

24시간이 부족한 개발자
손지승입니다.

Portfolio

CONTACT
miga98@naver.com
010-2672-1095

About Me



손지승
Jiseung Son
1998.09.26

“You are better off trying something and having it not work and learning from that than not doing anything at all.”

– Mark Zuckerberg

EDUCATION

- 2017 강원대학교 행정학과 입학
- 2020 강원대학교 컴퓨터공학과 복수전공
- 2023 강원대학교 컴퓨터공학과 졸업

SKILL

- 언어 : Java, Javascript, HTML, CSS, SQL, Ocaml
- 프레임 워크 : Jquery, bootstrap, Spring boot, Mybatis
- Tool : Tomcat, Mysql

PROJECT

- 2021.09 첫 번째 메모장 프로젝트
- 2021.12 두 번째 Inatakilogram 프로젝트
- 2022.02 세 번째 SHUSED 프로젝트

CONTACT

- | | |
|--------|---|
| Phone | 010-2672-1095 |
| E-mail | miga98@naver.com |
| Github | https://github.com/hoppang98 |



기술 정보

Project

MEMO

Github

https://github.com/hoppang98/Memo_211228



Instakilogram

Github

https://github.com/hoppang98/Instakilogram_211228



SHUSED

Github

<https://github.com/hoppang98/SHUSED>



기술 스택

Tomcat, Mysql

Spring framework, Spring boot, Mybatis, JSTL

Bootstrap, jQuery

Java, javascript, html

라이센스

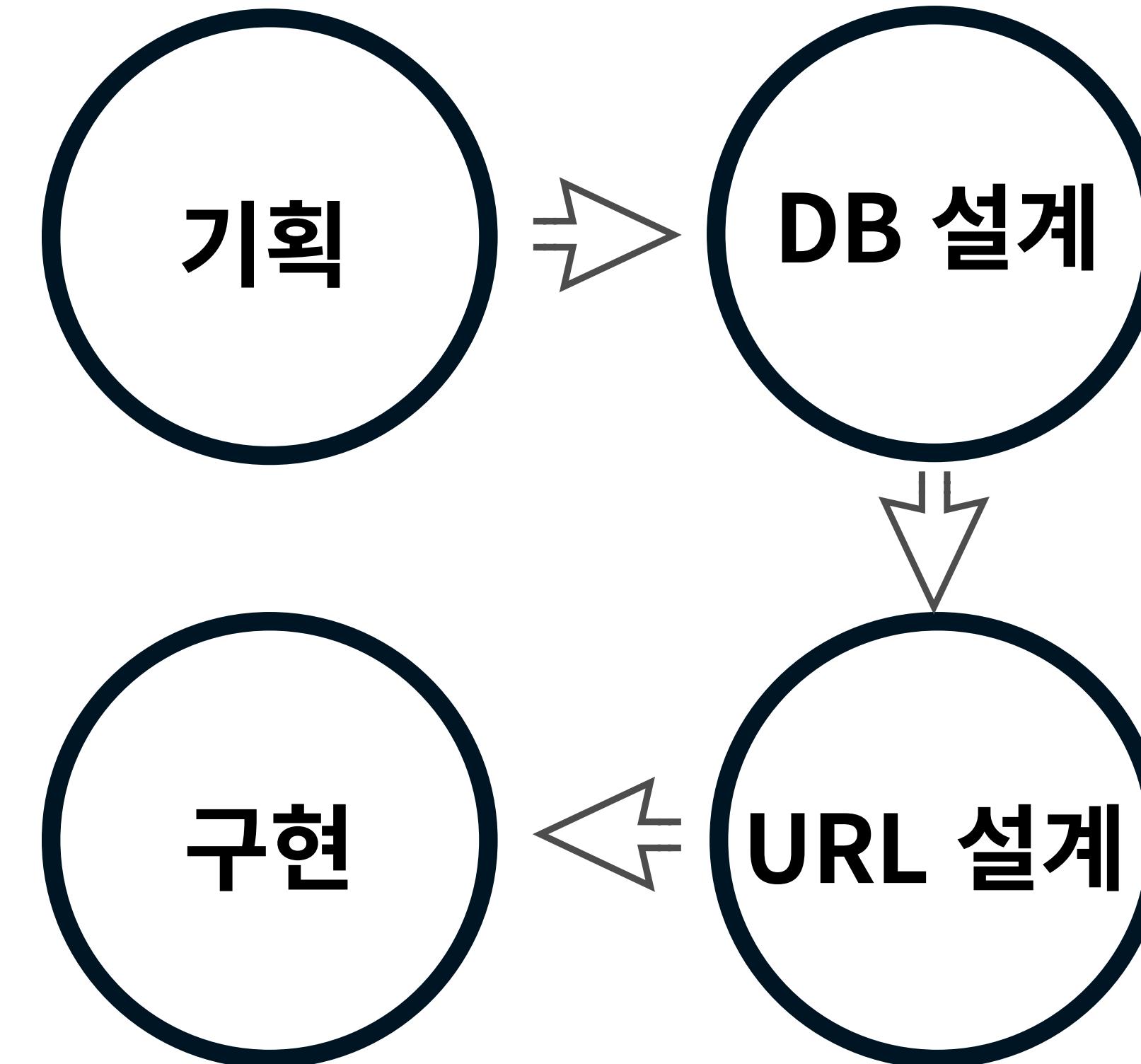
Tomcat(Apache License2.0), Mysql/GPLv2 or proprietary)

Spring framework(Apache license 2.0),

Mybatis(Apache License2.0),

Bootstrap(MIT License), jQuery(MIT License)

프로젝트 진행 단계



Index

Project

MEMO

개인 메모 저장이 가능한 웹 서비스

6 - 13

Instakilogram

최근 가장 많이 사용되는 SNS를 COPY한 프로젝트

14 - 21

SHUSED

개인 간의 중고 신발 거래를 위한 웹 플랫폼

22 - 36

Project MEMO

Memo

손지승 님 [로그아웃](#)

메모게시판

NO.	제목	시간
30	4월 1일 할 일 (부모님 생신)	2022-03-30 10:01:26
29	3월 31일 할 일	2022-03-30 10:00:58
28	3월 30일 할 일 (+추가)	2022-03-30 09:20:15

글쓰기

2021.09

Copyright 2018. memo all rights reserved.

Memo 게시판 프로젝트 소개

웹 서비스와 데이터베이스 활용에 대한 전반적인 이해를 위해서 제작한 **1인 프로젝트 Memo 게시판 프로젝트**입니다. 누구나 편리하게 사용할 수 있도록 직접적인 ui를 기반으로 **기획부터 설계, DB제작까지 직접 진행**한 프로젝트입니다. 회원가입과 로그인을 통해 타인이 볼 수 없는 나만의 비밀 메모 게시판이라는 컨셉으로 진행했습니다. 주요한 기능으로는 **게시글 작성, 파일 업로드, 메모 수정과 삭제** 기능이 있습니다.

Memo 게시판 프로젝트를 진행하며 느낀점

처음으로 기획부터 개인 프로젝트를 진행하며 내가 정말 이 서비스를 완성 할 수 있을지 기획 과정에서부터 걱정이 많았습니다.

그러나 **기획에서의 걱정은 오히려 안정으로 다가왔습니다.** 책과 인터넷을 통한 공부를 진행할 때는 다른 개발자의 구상대로 제작된 하나하나의 모듈을 억지로 내 머리 속에 끼워맞추는 느낌이었다면 직접 진행하는 프로젝트는 각각의 기능을 **기획하면서 어떤 모듈형태로 제작하고 어떻게 유기적으로 연결** 시켜야 할지 스스로 생각할 수 있게 해줬습니다.

이렇게 기능 하나하나를 작게 쪼개서 제작하다보니 웹 서비스라는 큰 흐름을 이해하는데 큰 도움이 되었습니다. **서블릿을 통해 먼저 경험했던 웹 서비스를 Spring 프레임워크를 통해 제작하면서 Controller -> Service -> Repository 가 어떤 구조로 연결되는지 체감** 할 수 있게 되었고 **모듈화**가 얼마나 중요한지에 대한 인식을 다시 한번 심어주는 계기가 되었습니다.

Database의 테이블을 기획하면서 각기 다른 테이블의 **데이터를 연결시키고 조합하면서 필요한 데이터를 생성**하는 과정도 큰 도움이 되었습니다. 방대한 양의 데이터를 어떤 class로 설계해야 최소한의 데이터로 가공이 가능한지, 도메인을 어떻게 설계해야 하는지 알 수 있게 되었습니다.

처음 진행하는 프로젝트이다보니 막히는 부분도 많았습니다. 왜 내 설계대로 작동하지 않는지 고민하면서 대학교 수업에서 들었던 교재를 펼쳐보고 배웠던 알고리즘을 활용해서 보다 쉽게 해결했던 경험도 있었습니다. **에러상황과 마주하면 에러 메시지를 천천히 읽으며 분석했습니다.** 초반에는 이해하기 어려웠던 에러 메시지들도 분석을 통해 해결하면서 이후에는 같은 에러 메시지가 발생하면 빠르게 명확한 이유를 파악하고 해결할 수 있게 되었습니다. 디버깅을 하는 과정에서는 **로그를 통해 어떤 데이터가 전달 되었고 어떤 결과가 나왔는지 파악**하며 버그를 해결하고자 했습니다. 데이터의 전달에 대해서 알게 되니 버그를 해결하는 것이 점점 더 쉬워졌습니다.

기획

Ovenapp.io 서비스로 페이지별 기획서 작성

로그인

메모 게시판

회원가입

메모 게시판

* 아이디	marobiana	중복확인
사용 가능한 아이디입니다.		
* 비밀번호	*****	
* 비밀번호 확인	*****	
* 이름	이름을 입력하세요	
* 이메일 주소	이메일 주소를 입력하세요	

메인페이지

메모 게시판

마른달님 안녕하세요 [로그아웃](#) [로그아웃](#)

No	제목	날짜
2	메모	2021-03-05
1	테스트	2021-02-28

글쓰기

메모 입력

메모 게시판

제목을 입력해주세요
내용을 입력해주세요

파일 선택 coffee.png

목록

모두지우기

저장

상세보기

메모 게시판

제목입니다
안녕하세요
메모
내용입니다
우하하하
○ㅎㅎㅎㅎ
후후후



파일 선택 coffee.png

삭제

목록으로

수정

DB 설계

Google spreadsheet로 데이터베이스 테이블 설계

컬럼명, 타입, Null 가능 여부, auto_increment 여부, 설명, 연결되는 컬럼 표시 등을 작성해서 표를 기반으로 테이블 생성 쿼리 작성



테이블 이름 : user				
설명 : 회원가입을 통한 회원 정보 관리 테이블				
컬럼명	타입	NULL가능 여부	auto_increment	설명
id	int	N	Y	primary key
loginId	varchar(16)	N	N	사용자 id
password	varchar(32)	N	N	사용자 비밀번호, 암호화 저장
name	varchar(16)	N	N	사용자 이름
email	varchar(64)	N	N	사용자 이메일 주소
createdAt	timestamp	N	N	생성 날짜
updatedAt	timestamp	N	N	수정 날짜

테이블 이름 : post				
설명 : 작성한 메모를 저장하는 테이블				
컬럼명	타입	NULL가능 여부	auto_increment	설명
id	int	N	Y	primary key
userId	int	N	N	메모 작성 사용자(user table의 primary key)
subject	varchar(128)	N	N	메모 제목
content	text	N	N	메모 내용
imagePath	varchar(128)	Y	N	첨부 이미지, 접근 가능한 이미지 경로
createdAt	timestamp	N	N	생성 날짜
updatedAt	timestamp	N	N	수정 날짜

URL 설계

Google spreadsheet로 URL 설계

View를 위한 URL과 API를 위한 URL을 분리해서 설계

View URL에는 url과 parameter, API URL은 url, method, parameter, response data를 작성



View URL

View URL 설계			C:\손지승\New_Study\1-2. Project 폴더 안에 메모게시판 개발 설계.txt 참고
제목	url	parameter	설명
로그인	/user/signin_view		로그인 화면
회원가입	/user/signup_view		회원가입 화면
글쓰기	/post/create_view		메모 쓰기 화면 로그인 상태에서만 접근 가능
메모 목록	/post/list_view		메모 목록 화면 로그인한 사용자 메모만 노출 로그인 상태에서만 접근 가능
메모 상세	/post/detail_view	id : 해당 메모의 id 많은 리스트에서 하나만 뽑아서 가져오는거니까	메모 목록에서 선택 이후 화면 메모 id를 parameter로 전달받고 해당 데이터를 노출 로그인 상태에서만 접근 가능

URL 설계 (2)



API URL

API URL 설계				C:\손지승\New_Study\1-2. Project 폴더 안에 메모게시판 개발 설계.txt 참고
1) 로그인				
url : /user/sign_in				
method : post				
parameter				
파라미터명	데이터 타입	NULL 여부	설명	
loginId	String	N	로그인 id	
password	String	N	로그인 비밀번호	
응답값(json)				
성공시	{ "result" : "success" }			
실패시	{ "result" : "fail" }			
2) 회원가입				
url : /user/sign_up				
method : post				
parameter				
파라미터명	데이터 타입	NULL 여부	설명	
loginId	String	N	로그인 id	
password	String	N	로그인 비밀번호	
name	String	N	사용자 이름	
email	String	N	사용자 이메일	
응답값(json)				
성공시	{ "result" : "success" }			
실패시	{ "result" : "fail" }			

https://docs.google.com/spreadsheets/d/1hKyRznj0lDdYlNCcJ_wd_bvFxQoKwVbN7G50JAz85XU/edit#gid=201988526

구현 (1) - 회원가입과 로그인

12

Memo

로그인

회원가입

회원가입

Copyright 2018. memo all rights reserved.

Memo**소개**

- 사용자가 입력한 값을 ajax를 통해 DB의 user 테이블에 저장해서 유저의 정보를 insert한다.
- 로그인 이후에는 session을 활용해서 header 부분에 유저의 이름과 로그아웃 정보가 나오도록 했다.
- form을 활용해서 submit방식을 통해 마우스 클릭이 아닌 enter키를 타이핑하는 방식으로도 로그인이 가능하도록 했다.
- 로그인, 회원가입시 input에 공백이 있는 경우 alert를 띄워 사용자가 알 수 있도록 했다.
- 로그인 이후에는 유저의 id값으로 유저가 남긴 메모의 정보가 모여있는 메인페이지로 이동하게 했다.

로그인

로그인[회원가입](#)

Copyright 2018. memo all rights reserved.

troubleshooting

- 비 로그인 상태에서 메모에 접근 가능한 경우가 있었다. WebMvcConfigurer에 기능을 추가하는 방법으로 interceptor를 활용해서 모든 페이지가 interceptor를 거쳐가게 해서 해결했다.
- PermissionInterceptor로 비 로그인 상태에서는 메모리스트에 접근 못하게, 로그인 상태에서는 로그인화면과 회원가입 화면에 접근 못하게 설정했다.
- 데이터베이스에 유저의 비밀번호가 그대로 저장되는 문제가 있었다. md5 알고리즘을 통해 비밀번호를 암호화해서 저장했다.

보완점

- 아이디 중복체크 기능 추가 필요

구현 (2) - 메모 입력, 상세보기, 삭제, 수정

Memo

손지승님 로그아웃

메모 입력

제목: 3월 30일 할 일

내용:
1. 공개 SW 개발자 대회 DB 구축
2. 부모님 생신 선물 쟁겨드리기
3. 택배 보내기

파일 선택: develop.png
목록으로

저장

Co Memo

손지승님 로그아웃

메모게시판

NO.	제목	시간
28	3월 30일 할 일	2022-03-30 09:20:15

글쓰기

Memo

손지승님 로그아웃

메모 보기

제목: 3월 30일 할 일

내용:
1. 공개 SW 개발자 대회 DB 구축
2. 부모님 생신 선물 쟁겨드리기
3. 택배 보내기

The poster features the text "오픈소스는 더하 GO. 4차 산업혁명은 누리 GO" and "11th 공개SW개발자대회". It also includes the dates "2017.4.28-6.19".

목록으로 **삭제** **수정**

소개

- 사용자가 글쓰기 버튼을 통해 이동한 view에서 입력한 값을 ajax를 통해서 post 테이블에 insert시켰다.
- sql의 update문을 활용해서 메모를 수정하는 기능이 있고 delete문을 활용한 삭제기능도 구현되어 있다.
- session을 활용해서 메모를 insert하면서 입력한 user의 pk값을 함께 저장하도록 했다. 이후 이 pk값을 통해 특정한 유저가 생성한 메모를 리스트로 불러오도록 했다.
- 메모 삭제시에 로그인한 유저의 pk값과 메모에 저장된 id값을 비교해서 메모를 작성한 작성자만 삭제가 가능하도록 했다.
- 메모 삭제시에 서버에 저장되어 있던 파일과 디렉터리를 함께 삭제하도록 했다.

troubleshooting

- MultiPartFile의 경우에는 formData를 append하는 방식으로 저장해야 했다.
- 서버에 파일을 저장하기 위해서 FileManagerService class를 생성해서 파일 관리만을 담당하게 했다.

보완점

- 다중 파일 업로드 기능

Project Instakilogram

2021.12

Instakilogram SNS 프로젝트 소개

현재 가장 많이 사용되는 웹 서비스인 Instagram을 COPY해 웹 서비스에 대한 이해도를 높이기 위해서 제작한 **1인 프로젝트**

Instakilogram SNS 프로젝트입니다.

기획부터 설계, DB제작까지 직접 진행했으며 유저의 로그인을 기반으로 timeline의 기능들을 구현하는데 중점을 뒀습니다.

주요한 기능으로는 **피드 작성, 파일 업로드, 댓글 입력, 좋아요 기능, 삭제 기능**이 있습니다.



Instakilogram SNS 프로젝트를 진행하며 느낀점

Memo 게시판 프로젝트를 마무리하고 나의 데이터베이스에 갇힌 서비스가 아닌 다른 사람과 소통할 수 있는 SNS 같은 서비스를 개발해보고 싶었고 현재 가장 선호도가 높은 SNS인 Instagram을 COPY해서 프로젝트를 제작했습니다.

Memo 프로젝트와는 달리 데이터베이스를 설계할 때 각 테이블관의 연계가 더욱 중요했고 이를 통해 다시 한번 모듈화가 얼마나 중요한지 깨닫게 되었습니다. 업로드, 댓글, 좋아요 등등 각각의 기능들을 모듈형태로 설계하고 이전 프로젝트에서 파악한 Spring 프레임워크의 구조와 흐름을 통해 개발을 진행했습니다.

데이터베이스의 여러 테이블의 데이터를 조합해 하나의 데이터로 탄생시키는 과정에서 많은 배움이 있었습니다. URL을 설계하면서 데이터를 어떻게 설계해서 구상해야 하는지 고민하는 과정이 이후의 프로젝트에도 큰 도움이 되었습니다.

또한 Memo 프로젝트에서 더욱 발전한 형태로 예외처리를 만들었습니다. Memo 프로젝트에서 문제가 발생했던 아이디 중복확인, 파일 용량 제한 등 실사용 상황에서 사용자가 마주할 수 있는 문제점들을 예측하고 그에 대한 대비책을 구상하고 설계했습니다.

에러가 발생하는 경우에는 Memo 프로젝트처럼 에러 메시지를 집중적으로 분석하고 데이터의 흐름에 대해서 파악했습니다. 또한 이클립스 디버깅을 활용해서 오류를 찾기도 했습니다. 한쪽에서의 에러를 해결하면 그 영향으로 다른 시스템에 문제가 생기는 경우들이 있었습니다. 이를 해결하기 위해서 당장 발생한 에러를 넘어서서 근본적인 문제점을 찾는데 중점을 뒀습니다. 시간이 다소 소요되었지만 전체적인 개발 시간을 단축하는데는 큰 도움이 되었습니다.

개발 과정 중간에 데이터베이스를 수정하고 코드를 수정하면서 기능에는 문제가 없지만 간결하게 줄일 수 있는 코드들이 보이게 되었습니다. 리팩토링 과정을 진행하면서 전체적인 구조를 간결하게 다듬고 이 또한 전체적인 개발 시간을 단축하는데 큰 도움이 되었습니다.

기획

Ovenapp.io 서비스로 페이지별 기획서 작성

<https://ovenapp.io/view/sNKgV2UzEsy15FKpggTP515NhaM2ISrd/>



Instakilogram

친구들의 사진과 동영상을 보려면
가입하세요.

아이디 중복확인

비밀번호

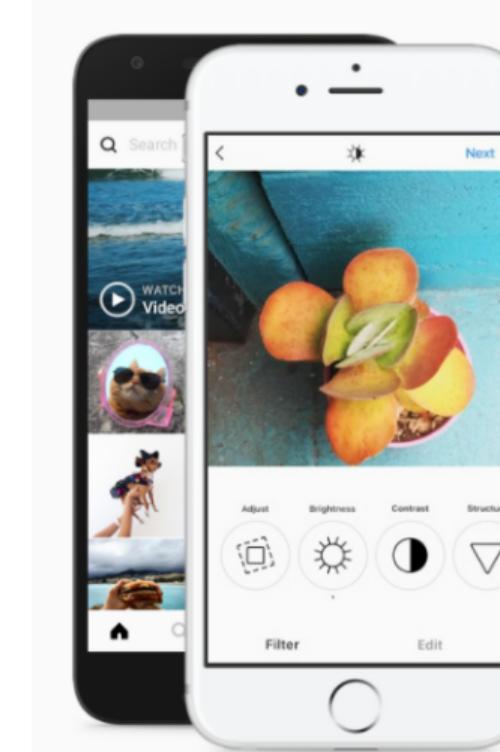
비밀번호 확인

이름

이메일

회원가입

[로그인하기](#)



Instakilogram

친구들의 사진과 동영상을 보려면
로그인하세요.

전화번호, 사용자 이름 또는 이메일

비밀번호

로그인

— 또는 —

계정이 없으신가요? [가입하기](#)

회원가입

로그인

Instakilogram

검색

sonji98 [로그아웃]

내용을 입력해주세요

업로드

sonji98

<- 클리시
삭제
버튼 출력

좋아요 18개
sonji98 나도 스카트 조던 갖고싶다...
2022-01-19 17:13

댓글
qpqp133 오 난 이미 샀지!
bakem169 오 저도 갖고싶어요

댓글 달기 게시

qpqp133

타임라인

DB 설계

Google spreadsheet로 데이터베이스 테이블 설계

컬럼명, 타입, Null 가능 여부, auto_increment 여부, 설명, 연결되는 컬럼 표시 등을 작성해서 표를 기반으로 테이블 생성 쿼리 작성



테이블 이름 : post				
설명 : 작성한 게시글을 저장하는 테이블				
들어갈 정보 : 게시글의 primary key, 게시글을 적는 사용자의 PK(primary key), 게시자의 이름, 게시글 내용과 사진, 생성, 수정 날짜				
컬럼명	타입	NULL가능 여부	auto_increment	설명
id	int	N	Y	primary key
userId	int	N	N	게시글 작성 사용자(user table의 id(primary key))
userName	varchar(16)	N	N	게시글 작성자 이름
content	text	N	N	게시글 내용
imagePath	varchar(128)	N	N	첨부 이미지, 접근 가능한 이미지 경로
createdAt	timestamp	N	N	생성 날짜
updatedAt	timestamp	N	N	수정 날짜

테이블 이름 : comment				
설명 : 게시글에 달린 댓글				
들어갈 정보 : 댓글의 primary key, 댓글이 적히는 게시글의 pk, 댓글 작성자의 pk와 이름, 댓글 내용과 생성, 수정 날짜				
컬럼명	타입	NULL가능 여부	auto_increment	설명
id	int	N	Y	primary key
postId	int	N	N	댓글을 올릴 게시글의 primary key
userId	int	N	N	댓글 작성자(user table의 id(primary key))
userName	varchar(16)	N	N	댓글 작성자 이름
content	varchar(256)	N	N	댓글 내용
createdAt	timestamp	N	N	생성 날짜
updatedAt	timestamp	N	N	수정 날짜

URL 설계

Google spreadsheet로 URL 설계

View를 위한 URL과 API를 위한 URL을 분리해서 설계

View URL에는 url과 parameter, API URL은 url, method, parameter, response data를 작성



View URL

View URL 설계			
제목	url	parameter	설명
로그인	/user/signin_view		로그인 화면
회원가입	/user/signup_view		회원가입 화면
메인화면	/post/timeline		메인화면 로그인 상태에서만 내용입력창과 좋아요버튼 나타남

URL 설계 (2)



API URL

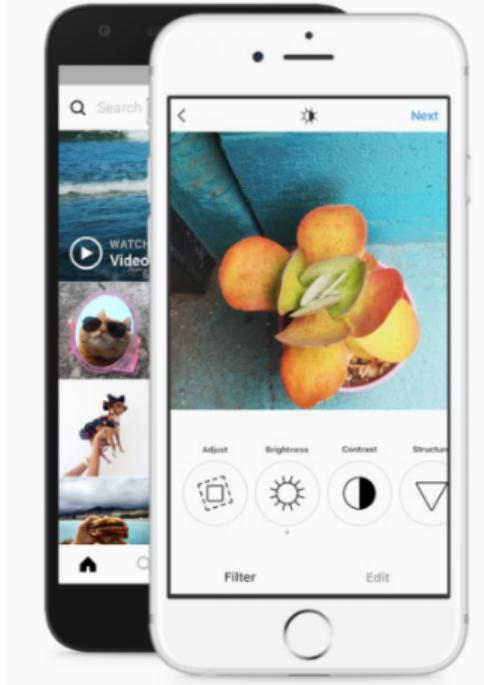
API URL 설계																							
1) 로그인																							
url : /user/sign_in method = post parameter																							
<table border="1"> <thead> <tr> <th>파라미터명</th><th>데이터 타입</th><th>NULL 여부</th><th>설명</th></tr> </thead> <tbody> <tr> <td>loginId</td><td>String</td><td>N</td><td>로그인 id</td></tr> <tr> <td>password</td><td>String</td><td>N</td><td>로그인 비밀번호</td></tr> </tbody> </table>				파라미터명	데이터 타입	NULL 여부	설명	loginId	String	N	로그인 id	password	String	N	로그인 비밀번호								
파라미터명	데이터 타입	NULL 여부	설명																				
loginId	String	N	로그인 id																				
password	String	N	로그인 비밀번호																				
응답값(json)																							
성공시 {"result" : "success"} 실패시 {"result" : "fail"}																							
2) 회원가입																							
url : /user/sign_up method : post parameter																							
<table border="1"> <thead> <tr> <th>파라미터명</th><th>데이터 타입</th><th>NULL 여부</th><th>설명</th></tr> </thead> <tbody> <tr> <td>loginId</td><td>String</td><td>N</td><td>로그인 id</td></tr> <tr> <td>password</td><td>String</td><td>N</td><td>로그인 비밀번호</td></tr> <tr> <td>name</td><td>String</td><td>N</td><td>사용자 이름</td></tr> <tr> <td>email</td><td>String</td><td>N</td><td>사용자 이메일</td></tr> </tbody> </table>				파라미터명	데이터 타입	NULL 여부	설명	loginId	String	N	로그인 id	password	String	N	로그인 비밀번호	name	String	N	사용자 이름	email	String	N	사용자 이메일
파라미터명	데이터 타입	NULL 여부	설명																				
loginId	String	N	로그인 id																				
password	String	N	로그인 비밀번호																				
name	String	N	사용자 이름																				
email	String	N	사용자 이메일																				
응답값(json)																							
성공시 {"result" : "success"} 실패시 {"result" : "fail"}																							

<https://docs.google.com/spreadsheets/d/1FNM9YvAoHEc203VA9yW8Ralwm7B0lAljhD5nUdRXurU/edit#gid=80182631>

구현 (1) - 회원가입과 로그인

20

Instakilogram



Instakilogram

친구들의 사진과 동영상을 보려면
가입하세요

miga2400
중복된 ID입니다.

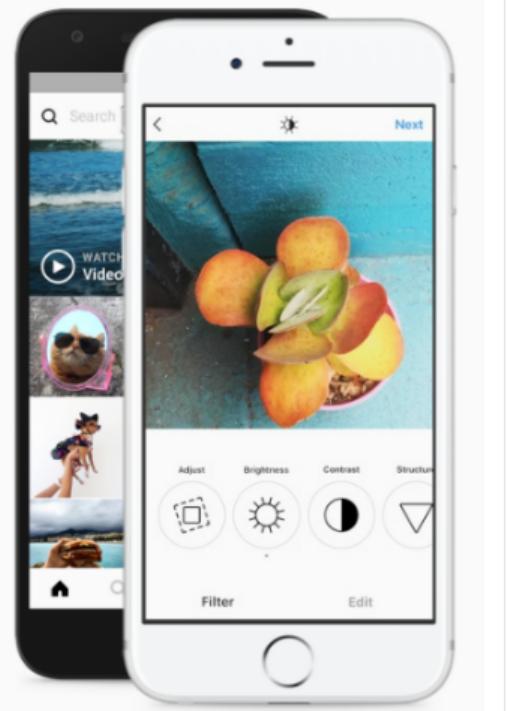
증복확인

비밀번호
비밀번호 확인
이름
이메일

회원가입

로그인하기

Instakilogram



Instakilogram

친구들의 사진과 동영상을 보려면
가입하세요

아이디
비밀번호

로그인

----- 또는 -----

계정이 없으신가요? [회원가입하기](#)

소개

- 사용자가 입력한 값을 ajax를 통해 DB의 user 테이블에 저장해서 유저의 정보를 insert한다.
- 로그인 이후에는 session을 활용해서 header 부분에 유저의 이름과 로그아웃 정보가 나오도록 했다.
- form을 활용해서 submit방식을 통해 마우스 클릭이 아닌 enter키를 타이핑하는 방식으로도 로그인이 가능하도록 했다.
- 로그인, 회원가입시 input에 공백이 있는 경우 alert를 띄워 사용자가 알 수 있도록 했다.
- 아이디를 중복체크하는 API를 만들어 중복이 발생하는 경우 빨간 글씨를 출력하게 했다.

troubleshooting

- PermissionInterceptor와 session을 활용해서 비 로그인 상태에서는 타임라인에 접근 못하게, 로그인 상태에서는 로그인화면과 회원가입 화면에 접근 못하게 설정했다.
- 아이디 중복상태에서 재입력시 빨간 글씨가 사라지지 않는 현상이 있었다. 편의성을 위해 아이디값에 입력이 있으면 중복체크 상태를 초기화시켰다.

보완점

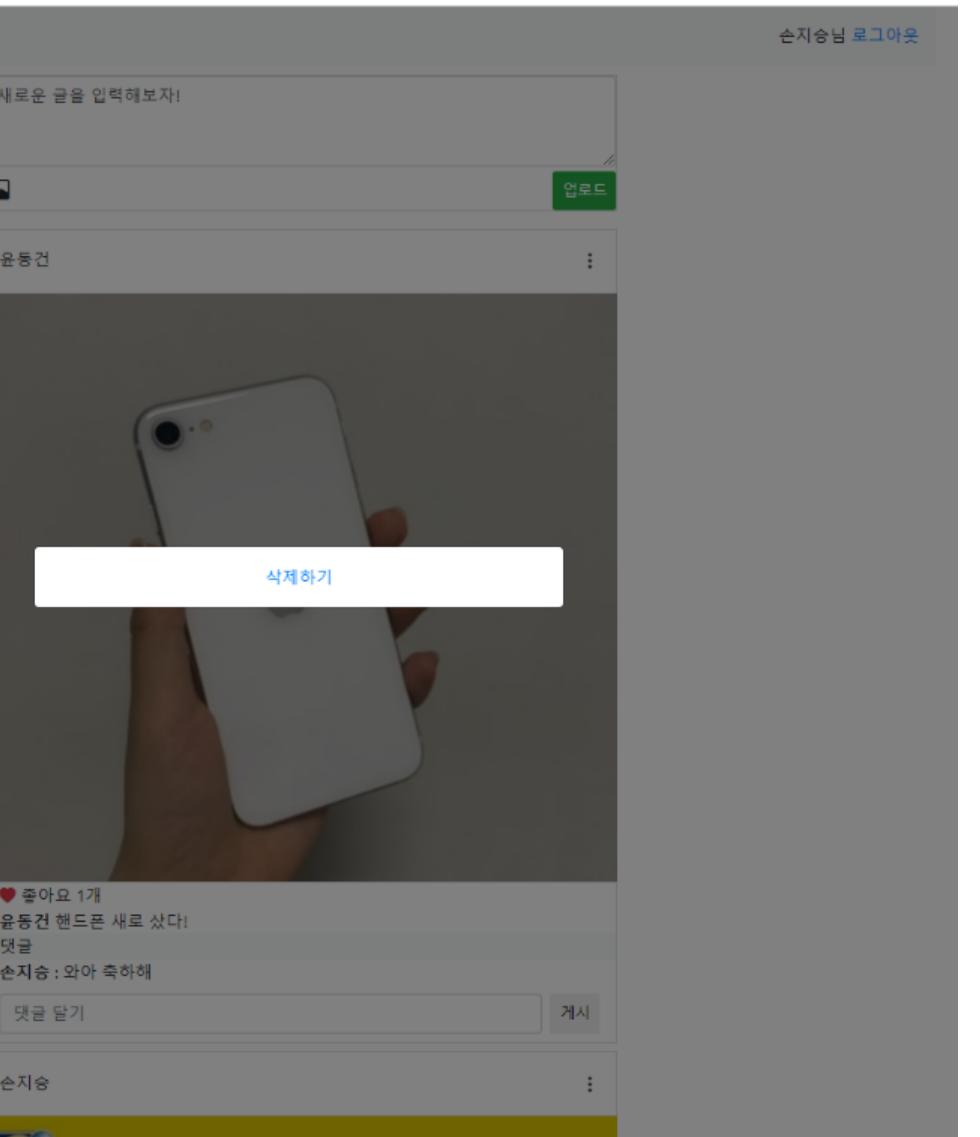
구현 (2) - 타임라인

21

Instakilogram



Instakilogram



소개

- 사용자가 업로드한 내용이 카드 형식으로 타임라인에 나타난다.
- 상단에 위치한 입력창으로 내용을 입력하는데 이미지 파일 첨부시에 input file버튼을 클릭하는게 아닌 Bootstrap의 icon을 활용해 아이콘이 input file과 같은 이벤트를 처리하게 했다.
- ajax를 활용해 게시글 입력, 댓글, 좋아요 기능을 구현했다.
- Bootstrap의 modal 기능을 활용해서 더보기 버튼을 클릭할 경우 삭제가 가능하도록 버튼이 나오도록 했다.
- 게시글을 작성한 유저만 삭제가 가능하도록 처리했다.
- 게시글 삭제시에 서버에 저장되어 있던 파일과 디렉터리를 함께 삭제하도록 했다.

troubleshooting

- 댓글과 좋아요, 게시글을 각각의 객체로 저장했더니 게시글 하나의 댓글을 보기 위해 데이터베이스의 모든 댓글을 가져오는 문제가 있었고 시간이 지연되는 것을 확인했다.
- 게시글을 중심으로 게시글의 댓글과 좋아요 여부를 하나의 객체로 저장할 수 있도록 클래스를 재설계했고 게시글마다 댓글을 조회하도록 변경하고 댓글과 게시글을 묶는 클래스를 만들었다.

보완점

- 실제 Instagram처럼 다중 파일 업로드 기능

Project SHUSED

2022.02

SHUSED 프로젝트 소개

현재 MZ세대의 관심도가 높은 스니커스 시장과 관련된 서비스를 이용하던 중 중고 신발을 전문으로 중계해주는 웹 서비스가 없는 것을 파악했습니다. 또한 신발 거래 뿐만이 아닌 커뮤니티 기능도 포함되어 있어 스니커즈 매니아들의 모임 장소로 활용할 수 있는 서비스입니다.

유저의 로그인을 기반으로 **중고 거래와 커뮤니티 기능**을 구현하는데 중점을 뒀습니다.

주요한 기능으로는 **판매 신발 등록, 신발 검색, 판매자 정보 확인, 커뮤니티 게시글 입력, 댓글 입력, 추천 기능, 삭제 기능**이 있습니다.

SHUSED

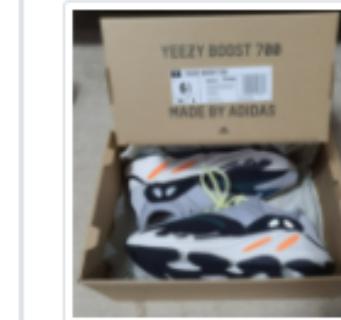


SELL

COMMUNITY

Recently Added

최근 등록 상품



adidas

Yeezy Boost 700 Wave Ru...

사이즈 : 285

상태 : 새 것

판매가 : 500000원

거래방식 : 택배거래

판매여부 : 판매중



nike

Peaceminusone Kwondo1 W...

사이즈 : 260

상태 : 보통

판매가 : 306000원

거래방식 : 직거래

판매여부 : 판매중



newbalance

992 Made in USA Grey

사이즈 : 235

상태 : 새 것

판매가 : 433000원

거래방식 : 택배거래

판매여부 : 판매중



jordan

Jordan 1 Mid Light Smok...

사이즈 : 255

상태 : 새 것

판매가 : 261000원

거래방식 : 직거래

판매여부 : 판매중



adidas

Yeezy Boost 350 V2 Zebr...

사이즈 : 260

상태 : 새 것

판매가 : 529000원

거래방식 : 택배거래

판매여부 : 판매중



nike

Dunk

사이즈 :

상태 :

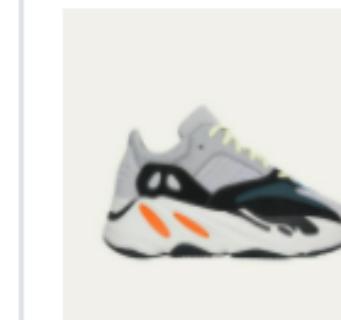
판매가 :

거래방식 :

판매여부 :

Just Dropped

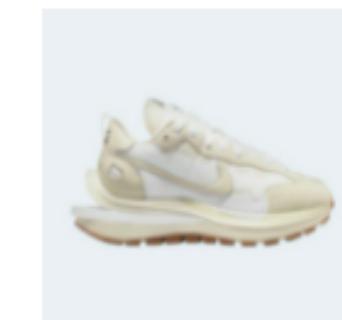
최근 발매 상품



adidas

B75571

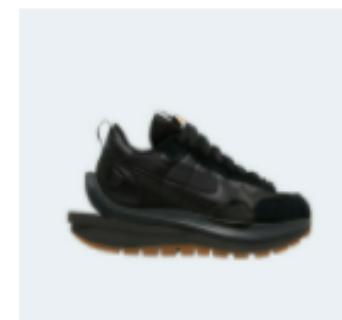
Yeezy Boost 700 Wave Ru...



nike

DD1875-100

Sacai VaporWaffle White...



adidas

DD1875-001

Sacai VaporWaffle Black...



adidas

DH2482-100

Peaceminusone Kwondo1 W...



nike

DJ6188-001

Dunk Low Retro Wolf Gre...

SHUSED 프로젝트를 진행하며 느낀점

Instakilogram 프로젝트 이후 조금은 더 복잡한 프로젝트를 진행해보고 싶다는 욕심이 생겼습니다. 정해진 하나의 서비스가 있는 형태가 아닌 SHUSED처럼 신발 거래 플랫폼과 스니커즈 매니아들이 서로 소통할 수 있는 창구를 만들어보고 싶다는 생각을 했고 바로 설계를 시작했습니다. 이전 프로젝트를 진행하면서 **모듈화가 얼마나 중요한지 깨달았기 때문에 설계 과정에서부터 모듈화에 집중**해서 진행했습니다.

Controller Service Repository로 이어지는 구조에서 한번 사용했던 코드를 재활용해 손실을 줄였고 객체 간의 연결에 중점을 두고 프로젝트를 진행했습니다.

이전에 했던 프로젝트보다 훨씬 큰 서비스를 제작했기 때문에 역시 에러상황 또한 이전보다 많이 발생했습니다. 특히 중고 신발 거래 플랫폼인 만큼 **사진을 여러장 업로드** 하는 것이 필수적으로 들어가야 한다고 생각했는데, 이 부분에서 **구글링**을 통한 문제 해결 능력을 기를 수 있었습니다. 신발과 여려장의 이미지를 하나의 테이블로 관리하려고 했지만 리스트 형태인 이미지의 특성상 같은 테이블에서 관리하면 자연이 발생했기 때문에 테이블을 나눠서 관리했습니다.

처음에는 국내 블로그의 다중파일 업로드 관련된 글들을 보다가 한계를 느끼고 **구글과 stackoverflow의 영문자료**를 참고해서 문제를 해결했습니다. 같은 방법이더라도 **오픈소스의 버전**에서 생기는 차이점도 존재해서 이전에 올라온 자료와 최신자료를 **크로스 체크**하면서 명확한 검증을 했습니다.

또한 개발 진행 상황을 **spreadsheet**에 표로 만들어 관리하고 **최대한 기록**했습니다. Github와 연동한 **zenhub** 툴을 활용해서 개발을 진행해야 할 기능들을 **issue**로 정리해서 **등록**하고 하루하루 기록해나갔습니다. 진행 과정에서 에러가 발견되고 이를 해결 한 경우에도 **기록**을 해 놓아서 나중에 비슷한 오류가 발생했을 경우 git의 commit 메시지로 기록을 확인함으로써 오류를 쉽게 해결할 수 있었습니다.

기획

Ovenapp.io 서비스로 페이지별 기획서 작성

<https://ovenapp.io/project/XiEN5FFegj0XQLQYRhHDWd7ggtBFnfj#njdS5>



SHUSED

관심상품 마이페이지 로그인

아이디
비밀번호
비밀번호 확인
비밀번호가 일치하지 않습니다.
닉네임
이미 사용 중인 닉네임입니다.
전화번호
ex) 010-0000-0000
형식이 일치하지 않습니다.
이미 사용 중인 전화번호입니다.
중복확인
회원가입
또는
이미 계정이 있으신가요? [로그인](#)

메인페이지

SHUSED

SELL 클리시 - 판매 물품 등록 화면으로 이동
COMMUNITY 클리시 - 커뮤니티 메인으로 이동

NIKE adidas new balance

SELL **COMMUNITY**

Recently Added
최근 등록 상품

- Nike Dunk Low Retro 사이즈 : 290 상태 : 좋음 판매가 : 200,000원 거래 방식 : 택배거래
- New Balance New Balance 993 USA 사이즈 : 260 상태 : 나쁨 판매가 : 120,000원 거래 방식 : 직거래
- Adidas Yeezy Boost 350 V2 사이즈 : 240 상태 : 보통 판매가 : 170,000원 거래 방식 : 직거래
- Jordan Jordan 1 High 85 Navy 사이즈 : 300 상태 : 좋음 판매가 : 200,000원 거래 방식 : 택배거래

Recently Dropped
최근 발매 상품

- Nike Dunk Low Retro 발매일 : 2022-02-22
- New Balance New Balance 993 USA 발매일 : 2022-02-21
- Adidas Yeezy Boost 350 V2 발매일 : 2022-02-20
- Jordan Jordan 1 High 85 Navy 발매일 : 2022-02-19
- New Balance New Balance 327 발매일 : 2022-02-18

SHUSED

판매정보

카테고리
NIKE adidas new balance Jordan

모델번호 ex) DD1391-100

제품명 ex) Dunk Low Retro Black

사이즈 숫자만 입력해주세요

가격 숫자만 입력해주세요 원

상태 새것 거의 새것 보통 나쁨

거래방식 직거래 매페이지

거래지역 ex) 서울 강남구

상품설명 상품 설명을 입력해주세요. (10글자 이상) ex) 구입 시기, 적용 혜수

커뮤니티

SHUSED

COMMUNITY

번호	제목	작성일	글쓴이
6421	정기풀 문의	2022-01-28	신발조양
5217	신발이 너무 비싸	2021-07-11	신발조양
4452	신발 추천해드립니다.	2021-07-10	추천전문가
3153	가입 인사 올립니다.	2021-03-12	신발조양

글쓰기

일정 설계

Google spreadsheet로 프로젝트 진행 상황을 손쉽게 확인할 수 있도록 일정 설계표를 작성했다.



프로젝트 진행 상황						
담당	내용	시작일	종료일	기간	진행상황	
		2022-02-21	2022-02-22	2	100%	
기획	SHUSED 서비스 기획	2022-02-21	2022-02-22	2	100%	
설계	DB 설계	2022-02-22	2022-02-23	2	100%	
설계	URL 설계	2022-02-23	2022-02-23	1	100%	
구현	회원가입 view 구성	2022-02-24	2022-02-25	2	100%	
구현	회원가입 api 구성	2022-02-25	2022-02-28	2	100%	
구현	로그인 view, api, 로그아웃 api 구성	2022-02-28	2022-02-28	1	100%	
구현	최근 발매 페이지 view 구성	2022-03-01	2022-03-02	2	100%	
구현	최근 발매 페이지 api 구성	2022-03-02	2022-03-04	3	100%	
구현	메인페이지 view 구성	2022-03-03	2022-03-10	5	100%	
구현	최근 발매 상품 select api 구성	2022-03-04	2022-03-04	1	100%	
구현	판매 물품 들록 화면 view 구성	2022-03-07	2022-03-10	2	100%	
구현	판매 물품 들록 api 구성	2022-03-10	2022-03-11	2	100%	
구현	판매 물품 들록 select api 구성	2022-03-11	2022-03-14	2	100%	
구현	판매 물품 상세보기 view 구현	2022-03-11	2022-03-14	2	100%	
구현	판매 물품 상세보기 api 구현	2022-03-14	2022-03-15	2	100%	
구현	커뮤니티 메인페이지 main view 구현	2022-03-16	2022-03-16	1	100%	
구현	커뮤니티 게시글 입력 createPost view 구현	2022-03-16	2022-03-16	1	100%	
구현	커뮤니티 메인페이지 api 구현	2022-03-16	2022-03-16	1	100%	
구현	커뮤니티 게시글 입력 createPost api 구현	2022-03-16	2022-03-16	1	100%	
구현	커뮤니티 게시글 상세보기 detailPost api 구현	2022-03-16	2022-03-16	1	100%	
구현	커뮤니티 게시글 상세보기 detailPost view 구현	2022-03-16	2022-03-16	2	100%	
구현	커뮤니티 게시글 상세보기 추천 api 구현	2022-03-17	2022-03-17	1	100%	
구현	커뮤니티 게시글 상세보기 댓글입력 api 구현	2022-03-17	2022-03-17	1	100%	
구현	커뮤니티 게시글 상세보기 댓글 select api 구현	2022-03-17	2022-03-17	1	100%	
구현	커뮤니티 게시글 삭제(추천, 댓글 포함) api 구현	2022-03-18	2022-03-18	1	100%	
구현	상품 구매하기 버튼 클릭시 판매자 정보 확인 구현	2022-03-18	2022-03-18	1	100%	
구현	마이페이지 view 구현	2022-03-18	2022-03-21	2	100%	
구현	마이페이지에 내 커뮤니티 게시글, 판매상품 블러오기	2022-03-21	2022-03-21	1	100%	
구현	마이페이지에 내 판매상품 블러오기, 삭제 api 구현	2022-03-21	2022-03-21	1	100%	
구현	판매물품 상태변경 api 구현	2022-03-22	2022-03-22	1	100%	
구현	검색 기능 api 구현	2022-03-22	2022-03-22	1	100%	
구현	검색화면 view 구현	2022-03-22	2022-03-23	1	100%	
구현	다중 파일 업로드	2022-03-23	2022-03-28	4	100%	

DB 설계

Google spreadsheet로 데이터베이스 테이블 설계

컬럼명, 타입, Null 가능 여부, auto_increment 여부, 설명, 연결되는 컬럼 표시 등을 작성해서 표를 기반으로 테이블 생성 쿼리 작성

테이블 이름 : user				
컬럼명	타입	NULL 가능 여부	auto_increment	설명
id	int	N	Y	user의 primary key
loginId	varchar(16)	N	N	사용자 id
password	varchar(32)	N	N	사용자 비밀번호, 암호화 저장
nickname	varchar(16)	N	N	사용자 닉네임
phoneNumber	varchar(16)	N	N	사용자 전화번호
createdAt	timestamp	N	N	생성 날짜
updatedAt	timestamp	N	N	수정 날짜

테이블 이름 : usedShoes				
컬럼명	타입	NULL 가능 여부	auto_increment	설명
id	int	N	Y	usedShoes의 primary key
userId	int	N	N	판매자 id (user table의 id(primary key))
nickname	varchar(16)	N	N	판매자 닉네임 (user table의 nickname)
phoneNumber	varchar(16)	N	N	판매자 전화번호 (user table의 phoneNumber)
category	varchar(16)	N	N	신발 카테고리
modelNumber	varchar(64)	N	N	신발 모델 번호
shoesName	varchar(128)	N	N	신발 제품명
size	int	N	N	신발의 사이즈
price	int	N	N	신발의 가격
condition	varchar(16)	N	N	신발의 상태
dealMethod	varchar(16)	N	N	거래 방식
place	varchar(32)	Y	N	거래 지역
explanation	text	N	N	신발 설명
state	int	N	N	판매완료 상태여부
imagePath	varchar(128)	N	N	첨부 이미지, 접근 가능한 이미지 경로
createdAt	timestamp	N	N	생성 날짜



URL 설계

Google spreadsheet로 URL 설계

View를 위한 URL과 API를 위한 URL을 분리해서 설계

View URL에는 url과 parameter, API URL은 url, method, parameter, response data를 작성



View URL

View URL 설계			
제목	url	parameter	설명
로그인	/user/signin_view		로그인 화면
회원가입	/user/signup_view		회원가입 화면
마이페이지	/user/mypage_view		마이페이지 화면 로그인 상태에서만 접근 가능
메인화면	/mainPage_view		메인화면 상품과 커뮤니티 상세보기, 글쓰기에 접근하기 위해서는 로그인 상태에서만 접근 가능
판매 물품 등록	/shoes/usedShoes_view		판매 물품 등록 화면 로그인 상태에서만 접근 가능
최근 발매 상품 등록	/shoes/droppedShoes_view		최근 발매 상품 등록 화면 로그인 상태에서만 접근 가능
물품 검색	/shoes/search_view	id : 해당 물품들의 id / 검색한 내용	물품 검색 화면 로그인 상태에서만 접근 가능
물품 상세보기	/shoes/detail_view	id : 해당 물품의 id	물품 상세보기 화면 로그인 상태에서만 접근 가능
커뮤니티 메인화면	/community_main_view		커뮤니티 메인 화면
커뮤니티 글쓰기	/community_create_view		커뮤니티 글쓰기 화면 로그인 상태에서만 접근 가능
커뮤니티 상세	/community_detail_view	id : 클릭한 게시글의 id	커뮤니티 게시글 상세화면 로그인 상태에서만 접근 가능

URL 설계 (2)



API URL

API URL 설계																							
1) 로그인																							
url : /user/sign_in method : post parameter																							
<table border="1"> <thead> <tr> <th>파라미터명</th><th>데이터 타입</th><th>NULL 여부</th><th>설명</th></tr> </thead> <tbody> <tr> <td>loginId</td><td>String</td><td>N</td><td>로그인 id</td></tr> <tr> <td>password</td><td>String</td><td>N</td><td>로그인 비밀번호</td></tr> </tbody> </table>				파라미터명	데이터 타입	NULL 여부	설명	loginId	String	N	로그인 id	password	String	N	로그인 비밀번호								
파라미터명	데이터 타입	NULL 여부	설명																				
loginId	String	N	로그인 id																				
password	String	N	로그인 비밀번호																				
응답값(json)																							
성공시 {"result" : "success"} 실패시 {"result" : "fail"}																							
2) 회원가입																							
url : /user/sign_up method : post parameter																							
<table border="1"> <thead> <tr> <th>파라미터명</th><th>데이터 타입</th><th>NULL 여부</th><th>설명</th></tr> </thead> <tbody> <tr> <td>loginId</td><td>String</td><td>N</td><td>회원가입 id</td></tr> <tr> <td>password</td><td>String</td><td>N</td><td>회원가입 비밀번호</td></tr> <tr> <td>nickname</td><td>String</td><td>N</td><td>회원가입 닉네임</td></tr> <tr> <td>phoneNumber</td><td>String</td><td>N</td><td>회원가입 전화번호</td></tr> </tbody> </table>				파라미터명	데이터 타입	NULL 여부	설명	loginId	String	N	회원가입 id	password	String	N	회원가입 비밀번호	nickname	String	N	회원가입 닉네임	phoneNumber	String	N	회원가입 전화번호
파라미터명	데이터 타입	NULL 여부	설명																				
loginId	String	N	회원가입 id																				
password	String	N	회원가입 비밀번호																				
nickname	String	N	회원가입 닉네임																				
phoneNumber	String	N	회원가입 전화번호																				
응답값(json)																							
성공시 {"result" : "success"} 실패시 {"result" : "fail"}																							

구현 (1) - 회원가입과 로그인

SHUSED

로그인

SHUSED
I've never been afraid to fail.

아이디

비밀번호

로그인

또는

계정이 없으신가요? [회원가입](#)

SHUSED

SHUSED
I've never been afraid to fail.

아이디

비밀번호

비밀번호 확인

닉네임

전화번호
 ex) 010-0000-0000

회원가입

또는

이미 계정이 있으신가요? [로그인](#)

소개

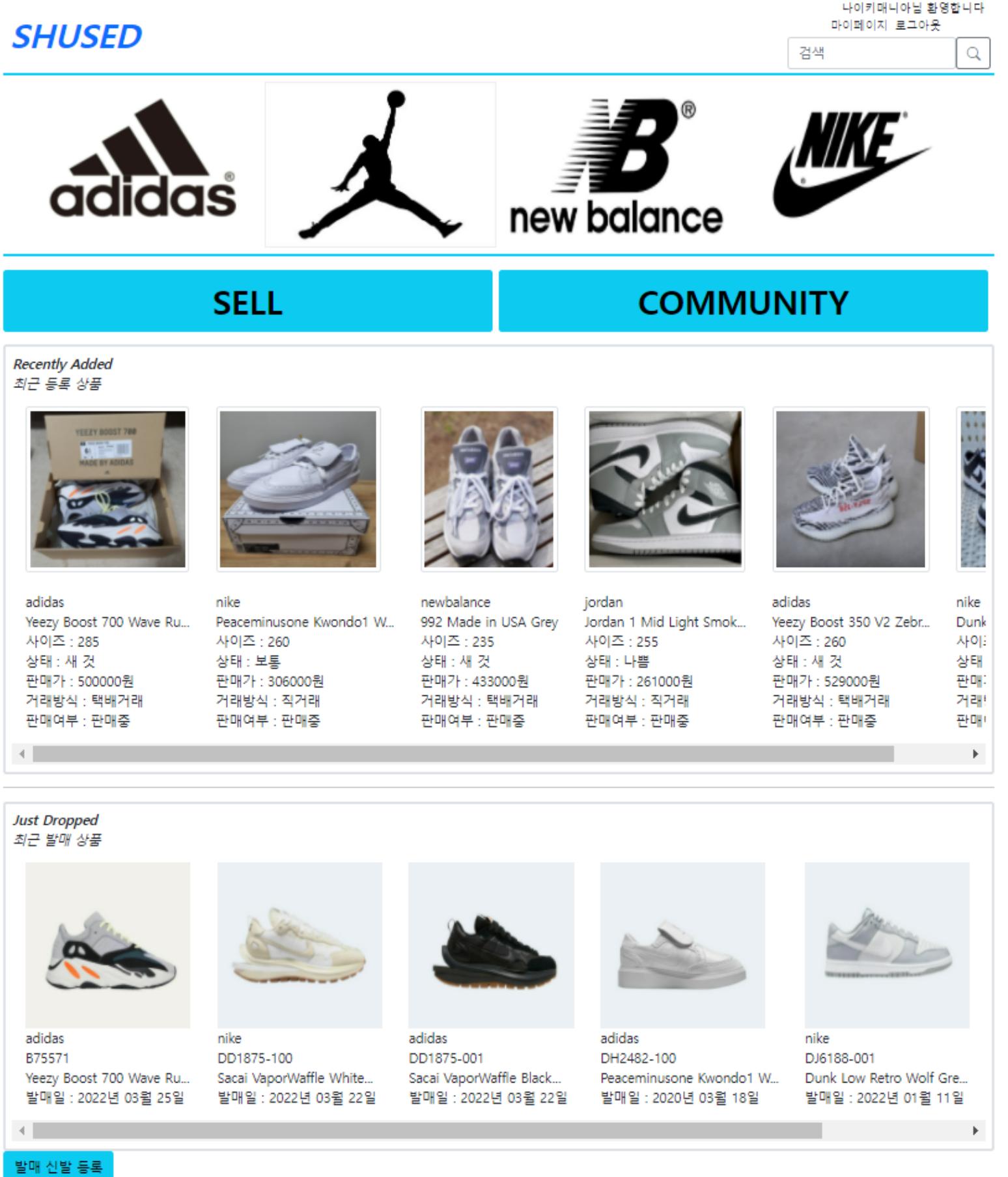
- 사용자가 입력한 값을 ajax를 통해 DB의 user 테이블에 저장해서 유저의 정보를 insert한다.
- 로그인 이후에는 session을 활용해서 header 부분에 유저의 닉네임과 로그아웃 정보가 나오도록 했다.
- form을 활용해서 submit방식을 통해 마우스 클릭이 아닌 enter키를 타이핑하는 방식으로도 로그인이 가능하도록 했다.
- id과 닉네임, 전화번호의 중복체크 API를 각각 생성하고 버튼 클릭시에 사용 가능한 경우에는 버튼 색상이 변경되어 쉽게 볼 수 있도록 했다. 또한 전화번호에 형식을 지정했다.
- 사용자의 비밀번호는 md5 알고리즘을 통해 데이터베이스에 암호화해서 저장했다.

troubleshooting

보완점

- 아이디 / 비밀번호 찾기 기능

구현 (2) - 메인페이지



- 메인화면으로 로그인 유저 정보, 검색버튼, 카테고리, 판매와 커뮤니티 버튼, 최근 등록 신발, 최근 발매 상품으로 구성되어 있다.

- 각각의 카테고리 버튼은 value를 삽입해서 카테고리를 클릭할 경우 value값이 포함된 신발들을 검색해서 검색결과view로 넘어가게 했다.

- 등록된 중고신발을 최근순으로 정렬해서 메인 페이지에 Bootstrap을 이용한 가로 스크롤로 확인할 수 있도록 했다.
이미지에 신발의 pk값이 포함된 링크를 적용해 이미지를 클릭할 경우 그 신발의 상세정보로 이동하게 했다.

- 최근 발매 신발도 최근순으로 정렬해서 가로 스크롤로 나타냈다. 신발의 이미지에 신발의 ModelName을 검색하는 링크를 적용해 이미지를 클릭해서 검색할 수 있도록 했다.

- 검색기능도 submit방식을 활용해 enter키로 동작할 수 있도록 했다.

소개

troubleshooting

보완점

- 등록된 신발에는 리스트 형태로 이미지의 경로가 삽입되어 있는데 사진을 한장만 불러오는데 어려움이 있었다. sql의 GROUP BY를 활용해서 같은 id값을 가진 신발의 이미지를 하나로 통합해서 해결할 수 있었다.

구현 (3) - 최근 발매 물품 등록

SHUSED

발매정보

카테고리



모델번호

DD1391-100

제품명

Dunk Low Retro

발매일자

2022-03-11

제품 사진

2022	년	3월				
일	월	화	수	목	금	토
1	2	3	4	5		
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

파일 선택 yeezy4.png



등록하기

나이키매니아님 환영합니다
마이페이지 로그아웃

검색

소개	<ul style="list-style-type: none"> 최근 발매한 신발의 정보를 등록하는 화면으로 로그인한 사용자라면 누구나 등록할 수 있다. 각각의 입력 값들에 value값과 input값을 첨부한 파일과 함께 formData로 뭉어 ajax를 통해 데이터베이스에 저장했다. datepicker 활용해서 날짜 정보 등록하고 이후 JSTL을 통해 메이네이지 형식에 맞게 FORMATTING을 했다.
troubleshooting	
보완점	

구현 (4) - 판매 물품 등록

SHUSED

회원님 환영합니다
マイページ 로그아웃

검색

발매정보

카테고리

- adidas
- Jordan
- new balance
- NIKE

모델번호: DD1391-100

제품명: Dunk Low Retro

사이즈: 290

가격: 200000 원

상태

- 새 것
- 거의 새 것
- 보통
- 나쁨

거래방식

- 직거래
- 택배거래

제품설명

정말 딱 두번 신었습니다.
흙길에서 걸은 것도 아니라 상태 정말 좋습니다.
실착한 경험 있어서 시세보다 저렴하게 판매합니다.
직거래는 강남역 1번출구에서 가능합니다.

제품사진

파일선택 파일 3개

등록하기

소개	<ul style="list-style-type: none"> ▪ 판매 물품 등록 페이지로 중고 물품 특성에 맞게 다중 파일 업로드 기능과 썸네일 미리보기 기능을 구현했다. ▪ 각각의 값들에 value와 id를 부여해서 ajax를 통해 session에서 받아온 유저의 정보와 함께 데이터베이스에 저장할 수 있도록 했다. 신발의 정보와 이미지 파일을 별도의 객체로 저장했다. ▪ 직거래 버튼을 클릭한 경우에만 거래지역을 입력하는 input이 나오도록 했다.
troubleshooting	<ul style="list-style-type: none"> ▪ 다중파일 업로드를 구현하는 과정에서 어려움이 많았다. MultipartFile을 리스트 형태로 가져와서 FileManagerService를 통해 파일 경로의 리스트로 가공한 다음 이를 sql문에서 반복문을 실행해 데이터베이스에 저장하는 방법으로 해결했다.
보완점	<ul style="list-style-type: none"> ▪ 신발의 각 부분을 올릴 수 있도록 설정

구현 (5) - 판매 물품 상세보기

SHUSED

상품정보

모델번호 DD1391-100

제품명 Dunk Low Retro

가격 200000 원

사이즈 290

상태 새 것

거래방식 직거래

상품설명 정말 딱 두번 신었습니다. 흙길에서 걸은 것도 아니라 상태 정말 좋습니다. 실착한 경험 있어서 시세보다 저렴하게 남액 1번풀구에서 가능합니다.

상품이미지



[상품 삭제하기](#)

[판매완료 설정하기](#)

SHUSED

상품정보

모델번호 DH2482-100

제품명 Peaceminusone Kwondo1 White

가격 306000 원

사이즈 260

상태 보통

거래방식 직거래

거래지역 문산역

상품설명 나이키와 지드래곤의 협업상품으로 다시는 안 나올 한정판 신발입니다. 나중에 두면 분명 가치가 오를 신발이라고 생각합니다. 직거래 선호합니다.

상품이미지



판매가 완료된 상품입니다.

소개	<ul style="list-style-type: none"> 판매 물품의 정보를 리스트 형식으로 보여주고 판매 물품 상세보기 페이지로 삭제, 판매완료 상태로 변경, 구매하기 버튼이 있다. 신발을 등록한 유저만 신발을 삭제하거나 판매완료 상태로 변경할 수 있다. 판매완료 상태로 변경되면 구매하기 버튼이 사라지도록 했다. 신발 구매하기 버튼을 누를 경우 Bootstrap의 modal 기능을 활용해 다시 한번 의사를 묻고 판매를 등록한 유저의 연락처가 modal에 나타나게 했다.
troubleshooting	<ul style="list-style-type: none"> 신발의 상세정보를 클릭했을 때 신발의 정보를 저장한 객체와 이미지의 경로를 저장하고 있는 객체에 동시에 접근해야 했다. 리스트 형태로 저장되어 있는 객체를 불러오는데 어려움이 있었지만 리스트를 model에 담아 JSTL의 반복문을 통해 사용했다.
보완점	<ul style="list-style-type: none"> 구매하기 클릭시 채팅기능 구현

구현 (6) - 검색

34

SHUSED

검색어 : **nike**

	
판매 ID : 24	판매 ID : 28
Dunk Low Retro	Peaceminusone Kwondo...
카테고리 : nike	카테고리 : nike
모델명 : DD1391-100	모델명 : DH2482-100
사이즈 : 290	사이즈 : 260
가격 : 200000원	가격 : 306000원
거래방법 : 직거래	거래방법 : 직거래
판매여부 : 판매중	판매여부 : 판매완료
상세보기	상세보기

나이키매니아님 환영합니다
마이페이지 로그아웃

검색



소개

- 검색어를 입력하거나 카테고리 클릭, 발매정보 클릭을 통해 이동한 신발 검색화면이다.
- BootStrap의 card 기능을 활용해서 사용자가 보기 좋게 배치했다.
- 특정 단어가 포함된 데이터를 데이터베이스에서 select하기 위해서 sql문의 LIKE CONCAT 속성을 활용했다.

troubleshooting

- 상세보기와 마찬가지로 상세정보를 클릭했을 때 신발의 정보를 저장한 객체와 이미지의 경로를 저장하고 있는 객체에 동시에 접근해야 했다.
- sql문의 GROUP BY를 활용해 같은 ID값을 가진 신발들을 하나로 묶어 이미지를 한장만 가져올 수 있었다.

보완점

구현 (7) - 마이페이지

SHUSED

판매 ID : 28	판매 ID : 29
Peaceminusone Kwondo...	Yeezy Boost 700 Wave...
카테고리 : nike	카테고리 : adidas
모델명 : DH2482-100	모델명 : 875571
사이즈 : 260	사이즈 : 285
가격 : 306000원	가격 : 500000원
거래방법 : 직거래	거래방법 : 택배거래
판매여부 : 판매완료	판매여부 : 판매중

[상세보기](#) [삭제하기](#) [상세보기](#) [삭제하기](#)

SHUSED

NO.	제목	작성일
11	요즘 어떤 신발이 유행인가요?	2022-03-31 04:32:30
12	신발 밀창이 떨어졌습니다	2022-03-31 04:33:20

나이키매니아님 환영합니다
마이페이지 로그아웃
 검색

나이키매니아님 환영합니다
마이페이지 로그아웃
 검색

- 마이페이지 화면으로 내가 등록한 신발과 내가 쓴 커뮤니티의 게시글의 목록을 한 화면에서 확인할 수 있도록 했다. 각 버튼을 클릭하면 버튼에 맞는 내용을 확인 할 수 있다.

소개

- BootStrap의 card 기능을 활용해서 사용자가 보기 좋게 배치했다.
- 신발 목록의 삭제버튼을 누르면 상세보기로 들어가지 않고도 신발을 삭제할 수 있도록 했다.

troubleshooting

- 마이페이지로 이동하면서 신발의 정보와 커뮤니티의 정보에 동시에 접근해야하는 문제가 있었다. session에 저장되어 있는 유저의 pk값으로 select해서 불러오도록 했다.

보완점

구현 (8) - 커뮤니티

SHUSED

COMMUNITY

NO.	제목	작성일	작성자
13	요즘 멍크 시세 어떤가요?	2022-03-31 06:18:15	호빵편지
12	신발 밀창이 떨어졌습니다	2022-03-31 04:33:20	나이키매니아
11	요즘 어떤 신발이 유행인가요?	2022-03-31 04:32:30	나이키매니아

[글쓰기](#)

SHUSED

COMMUNITY

제목 : [요즘 어떤 신발이 유행인가요?](#)

최근주에 이지 700구입했는데 생각보다 안 예뻐서 팔고 다른 신발 구입할까 고민중입니다. 예쁜 신발 추천 부탁드려요



작성자 | 내용 | 작성일자

호빵편지 | 이지 700 예쁜 신발입니다. 다른거 사신다면 초년이 요즘 예쁘게 잘 나오더라구요 | 2022-03-31 04:33:59

댓글 달기 | 게시 | 삭제

[목록으로](#)

소개	<ul style="list-style-type: none"> 유저들이 게시글을 작성하는 커뮤니티로 제목과 내용, 이미지파일 업로드, 추천과 댓글 달기를 할 수 있다. 작성자만 글 삭제가 가능하다. 첨부파일 버튼을 Bootstrap의 icon이 이벤트를 받도록 했다. ajax를 통해 게시글 입력, 댓글과 추천기능을 구현했고 각 게시글마다 댓글과 추천기능을 하나의 객체로 저장하는 클래스로 만들었다.
	troubleshooting <ul style="list-style-type: none"> 이전 프로젝트와 마찬가지로 게시글 하나를 확인하기 위해 모든 추천, 댓글 버튼을 불러오면 자연이 발생하는 문제가 있었고 요소들을 하나의 객체로 묶어 해결했다.
보완점	<ul style="list-style-type: none"> 추천이 가장 많은 글 상단에 노출

24시간이 부족한 개발자
손지승입니다.

감사합니다.

CONTACT
miga98@naver.com
010-2672-1095