

# 0309 회의록

## 아침 데일리 스크럼

- 1주차에는 A형 특강으로 인해 발표가 없다. 2주차에 기획안 및 개발내용을 발표
- 프론트 : Vue 디렉토리 구성 설정 및 와이어 프레임 제작 시작 ( 반응형을 고려하여 꼼꼼히 할 예정 )
- 백엔드 : 아나콘다 가상환경에서 flaskAPI 까는게 어렵다. 메뉴얼 만들던가 이슈를 남겨서 처리한다.
- 서버 : 공부중에 있다. 멘토링 ppt 준비

## 오후 회의

- 팀장 미팅 내용 : 매일같이 하는 상황보고, 시퀀스 다이어그램 추천 이외 특이사항 없음
- 검색 기능을 넣을려고 하였으나 그러면 너무 무거운 감이 있어서 상황 보고 넣기로 함
- 디테일 페이지에 추천기능을 추가하기로 함 ( 중복 검사 가능하게~ )
- Nav바를 구성하여 페이지간 이동이 편하게 하기로 변경
- 추후 세부 UI/UX는 만들면서 논의



지라와 git 연동 간 문제 → Q&A를 통해 비밀번호 초기화로 이슈 해결!

## Front 정의

- Props은 보는 사람을 위해 자세히 정의한다. ( type : Object 사용 X )

```
props: {  
  id: { type: Number },  
  badcomment: { type: Number },  
  opinion_type: { type: Boolean },  
  content: { type: String },  
  created_at: { type: String },  
  updated_at: { type: String },  
  user: { type: Number },  
  article: { type: Number },  
  emotion: { type: String },  
  like_users_count: { type: Number },  
  like_users: { type: Array },  
  username: { type: String },  
  board_type: { type: Boolean },  
},
```

### 1. 컴포넌트 이름

- 컴포넌트 파일 이름은 PascalCase를 이용 한다.
- Abbreviation (축약어)를 사용하지 않는다.
- 예시: MyComponent.vue

### 2. 베이스 컴포넌트 이름

- 한 페이지의 특정 기능/부분을 이용할 경우 Prefix (접두사)를 통일한다.
- 예시: VButton, VTable, VIcon

### 3. 강한 연관성을 가진 컴포넌트 이름

- 부모 컴포넌트와 연결되어 있는 자식 컴포넌트들의 이름의 경우 접두사를 통일한다.
- 예시: TodoList, TodoListItem, TodoListItemButton, SearchSideBar, SearchSidebarNavigation

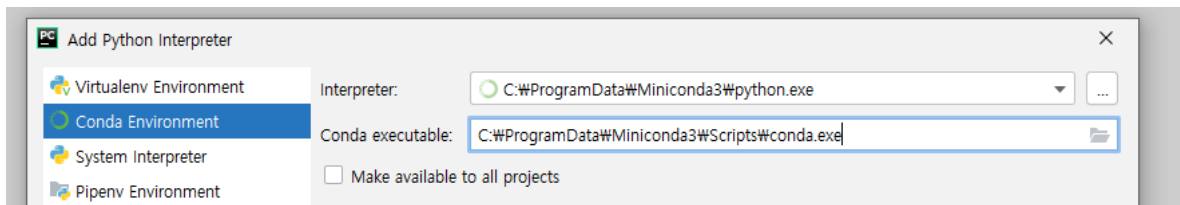
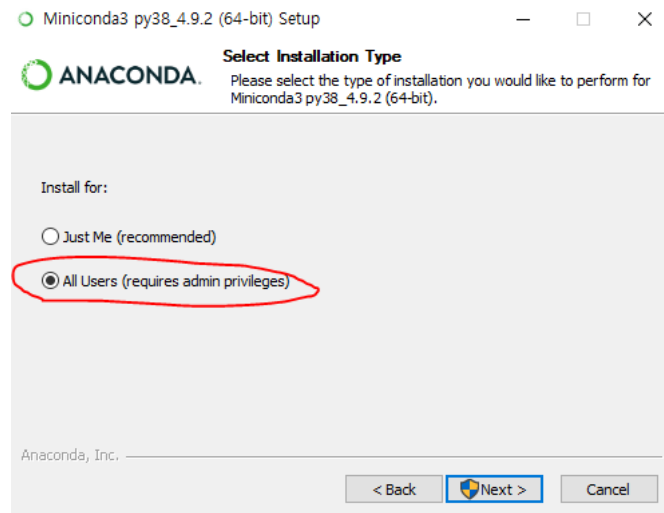
## Vuex Convention

1. Module은 기능 단위로 자른다.
2. state은 중첩해서 사용하지 않는다 (should not be nested)
  - state은 언더바를 사용해 다음과 같이 지정한다: products\_name, this\_is\_convention
3. Getters
  - 참인지 거짓인지 판단할 경우 `is` 를 사용한다. (예시: isUserLoggedIn, isUser)
  - Boolean이 아니면 `get` 으로 시작한다.
4. Actions
  - 모듈 밖에서 상태 변화가 일어날 경우, action에서 일어나야 한다.
  - 여러개를 불러올 때는: `fetchProducts`
  - 한 개를 불러올 때는: `findProduct`
  - 생성 시에는: `createProduct`
  - 수정 시에는: `updateProduct`
  - 삭제 시에는: `deleteProduct`
5. Mutations
  - Mutation만 module의 state를 변경할 수 있다.
  - 대문자를 사용한다. (예시. ADD\_PRODUCT, SET\_CURRENT\_PRODUCT\_ID)
  - `SET`: 비어있는 state에 데이터를 배정한다.
  - `ADD`: 이미 배정되어 있는 state에 새로운 데이터를 추가한다.
  - `REMOVE`: `ADD` 의 반대이다.

---

## 백엔드 기본 설정에 대해 진행 상황 요약

1. 미니콘다 설치 (아나콘다의 기본라이브러리를 제거한 형태)
  2. 미니콘다 환경에서 fastAPI 설치 및 pycharm 연동
  3. 미니콘다 환경에서 모듈 설치와 실행 자동화 배치 파일 생성
  4. DB 스키마 설계
- <https://docs.conda.io/en/latest/miniconda.html#windows-installers> 미니콘다 설치



- fastAPI 테스트를 위한 데모 코드 작성

```
from fastapi import FastAPI, File, Form, UploadFile

from typing import Optional, List
import os

app = FastAPI()

@app.get("/")
async def read_root():
    return {"Hello": "World"}

@app.get("/items/{item_id}")
async def read_item(item_id: int, q: Optional[str] = None):
    return {"item_id": item_id, "q": q}

@app.post("/files_old/")
async def create_file(
    file: UploadFile = File(...)
):
    return {
        "file_name": file.filename,
        "file_size": len(file.file),
        "file_content_type": file.content_type,
    }

@app.post("/files/")
async def create_files(files: List[bytes] = File(...)):
    return {"file_sizes": [len(file) for file in files]}

@app.post("/uploadfiles")
async def create_upload_files(files: List[UploadFile] = File(...)):
    UPLOAD_DIRECTORY = "."
    for file in files:
        contents = await file.read()
        with open(os.path.join(UPLOAD_DIRECTORY, file.filename), "wb") as fp:
            fp.write(contents)
        print(file.filename)
    return {"filenames": [file.filename for file in files]}
```

```
# @app.post("/files/")
# async def create_file(
#     file: bytes = File(...), fileb: UploadFile = File(...), token: str = Form(...)
# ):
#     return {
#         "file_size": len(file),
#         "token": token,
#         "fileb_content_type": fileb.content_type,
#     }
```

- 서버 실행을 위한 배치파일 생성 run.bat

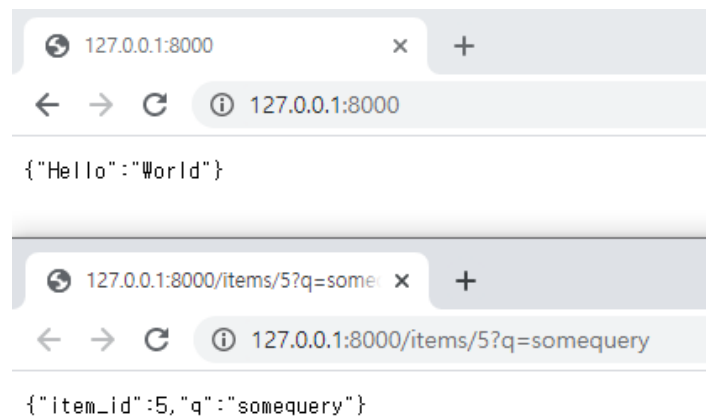
1. run.bat 위치의 폴더로 현재 작업 위치 이동
2. 미니콘다 "C:\ProgramData\Miniconda3" 경로에 pip로 requirements에 있는 모듈 설치
3. activate.exe base의 미니콘다 가상환경 실행
4. uvicorn 모듈을 통해 main.py 코드 실행하여 서버 동작

```
pushd %~dp0
REM 관리자 권한으로 실행하여야 문제가 없습니다.
title JJAL Python SERVER
REM pip freeze > requirements.txt
set conda=C:\ProgramData\Miniconda3
%conda%\Scripts\pip.exe install -r requirements.txt
call %conda%\Scripts\activate.bat %conda%
call uvicorn main:app --reload --host=0.0.0.0 --port=8000
```

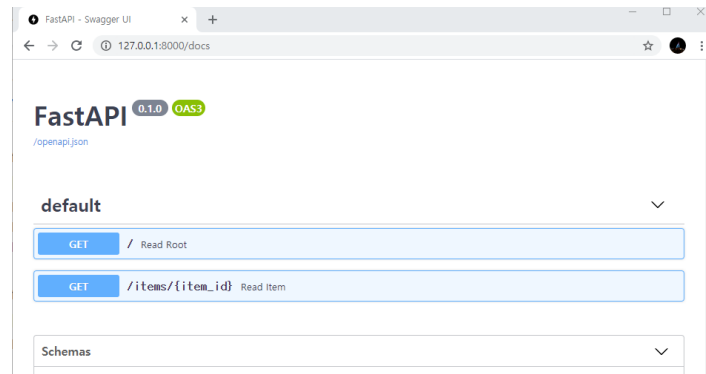
- 서버 결과

http://localhost:8000

http://localhost:8000/docs



fastAPI 기본동작 확인



fastAPI 스웨거 정상동작

- 데이터베이스 스키마 설계

board						
PK	AI	FK	Null	Logical Name	Name	Type
✓	✓			board_no	board_no	INT
				content_type	content_type	VARCHAR(200)
				content	content	LONGTEXT
				nickname	nickname	VARCHAR(100)
				password	password	VARCHAR(100)
				ip	ip	VARCHAR(16)
				good	good	INT
						0

comment						
PK	AI	FK	Null	Logical Name	Name	Type
✓	✓			comment_no	comment_no	INT
		✓		board_no	board_no	INT
				comment	comment	LONGTEXT
				nickname	nickname	VARCHAR(45)
				password	password	VARCHAR(100)
				ip	ip	VARCHAR(16)

most						
PK	AI	FK	Null	Logical Name	Name	Type
✓	✓			most_no	most_no	INT
		✓		count	most_count	INT
				name	most_name	VARCHAR(100)
				type	most_type	VARCHAR(100)

```
-- MySQL dump 10.13  Distrib 8.0.21, for Win64 (x86_64)
--
-- Host: 127.0.0.1  Database: jjal
-----

DROP TABLE IF EXISTS `board`;
CREATE TABLE `board` (
  `board_no` int NOT NULL AUTO_INCREMENT COMMENT 'board_no',
  `content_type` varchar(200) COLLATE utf8_unicode_ci NOT NULL COMMENT 'content_type',
  `content` longtext COLLATE utf8_unicode_ci NOT NULL COMMENT 'content',
  `nickname` varchar(100) COLLATE utf8_unicode_ci NOT NULL COMMENT 'nickname',
  `password` varchar(100) COLLATE utf8_unicode_ci NOT NULL COMMENT 'password',
  `ip` varchar(16) COLLATE utf8_unicode_ci NOT NULL COMMENT 'ip',
  `good` int NOT NULL DEFAULT '0' COMMENT 'good',
  `regdate` DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`board_no`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci COMMENT='board';

DROP TABLE IF EXISTS `comment`;
CREATE TABLE `comment` (
  `comment_no` int NOT NULL AUTO_INCREMENT COMMENT 'comment_no',
  `board_no` int NOT NULL COMMENT 'board_no',
  `comment` longtext COLLATE utf8_unicode_ci NOT NULL COMMENT 'comment',
  `nickname` varchar(45) COLLATE utf8_unicode_ci NOT NULL COMMENT 'nickname',
  `password` varchar(100) COLLATE utf8_unicode_ci NOT NULL COMMENT 'password',
  `ip` varchar(16) COLLATE utf8_unicode_ci NOT NULL COMMENT 'ip',
  `regdate` DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`comment_no`),

```

```

    KEY `FK_comment_board_no_board_board_no` (`board_no`),
    CONSTRAINT `FK_comment_board_no_board_board_no` FOREIGN KEY (`board_no`) REFERENCES `board` (`board_no`) ON DELETE RESTRICT ON UPDATE RESTRICT
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci COMMENT='comment';

DROP TABLE IF EXISTS `most`;
CREATE TABLE `most` (
  `most_no` int NOT NULL AUTO_INCREMENT COMMENT 'most_no',
  `most_count` int NOT NULL DEFAULT '0' COMMENT 'count',
  `most_name` varchar(100) COLLATE utf8_unicode_ci NOT NULL COMMENT 'name',
  `most_type` varchar(100) COLLATE utf8_unicode_ci NOT NULL COMMENT 'type',
  PRIMARY KEY (`most_no`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci COMMENT='most';

```