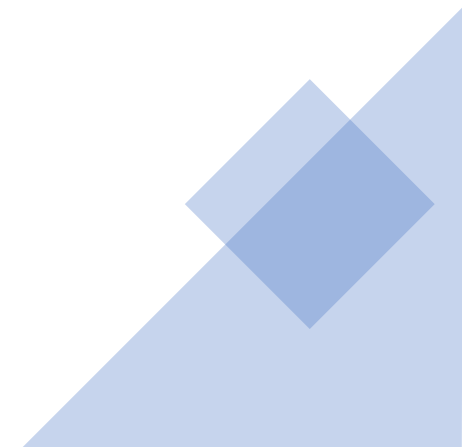


5월 16일 발표자료 - iOS

1871069 김진웅



목차

- 1. 한 주간의 진행 상황
 - 한 주 간의 진행상황
 - 부딪힌 이슈 및 해결 사항
 - 2. 고민하고 있는 issue
 - 컬렉션 셀을 이용한 캘린더의 일정을 삽입하기
 - 3. 앞으로의 계획
- 

1. 한 주간의 진행상황

- DB로부터 받은 주요 데이터 저장

- UserDefaults

- Token, userId, password

- Schedule

- Format을 struct화 하여 배열에 format들을 저장
 - DB -> Schedule -> Filtered...
 - Filter : 각 날짜에 맞는 일정들

- Group

- Format을 struct화 하여 배열에 저장

1. 한 주간의 진행상황

- 일정 UI

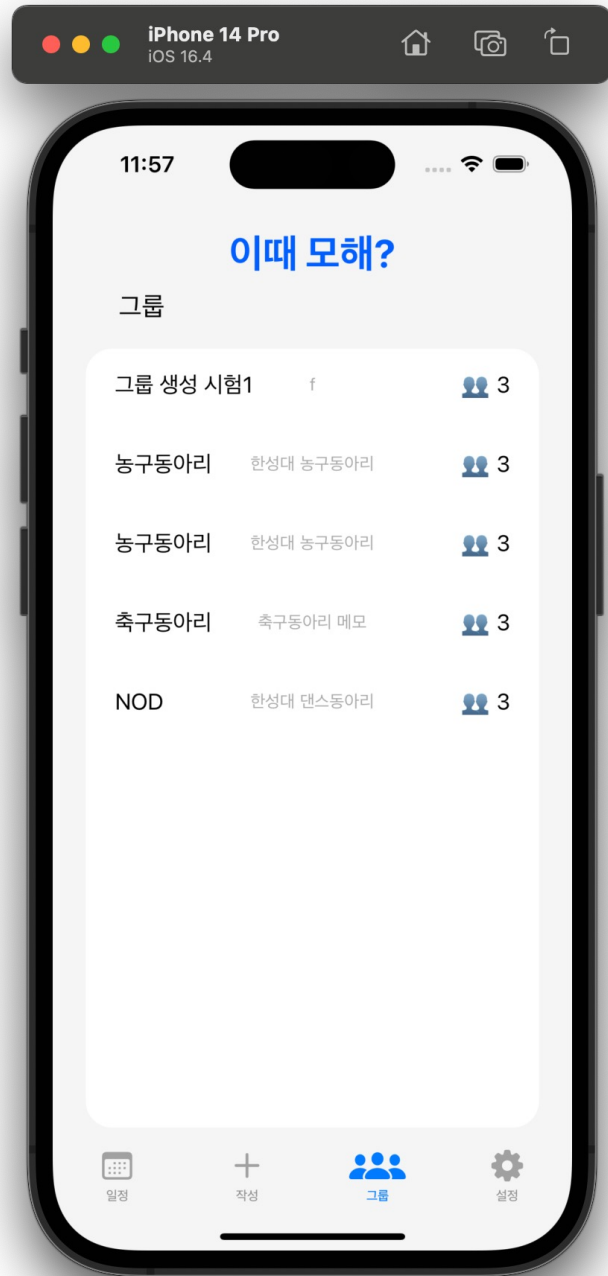
- 일정을 리스트 형태로 띄우고
- 다시 일정 수정 모달에 데이터 맵핑 완료
- 현재 update, delete에 대한 로직 처리만 남았음.



1. 한 주간의 진행상황

- 그룹

- 그룹의 리스트를 가져와 리스트 형태로 출력 완료
- 각 그룹을 클릭할 시에 반응하는 작업 진행 중 (그룹의 일정 띄우기 등)



1. 한 주간의 진행상황

- 부딪힌 이슈 + 해결사항

- 서버에서 데이터를 받기 위해 다음과 같은 함수가 실행되었을 때,
- `Group.shared2.groups`가
 - 그룹 리스트를 저장하는 배열
- 초기화되는 문제가 발생

```
ss GroupViewController: UIViewController {
    override func viewWillAppear(_ animated: Bool) {
        server.getAllData(requestURL: "group", token: UserDefaults.standard.string(forKey:
            "token"))! { (data, response, error) in
            if let error = error {
                print("Error: \(error)")
                return
            }

            if let data = data {
                if let jsonObject = try? JSONSerialization.jsonObject(with: data, options:
                    []),
                    let json = jsonObject as? [String: Any],
                    let dataArray = json["data"] as? [[String: Any]] {
                    for dataEntry in dataArray {
                        if let originKey = dataEntry["originKey"] as? String {
                            let groupName = dataEntry["groupName"] as? String ?? ""
                            let description = dataEntry["description"] as? String ?? ""
                            let numOfUsers = dataEntry["numOfUsers"] as? Int ?? 0
                            let leaderId = dataEntry["leaderId"] as? String ?? ""
                            let groupUsers = dataEntry["groupUsers"] as? [String] ?? []

                            let groupData = Group.Format(
                                originKey: originKey,
                                groupName: groupName,
                                description: description,
                                numOfUsers: numOfUsers,
                                leaderId: leaderId,
                                groupUsers: groupUsers

                            )

                            Group.shared2.updateGroupData(data: groupData)
                            print(">>>: \(Group.shared2.groups)")
                        }
                    }
                }
            }
        }
    }
}
```

1. 한 주 간의 진행상황

- 부딪힌 이슈 + 해결사항

- 서버에서 데이터를 받을 때, 비동기적으로 동작
- 허나 View를 구성하는 과정에서, CollectionView의 Cell을 update할 때 비동기로 동작하는 데이터 로드 과정과 달리 동기적으로 동작하여 빈 배열을 먼저 접근
- 추후에 배열에 서버로부터 받은 데이터를 update 하는 과정을 거치는 것을 확인하였으며,
- 'DispatchQueue.main.async' 구문을 삽입하여 서버의 데이터를 받았을 시에, CollectionView를 다시 update 할 수 있도록 하였음.

```
}  
DispatchQueue.main.async {  
    self.groupListView.reloadData() // 데이터 업데이트 후 UICollectionView  
    다시 로드  
}  
}
```

2. 고민하고 있는 사항

- 그룹의 타임 테이블을 보여줄 View에서 배경을 이미지로 두고, 막대 바를 좌표로 삽입할 것인지,
- CollectionView의 셀을 구성하여 (시간표 형태) 셀 안의 일부 영역을 채우는 것으로 할지,
- 현재 CollectionView를 구성하기로 결정하고, 금일 구현해볼 예정

3. 앞으로의 계획

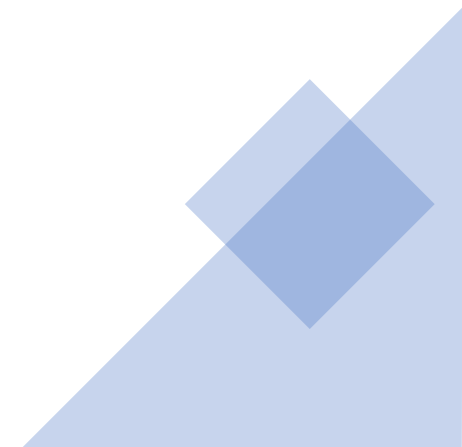
- 일정 Update, Delete 연산 완성
- 그룹의 Time Table View 완성
- 해당 그룹이 만든 일정만을 확인하는 리스트 뷰 만들기

5월 16일 발표자료 - Web

1871062 김정한



목차

- 1. 한 주간의 진행 상황
 - 1-1. 어떠한 것을 공부했는가
 - 1-2. 부딪힌 이슈 & 해결사항
 - 2. 고민하고 있는 issue
 - 3. 앞으로의 계획
 - 4. 기타
- 

1. 한 주간의 진행 상황

- 1-1. 어떠한 것을 공부했는가
 - 그룹 모달 구현

그룹 추가

```
<div style={{ display: 'flex', justifyContent: 'center' }}>  
  <Button  
    variant="contained"  
    onClick={handleClick}  
  >  
    그룹 추가  
  </Button>  
</div>
```

그룹 추가 버튼 클릭시 handleClick 핸들러 함수 작동

```
const [open, setOpen] = useState(false); // true면 모달 열림, false면 모달 닫힘  
const [editMode, setEditMode] = useState(false) // 수정할지 말지 알려줌  
const handleClick = () => {  
  setOpen(true)  
}
```

handleClick 핸들러에서 open의 상태를 true로 변경

그룹 생성 ?

그룹이름 *

메모

사용자 검색하기

그룹 추가

그룹모달 생성 완료

1. 한 주간의 진행 상황

• 1-1. 어떠한 것을 공부했는가

- 그룹 정보 가져오기 완료
- 그룹 추가하기 완료

▼ Group CRUD

- Create **POST**

```
// 그룹 생성 요청, POST
/group
{
  "groupName": "한성대 동아리",
  "description": "2023학년도 새로운 동아리",
  "numOfUsers": 3,
  "leaderId": "terry8408@gmail.com",
  "groupUsers": [ "terry8408@gmail.com" ]
}

// 그룹 생성 응답
{
  "originKey": "4028cb8187eb6b29",
  "groupName": "한성대 동아리",
  "description": "2023학년도 새로운 동아리",
  "numOfUsers": 3,
  "leaderId": "terry8408@gmail.com",
  "lastModifiedAt": "2023-05-05T19:30",
  "createdAt": "2023-05-05T19:30",
  "groupUsers": [
    "terry8408@gmail.com",
    "aaa@gmail.com",
    "bbb@gmail.com"
  ]
}
```

그룹 프로토콜

이때 모해?

이윤재 님 | 로그아웃

그룹 추가 Button

그룹 생성 시험1	그룹 메모 시험1	2
그룹 생성 시험1	f	3
농구동아리	한성대 농구동아리	3
농구동아리	한성대 농구동아리	3
축구동아리	축구동아리 메모	3
그룹 생성 시험1	f	2
농구동아리	한성대 농구동아리	2
농구동아리	한성대 농구동아리	2
펜싱동아리	한성대 펜싱동아리	2
NOD	한성대 댄스동아리	3

group Name

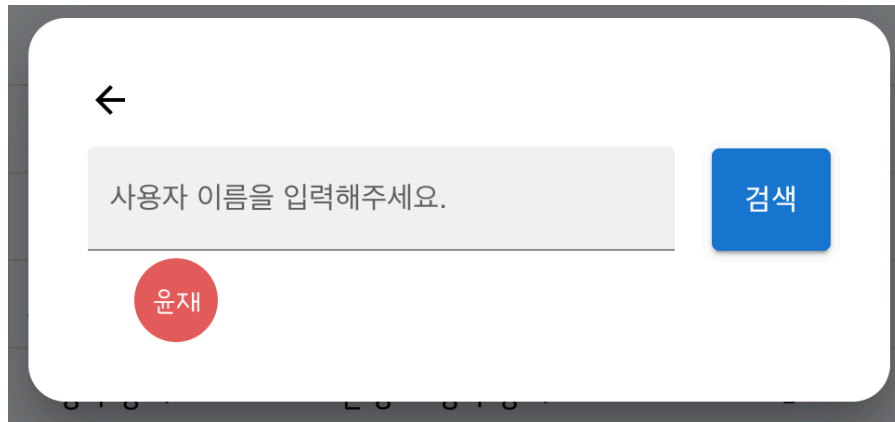
NumOfUsers

Description

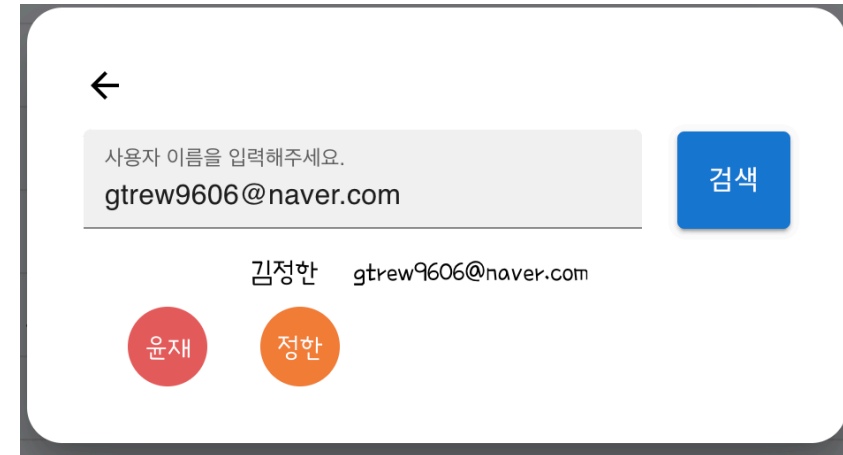
1. 한 주간의 진행 상황

• 1-1. 어떠한 것을 공부했는가

- 사용자 검색하기 로직 구현 완료



사용자 검색하기 모달

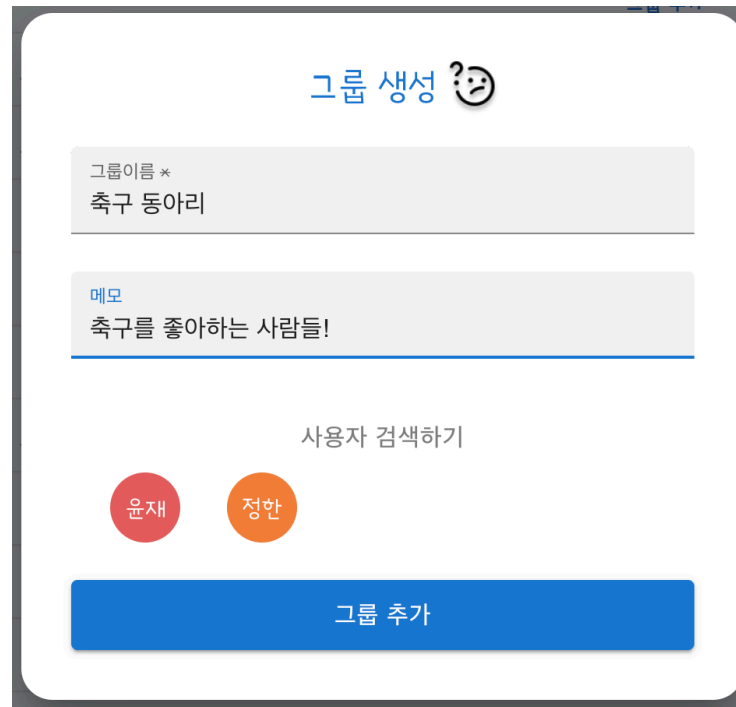


다른 사용자를 추가할 경우 그 사람의 이름이 적힌
아이콘이 다른 색상으로 추가.
아이콘은 클릭하면 그 아이콘은 삭제(본인 제외)

1. 한 주간의 진행 상황

• 1-1. 어떠한 것을 공부했는가

- 사용자 검색하기 로직 구현 완료



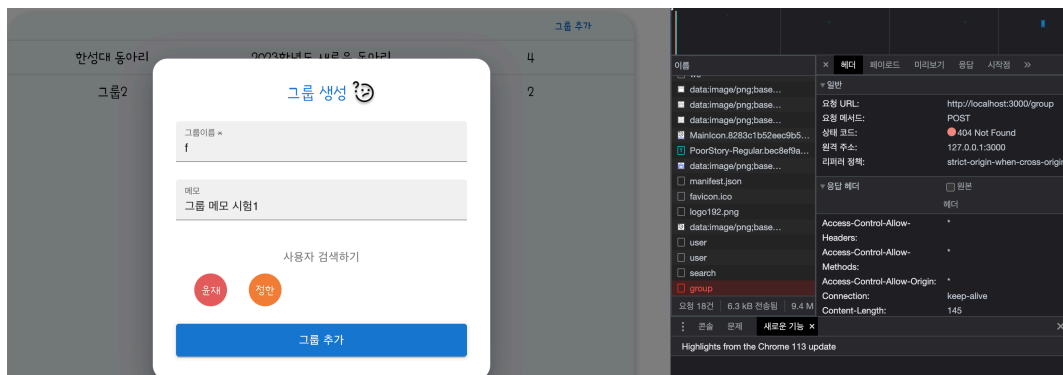
A screenshot of a 'Group Creation' modal window. At the top, it says '그룹 생성 ?😊'. Below this, there are two input fields. The first is labeled '그룹이름 *' and contains the text '축구 동아리'. The second is labeled '메모' and contains the text '축구를 좋아하는 사람들!'. Below the input fields, there is a section titled '사용자 검색하기' which displays two user avatars: one red circle with the name '윤재' and one orange circle with the name '정한'. At the bottom of the modal is a large blue button labeled '그룹 추가'.

추가한 사용자 아이콘이 그룹 생성 모달에 남아있다.

1. 한 주간의 진행 상황

• 1-2. 부딪힌 이슈 & 해결사항

- 서버로 데이터 전송 시 나타나는 404오류 해결



요청 url을 보니 서버에서 요청한 url이 아닌 localhost로 설정되어있다.

```
1 import axios from "axios"
2 import cookie from "js-cookie";
3
4 const instance = axios.create({ //동일 형태로 url 설정
5   baseURL: 'http://43.201.251.14:8080'
6 })
7
8 //인터셉터
9 instance.interceptors.request.use(function (config) {
10   const token = cookie.get('token') //쿠키의 토큰값을 가져와서 header에 넣어줌
11   console.log('token', token)
12   config.headers = {
13     Authorization: `Bearer ${token}` //Authorization에 token값 넣기
14   }
15
16   // console.log(config)
17   //요청이 전달되기 전에 작업 수행
18   return config;
19 })
20
21 // instance.interceptors.response.use(console.log, console.log)
22
23 export default instance
```

우리는 직접 작성한 baseURL을 사용하고 있다.

```
.import React, { useState } from 'react';
.import { Modal, Box, TextField } from '@mui/material';
.import Button from '@mui/material/Button'
.import MiniIcon from '../images/MiniIcon.png';
.import axios from 'axios';
.import cookie from 'js-cookie'
.import SearchModal from './SearchModal.jsx';
.import Avatar from './Avatar';
.import { COLOR_CODE_LIST } from '../constants';
```

따라서 axios를 import 해오는 경로가 axios 가 아니라 ../../axios.js 여야 한다.

1. 한 주간의 진행 상황

• 1-2. 부딪힌 이슈 & 해결사항

- url이 달라짐에 따라 아이콘 사진을 바꾸는 법 + 비슷한 url을 하나로 묶는 방법 해결

Url이 /groups/: 로 시작하는 경우 + /groups 일 경우는 전부 파란 아이콘으로 변경

```
<div style={{ display: 'flex', columnGap: '10px' }}>
  <img src={tab === '/calendar' ? CalendarTab : Calendar} style={{ ...headerStyle }} onClick={handleClickCalendarIcon} />
  <img src={tab === '/groups' || tab.startsWith('/groups:') ? GroupTab : Group} style={{ ...headerStyle }} onClick={handleClickGroupIcon} />
  <img src={tab === '/settings' ? SettingTab : Setting} style={{ ...headerStyle }} onClick={handleClickSettingIcon} />
</div>
```

삼항연산자 + startsWith 함수를 이용해 해결.

localhost:3000/groups/:groupId



이때 모해?

이윤재 님 | 로그아웃



1. 한 주간의 진행 상황

• 1-2. 부딪힌 이슈 & 해결사항

- 반응형 웹 방식으로 브라우저의 크기 변경 시에도 컴포넌트의 위치가 고정되어있게 만들기

```
<div style={{ display: 'flex' }}>  
  <div style={{ margin: '10px', width: '33%', textAlign: 'center' }}>{group.groupName}</div>  
  <div style={{ margin: '10px', width: '34%', textAlign: 'center' }}>{group.description}</div>  
  <div style={{ margin: '10px', width: '33%', textAlign: 'center' }}>{group.numOfUsers}</div>  
</div>
```

width값을 px같은 단위가 아닌 %로 표현해주면 브라우저의 크기가 변경되어도 해당 브라우저 크기의 %만큼의 위치에 고정되어있다.

그룹 추가

한성대 동아리

2023학년도 새로운 동아리

4

그룹2

그룹 2입니당

2

그룹 추가

한성대 동아리

2023학년도 새로운 동아리

4

그룹2

그룹 2입니당

2

브라우저의 크기가 바뀌어도 컴포넌트 고정!

2. 고민하고 있는 이슈들

- 2-1.

- Group Detail화면과 Notification을 받아오는 로직에 대한 고민
- Sse emitter 를 받아오는 형식에 대한 공부 필요함

3. 앞으로의 계획

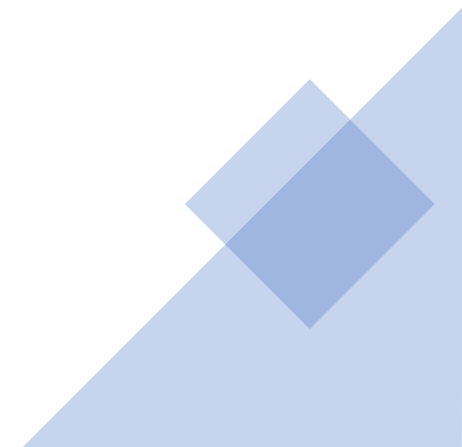
- Group Detail 페이지, 그룹 일정 추가 기능, 그룹 일정 열람 기능 추가 예정
- Notification 알람 구현 예정
- Server와 최종 테스트 예정

5월 16일 발표자료 - Server

1811072 유영재
1871197 이윤재



목차

- 1. 한 주간의 진행 상황
 - 1-1. 어떠한 것을 공부했는가
 - 1-2. 부딪힌 이슈 & 해결사항
 - 2. 앞으로의 계획
- 

1. 한 주간의 진행상황

- 1-1. 어떠한 것을 공부했는가
 - Group CRUD 구현 완료

Group Create

```
no usages  genius +1
@PostMapping("/createRequest")
public ResponseEntity<?> createGroup(@AuthenticationPrincipal String userId, @RequestBody GroupDTO dto)
```

Group Retrieve

```
no usages  genius +1
@GetMapping
public ResponseEntity<?> retrieveGroup(@AuthenticationPrincipal String userId)
```

Group Update

```
@PutMapping
public ResponseEntity<?> updateGroup(@AuthenticationPrincipal String userId, @RequestBody GroupDTO dto)
```

Group Delete

```
@DeleteMapping
public ResponseEntity<?> deleteGroup(@AuthenticationPrincipal String userId, @RequestBody GroupDTO dto)
```

1. 한 주간의 진행상황

- 1-1. 어떠한 것을 공부했는가
 - Group Schedule CRUD 구현 완료

Group Schedule Create

```
@PostMapping  
public ResponseEntity<?> createGroupSchedule(@RequestBody GroupScheduleDTO dto)
```

Group Schedule Retrieve

```
@PutMapping  
public ResponseEntity<?> updateGroupSchedule(@RequestBody GroupScheduleDTO dto)
```

Group Schedule Update

```
@GetMapping  
public ResponseEntity<?> retrieveGroupSchedule(@RequestBody GroupScheduleDTO dto)
```

Group Schedule Delete

```
@DeleteMapping  
public ResponseEntity<?> deleteGroupSchedule(@RequestBody GroupScheduleDTO dto)
```


1. 한 주간의 진행상황

- 1-1. 어떠한 것을 공부했는가
 - SSE emitter를 이용한 실시간 알림 구현

푸시 알림 예시



실시간 알림을 보내는 경우

1. 그룹 생성 시 => 그룹원에게 그룹 초대 알림

ex) `entity.getLeaderId() + "님이 당신을 " + entity.getGroupName() + " 그룹에 초대하셨습니다."`

2. 그룹 일정 생성 => 그룹원에게 그룹 일정 생성 알림

3. 그룹 일정 수정 => 그룹원에게 그룹 일정 변경 알림

4. 그룹 일정 삭제 => 그룹원에게 그룹 일정 삭제 알림

Ex) `group.getGroupName() + "그룹에서 " + groupScheduleEntity.getName() + " 일정을 생성/수정/변경하였습니다."`

1. 한 주간의 진행상황

- 1-1. 어떠한 것을 공부했는가
 - SSE emitter를 이용한 실시간 알림 구현

```
@PostMapping("/auth/signin")
public ResponseEntity<?> authenticate(@RequestBody UserDTO userDTO){
    UserEntity user = userService.getByCredentials(userDTO.getUserId(), userDTO.getPassword(), passwordEncoder);

    if(user != null) {
        final String token = tokenProvider.create(user);
        log.info("token create!");
        log.info(token);
        final UserDTO responseUserDTO = UserDTO.builder()
            .userId(user.getUserId())
            .userName(user.getUserName())
            .token(token)
            .build();

        // 사용자가 로그인하면, Emitter 만들기
        SseEmitter emitter = emitterService.subscribe(userDTO.getUserId());
        try {
            emitter.send(SseEmitter.event()
                .name("login")
                .data(responseUserDTO));
        } catch (IOException e) {
            throw new RuntimeException(e);
        }
        return ResponseEntity.ok().body(emitter);
    }
}
```

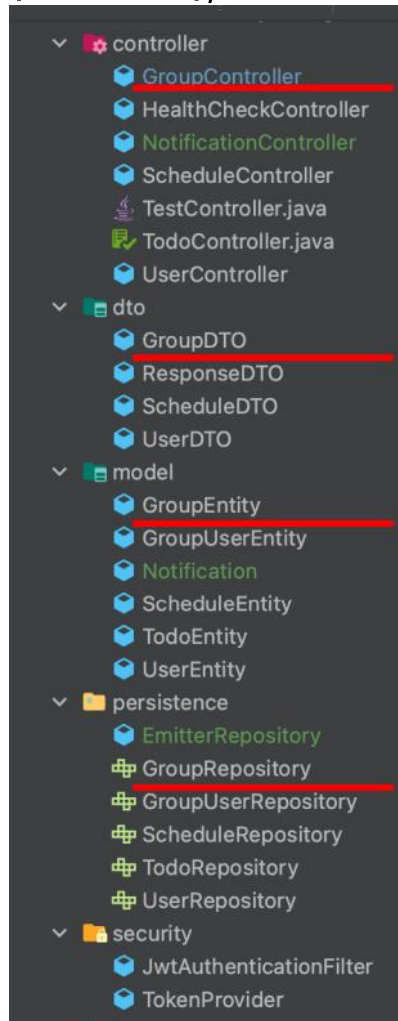
로그인 시 SSE Emitter 생성

Emitter로 event를 보내는 부분

생성 후 반환 하여 SSE Emitter 연결 설정

1. 한 주간의 진행상황

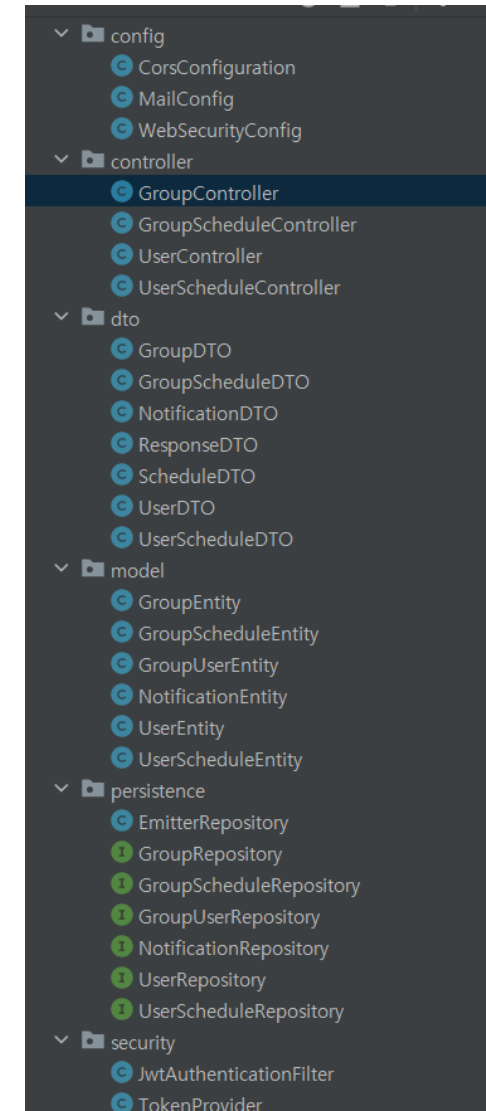
- 1-1. 어떠한 것을 공부했는가
 - 클래스 정리, Protocol 정리



쓸모 없는 Controller 정리



클라이언트로 다양한 형태의 데이터를 보내 주기 위해 DTO 추가



일정 조회 응답 => Response DTO에 담아서 보냄

1. 한 주간의 진행상황

- 1-1. 어떠한 것을 공부했는가
 - 클래스 정리, Protocol 정리

클라이언트 응답용 데이터 구조
Response DTO

```
public class ResponseDTO<T> {  
    // DTO가 여러개의 형태일 수 있으므로 List 형태로  
    no usages  
    private List<T> data;  
  
    // 성공, 실패 시 String으로 succeed, failed 전송  
    no usages  
    private String status;  
  
    no usages  
    private String error;  
}
```

```
"data" : [  
  {  
    "originKey" : "어쩌구저쩌구",  
    "name" : "밥먹기",  
    "startAt" : "2023-03-17-19:11",  
    "endAt" : "2023-03-17-19:11",  
    "userId" : "abc@naver.com",  
    "memo" : "오늘 저녁은 고기!",  
    "notification" : 1,  
    "allDayToggle" : 1,  
    "createdAt" : "2023-03-17-19:11",  
    "lastModifiedAt" : "2023-03-17-19:11"  
  }  
],  
"status": "succeed",  
"error" : null
```

로그인 응답 => User 객체를 보냄

```
"userId" : "terry8408@gmail.com",  
"password": null,  
"userName": null,  
"birth": null,  
"createdAt": null,  
"lastModifiedAt": null,
```

Response
DTO에 담아서
보내는 것으로
통일

1-2. 부딪힌 이슈 & 해결사항

- 1-2. 부딪힌 이슈

- Protocol 테스트 시 서버는 Postman을 이용해 왔음.
- Http 통신에는 어려움이 없었지만, SSE 통신 테스트를 할 수 없음.
- (SSE 통신기능이 Postman 비교적 최근에 추가된 기능인데, 사용법을 잘 모르겠음.)
- 또한 SSE emitter 가 구현되어 있는 서버의 main코드를 EC2에 올려서 클라이언트가 테스트 할 경우, emitter 가 아직 구현되어 있지 않은 클라이언트에서 SSE emitter를 제외한 다른 부분도 테스트 할 수 없을 수도 있음.
- 따라서 branch 를 하나 분기 해서 SSE 가 구현되어 있는 서버 프로그램과 , SSE가 구현되어 있지 않은 서버 프로그램 두개를 branch를 왔다 갔다 하면서 EC2 에 업로드 하여서, 테스트 환경을 우선적으로 구축

2. 앞으로의 계획

- 2. 계획

- 클라이언트의 요청 처리(API 추가)
- 클라이언트와 RestAPI 테스트
- 클라이언트와 SSE 테스트

3주차 (3/12 ~ 3/18)	요청 프로토콜 만들기, Controller와 Entity 먼저 구성
4주차 (3/18 ~ 3/25)	EC2 서비스 환경구축, DB 연결, 회원가입/로그인 기능 구현
5주차 (3/25 ~ 4/1)	Firebase Authentication
6주차(4/2 ~ 4/8)	일정 CRUD
7주차(4/9 ~ 4/15)	일정 CRUD
8주차(4/16 ~ 4/22)	그룹 CRUD
9주차(4/23 ~ 4/29)	그룹 CRUD
10주차(4/30 ~ 5/6)	그룹 일정 CRUD
11주차(5/7 ~ 5/13)	그룹 일정 CRUD
12주차(5/14 ~ 5/20)	테스팅
13주차(5/21 ~ 5/27)	테스팅