

To Do

- ✓ 정보 (정책/생활백서)
 - ✓ 목록
 - ✓ 내용 (상세 화면)
 - ✓ 작성/수정/삭제
 - ✓ 좋아요
 - ✓ 검색 (이슈 수정 필요)
 - ✓ 인기 게시글 ⇒ △
- ✓ 거래 (공구/OTT)
 - ✓ 목록
 - ✓ 내용 (상세 화면)
 - ✓ 작성/수정/삭제
 - ✓ 거래
 - ✓ 검색 (이슈 수정 필요)
 - ✓ 동네정보 반영하여 글 불러오기
- ✓ FAQ
 - ✓ 목록
 - ✓ 내용
- ✓ 관심글 목록
- ✓ 댓글 작성/수정/삭제
- ✓ 답글 작성/수정/삭제
- ✓ 사용자 프로필 눌렀을 때 평점 확인/강퇴 (강퇴알림)

해서 (Backend)

- ✓ 이슈: 검색이 한 차례 늦게 됨
- ✓ 이슈: 댓글 달면 바로 안 보여짐
- ✓ 홈페이지(전체)검색
- ✓ 이슈: 거래 참여했을때 accumulate가 바로 반영이 안됨
- ✓ 이슈: 거래 거절했을때 명단이 바로 안 바뀜

은혜 (Frontend)

- 화면 보완
 - ☐ 폰트/여백 vh 단위 변경 (dropdown, modal, post, write)
 - ☐ 모달 크기 변경
 - ☐ 전체 글씨 크기 변경
 - ☐ 댓글/답글 표 테두리 차이
 - ☐ 키보드 올라오는 문제 해결 ⇒ 레이아웃 수정
 - 오류 수정
 - ☐ 웹 호스팅 웹에서 새로고침 404 문제 해결
-
- 반응형 웹
 - Component 세분화
 - 코드 간단하게 정리
 - css 충돌 방지 (.css ⇒ .module.css)
 - 주기적으로 build 후 dothome 업데이트

	웹 기존 진행 사항	웹 실제 진행 사항	웹 향후 개발 예정 사항
2주차	기본 UI 설계/개발	기본 UI 설계	
3주차	DB 설계/구축	DB 설계/구축	
4주차	정책/생활백서 게시판 설계	(프론트) 게시판 화면 설계/개발	
5주차	정책/생활백서 게시판 개발	(프론트) 게시글 작성 화면 설계/개발	
6주차	문의/FAQ 게시판 설계	(프론트) 게시글 검색 화면 설계/개발	
7주차	문의/FAQ 게시판 설계	(프론트) 게시글 화면 설계/개발	
8주차	웹앱 연결	PHP-DB연결, React-PHP 연결	
9주차	공동구매 게시판 설계	(백) 게시판 글 목록, 글 작성	
10주차	공동구매 게시판 개발	(백) 게시판 글 검색, 글 조회	
11주차	OTT 구독 게시판 설계	웹앱 연결 (프론트/백) 게시글 수정/삭제 (백) 댓글 작성/수정/삭제	
12주차	OTT 구독 게시판 개발		(프론트) 댓글 및 답글 작성/수정/삭제, 거래 취소, 인기 게시글, 관심글 (백) 글 수정/삭제, 관심글 목록 기능, 댓글 및 답글 작성/수정/삭제
13주차	부가 기능 설계/개발		(프론트) 웹앱 화면 보완 (백) 인기 게시글 설정, 모바일 환경 통신, 거래 게시판 기능 완성
14주차	UI 디자인 완성		독립적인 웹
15주차	마무리 통합 테스트		마무리 통합 테스트
16주차			
17주차			<ul style="list-style-type: none"> 6/21(화) 최종발표 6/24(금) 최종 보고서 제출

현재 개발 완료

- 모든 게시판의 글 목록, 글 작성, 글 조회, 글 검색

향후 개발 예정(프론트)

- 댓글/답글 작성/수정/삭제
- 게시글 수정/삭제
- 거래 취소 기능
- 거래 완료 화면
- 인기 게시글

향후 개발 예정(백엔드)

- 게시판 좋아요 누르기/취소
- 모든 게시판의 글 수정, 삭제
- 댓글/답글 달기, 수정, 삭제, 조회
- 모바일 환경과 통신
- 거래 완료/취소
- 좋아요 누른 게시글을 관심글로 불러오기
- 웹 로그인/회원가입 (웹 캐시)

독립적인 웹

- 웹 메인 화면
- 웹 로그인/회원가입 (웹 캐시)

모바일 환경 통신

- 거래) 사용자 위치 정보 보고 그 위치에 해당하는 게시글 출력
- 거래) 거래 신청/취소/완료
- 거래) 참여한 사람 정보
- 거래) DB 에서 목표 달성 ⇒ 했으면 완료 처리 및 앱한테 신호

거래 기능

- 사용자 위치 정보 확인 후 글 출력

회의

- 3월12일(토)
 - Web 레이아웃 완성
- 3월17일(목) 21:00
 - UI 디자인 수정
 - 개발 방향 결정 (반응형 웹 view port)
- 3월19일(토) 20:00
 - 진행 사항 및 이슈 토의
 - 일정 및 계획 정리
 - GitHub 협업 툴 관리
- 3월24일(목) 20:00
- 3월27일(금) 21:00
- 4월3일(일) 21:00
- 4월10일(일) 14:00
- 4월17일(일) 17:00
- (생략)
- 5월12일(목) 16:00

Memo

- 인기글 : 상위 3개 1page
- 인기 게시글 고정 기간 3일 (추천수 기준)
- ~~게시판 이미지 업로드 API 확인~~
- 웹에서는 거래 게시판 팝업 띄워서 동네 검색
- 웹에서 거래게시판을 이용할 경우, 게시글 열람만 가능하게 할 것. 동네정보는 내가 설정해서 볼 수 있도록 할 것
- Post.js에서 사용자의 like 유무 확인 후 setHeart(true) or setHeart(false)로 정보 세팅
- 추가 구현) 구매 일시를 DateTime으로 임시로 받고 공구 마감 시간이 되면 DB 삭제 + 사용자 알림 처리 ⇒ 현재는 char로 되어 있음
- 거래 완료되면 출력 X

Issue

- Q.

#현재진행된것

-var poliJson[] , var infoJson[] 변수 추가 -> 정책/백서 json을 저장할 array형 변수

-getPolicyJson(), getInfoJson() -> fetch()의 get 메소드로 json을 받아와서 각각 poliJson과 infoJson에 저장하였음

-infoBoard.js 에서 13,28번 줄에 보면 url 앞에 '<https://stark-savannah-03205.herokuapp.com/>' 가 붙어있는데,
이는 cors policy 정책을 위반하게 되어 나타나는 CORS disabled문제를 해결하기 위한 것이다.

-참고로 저 주소는 개인마다 다를 수 있으므로 **혹시나 에러가 뜬다면**

npm install -g npm

npm install -g heroku

git clone <https://github.com/Rob--W/cors-anywhere.git>

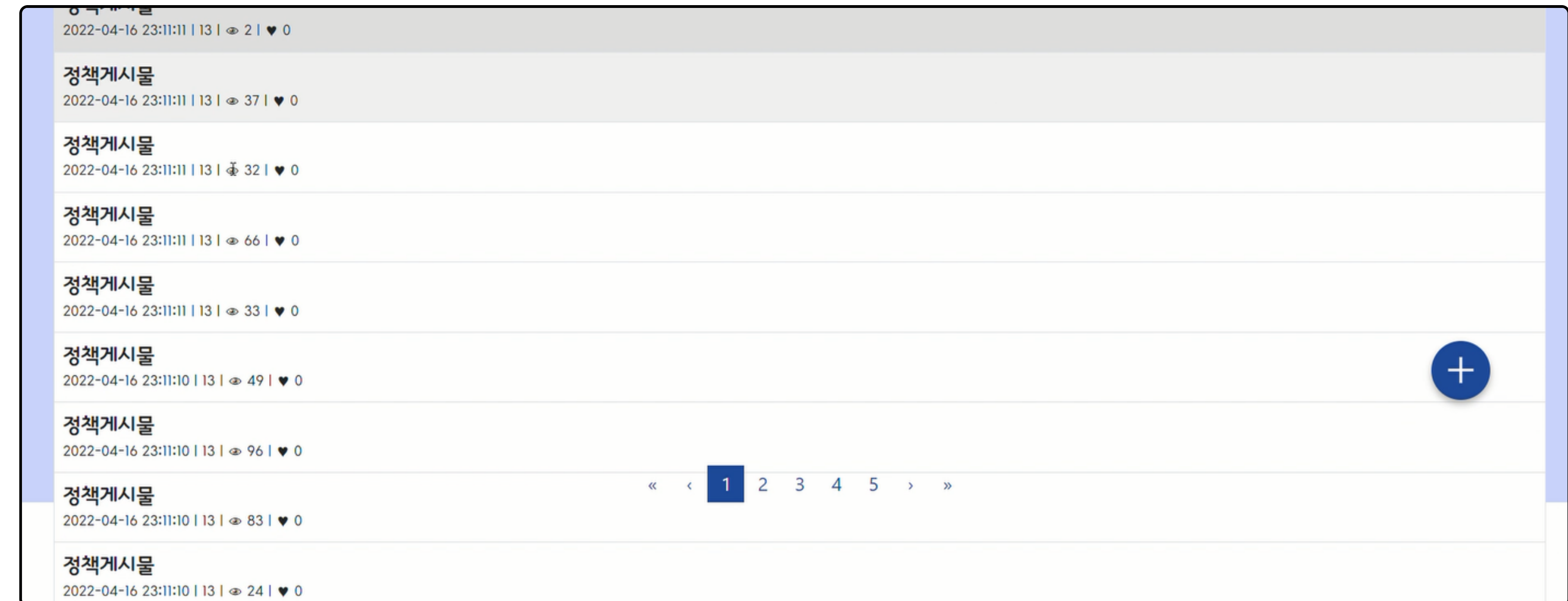
cd cors-anywhere/

npm install

heroku create

git push heroku master

를 하여 나타나는 주소로 대체해주면 될 것이다.



#이슈

-policyList→poliJson, infoList→infoJson으로 대체하고

-BoardTable.js에서

```
<div className="tableInfo">{item.date} | {item.writer} | 👁 {item.view} | ❤ {item.like}</div>
```

위의 코드를

```
<div className="tableInfo">{item.reg_date} | {item.nick_name} | 👁 {item.view} | ❤ {item.like}</div>
```

로 바꿀 경우 연동은 되는 것을 확인함

-다만 아무런 조치 없이 저 코드를 실행하면 7개씩 잘리지 않고 전체가 다 보여져서 표 밖을 탈출함

-따라서 Json(배열형) 변수안의 내용들을 **7개씩 잘라서** 최신순으로 보여줄수 있도록 조정해야 할듯... (참고로 json파일들은 게시일자가 제일 최신인 것부터 정렬되게 해놨음)

-이에 따라 pagination 코드도 조정이 필요할듯

1. REST API를 통해 받아온 데이터가 늦게 출력됨

- (메인 게시판 화면): 정책 게시판에서 바로 보드가 출력이 안됨. 클릭을 해야 내용이 출력됨
- (검색): 어떤 검색어를 검색할 경우 바로 나타나고 한 번 더 검색 버튼을 클릭해야 내용이 출력됨
- (FAQ): 처음 '궁금해요'를 누르면 아무것도 안 보여짐. 메인 화면으로 돌아갔다가 다시 들어와야 보드가 출력됨.
- axios나 fetch로 데이터를 마저 받아오기 전에 렌더링이 되버리는 것 같음.
- axios나 fetch에 await 사용 불가(에러남).

2. 생활백서 게시판에서 닉네임이 안 보여짐

- policyJson과 docJson 모두 user테이블과 JOIN 해서 닉네임까지 가져오도록 해놨음.
- 분명 docJson(<http://holo.dothome.co.kr/docJson.json>) 에서 nickname이 표시되는 걸 확인했는데
- 1번 이슈와 관련있는건지, react에서 axios통해 위 json을 받아오면
- json이 수정되기 이전의 데이터를 여전히 받아오고 있음 (nickname 없음)
- 똑같은 코드인데 정책 게시판은 닉네임이 보여지고 생활백서 게시판은 안됨.

그동안 동기화를 해결했던 방식은 클래스를 사용하여 componentDidMount()로 데이터 요소가 모두 갖춰져야만 mount를 할수 있는 식으로 했었다. 즉, 필요한 정보(포스트의 id값, db에 있는 모든 레코드들)를 mount 이전에 받아올 수 있었기에 가능한 것이었다.

그러나 검색의 경우, 검색화면이 화면에 띄워지고 나서 거기에서 작성된 검색어를 바탕으로 db에서 검색하기 때문에 그전에 했던 방식을 적용할 수가 없다. 함수형 컴포넌트에서 componetDidMount()와 같은 기능이 있는지 찾아봐야할듯...

정책

이게 무슨 오류임...?



먼지3

• 사각형 캡처(R)

2022-06-06 03:37:51

👁 2 ❤ 0

수정 삭제

💬 댓글 0

댓글을 입력해주세요.

등록



연인데요

2022-06-06 03:29:58

답글

수정

삭제

생활백서

DeliveryDeliveryDeliveryDeliveryDeliveryDeliveryDe



몰라

2022-06-14 05:08:49

DeliveryDeliveryDeliveryDeliveryDeliveryDeliveryDeDeliveryDeliveryDeliveryDeliveryDeliveryDeDeliveryDeliveryDeliveryDeliveryDeliveryDe

👁 1 ❤ 0

word-wrap: break-word;
word-break: break-all;