

포트폴리오

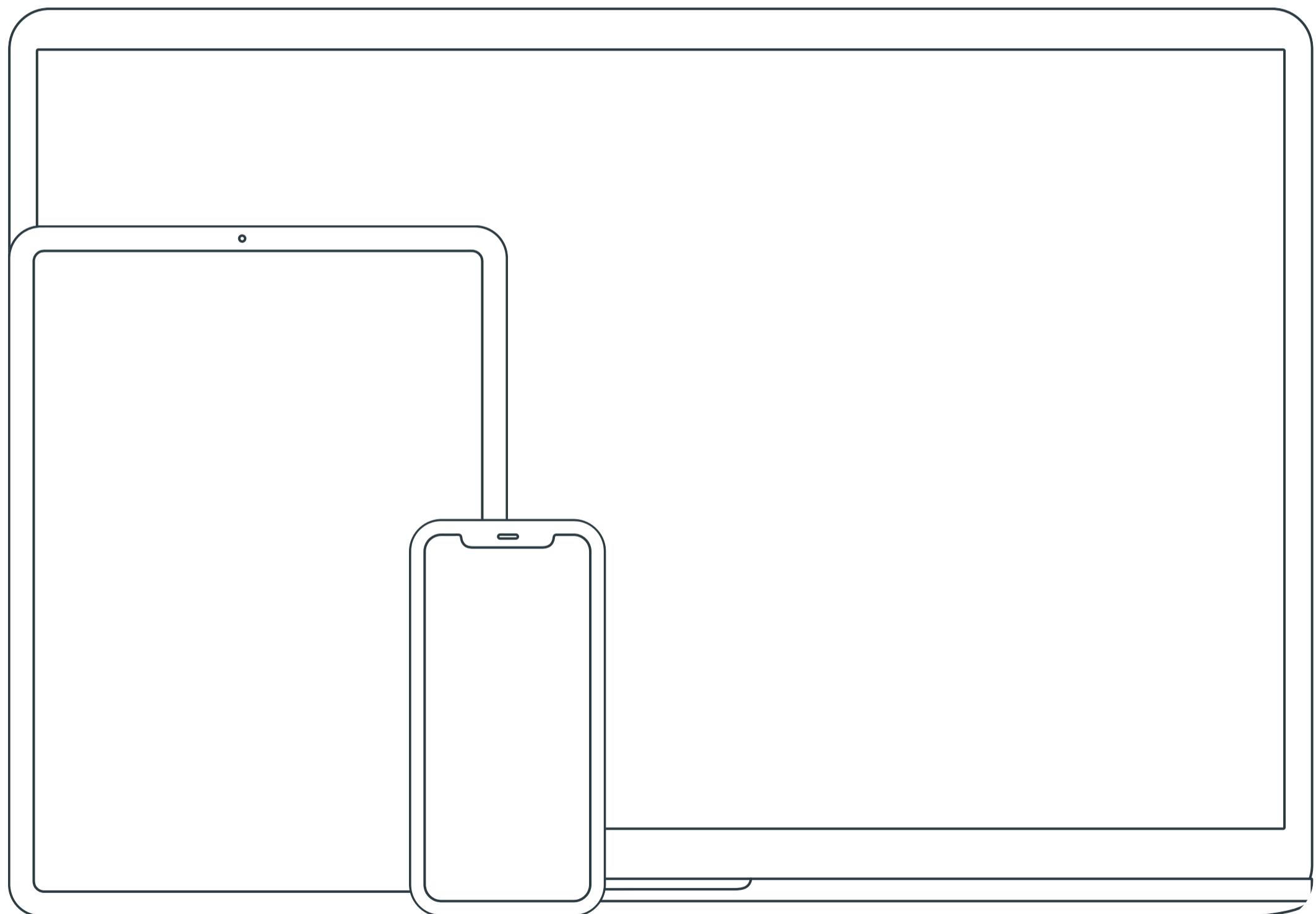
기업형 레이아웃 페이지

사용 스킬



사이트 기능

- Redux로 양방향 data 관리, Redux-Saga로 api 요청
- 게시판 data를 localstorage에 장하고 CRUD 기능
- Youtube Api data 관리
- Flickr Api data를 가져고 data 검색 기능
- 반응형 적용
- 컴포넌트 재사용 편의을 위해 폴더 분리
- 스크롤, 메뉴 등의 효과

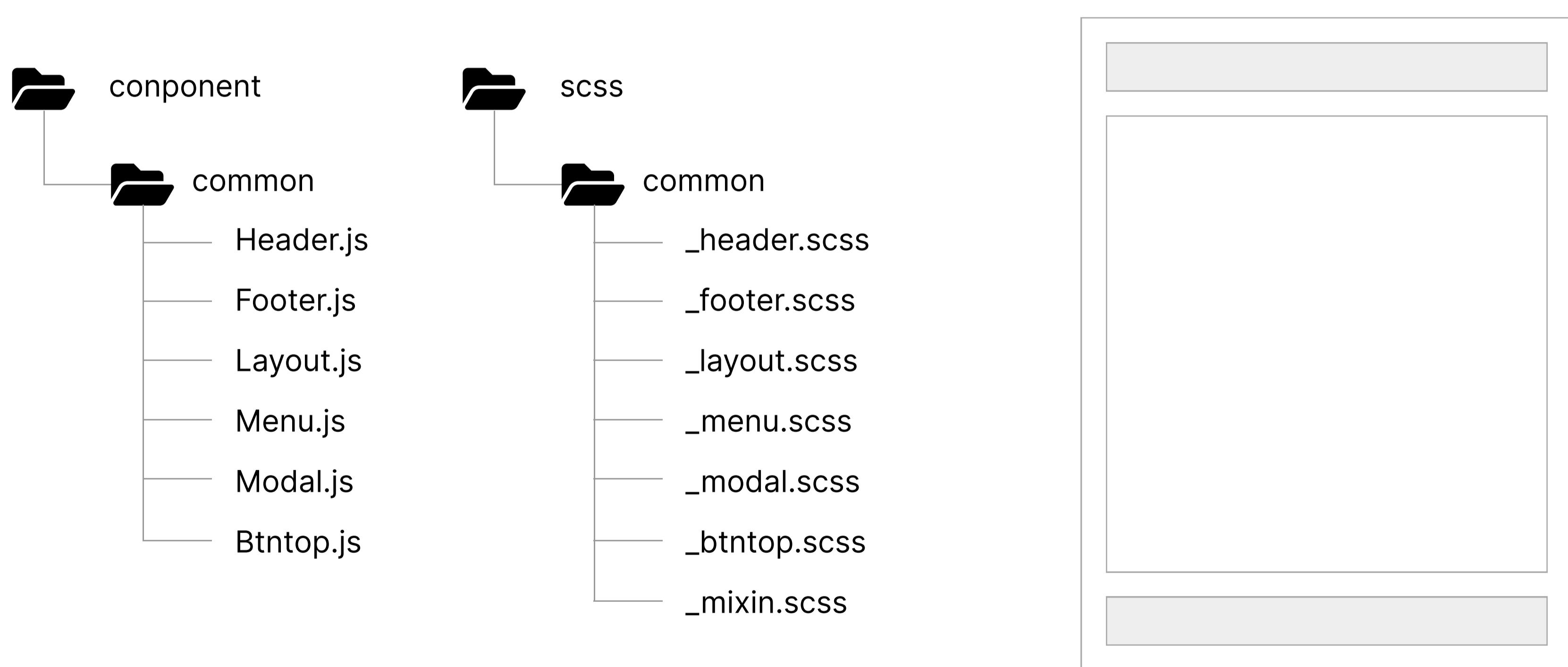


폴더 구조



공통 구조

Header, Footer, 효과, 공통 재사용 컴포넌트 등을 따로 각 common 폴더로 구성



메인 페이지 구조

main 공통 컴포넌트를 따로 만들고, 컨텐츠 기준 section으로 컴포넌트로 구성



component



main

Main.js

Visual.js

About.js

Company.js

Work.js

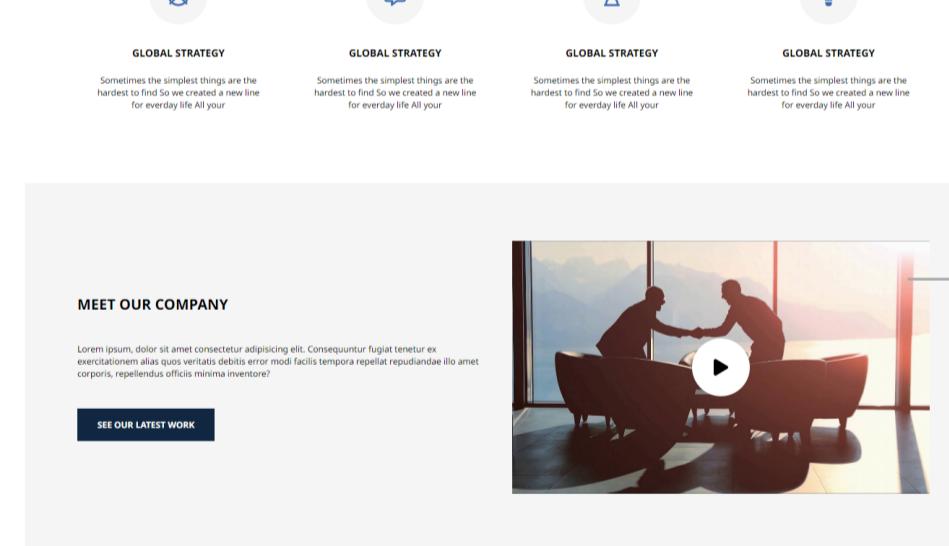
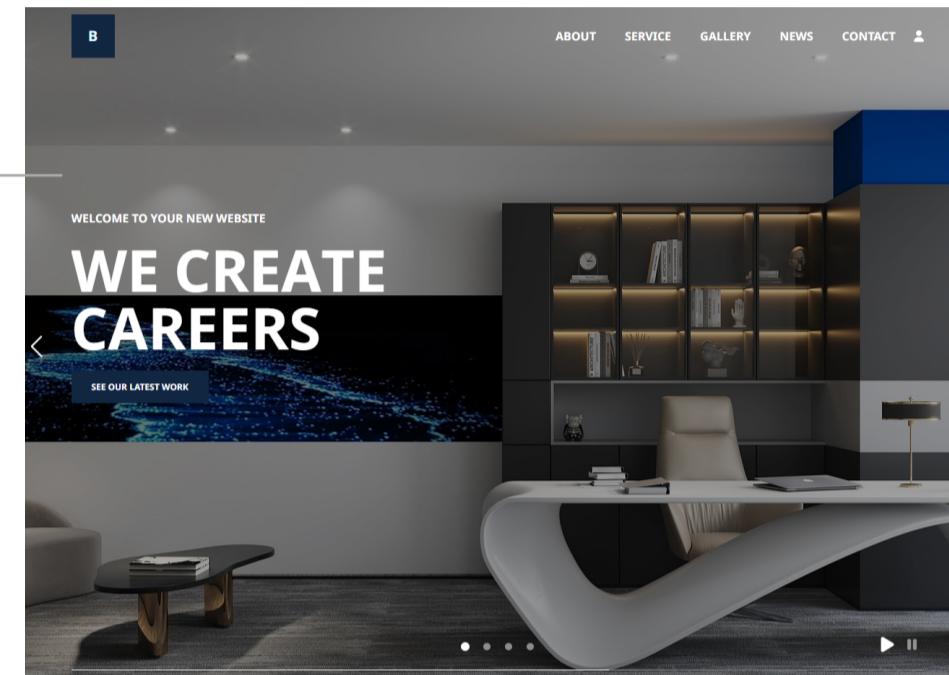
News.js

Contact.js

ScrollNav.js

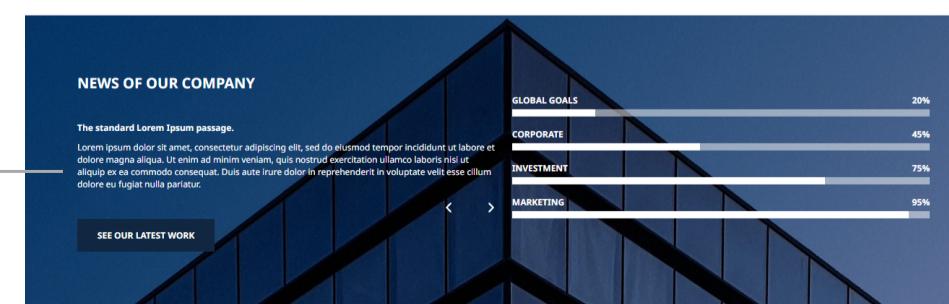
i Visual

- Visual.js / _visual.scss
- fade in - fade out 효과
- 이전, 다음 버튼 기능
- 페이지 버튼 기능
- 자동 재생 기능
- 진행바 모션 기능



i News

- News.js / _news.scss
- 가상 데이터를 만든 후 LocalStorage에 저장
- NEWS 페이지와 데이터 연동
- 슬라이드 기능
- 이전, 다음 버튼 기능



i Company

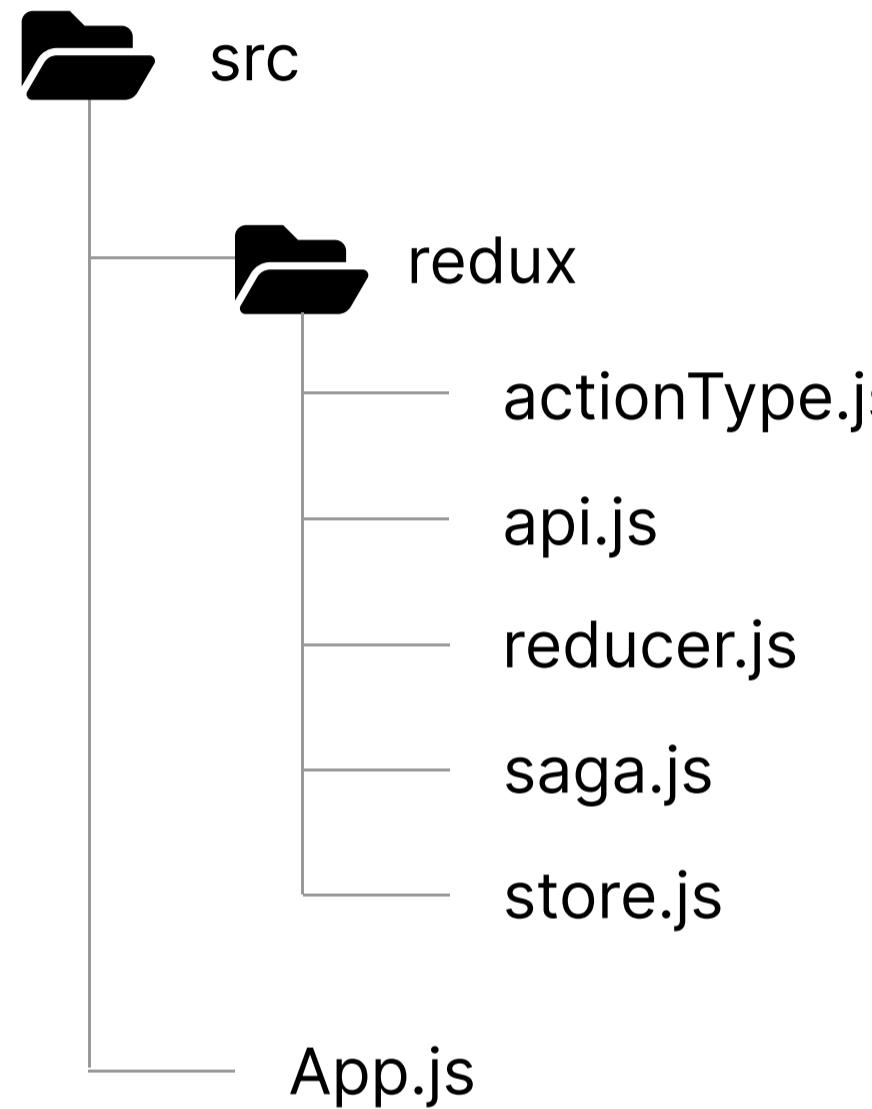
- Company.js / _company.scss
- Youtube Api 데이터 연동
- SERVICE 페이지와 데이터 같이 사용

i Work

- Work.js / _work.scss
- Flickr Api 데이터 연동
- GALLERY 페이지와 데이터 같이 사용

상태관리

Renux를 통해 전역으로 데이터 관리

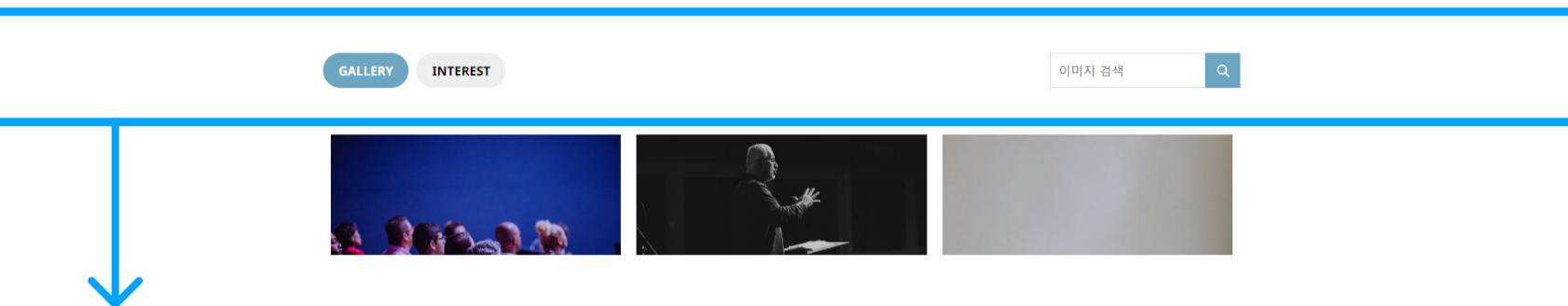
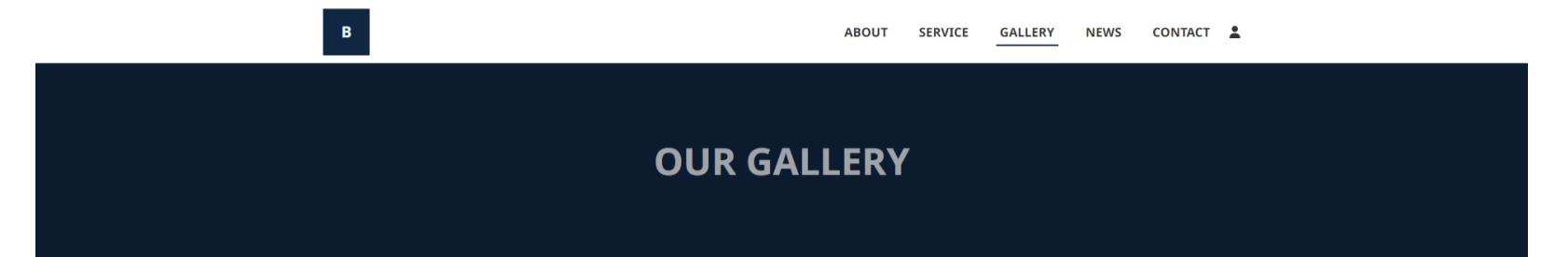


- App.js에서 useDispatch 적용
- 다른 컴포넌트에서도 데이터 연동

*App.js

```
const dispatch = useDispatch();

useEffect(() => {
  dispatch({ type: types.YOUTUBE.start });
  dispatch({ type: types.FLICKR.start, Opt: { type: 'user', user: 'user: my_id' } });
  dispatch({ type: types.MENU.close });
}, [dispatch]);
```



Redux Middleware - Redux Saga

- 외부 Api data를 호출
- GALLERY 페이지 검색어, 카테고리에 따라 데이터 요청
- redux-saga 적용 : Youtube, Gallery

Redux Saga code - Flickr

- saga.js에서 returnFlickr함수 call에서 action.Opt로 필요한 데이터 받음
- action.Opt 내용 : type, tags, user
- api.js - option을 받아서 data 요청

*saga.js

```
function* returnFlickr(action) {
  ...코드 중략...
  try {
    const response = yield call(getFlickr, action.Opt);
    ...코드 중략...
  }
}
```

*api.js

```
export const getFlickr = async (option) => {
  ...코드 중략...
  if (option.type === 'interest')
    ...코드 중략...
  return await axios.get(url);
};
```

LocalStorage 사용(1)

LocalStorage 이용한 CRUD 구현

NEWS 페이지 게시글 Create(생성), Read(읽기), Update(갱신), Delete(삭제) 기능 적용

The standard Lorem Ipsum passage.

EVENT

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

Read(읽기)

- LocalStorage에 data가 없으면 가상데이터 setItem로 data 추가, data가 있으면 getItem으로 조회

*News.js

```
const getLocalData = () => {
  const dummyPosts = [];
  const data = localStorage.getItem('post');
  if (data) return JSON.parse(data);
  else return dummyPosts;
};
```

Create(생성)

- LocalStorage data를 useState에 저장 setPosts로 기존 data에 추가
- setPosts({새게시글, ...기존게시글})

*Event.js

```
const [Posts, setPosts] = useState(getLocalData());

const creatPost = () => {
  setPosts([{ title: input.current.value, content: textarea.current.value },
  ...Posts]);
  resetForm();
};
```

Delete(삭제)

- 삭제 버튼을 누르면 setPosts로 기존 data 변경
- useEffect로 Post가 변경 될때마다 localStorage에 저장

*Event.js

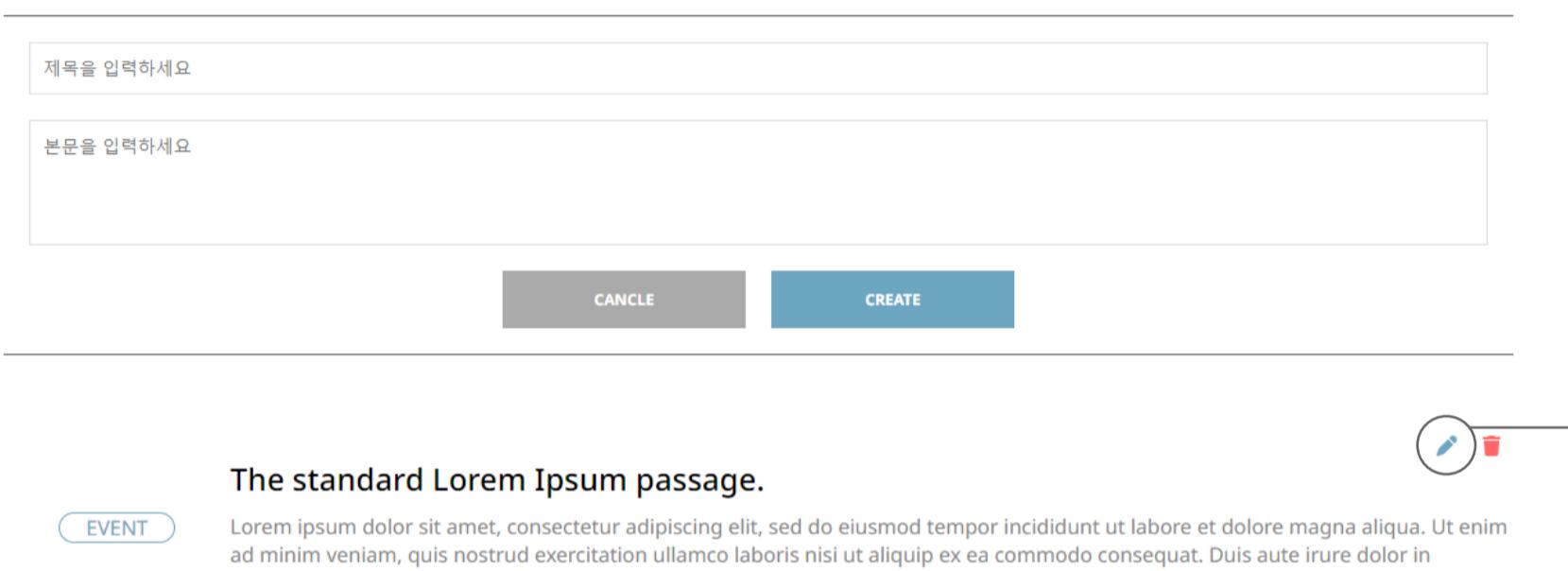
```
const deletePost = (index) => {
  setPosts(Posts.filter((_, idx) => idx !== index));
};

useEffect(() => {
  localStorage.setItem('post', JSON.stringify(Posts));
}, [Posts]);
```

LocalStorage 사용(2)

LocalStorage 이용한 CRUD 구현

NEWS 페이지 게시글 Create(생성), Read(읽기), Update(갱신), Delete(삭제) 기능 적용



수정 모드

- 수정 버튼을 누르면 enableUpdate 값에 따라 수정 모드로 변환

*Event.js

```
const enableUpdate = (index) => {
  setPosts(
    Posts.map((post, idx) => {
      if (idx === index) post.enableUpdate = true;
      return post;
    })
  );
};

return(
  {post.enableUpdate ? ( 수정 모드 ) : ( 출력 모드 )}
)
```

Update(갱신)

- 게시글의 index값으로 수정게시글을 구분, 해당 data를 불러옴
- 저장 버튼을 누르면 setPosts로 해당 input 값으로 변경하여 반환되며 저장, 출력 모드로 변환

*Event.js

```
const updatePost = (index) => {
  setPosts(
    Posts.map((post, idx) => {
      if (idx === index) {
        post.title = inputEdit.current.value;
        post.content = textareaEdit.current.value;
        post.enableUpdate = false;
      }
      return post;
    })
  );
};
```