



12주차 팀 주간회의

≡ 팀 주제	동국대 대관신청 시스템
≡ 팀명	5조 Prodigy Project
📍 유형	팀 주간 회의
📅 일시	@May 18, 2023
≡ 참석자	김유리, 홍범선, 홍서이
👤	



회의 주제

1. frontend - backend - databse 연결 문제 해결하기



회의 내용

0. 프로젝트 타임라인

Gantt Chart		3월~4월 2주	3주	4주	5월 1주	2주	3주	4주	6월 1주	2주	3주
일정							중간발표				최종발표
작업	담당자										
오픈소스 분석	김유리, 홍범선, 홍서이										
기존 컴포넌트 수정	김유리										
컴포넌트 구현	김유리										
기존 페이지 수정	김유리										
새 페이지 구현	김유리										
기능 구현	김유리, 홍범선										
API 설계	홍범선										
API 구현 및 테스트	홍범선										
front - db 연동	홍범선, 홍서이										
AWS 서버 연동 및 배포	홍범선										
DB 설계	홍서이										
DB 구현	홍서이										
발표준비	김유리, 홍범선, 홍서이										
피드백 및 수정	김유리, 홍범선, 홍서이										

1. 진행상황

홍서이(Database)

- 데이터베이스 수정
 - 관리자모드 사용을 위해 User 데이터베이스에 새로운 컬럼 type 추가
 - Room table의 floor 컬럼 내용 수정
 - 데이터베이스에 Building 테이블 추가하고 건물사진 데이터 추가
- 프로젝트 디렉토리에 Backend/DB 생성 후 데이터베이스 관련 코드 작성
 - express 이용해서 데이터베이스의 값을 원하는 형태로 가져오는 코드 작성

```

서버가 3000번 포트에서 실행 중입니다.
[
  { name: '과학관', rooms: [ [Object] ] },
  {
    name: '명진관',
    rooms: [
      [Object], [Object], [Object],
      [Object], [Object], [Object],
      [Object], [Object], [Object],
      [Object], [Object], [Object],
      [Object], [Object], [Object],
      [Object], [Object]
    ]
  },
  {
    name: '법학/만해관',
    rooms: [
      [Object], [Object],

```

홍범선(Backend)

- 건물, 강의실 정보를 데이터베이스로부터 받아와 프론트에 전송
- axios 오류 수정
 - porxy설정으로 데이터 전송되는 것 확인했었으나 다른 컴포넌트에서는 제대로 적용되지 않아 axios인스턴스 생성하여 설정 추가하는 방식으로 오류 수정

```
Front-End > utils > .js api.js > api > baseUrl
1  import axios from "axios";
2
3  const api = axios.create({
4    baseUrl: "http://localhost:5000/api", // 프록시 서버의 주소를 baseUrl로 설정
5  });
6
7  export default api;
```

- 이후 데이터 전송되어 건물명, 건물사진, 강의실정보 등 데이터 전송되고 프론트에 반영 됨
- 건물 이미지가 안뜨는 오류 발생하여 수정
 - next.config.js파일의 image 설정에 i.ibb.co 도메인을 추가

```
    ],
    images: {
      domains: ['i.ibb.co'],
    },
  },
}
```

김유리(Frontend)

- 회원가입 페이지 제작

회원가입

자격

이름

학번/ID

비밀번호

이메일

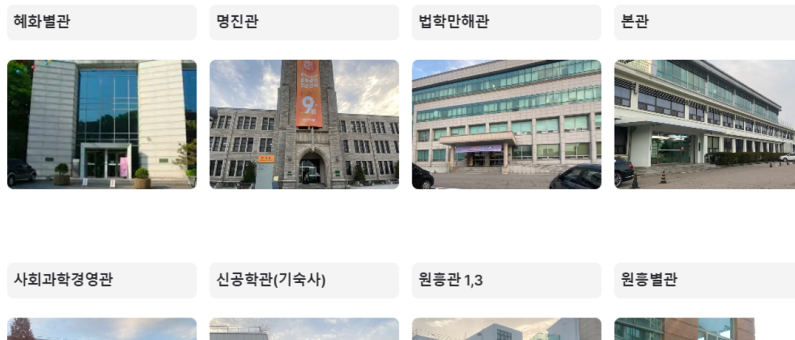
전화번호

- 건물 서버에서 불러와서 동국대학교 건물로 수정

2023/05/20

Buildings

[지도로 보기](#)



- 강의실 리스트 임의 배열로 되어있던 것 json형태로 받도록 코드 수정 후 서버에서 받아옴
- 상태관리 라이브러리 Redux를 이용하여 선택한 날짜 저장하도록 설정

```
import { configureStore, createSlice } from "@reduxjs/toolkit";

// 초기 상태
const initialState = {
  |   selectedDate: new Date(),
};

// createSlice를 사용하여 액션과 리듀서를 생성
const dateSlice = createSlice({
  |   name: "date",
  |   initialState,
  |   reducers: {
  |     setSelectedDate: (state, action) => {
  |       |   state.selectedDate = action.payload;
  |     },
  |   },
});

// 액션 생성자 함수를 추출
export const { setSelectedDate } = dateSlice.actions;

// rootReducer를 생성
const rootReducer = dateSlice.reducer;

// 스토어 생성
const store = configureStore({
  |   reducer: rootReducer,
});

export default store;
```

2. 앞으로의 계획

김유리(Frontend)

- 연동 코드 작성-서버에 데이터 전송 코드 작성(form 에서)
- 관리자 모드 페이지 제작
- 예약 시간 선택하는 체크박스

홍범선(Backend)

- 프론트에서 전송된 데이터(예약시간, 예약정보 등) 데이터베이스로 전달하는 코드 작성
- AWS배포 준비

홍서이(Database)

- 프론트엔드에서 유저가 입력한 정보를 데이터베이스에 저장하는 코드 작성하기
- README.md 파일 작성하기

3. 월요일 교수님 피드백 시간

- 발표자 : 홍서이
 - 회의록 내용 발표
- 서기 : 김유리

→ 다음 회의 내용

- 개인별 진행 상황 공유
- 5/22(월) 교수님 피드백 내용에 대한 내용 논의