Progress6(~6/12)

2조 김명곤 배상환 류호 이한결

1. 이번주에 구현한 behavior 목록 및 주요 사항

이번주에는 gps기능 등의 활용을 위한 https연결, 투표기능, 일정기능, 이미지 검열 모델 교체 등을 수행했다. 하였다. 전반적으로 이번주 및 다음주 초에 팀원들의 기말고사가 몰려있어 비교적 적은 양의 작업을 수행하였다

1.1. behavior 설명

1.1.1. https

AWS에서 SSL인증서를 이용해 https를 이용하기 위해서는 도메인 주소가 필요한 것으로 보인다. 이를 위해 도메인 주소를 구입해 AWS에서 SSL인증서를 얻었다. 이를 AWS의 ELB를 이용해 연결하고 https를 위한 AWS의 443포트를 3000포트로 redirect하였다. 이를 통해 프론트앤드를 https를 이용해 접근할 수 있게 되었다.

그런데 프론트앤드에서 https를 이용할 경우 백앤드에서도 마찬가지로 https를 이용해야 프론트앤드와 백앤드간의 통신이 가능하다. 따라서 백앤드의 포트인 8000포트 역시 https를 이용해 접근할 수 있도록 할 수 있어야 한다. 그런데 AWS의 443포트를 이용해 여러 포트로 다이나믹하게 redirect하는 방법을 찾지 못했다. 때문에 자가서명 SSL인증서를 이용하여 8443포트를 거쳐 8000포트에 접근하도록 하였다.

1.1.2. 이미지 검열 모델 교체

AWS 서버의 디스크 용량 부족으로 서버가 예상치 않게 종료되는 문제가 있어 이미지 검열 모델의 용량을 줄였다. 기존에 VGG16 모델을 transfer learning 방식으로 만들었던 모델에서 ResNet50을 transfer learning 방식으로 만든 모델로 교체했다. ResNet은 VGGNet에 비해 용량을 크게 줄인 것으로 알려져 있다. 약 100MB 정도의 용량을 절감했다. 또한 정확도도 94%에서 96%로 소폭 상승했다.

1.1.3. 투표기능

투표 형태의 포스트를 올릴 수 있다. 기존의 포스트를 올리는 공간에서 전환해 올릴 수 있고 투표는 중복투표가 불가능하다. 결과 확인은 투표를 마친 사람만 가능하다.

1.1.3. 일정 기능

React-big-calendar를 이용하여 달력을 보이도록 하였다. 드래그시 또는 직접 날짜를 입력해 일정을 설정할 수 있도록 하였다. month/week 단위로 달력을 볼 수 있다. 일정을 추가한 뒤에는 일정을 달력 페이지 또는 메인 페이지에서 확인할 수 있다.

1.2. behavior 배분 내용

|  |  |  |
| --- | --- | --- |
| 담당 | 기능 | 상세 |
| 류호 | https | AWS ELB세팅, 도매인 구입 및 도메인 연결 |
| 자가서명으로 백앤드 https 구동 |
| 그 외 https관련 작업 전반 |
| 배상환 | 이미지검열 | 이미지 검열 모델 교체  (이미지 검열 모델 교체(VGG16 -> ResNet50) |
| 일정기능 | 일정 기능 백엔드 구현 |
| 일정 기능 프론트엔드 일부 구현 |
| 백엔드 테스트 추가 |
| 이한결 | 투표기능 | 투표기능 프론트앤드 구현 |
| 투표기능 백앤드 구현 |
| 백앤드 및 프론트앤드 테스트 추가 |
| 김명곤 | 일정기능 | 프론트앤드 구현 |
| 프론트앤드 테스트 추가 |

2. backend 새로 구현한 api 스펙 및 테스트 코드

2.1. 신규 api 스펙

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| url | GET | POST | PUT | DELETE |
| schedule/ | 해당 유저의 모든 일정 리스트 반환 | 일정(start, end, content) 받아서 일정 생성 | - | - |
| vote/ | - | option id를 받아서 투표하고, 투표한 사람 목록에 반영 | - | - |

2.2. 세부 수정 api 스펙

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| url | GET | POST | PUT | DELETE |
| posts/ | 이전과 동일 | poll\_data(topic, option)을 받아서  투표 생성 | - | - |

2.3. 테스트코드

아래의 테스트 코드를 실행하면 된다.

2.3.1. $python3 pollTest.py

- 친구가 아닌 상태에서 상대 담벼락에 투표 만들기

- 친구 맺기

- 친구 상태에서 상대 담벼락에 투표 만들기

- 만들어진 투표 포스트에 투표하기

2.3.2. $python3 scheduleTest.py

- test0 – test9 10명 유저 회원가입

- test 유저들 모두 임의의 일정 추가

- test 유저들의 일정 GET

3. 프론트앤드 테스트코드

아래의 테스트 코드를 실행하면 된다.

3.1. $python3 poll\_test.py

- test1과 test2 친구 맺기

- test1이 자신의 담벼락에 투표 만들기

- 만든 투표에 투표하기

- test2가 test1이 만든 투표에 투표하기

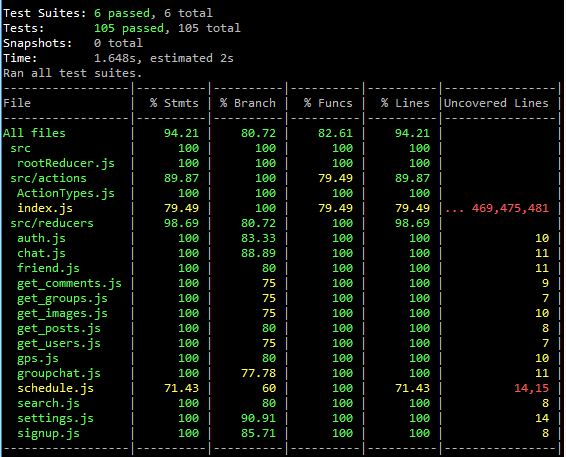
3.2. $python3 schedule\_test.py

- 유저 생성 및 로그인 후 캘린더 페이지 이동

- 스케쥴 날짜/내용 입력

- 스케쥴리스트에서 확인

4. jest 테스트



5. aws 링크

Frontend: <https://calendargrams.com>/

Backend: <https://34.208.93.214:8443/>