

프로젝트 소개

(2017 봄학기) 소프트웨어 개발의 원리 및 실제

2017. 04. 18 (화)

SWPP 프로젝트 - 페이스북

- 구현할 프로젝트 : 페이스북
- 개인이 아닌 조별로 구현
- 프로젝트 진행 순서
 - 페이스북 페이지 만들기 -> 지정된 기능 추가 -> 원하는 기능 추가

SWPP 프로젝트

- 프로젝트 첫번째 과제
 1. 페이스북 페이지 구현
 2. 로그인, 쓰기, 답글, 채팅 기능 구현
- 제출 기한 : 5월 12일 11시 59분 (예정)
- 매주 progress 보고서 제출
 - 분량 1페이지
 - 그림 or 글 or 아무 형태
 - 보고서를 보면서 설명 시 전달 가능하게 작성 (엄청 자세히 쓰지 않아도 됨)
 - 매주 무언가 돌아가는 형태
 - 보고서에 적힌 해당 주소에서 기능을 확인할 수 있어야 함

SWPP 프로젝트

- 프로젝트의 자세한 사항은 오목 숙제 제출 전후에 공지
- 프로젝트 데모 시연
 - 쓰기과 로그인 기능이 구현되어 있는 페이스북

SWPP 프로젝트

Login

Write a post

Content

Write

1

11:34 PM

jack

LIKE

REPLIES

1

3:01 AM

loglog

LIKE

REPLIES



Frontend : Store

- Schemas

Entities

POST

Id	Author	Created	Content
...			

REPLY

Id	Author	Created	Post	Content
...				

Frontend : Store

- Action

1. Async한 Action(Server과의 통신)

- Server 에 post 쓰기
- Server 에 post list 요청
- Server 에 id와 pwd를 post해주는 Action(Login)

2. Sync한 Action(Redux의 store과의 상호작용)

- Store 에 post 넣기
- Store 에 post list 그리기

Frontend : Store

- Reducers
 - 앞의 Sync한 action을 실행
- Saga : 하나의 일을 두개의 Saga로 나눠서 구현
 - Request 가 오는지 감시하는 saga
 - ex) 서버에 새로운 POST를 올려달라는 request가 오는지 감시
 - 위의 일을 실제로 하는 saga
 - ex) 서버에 새로운 POST를 실제로 올림

Frontend - components

ID

PWD

Login

여기에 글을 쓰세요..

입력

영희
4월 18일 7시 33분
음악은 나라가 허락한 유일한 마약

철수
4월 19일 3시 33분
중간고사 보기 싫다.

Frontend - components

ID	PWD	Login
여기에 글을 쓰세요..	입력	
<div><div>영희 4월 18일 7시 33분 음악은 나라가 허락한 유일한 마약</div><div>철수 4월 19일 3시 33분 중간고사 보기 싫다.</div></div>		

Frontend - components

ID	PWD	Login
여기에 글을 쓰세요..		입력
<div><div>영희 4월 18일 7시 33분 음악은 나라가 허락한 유일한 마약</div><div>철수 4월 19일 3시 33분 중간고사 보기 싫다.</div></div>		

Frontend - components

- 오늘 데모에서 Component의 구조는 다음과 같습니다.
 - 전에 설명해 드렸던 TodoList예제와 비교해보세요. 거의 다름없습니다.!
- PostList(Organism)
 - Store에 있는 Post들의 List를 받아서 Post들을 보여주는 부분
- PostForm(Organism)
 - Store에 Post를 추가하는 부분
- LoginForm(Organism)
 - Username과 Password를 받아서 서버에 Post하는 부분

Backend

- Model
 - Post, Reply
 - User, Group : `django.contrib.auth.models`
- View
 - 위의 Model들에게 get/post 해줄 수 있음
- Urls
 - 각 Model마다 urls 를 만듦

Backend

- Model
 - Post
 - Reply
 - User
 - Group

REPLY				
Id	Author	Created	Post	Content
...				

POST			
Id	Author	Created	Content
...			

User			
Id	user	email	groups
...			

Group	
id	name
...	

Backend

- Model
 - Post, Reply
 - User, Group : `django.contrib.auth.models`
- View
 - 위의 Model들에게 get/post 해줄 수 있음
- Urls
 - 각 Model마다 urls 를 만듦

```
{  
    "users": "http://127.0.0.1:8000/api/users/",  
    "groups": "http://127.0.0.1:8000/api/groups/",  
    "posts": "http://127.0.0.1:8000/api/posts/",  
    "replies": "http://127.0.0.1:8000/api/replies/"  
}
```

프로젝트 소개: 질문 시간

- 질문이 있으면 말씀해주세요.