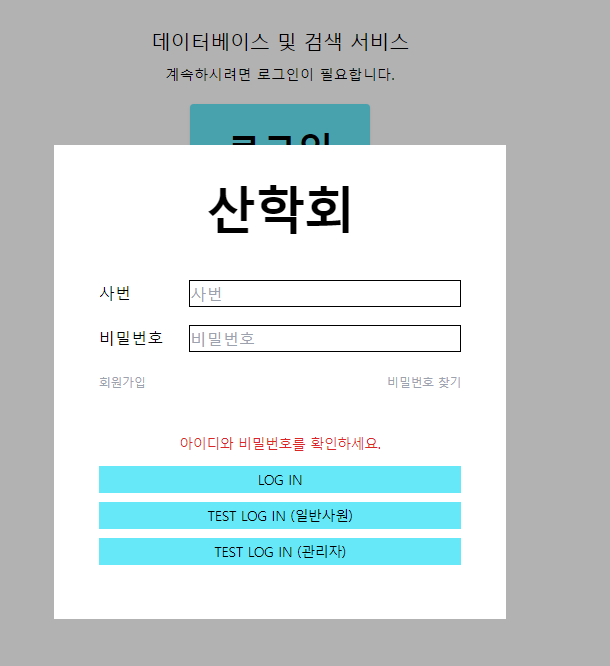
정상적으로 로그인 했을 시 아래와 같이 파일 검색 페이지로 이동하게 된다.

텍스트이(가) 표시된 사진

자동 생성된 설명

해당 페이지는 로그인 후 사용자가 보는 첫 메인 페이지이다.

로그인 화면에서 잘못된 값을 입력한 경우(DB에 사용자의 사번이 없는 경우 or 비밀번호를 잘못 입력한 경우) 아래와 같은 화면을 출력한다.



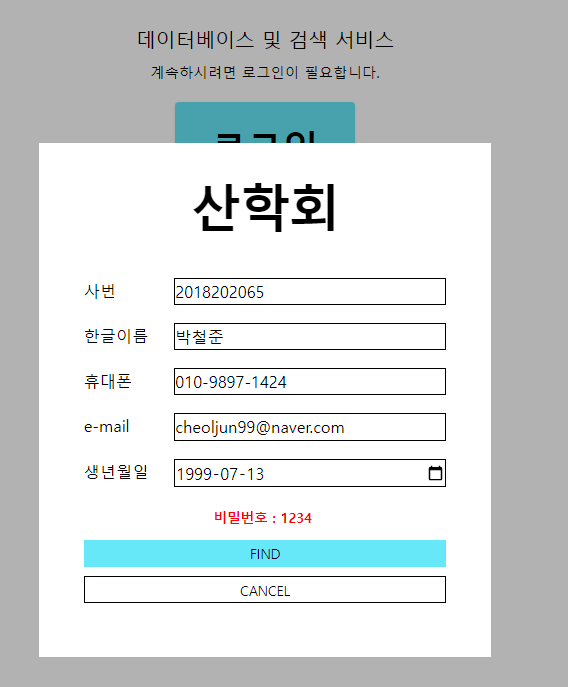
1. 비밀번호 찾기

초기화면은 아래와 같다.

테이블이(가) 표시된 사진

자동 생성된 설명

원하는 정보를 입력하여 FIND를 클릭시 아래와 같은 화면을 출력한다.

출력된 해당 비밀번호는 2초후 자동으로 지워지게 된다.

잘못된 값을 입력한 경우(로그인 서버에서 조회가 안될 경우 or 정보 입력을 누락한 경우) 아래와 같은 화면을 출력한다.



1. MY페이지 이동

MY페이지 배너를 클릭 시 아래와 같은 모달 화면을 볼 수 있다.

텍스트이(가) 표시된 사진

자동 생성된 설명

해당 모달 화면에 현재 로그인한 사용자의 비밀번호를 입력하게 되면 아래와 같이 MY페이지로 이동하게된다.



모달 화면에 잘못된 비밀번호를 입력한 경우 아래와 같은 화면을 출력한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

1. 개인정보수정

MY페이지에 접속하게 되면 로그인한 사용자의 정보를 아래와 같이 출력한다.

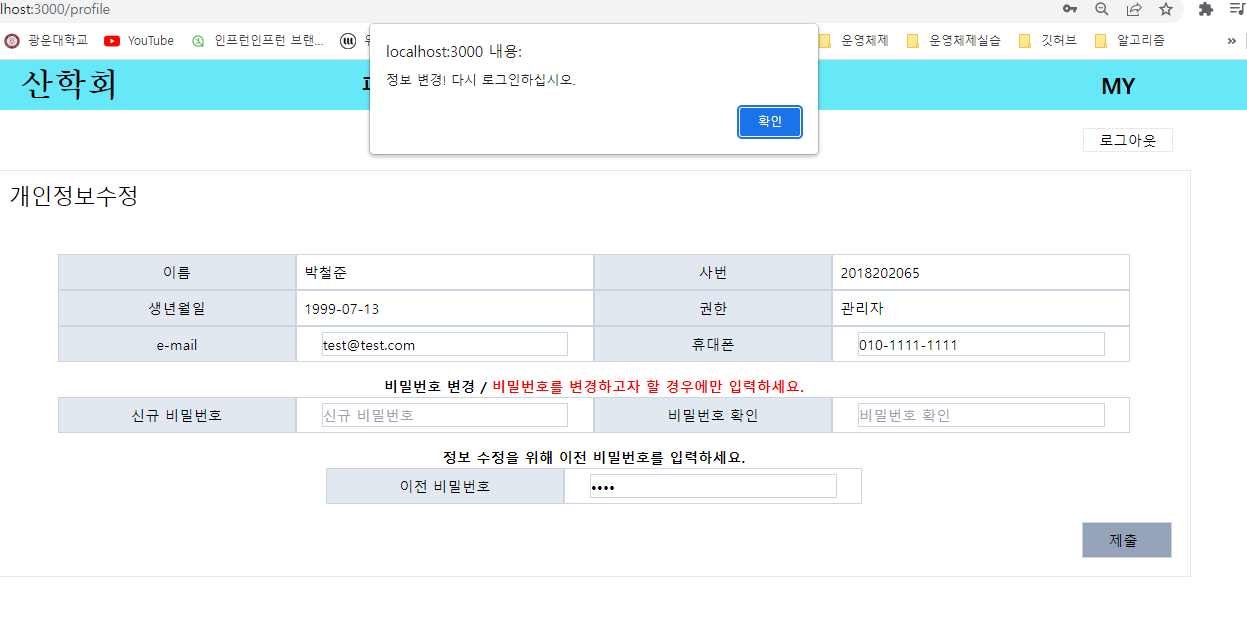
텍스트이(가) 표시된 사진

자동 생성된 설명

해당 사용자의 정보에서 수정할 수 있는 정보는 1. 이메일 2. 전화번호 3. 비밀번호로 방식은 이메일과 전화번호의 경우 변경하기 전 이전 비밀번호를 통해 사용자의 잘못된 클릭을 방지한다. 신규 비밀번호로 변경할 경우 확인 차 다시한번 신규 비밀번호를 입력받는다.

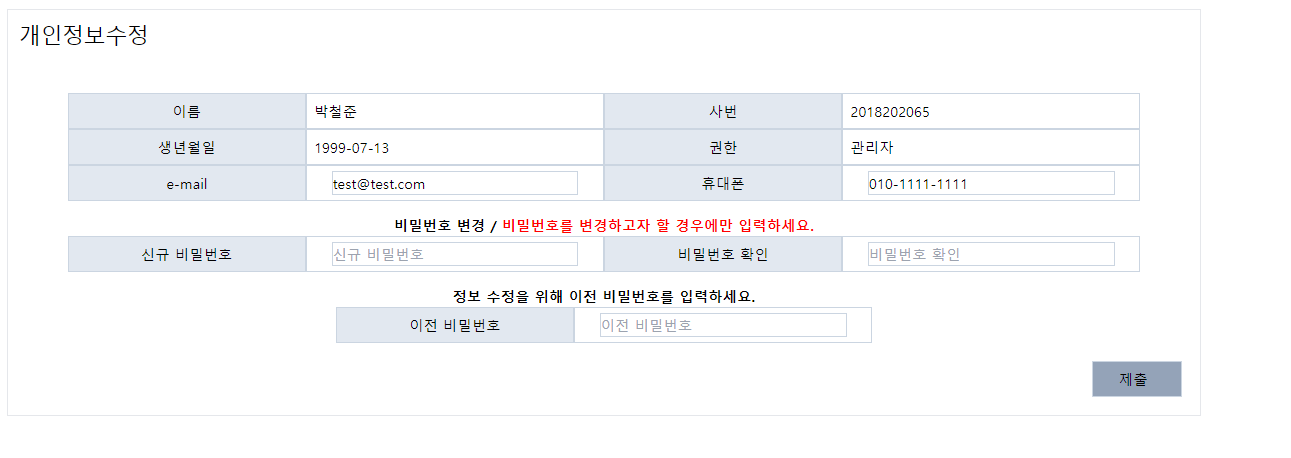
변경할 정보를 입력하여 제출을 누르면 아래와 같이 알림창이 발생한다.

변경할 정보(이메일: test@test.com 휴대폰: 010-1111-1111)



이후 사용자는 로그아웃 되고 다시 초기 화면으로 돌아가 다시 로그인할 수 있게 한다.

이후 다시 로그인 하여 확인하면 정보가 변경된 것으로 출력된다.



이전 비밀번호를 입력하지 않은 경우 아래와 같은 문장을 출력한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

신규 비밀번호와 신규 비밀번호 확인이 같지 않는 경우

텍스트이(가) 표시된 사진

자동 생성된 설명

* 로그인 기능 구현(프론트)

1. 변수 설정

로그인 테스트 백서버에 저장되어 있는 변수는 아래와 같다.

텍스트이(가) 표시된 사진

자동 생성된 설명

사용자의 입력을 프론트에서 서버로 넘겨줄 때 사용한 변수와 함수

텍스트이(가) 표시된 사진

자동 생성된 설명

서버로부터 받은 값을 프론트에서 저장할 때 사용하는 변수와 함수

텍스트이(가) 표시된 사진

자동 생성된 설명

로그인 기능을 사용하면서 이용되는 부가적인 변수와 함수

텍스트이(가) 표시된 사진

자동 생성된 설명

1. login함수

텍스트이(가) 표시된 사진

자동 생성된 설명

login함수의 경우 프론트에서 서버로 post요청을 통해 행해지는데 이때 서버 주소는 http://서버주소/auth/login의 경로로 요청을 보낸다. 이후 서버의 응답을 확인하는데 서버가 성공 응답을 보낼 때는 access\_token또한 같이 보내게 된다. 해당 토큰을 token 변수에 저장하기 위한 setToken함수를 실행하고 해당 토큰을 복호화 하여 adminkey에 저장되어있는 값을 확인하여 setAuth함수를 통해 권한을 부여한다. 이후 모달창을 종료하고 파일 검색 페이지로 이동한다.

1. Register함수

텍스트이(가) 표시된 사진

자동 생성된 설명

register함수의 경우 프론트에서 서버로 post요청을 통해 행해지는데 이때 서버 주소는 http://서버주소/auth/register의 경로로 요청을 보낸다. 이때 생성하기 위해 입력한 유저 정보 또한 같이 보낸다. 이후 서버의 응답을 확인하는데 성공일 경우 모달 창을 로그인 창으로 바꿔주고 실패한 경우 에러메시지를 출력 후 다시 입력받는다.

1. find함수(비밀번호를 찾는 기능)

텍스트이(가) 표시된 사진

자동 생성된 설명

Find함수의 경우 프론트에서 서버로 post요청을 통해 행해지는데 이때 서버 주소는 http://서버주소/auth/find의 경로로 요청을 보낸다. 이때 사용자에게 입력 받은 사용자 정보 또한 같이 보내게 되고 서버는 성공 응답과 비밀번호를 같이 보내게 되고 이것을 2초동안 모달창에 출력한다. 오류의 경우 에러 메시지를 출력하고 사용자에게 다시 입력 받는다.

1. check함수(MY페이지 접속 전 비밀번호 확인을 위한 함수)

텍스트이(가) 표시된 사진

자동 생성된 설명

check함수의 경우 프론트에서 서버로 post요청을 통해 행해지는데 이때 서버 주소는 http://서버주소/auth/check의 경로로 요청을 보낸다. 이때 로그인 후 저장하고 있던 사번 정보와 다시 한번 입력 받은 비밀번호를 서버로 넘겨 준다. 이때 저장하고 있던 토큰의 유효성도 검사하기 위해 토큰 또한 같이 보낸다. 성공 응답이 올 경우 MY페이지로 이동을 하고 모달을 종료한다. 실패 응답이 올 경우 에러 메시지를 출력 후 다시 사용자로부터 다시 입력 받는다.

1. MY페이지에서 개인정보 항목을 출력할 시

텍스트이(가) 표시된 사진

자동 생성된 설명

프론트에서 서버로 post요청을 통해 행해지는데 서버 주소는 http://서버주소/auth/check의 경로로 요청을 보낸다. 저장하고 있던 사번 정보와 토큰을 같이 보내게 되고 성공 시 서버로부터 받은 데이터를 set함수를 이용하여 유저 정보를 저장한다. 실패의 경우는 잘못된 토큰을 서버로 보냈을 경우 혹은 가장 높은 실패 확률은 저장하고 있던 토큰의 유효시간이 지난 후 서버로 보내 유효성 검사에서 이를 발견하였을 때이다.

* 임시 로그인 백서버 구현

1. 설정

텍스트이(가) 표시된 사진

자동 생성된 설명

1. 필요한 함수 선언

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 화면이(가) 표시된 사진

자동 생성된 설명

1. 미들웨어 정의