

기획력과 리더십을 겸비한 개발자, 권순형입니다.

최종학력	서울시립대학교 전자전기컴퓨터공학부
자격증	SQLD
배포한 앱	가계부자
git	https://github.com/SunHyongKwon
notion	https://neuroo.notion.site/

목 차[**1 사용한 프로그램**](#)[**2 App 프로젝트**](#)[**2.1 Flutter 프로젝트**](#)[**2.2 Swift 프로젝트**](#)[**3 App에 사용한 View Model**](#)[**3.1 Flutter**](#)[**3.2 Swift**](#)[**4 App에 사용한 DB**](#)[**4.1 RDBMS : MySQL, SQLite**](#)[**4.2 NoSQL : Firebase**](#)[**5 App에 사용한 Backend**](#)[**5.1 Spring Boot**](#)[**5.2 Flask**](#)[**6 App에 사용한 Open API**](#)[**7 App에 사용한 AI 모델**](#)[**7.1 강화학습을 이용한 딥러닝 모델링**](#)[**7.2 머신러닝 지도학습 회귀 모델**](#)[**7.3 머신러닝 지도학습 분류 모델**](#)[**8 데이터 모델링**](#)[**8.1 개념적 설계 \(ERD\)**](#)[**8.2 Scraping 이용한 물리적 설계**](#)

사용한 프로그램

App Tools

[언어](#)Swift
5.7.3Dart
2.18.7[IDE](#)Xcode
14.2[Framework](#)VS code
1.76.2Flutter
3.3.10

Database Tools

[RDBMS](#)MySQL
8.0.31SQLite
3.40.1[NoSQL](#)

Firebase

데이터 분석 & AI

[언어](#)Python
3.10.4R
4.2.2

Backend Tools

[framework](#)Spring Boot
3.0.2 / 2.7.5Flask
9.0.68[언어](#)Java
17 / 11Python
3.10.4

Cooperation Tools

[Web Service](#)

Github

[DVCS](#)

git

[Design](#)Notion
Miro
Figma

Flutter 프로젝트

목 차 |

[1 세상의 모든 신발 - 신발 판매 어플](#)

[1.1 프로젝트 개요](#)

[1.2 System Flow Diagram](#)

[1.3 Process Diagram](#)

[2 가계부자 - 간편하게 사용가능한 가계부 어플](#)

[2.1 프로젝트 개요](#)

[2.2 Process Diagram](#)

[3 솜솜 하우스 - 부동산 정보 및 예측 어플](#)

[3.1 프로젝트 개요](#)

[3.2 Process Diagram](#)

[4 성연\(星演\) - 소상공인을 위한 어플](#)

[4.1 프로젝트 개요](#)

[4.2 Process Diagram](#)

세모신 - user 앱

프로젝트 개요



주제

신발 판매 어플

담당 역할

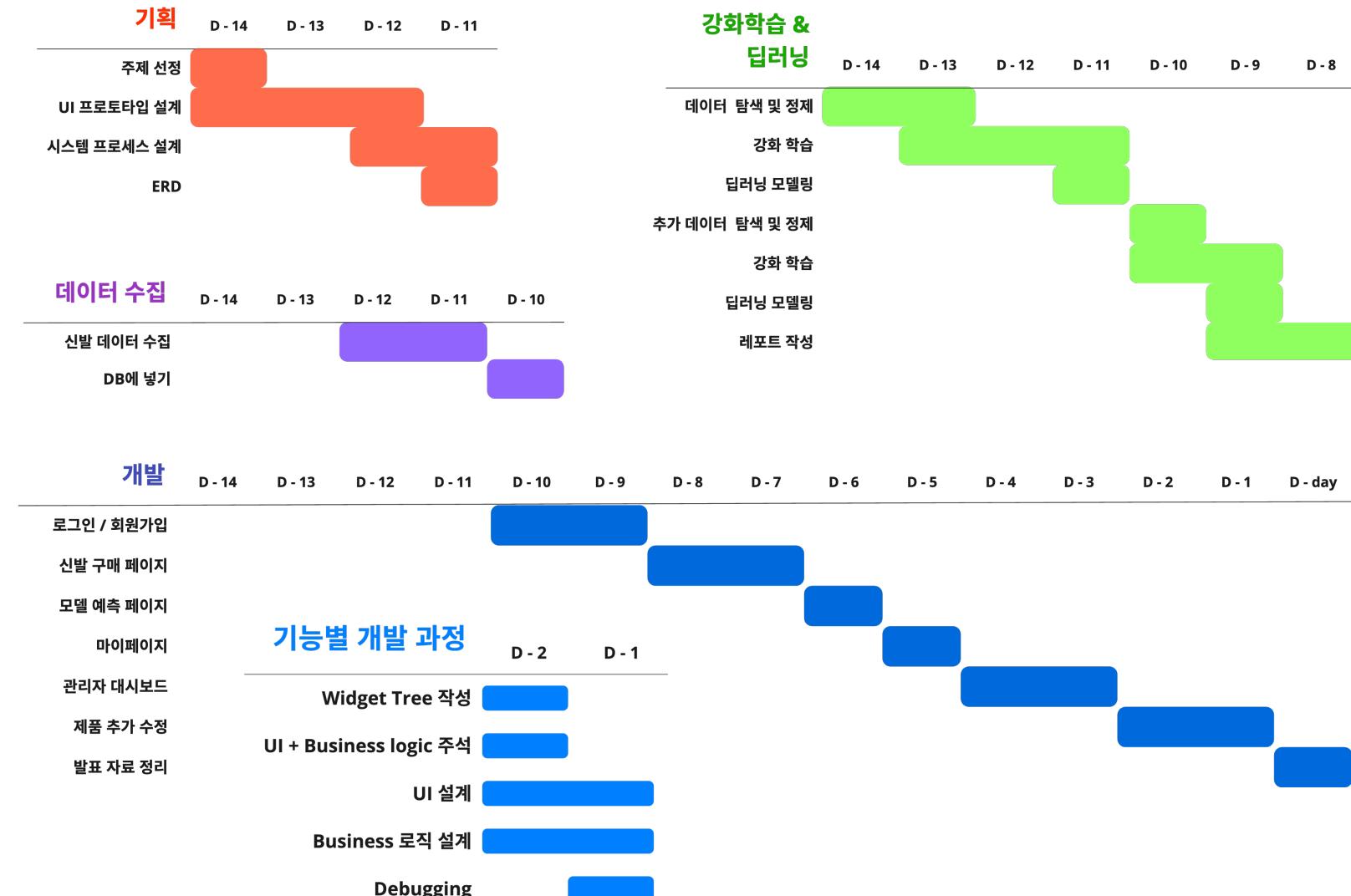
강화학습 20%

딥러닝 모델 50%

개발 33%

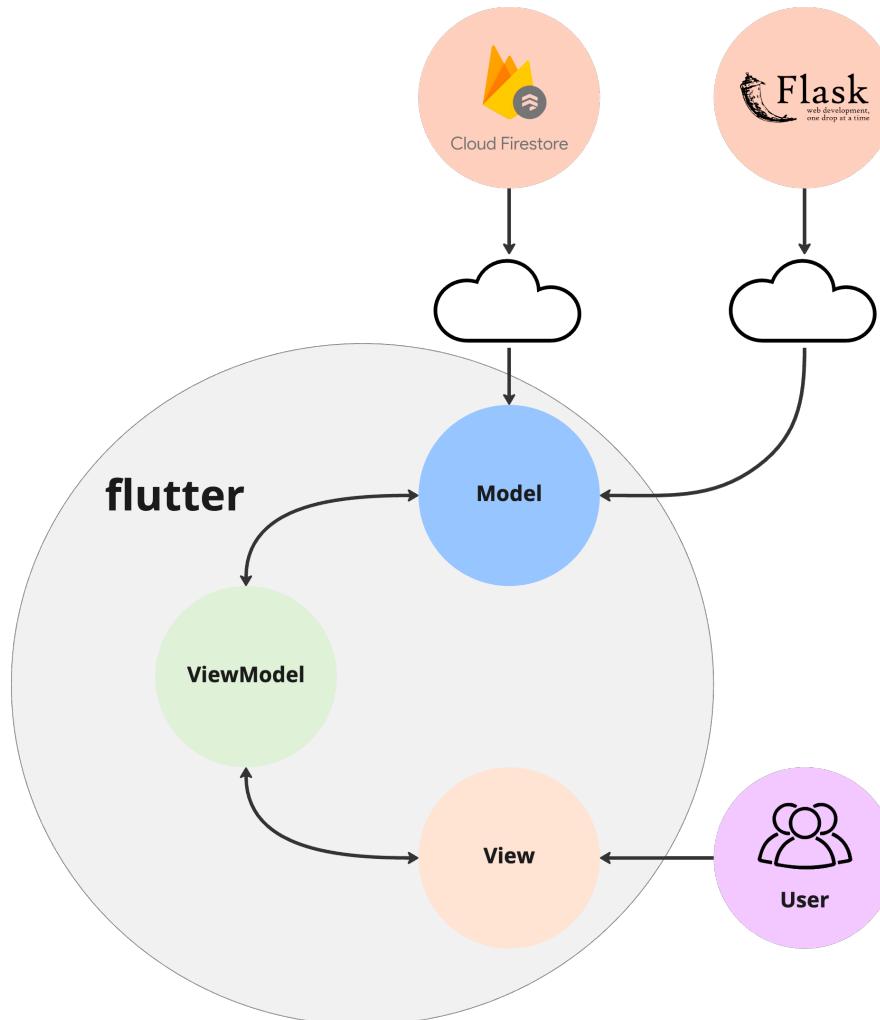
DB 구축 100%

개발 일정



세모신 - user 앱

System Flow Diagram



NoSQL ERD

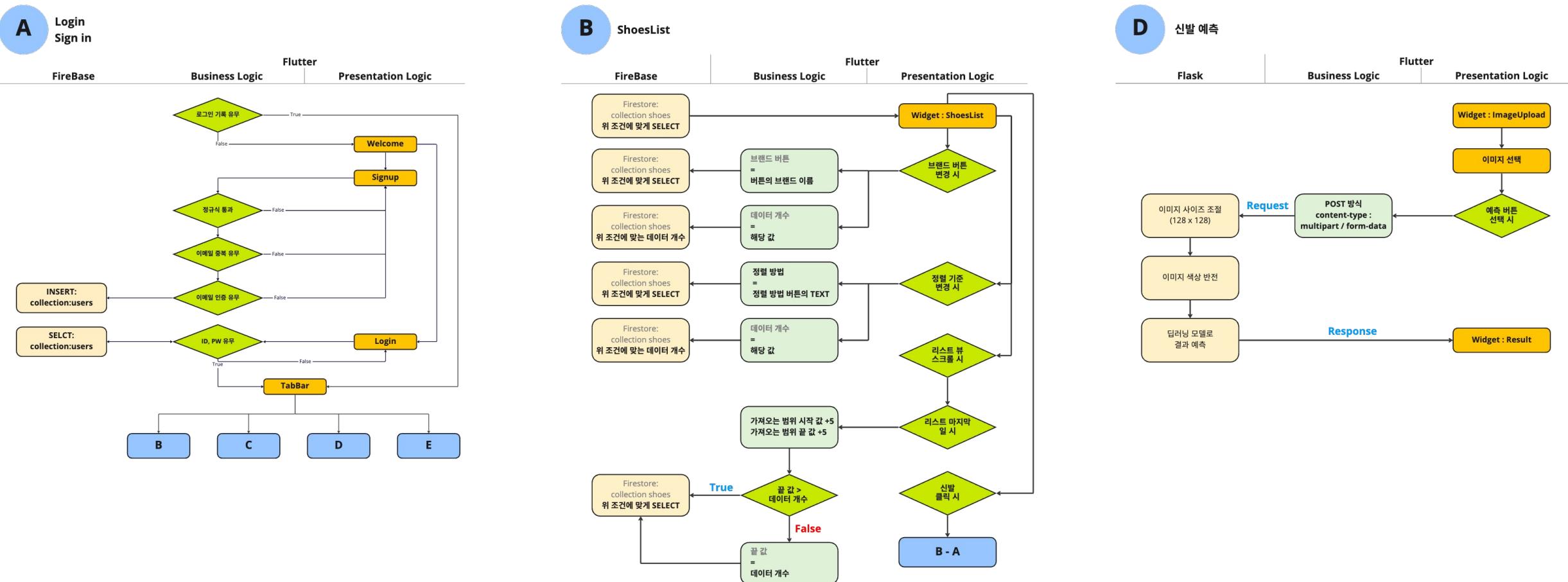
users	
name	orders
uid	+ {}
email	orderDate
nickname	+ [{}]
sex	orderNo
phone	modelName
address	orderedSize
addressDetail	amount
postcode	orderStatus
initDate	cancelStatus
updateDate	cancelNo
deleteDate	refundStatus
carts	refundNo
+ [{}]	exchangeStatus
checkStatus	exchangeNo
cartNo	changeStatusDate
cart modelName	changeStatusDoneDate
selectedSize	
amount	
favorites	
+ []	
favoriteModel	

shoes	
images	colors
+ []	+ []
image_url	color
model	sizes
brand	+ [{}]
price	size
material	amount
height	maker
method	country
initdate	

세모신 - user 앱

Process Diagram

기능들 중 담당한 부분의 Process Diagram



가계부자 - app store 배포 완료

프로젝트 개요



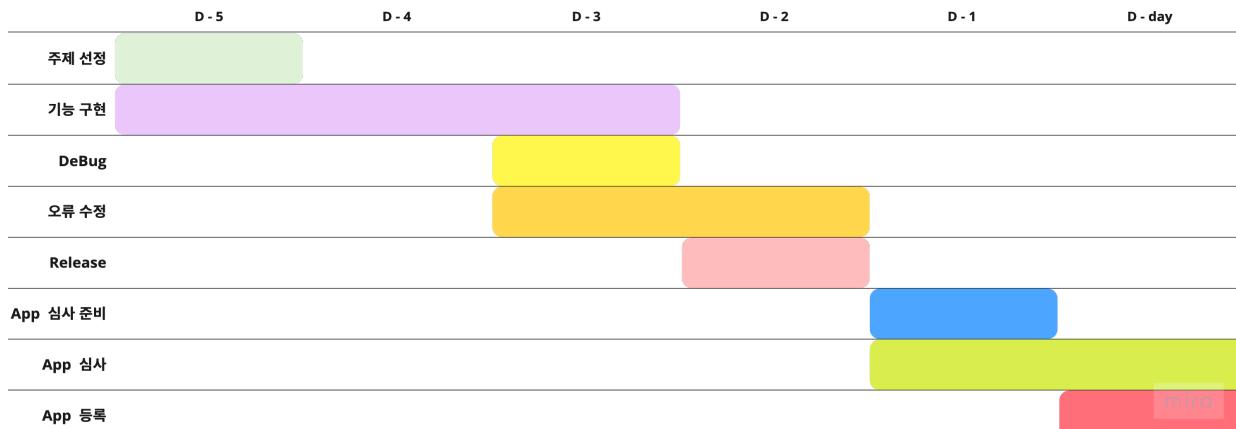
주제

회원가입없이 간편하게
사용할 수 있는 가계부

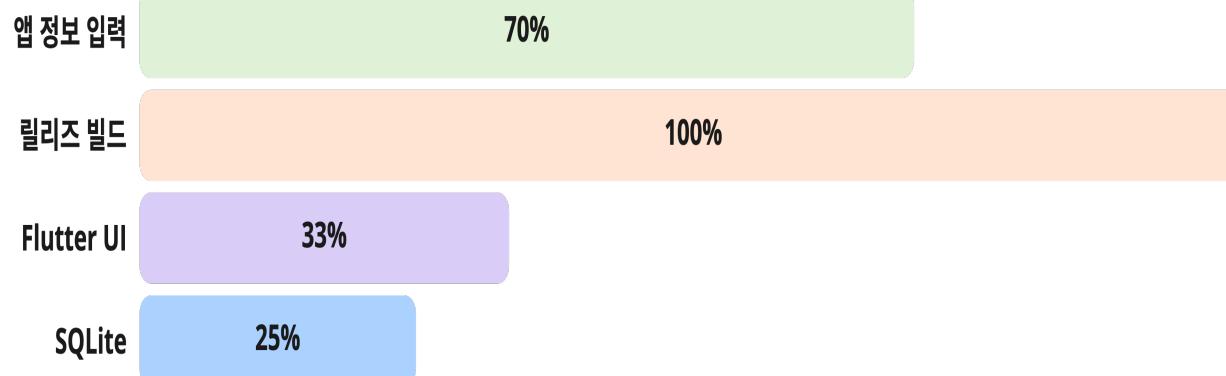
주제 선정 이유

성연 프로젝트에서 사용한 가계부가 SQLite만을 사용하여서 이를 개인이 사용할 수 있도록 카테고리 항목 추가와 일별, 카테고리별, 제목 별 검색 기능을 추가하여 app store에 배포하는 프로젝트를 추가적으로 진행하였습니다.

개발 일정



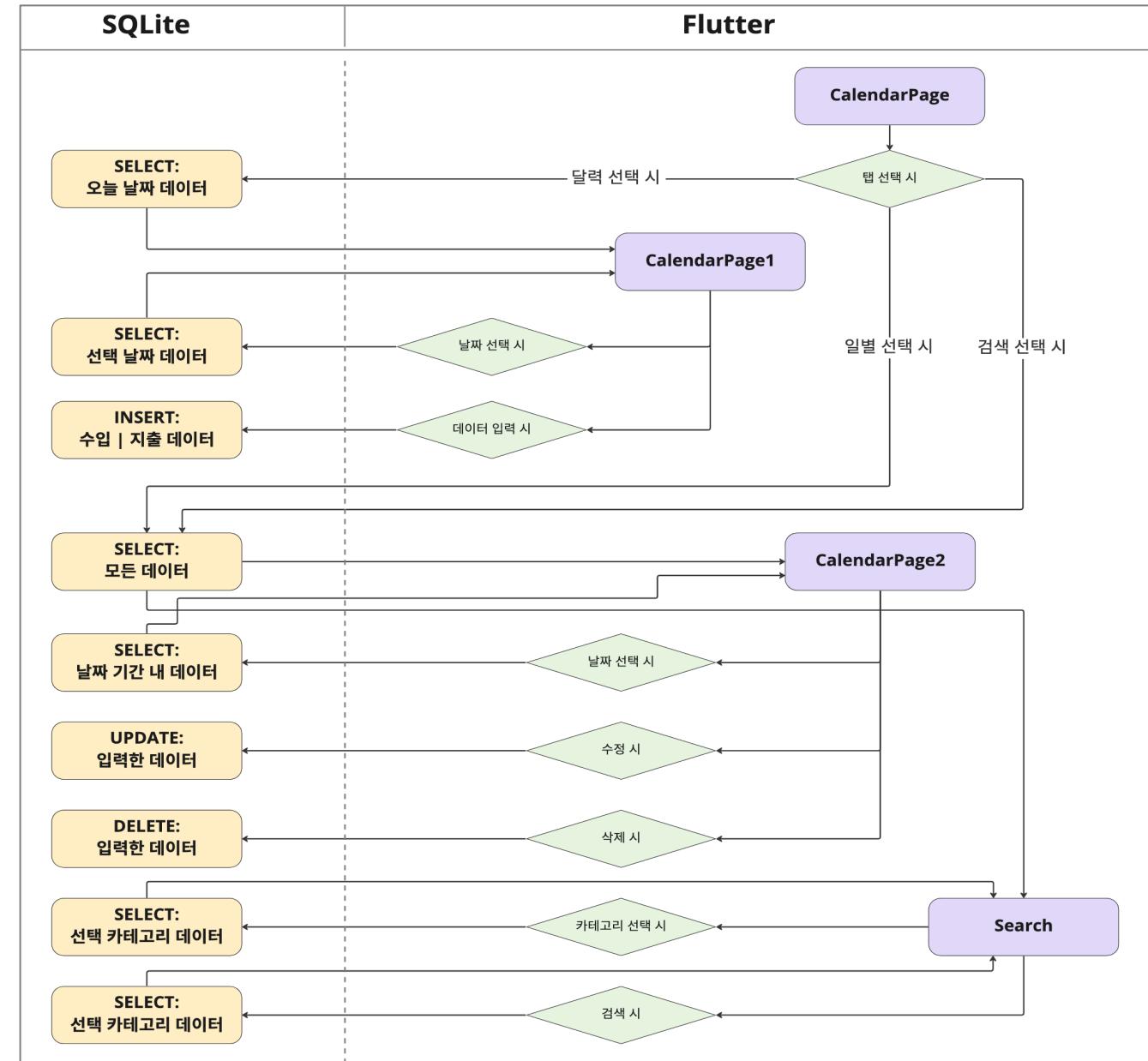
담당 역할



가계부자 - app store 배포 완료

Process Diagram

- Flutter를 이용하여 앱 UI 디자인
- SQLite를 이용하여 Device 내부 DB 구현



솜솜 하우스

프로젝트 개요



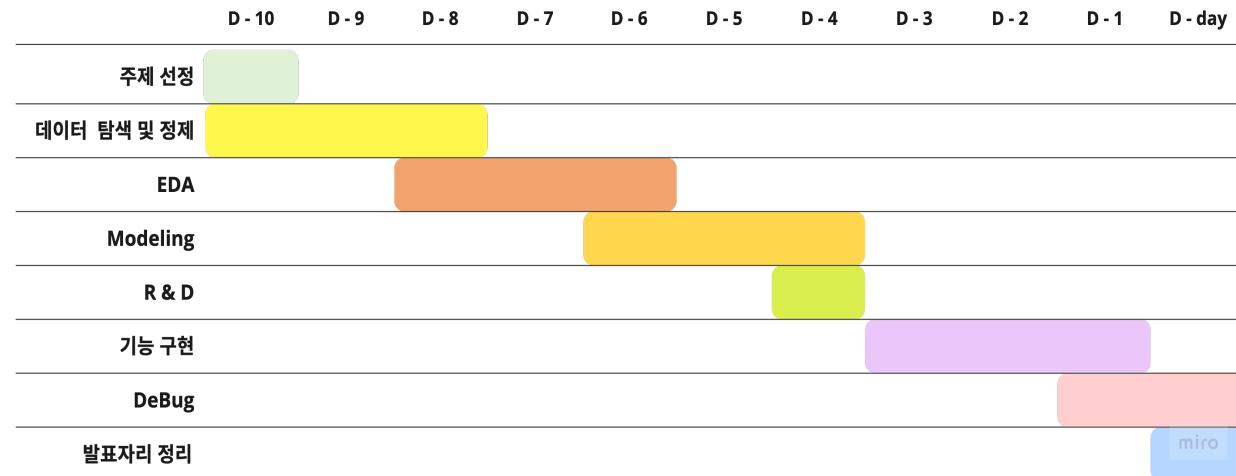
주제

서울 전월세 정보 조회 및
전세 가격 예측 서비스 제공

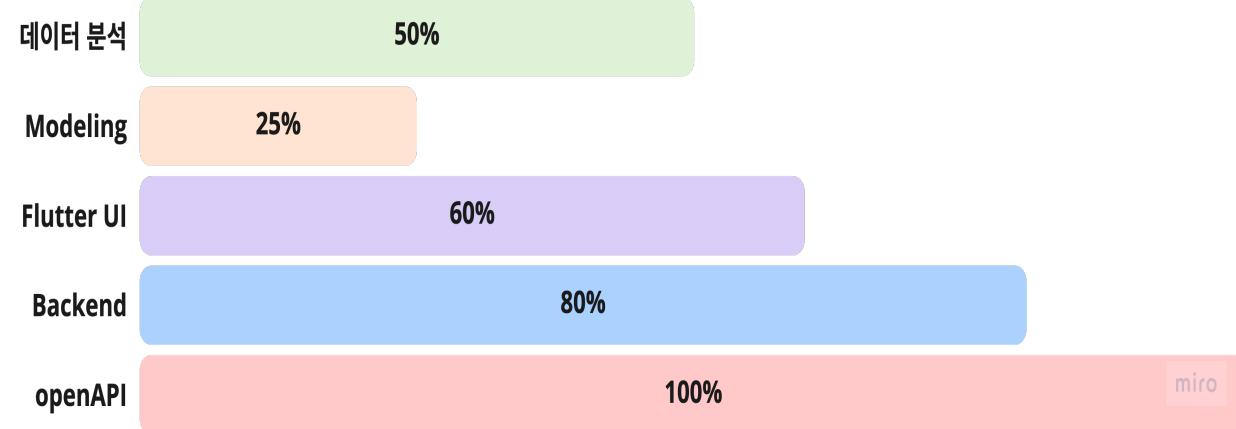
주제 선정 이유

치솟는 부동산 가격으로 인한 주거안정에 대한 불안감과
부동산 문제가 인생 전반의 계획에 큰 부분을 차지하기에
부동산 전세 가격 예측과 전월세 정보를 제공함으로써
주거안정에 대한 불안감 해소와
인생 계획에 도움이 되고자 주제를 선정했습니다.

개발 일정



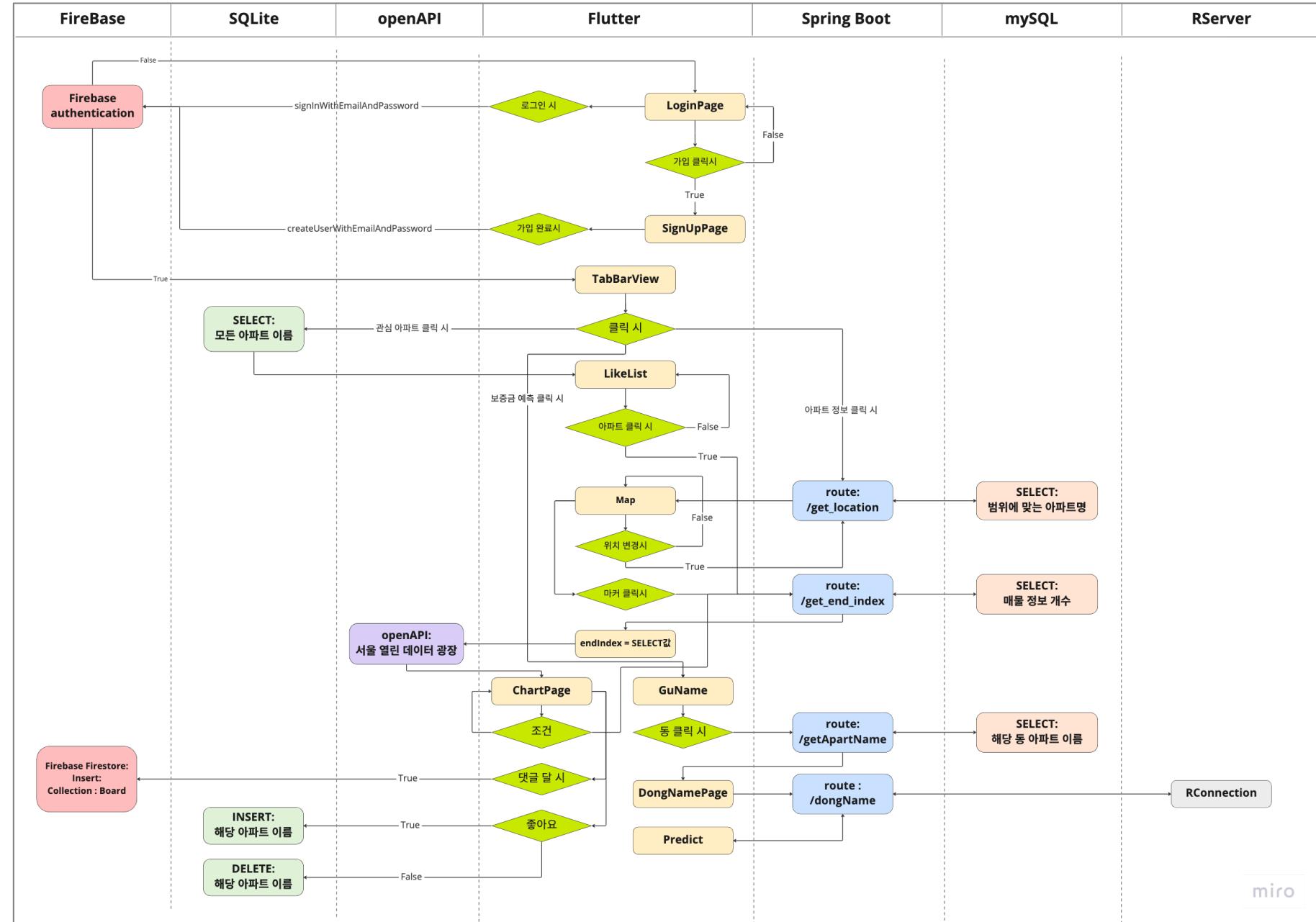
담당 역할



솜솜 하우스

Process Diagram

- Flutter 이용한 앱 UI 디자인
- SQLite를 이용하여
Device 내부 DB 구현
- Spring Boot를 이용하여
mySQL DB 연동과 RServer
연결하는 Backend 구성
- Firebase 이용하여
Login ,RealTime DB 구축
- openAPI 이용하여
필요한 데이터 가져오기



성연 (星演)

프로젝트 개요



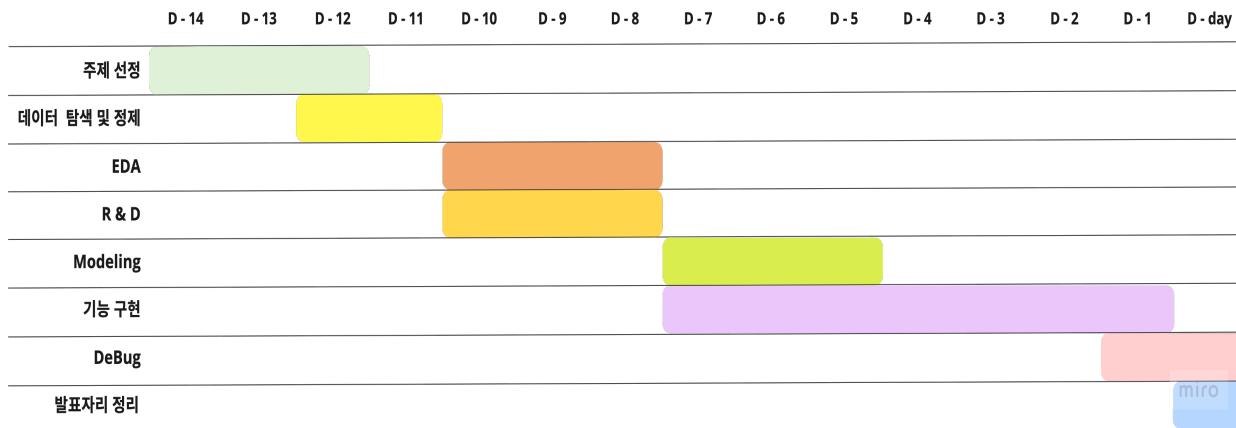
주제

예비 자영업자를 위한
상권 분석과 매출 예측 서비스 제공

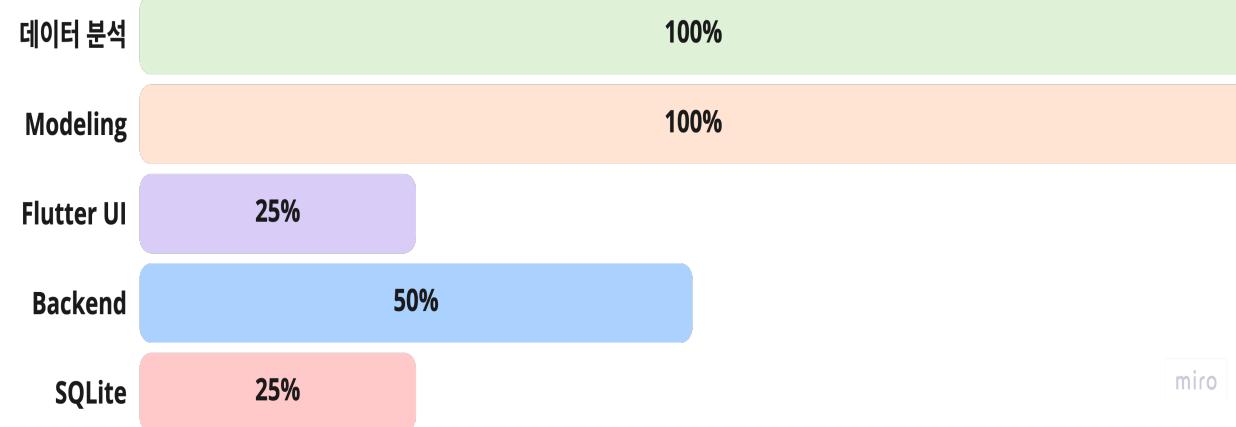
주제 선정 이유

전체 취업자 중 자영업자의 비율이 25% 정도입니다.
위와 같은 이유로 많은 사용자를 타겟으로 할 수 있고,
자영업의 경우, 상권에 따른 매출의 차이가 크다고 알고 있습니다.
이를 분석하여 정보를 제공함으로써 예비 자영업자한테
의미 있는 지표로 사용할 수 있을 것 같습니다.

개발 일정



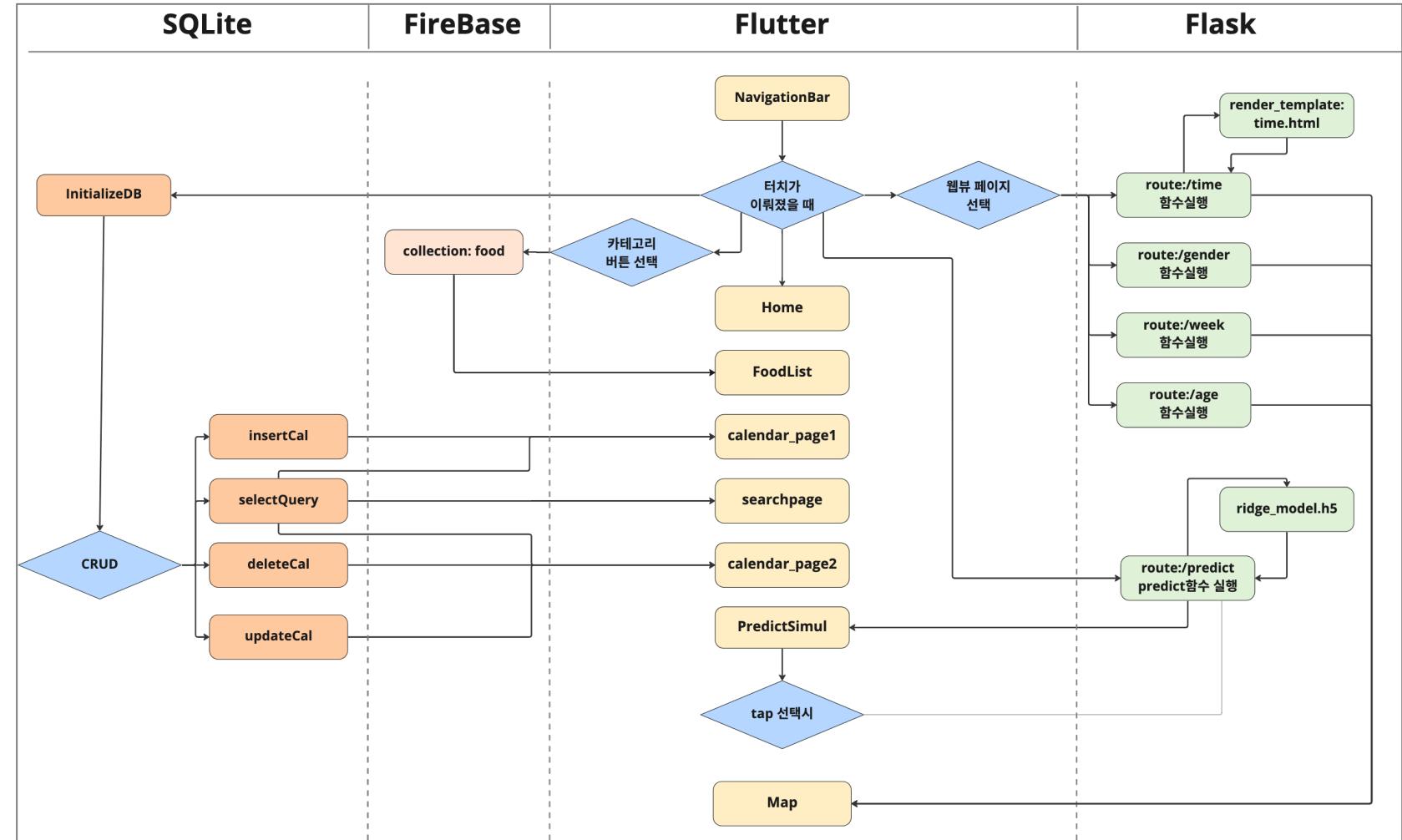
담당 역할



성연 (星演)

Process Diagram

- Flutter를 이용하여 앱 UI 디자인
- SQLite를 이용하여
Device 내부 DB 구현
- Flask를 이용해 AI 모델과 연동하여
매출액 예측 금액을 Flutter UI에 표시
- Firebase를 이용하여 crawling한
데이터를 Flutter 화면에 표현



Swift 프로젝트

목 차 | 1 [영어의 신 - 영어 단어 테스트 및 단어장 어플](#)

1.1 [프로젝트 개요](#)

1.2 [System Flow Diagram & Process Diagram](#)

1.3 [NoSQL ERD & Storyboard](#)

영어의 신 - 개인 프로젝트

프로젝트 개요



주제

영어 사전 기능과
영어 단어 퀴즈와 나만의 단어장을 통한
영어 실력 향상을 위한 앱

주제 선정 이유

지하철이나 버스에서 노트에 단어장을 만들어서 들고 다니는
사람들을 보면서 이걸 어플로 만들어서 핸드폰에서 볼 수 있으면
노트를 들고 다녀야 하는 불편함도 줄일 수 있고
언제 어디서나 볼 수 있다는 편의성을 사용자에게 제공할 수
있을 것 같아 주제를 정했습니다.

개발 일정

- 애자일 방법론 적용 하여 개발

DB 구축

D - 7

영어 단어 데이터 수집

Firestore DB 구축

개발

D - 7

D - 6

D - 5

D - 4

D - 3

D - 2

D - 1

D - day

로그인 / 회원가입

퀴즈 페이지

채점 페이지

나의 단어장 페이지

단어 검색 페이지

기능별 개발 과정

D - 2

D - 1

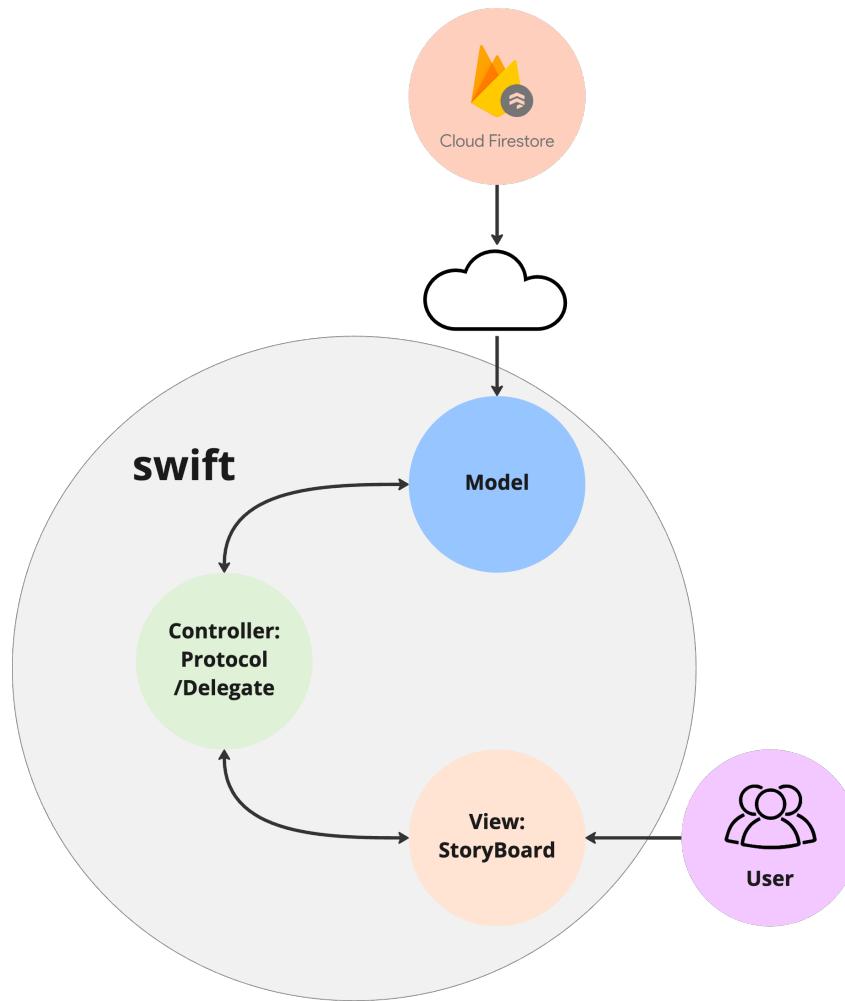
Storyboard UI 설계

Business logic 설계

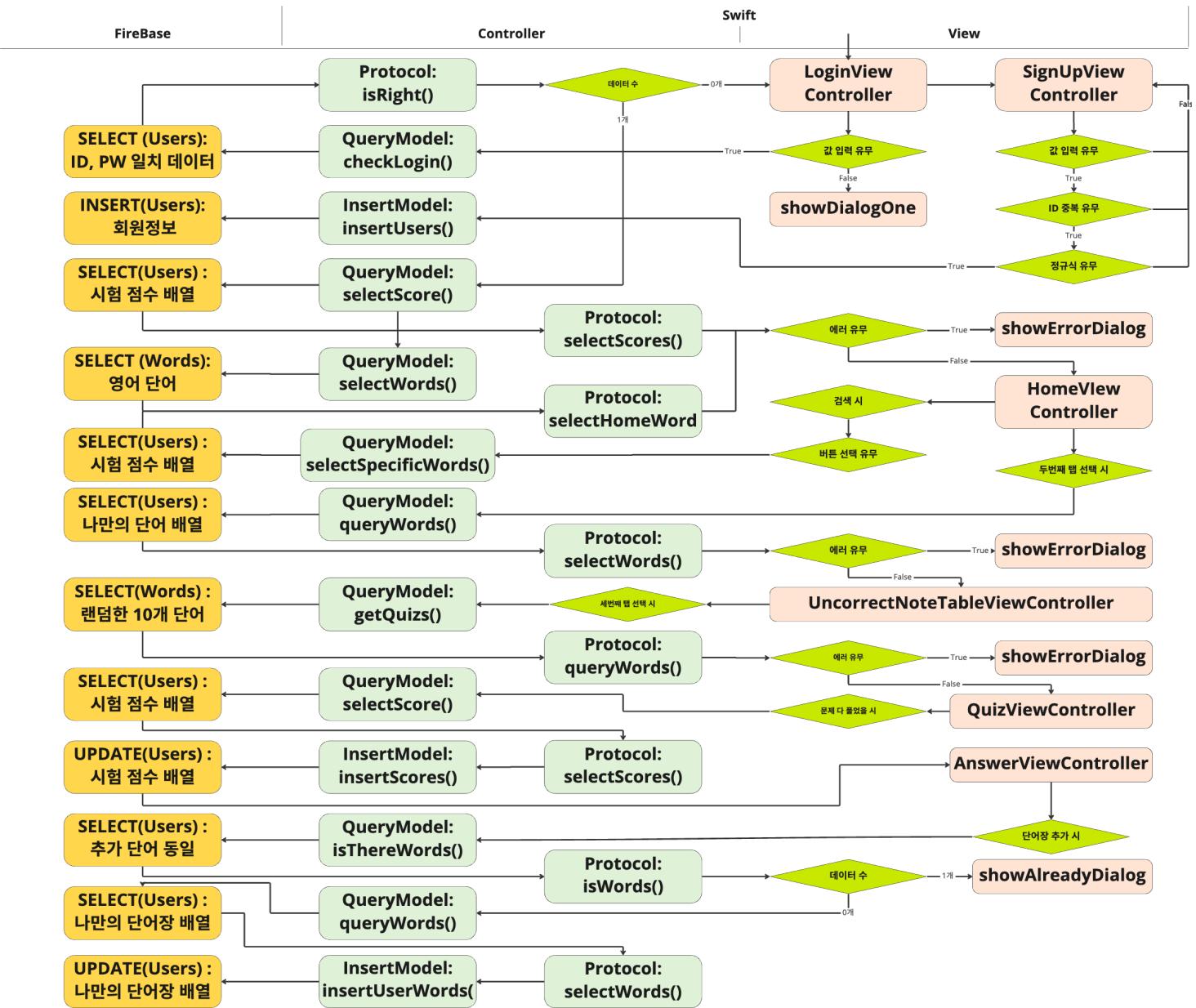
Debugging

영어의 신

System Flow Diagram



Process Diagram



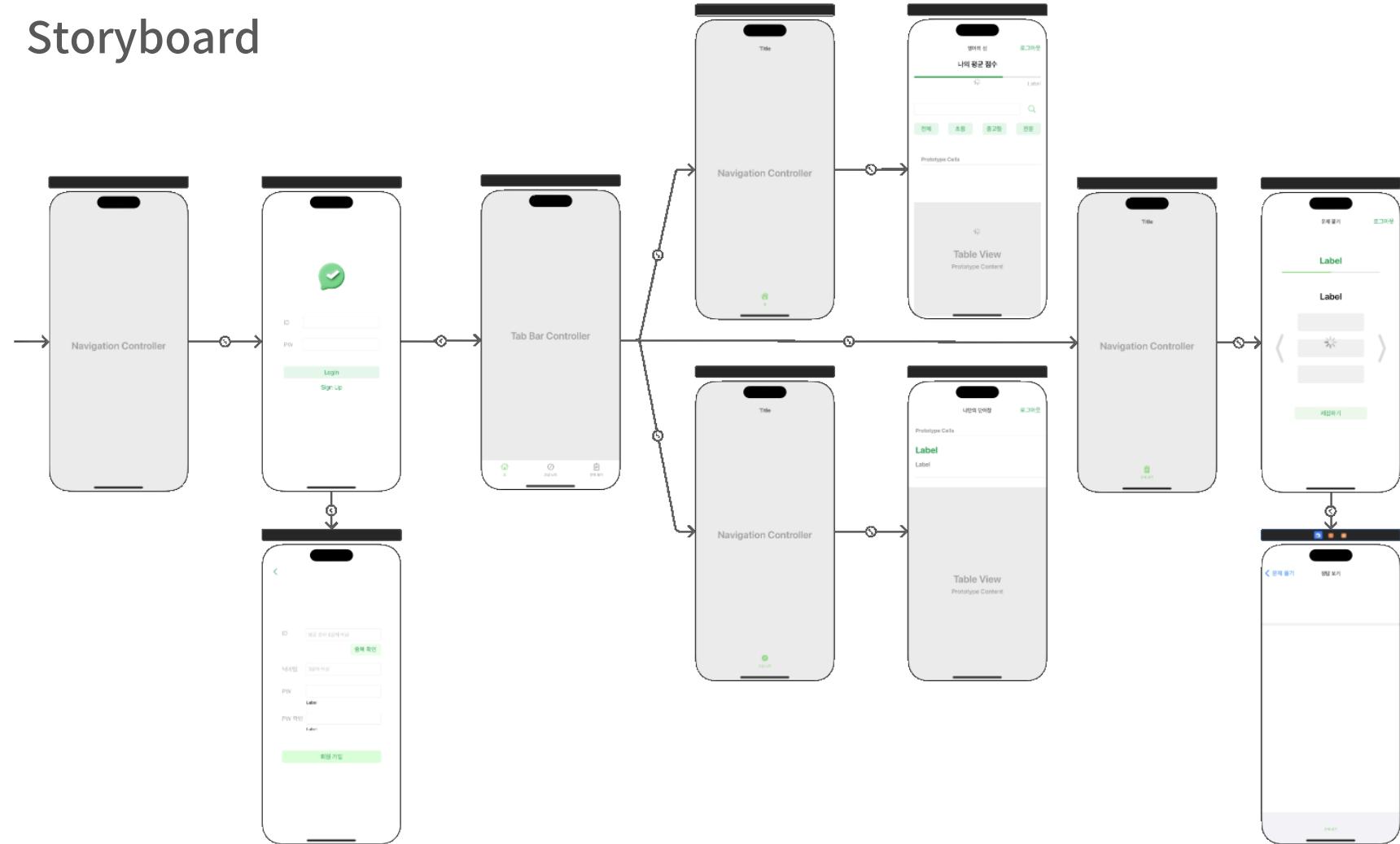
영어의 신

NoSQL ERD

```
users
id : String
nickname : String
pw : String
scores :
+ []
  score : int
words :
+ [{}]
  mean : String
  word : String
```

```
words
mean : String
step : String
word : String
wrong1 : String
wrong2 : String
```

Storyboard



App View Model

목 차 | 1 [Flutter](#)

1.1 [paging](#)

1.2 [위젯 분리](#)

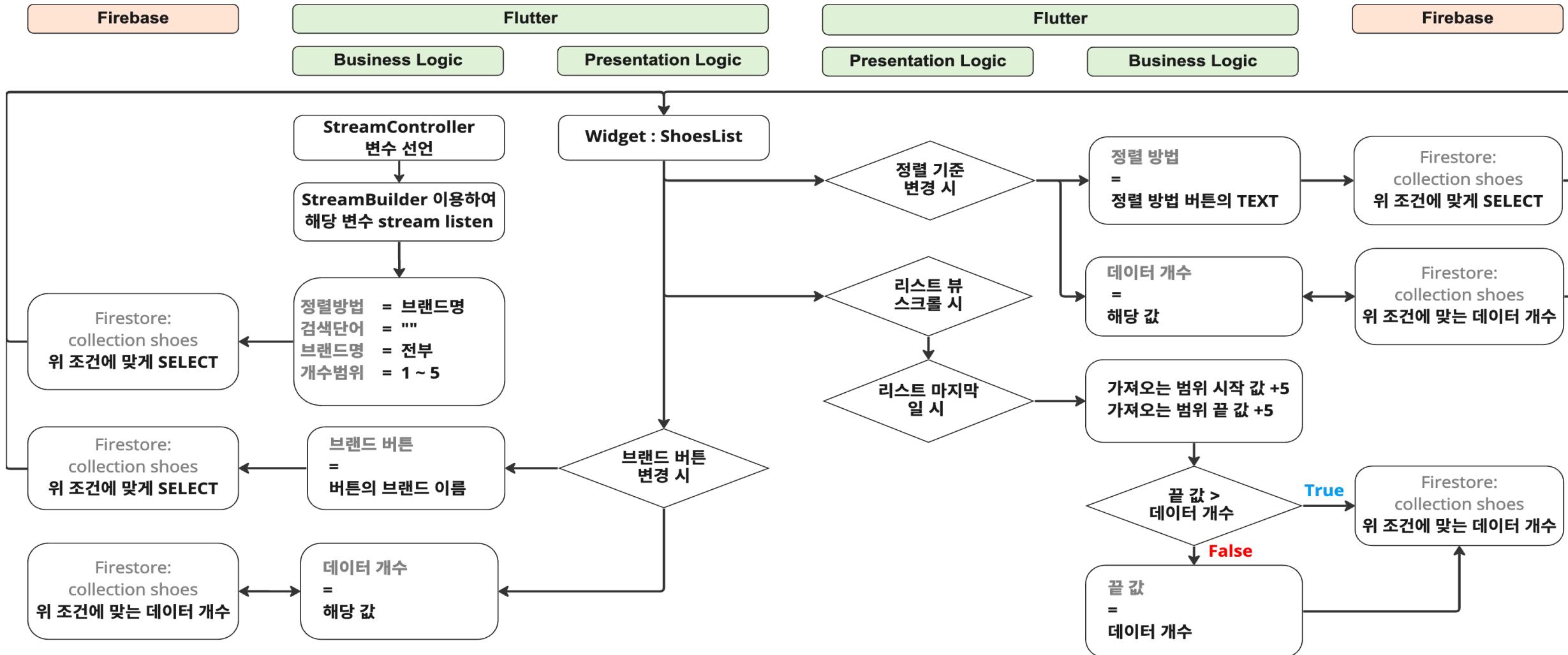
2 [Swift](#)

2.1 [protocol & delegate](#)

ListView Paging

[구현 영상
보러 가기](#)

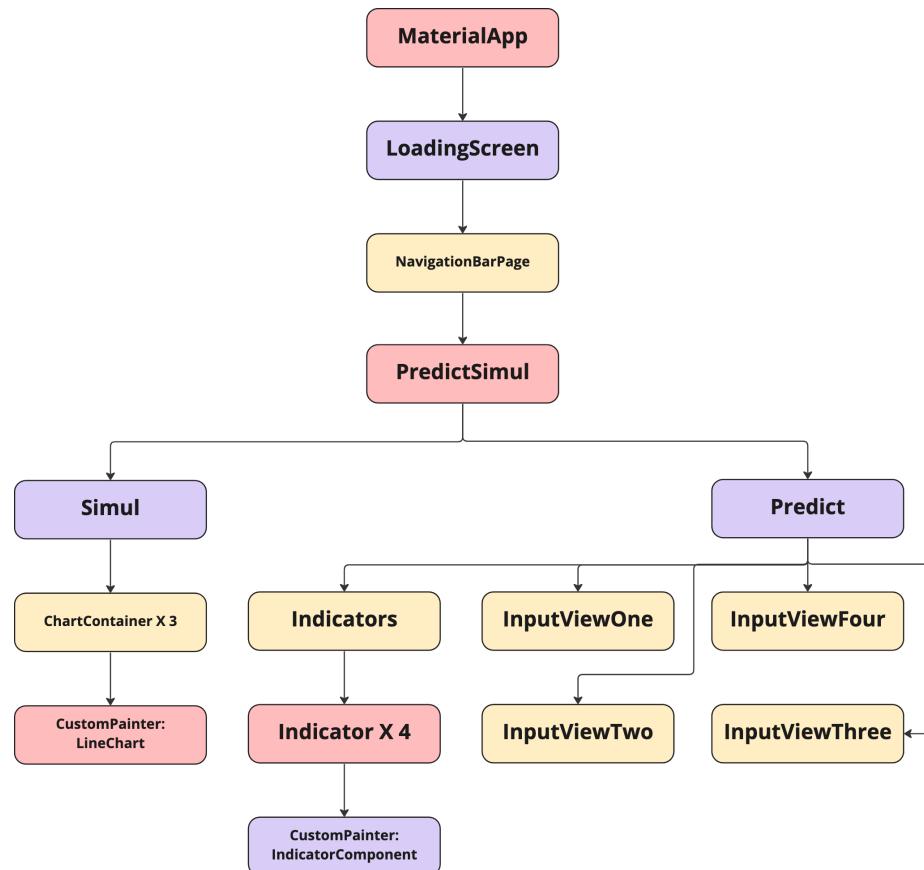
StreamBuilder와 ScrollController를 이용한 ListView paging 기능 구현



클래스 간 데이터 교환

StreamController를 이용하여 다른 클래스 간에 데이터를 전달할 수 있는 기능을 구현

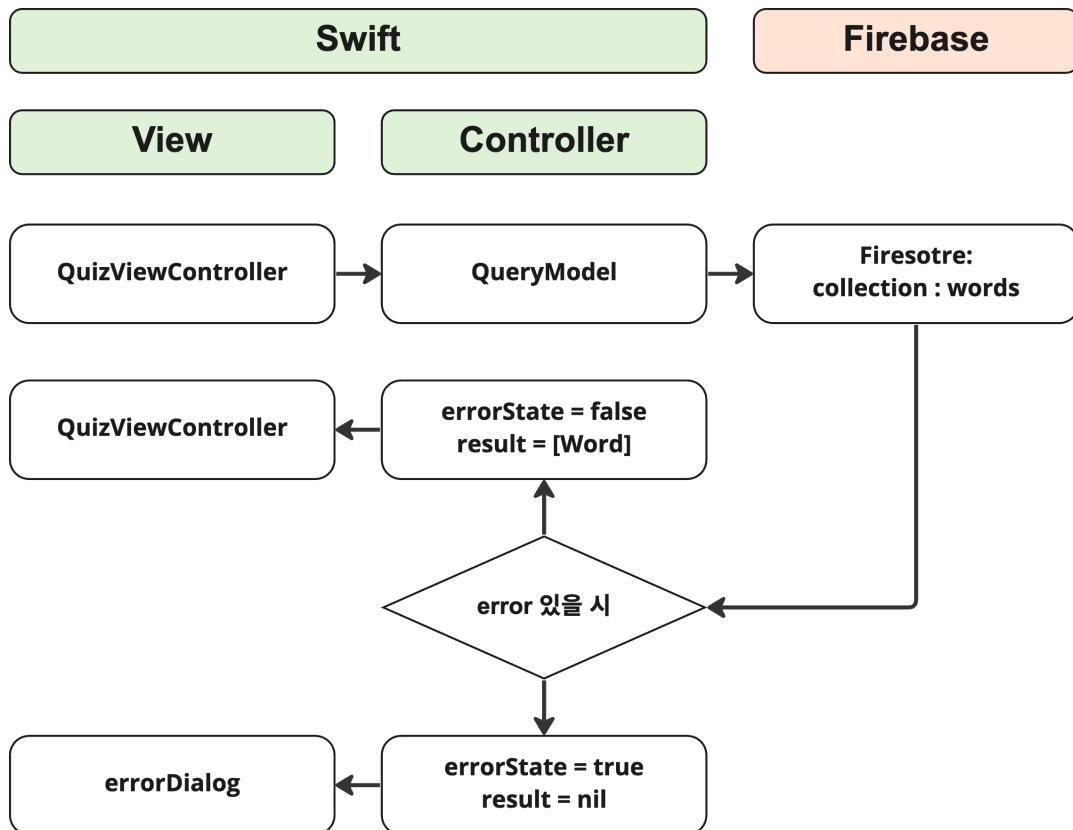
Widget Tree



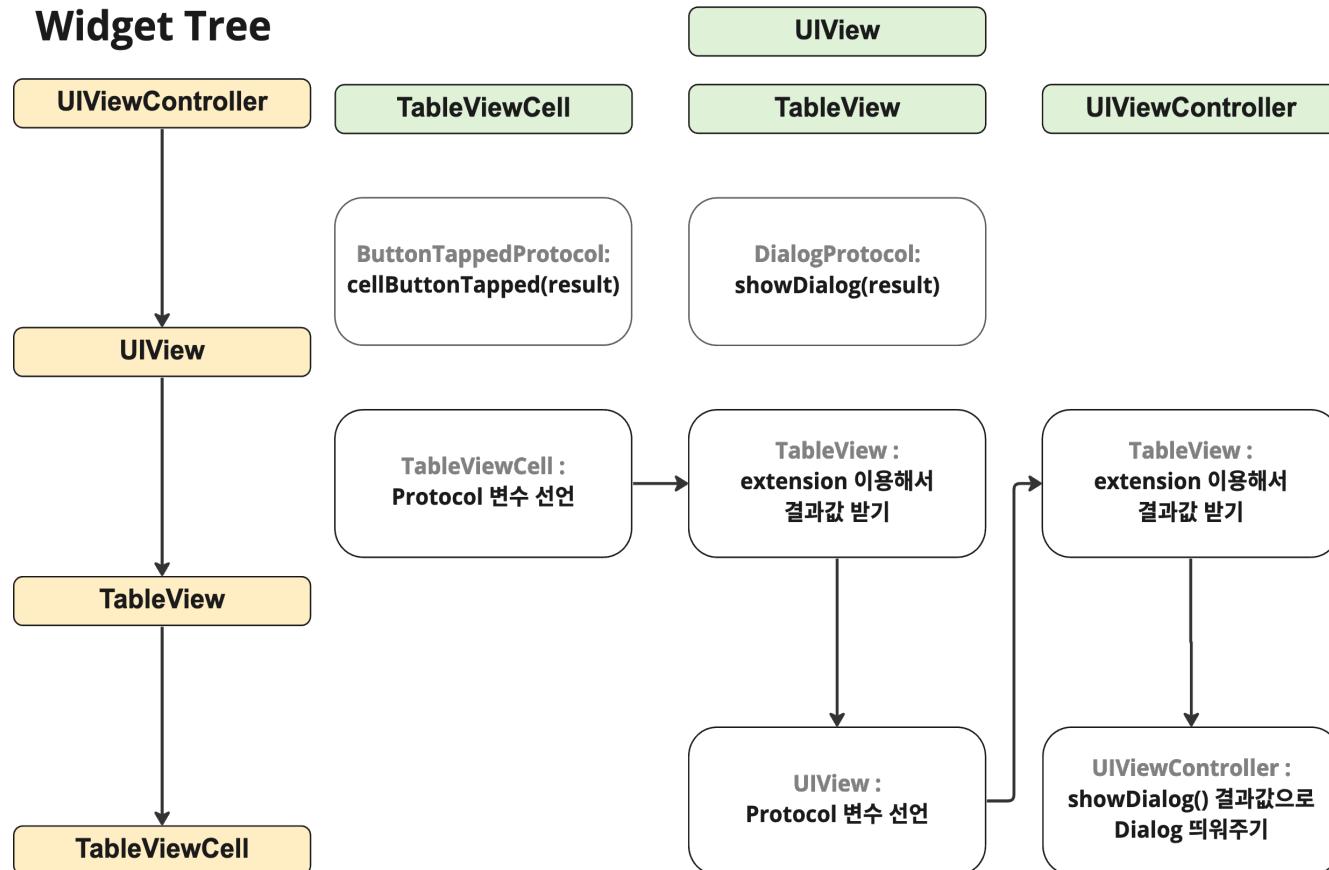
- 위젯 간의 분리를 통해 유지 보수 시 어느 부분에서 오류가 났는지 쉽게 파악할 수 있다.
- 한 화면을 여러 위젯으로 분리를 함으로써 한 화면 사이에서 데이터 교환이 필요할 때 클래스 간의 데이터가 필요한 경우가 발생
- 그 때에는 StreamController를 이용하여 controller의 listen 함수와 controller.stream의 add 함수를 통해 변화가 생기는 위젯에는 listen 함수를 변화를 주는 위젯에는 add 함수를 줘서 다른 클래스여도 데이터를 주고 받을 수 있게 하였다.

Protocol & Delegate

1. 네트워크 연결 시 비동기 처리를 위해서



2. child widget의 이벤트 처리를 위해서



DB

목 차 | 1 [RDBMS](#)

1.1 [MySQL](#)

1.2 [SQLite](#)

2 [NoSQL](#)

2.1 [Firebase](#)

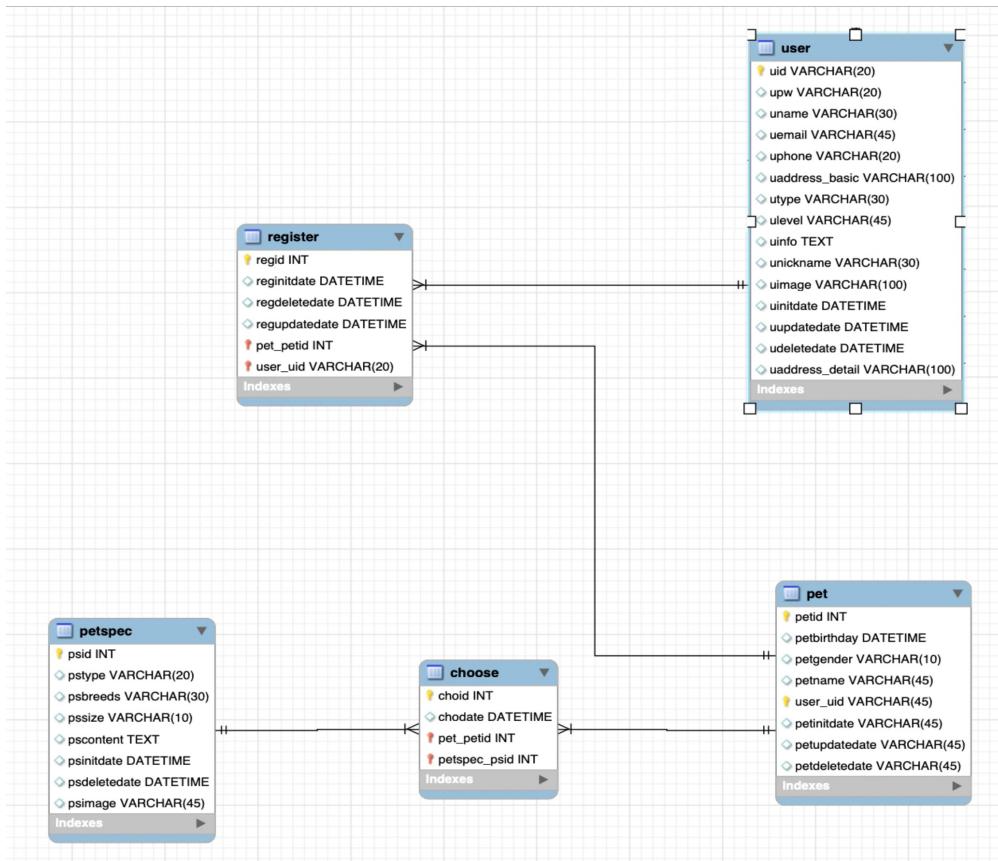
2.1.1 [Firestore](#)

2.1.2 [Storage](#)

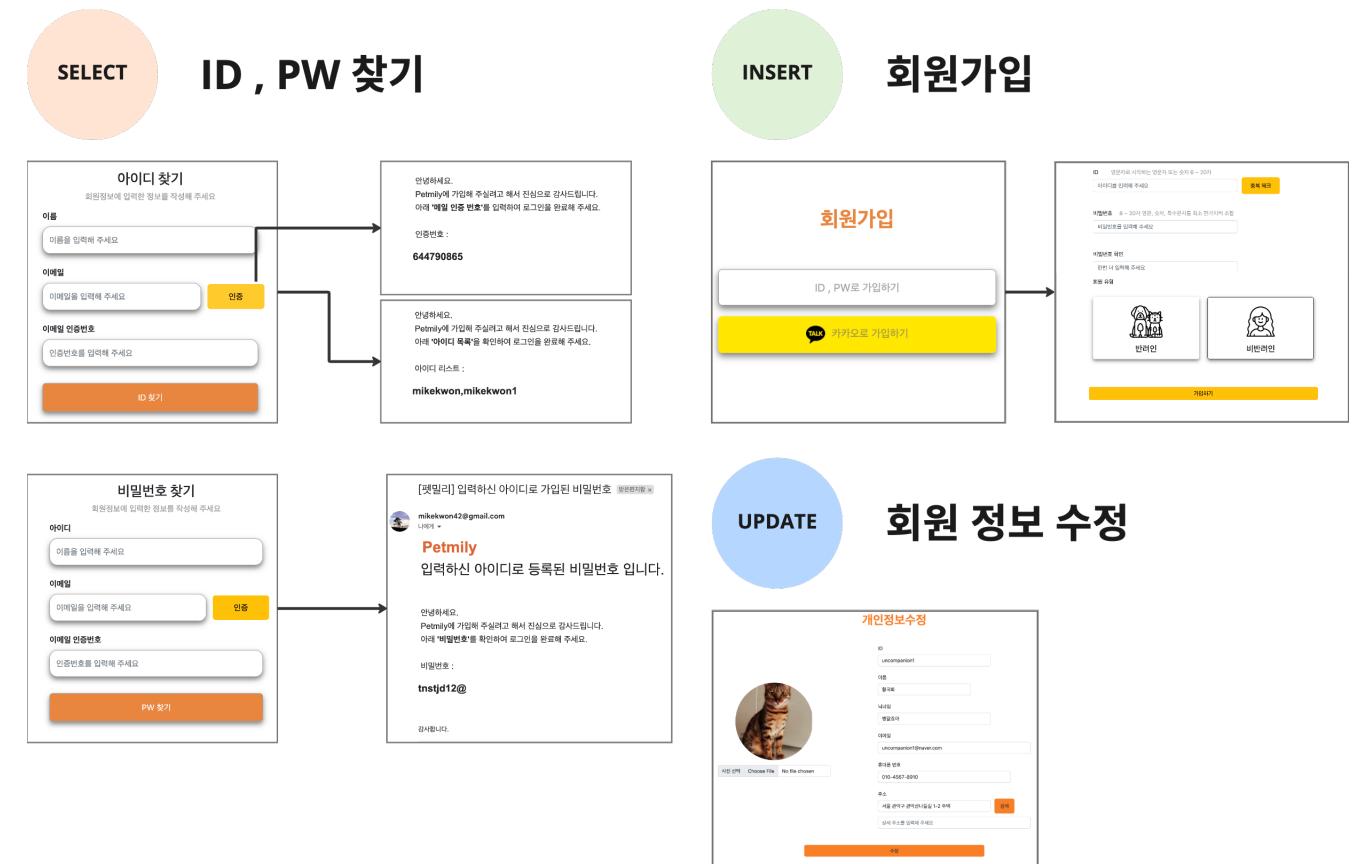
MySQL - 회원가입

회원 정보를 저장할 테이블 정의 (DDL), 회원가입 / 정보 수정 / 아이디 비밀번호 찾기 시에 필요한 DML

EER / DDL



DML

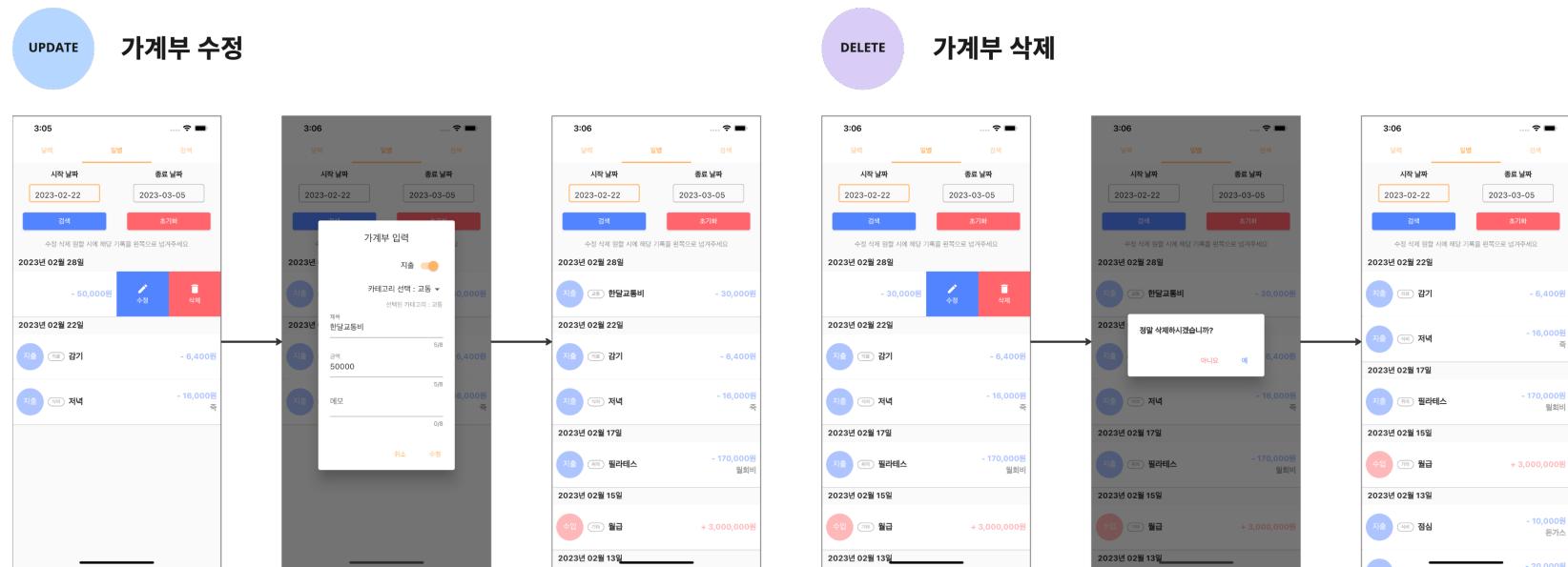
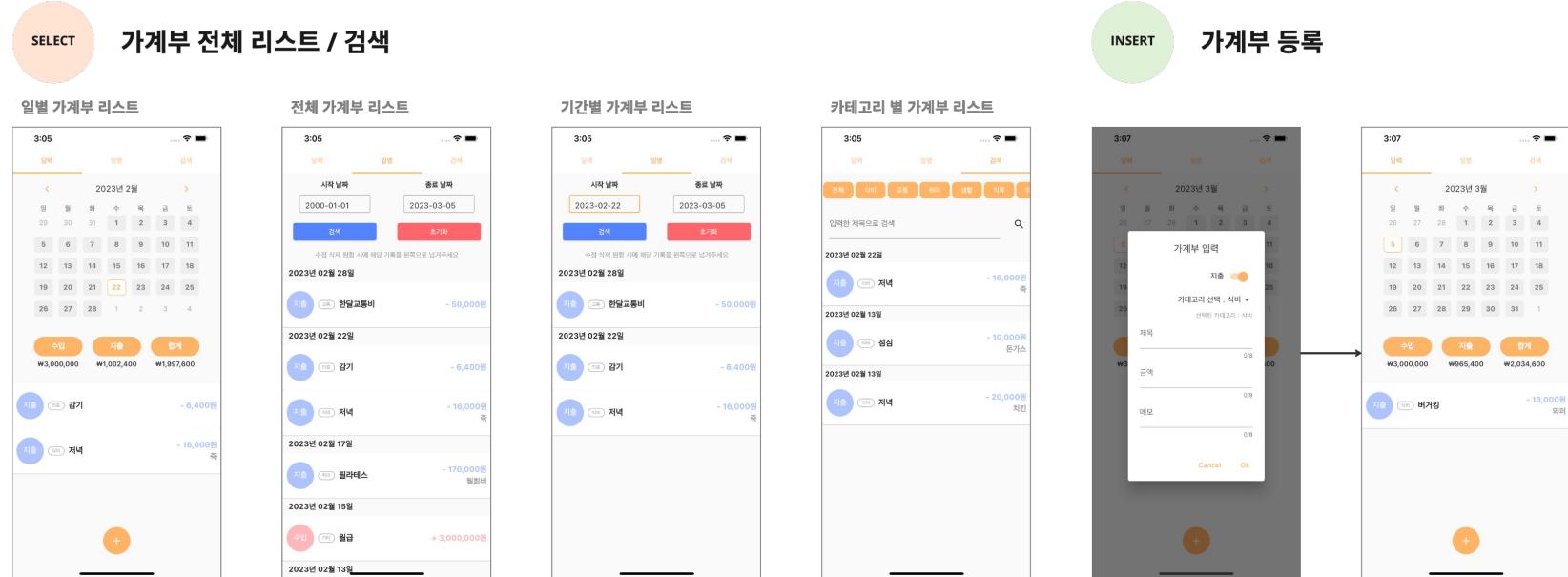


SQLite - 가계부

가계부를 저장할 테이블 정의 (DDL),
가계부 추가 / 수정 / 삭제 / 검색
기능이 가능하게 하는 DML

EER / DDL

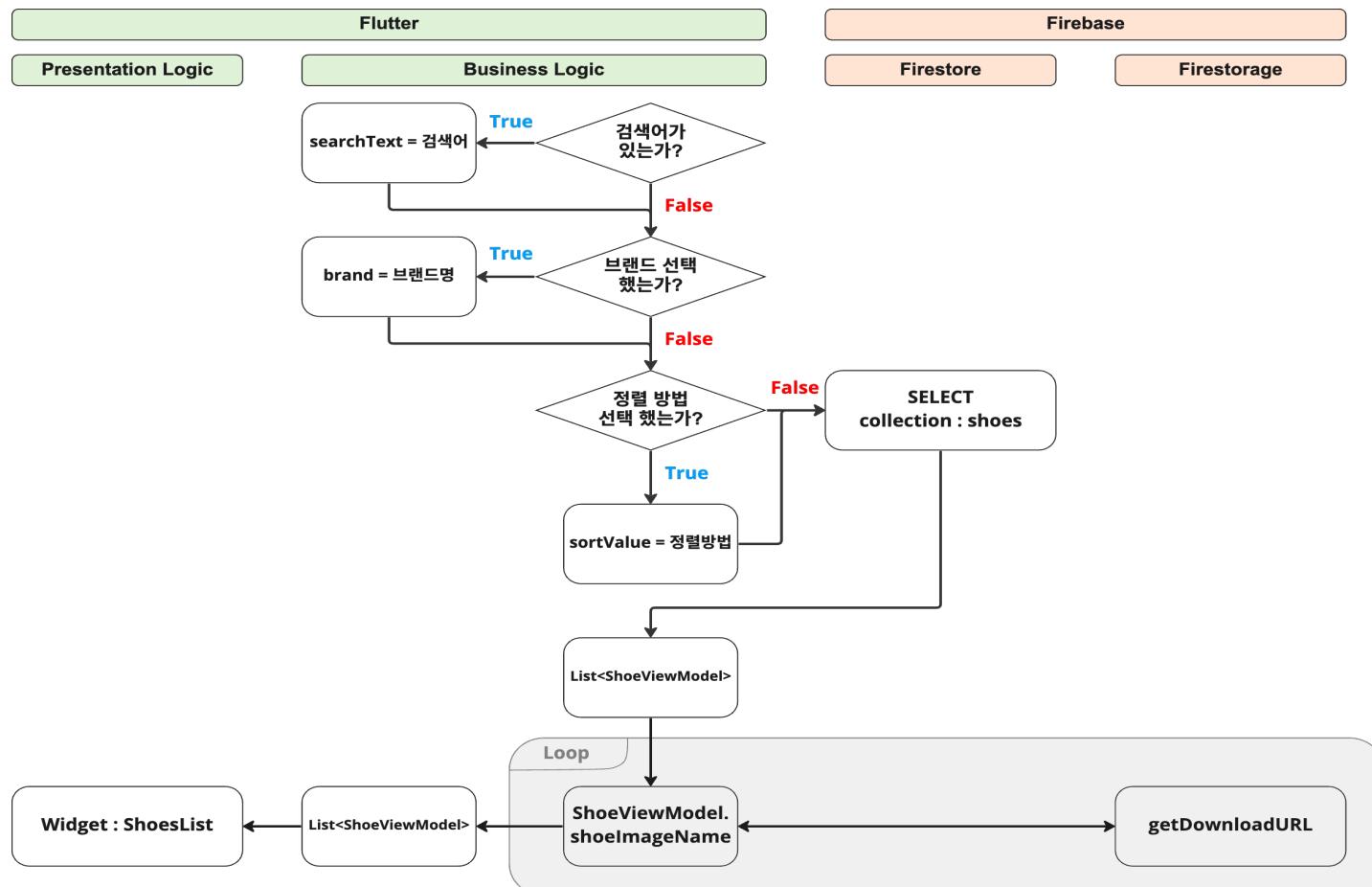
DML



Firebase Firestore & Storage

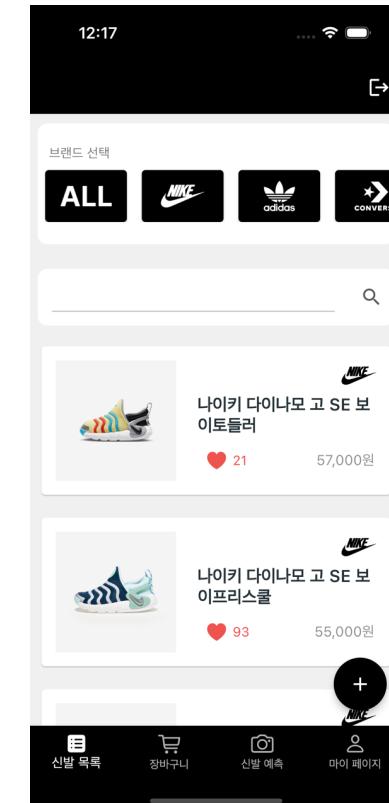
Firestore에 이미지 url을 저장하여 Storage의 이미지 파일을 불러오는 기능

Flow Diagram

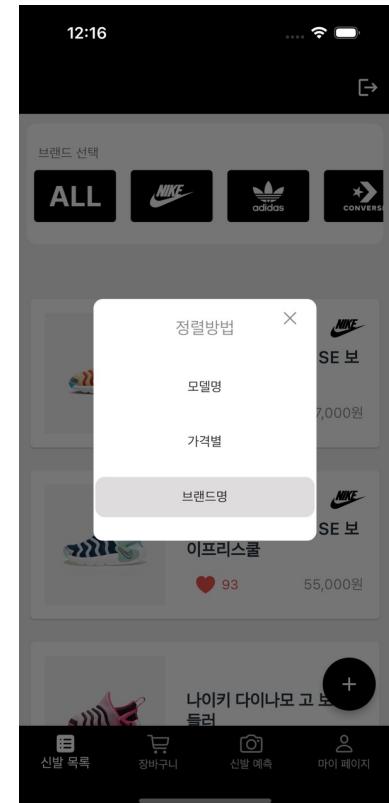


구현 화면

검색 브랜드 선택 화면



정렬 방법 선택 화면



구현 영상
보러 가기

Backend

목 차 | 1 [Spring Boot](#)

1.1 [Spring Boot + DB](#)

1.2 [Spring Boot + AI](#)

2 [Flask](#)

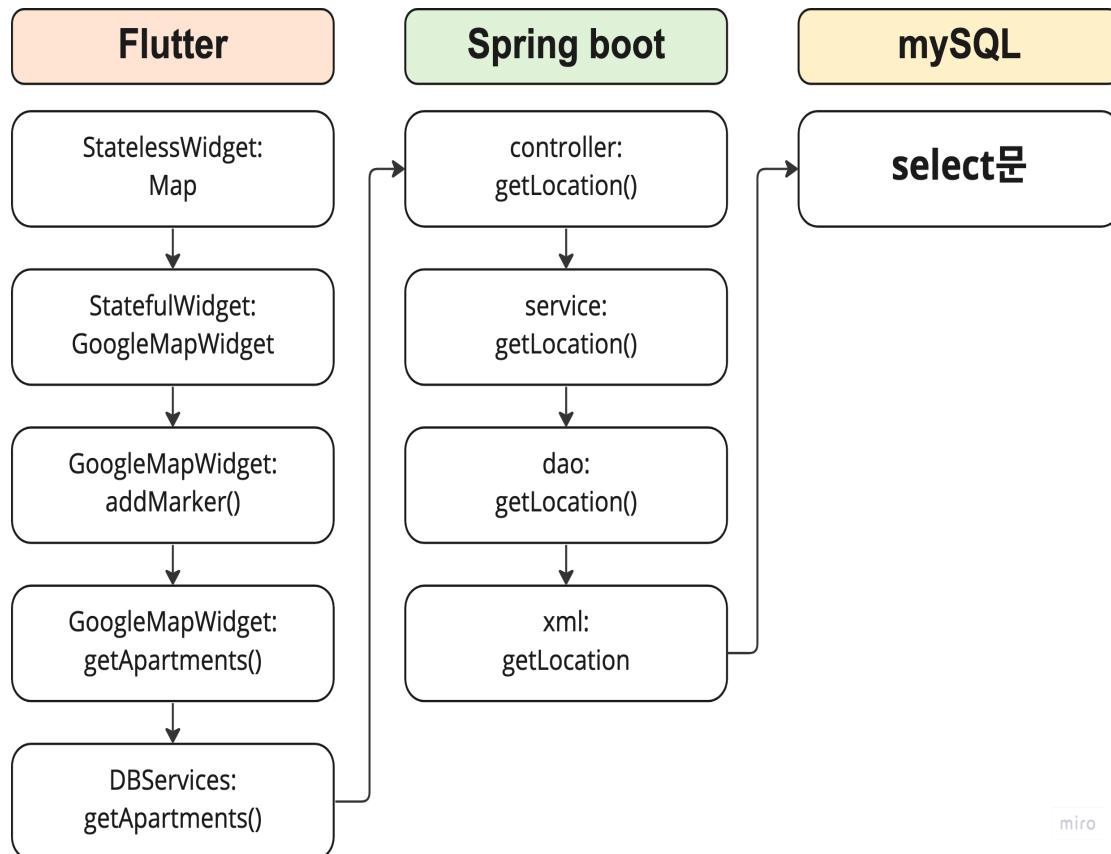
2.1 [Flask + AI \(회귀 분석 모델\)](#)

2.2 [Flask + AI \(딥러닝 모델\)](#)

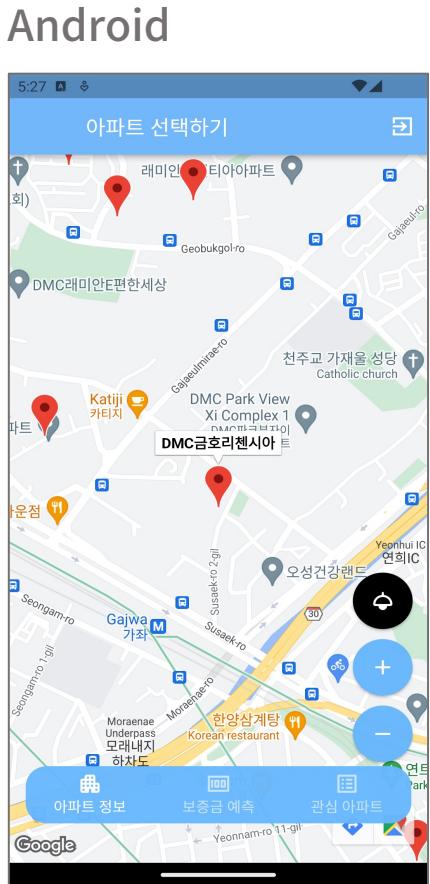
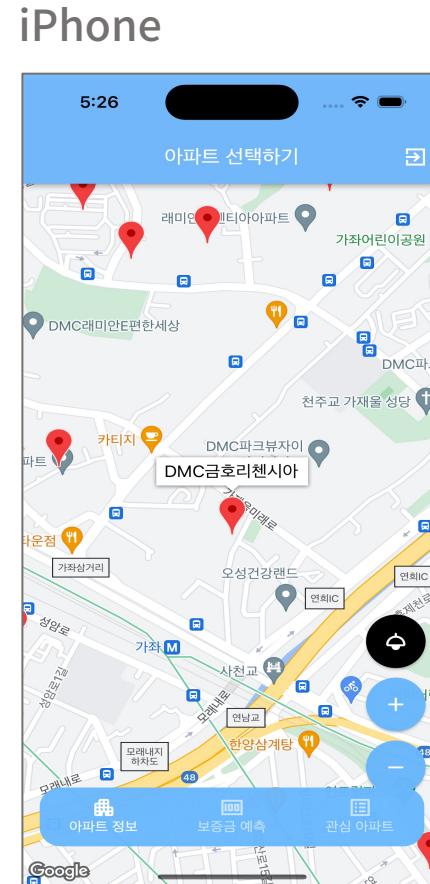
Spring Boot +DB by.MySQL

Spring Boot로 Rest API 구현 하여 DB의 아파트별 경도 위도 데이터를 App UI에 지도의 마커로 표시

Flow Diagram



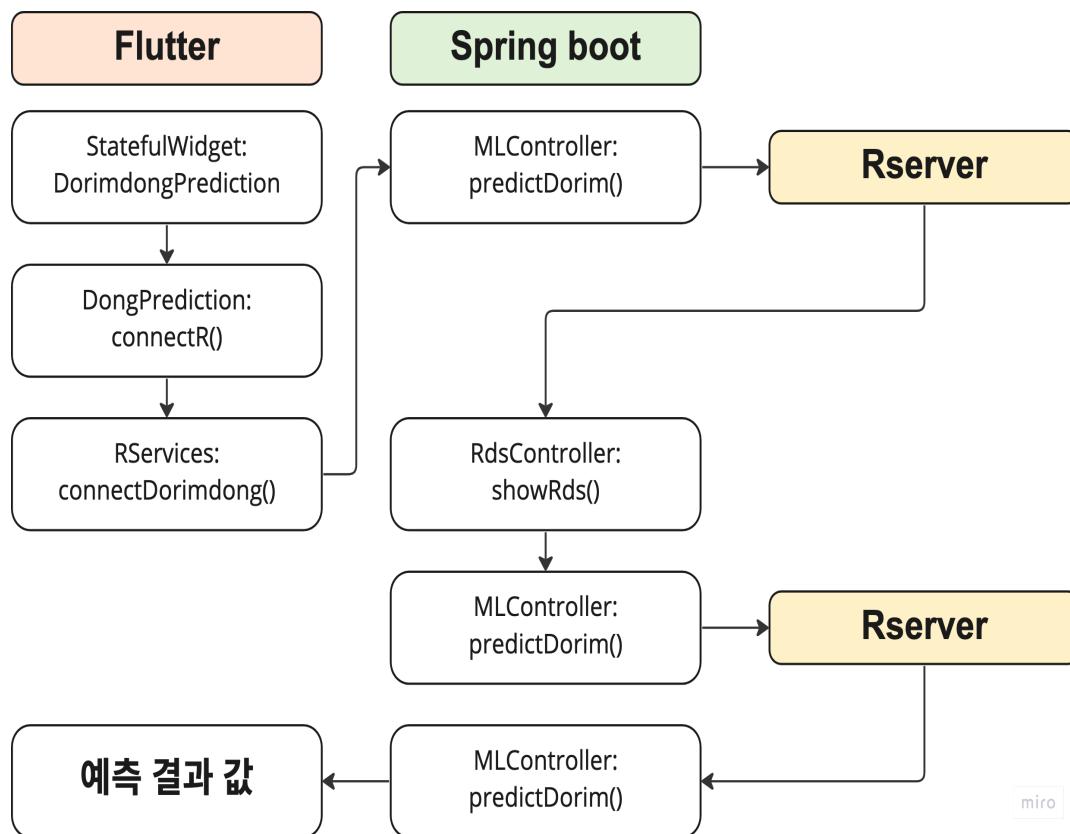
구현 화면



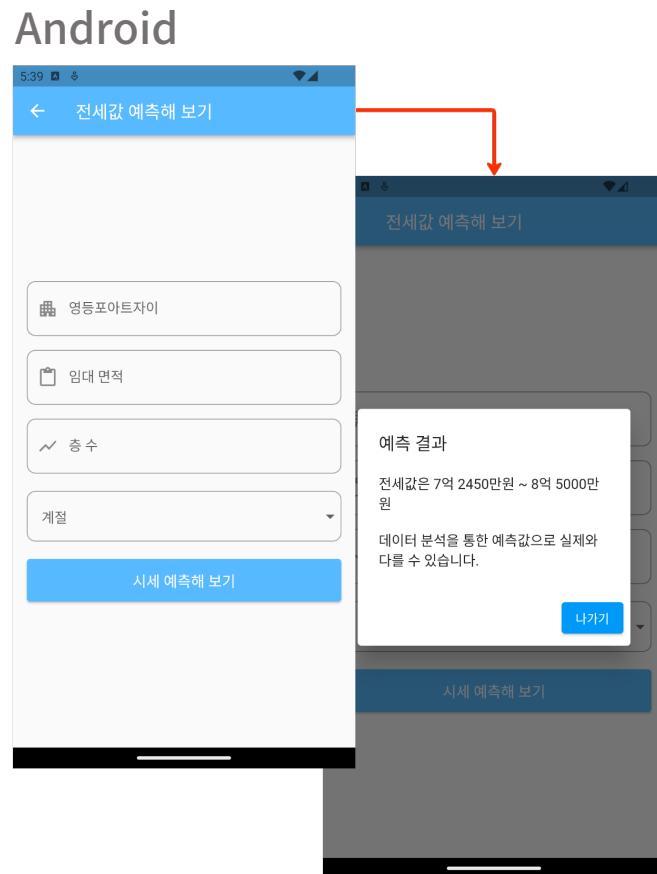
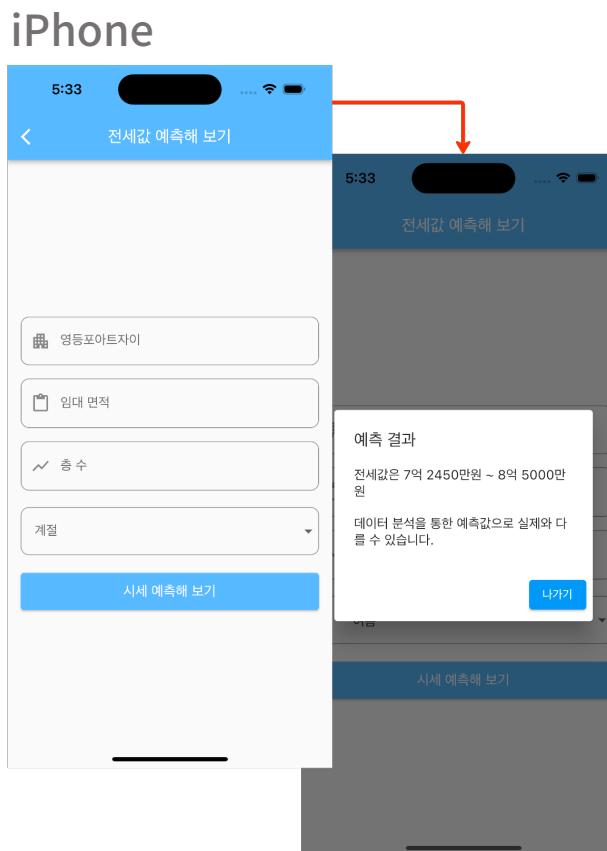
Spring Boot + AI by.R

Spring Boot로 Rserver와 연결하여 모델링한 AI에 입력값을 넣어 예측 결과 값을 UI에 보여주는 기능 구현

Flow Diagram



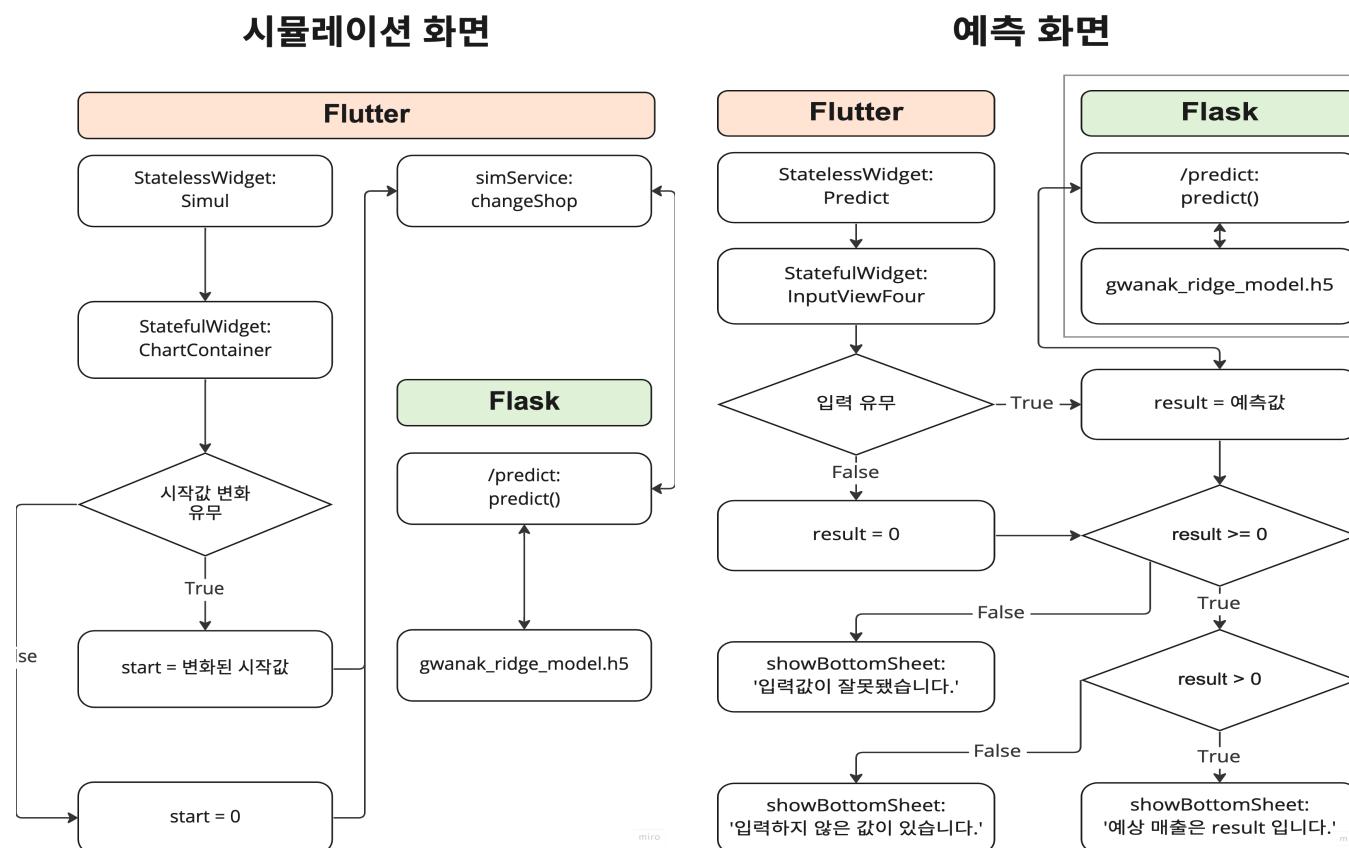
구현 화면



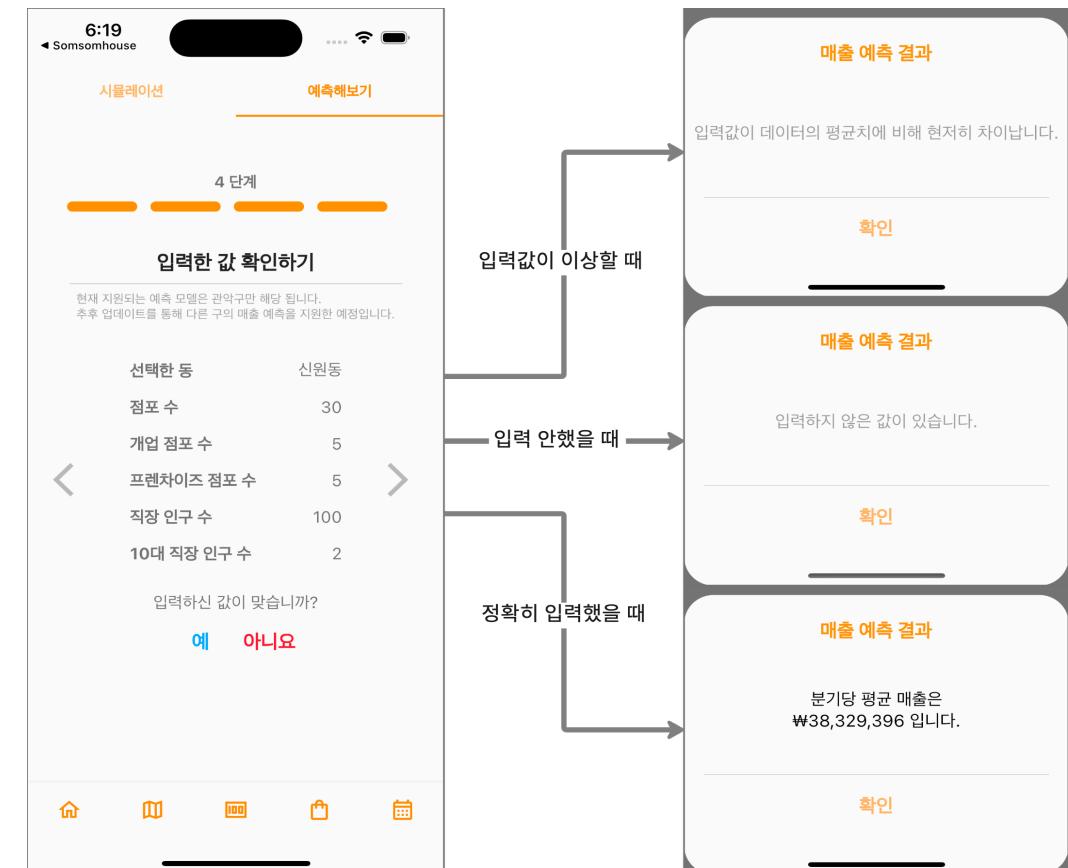
Flask + AI by.python(scikit-learn)

parameter 값을 받아 Web Server의 h5 파일을 이용해서 예측결과를 flutter 화면으로 불러오는 기능

Flow Diagram

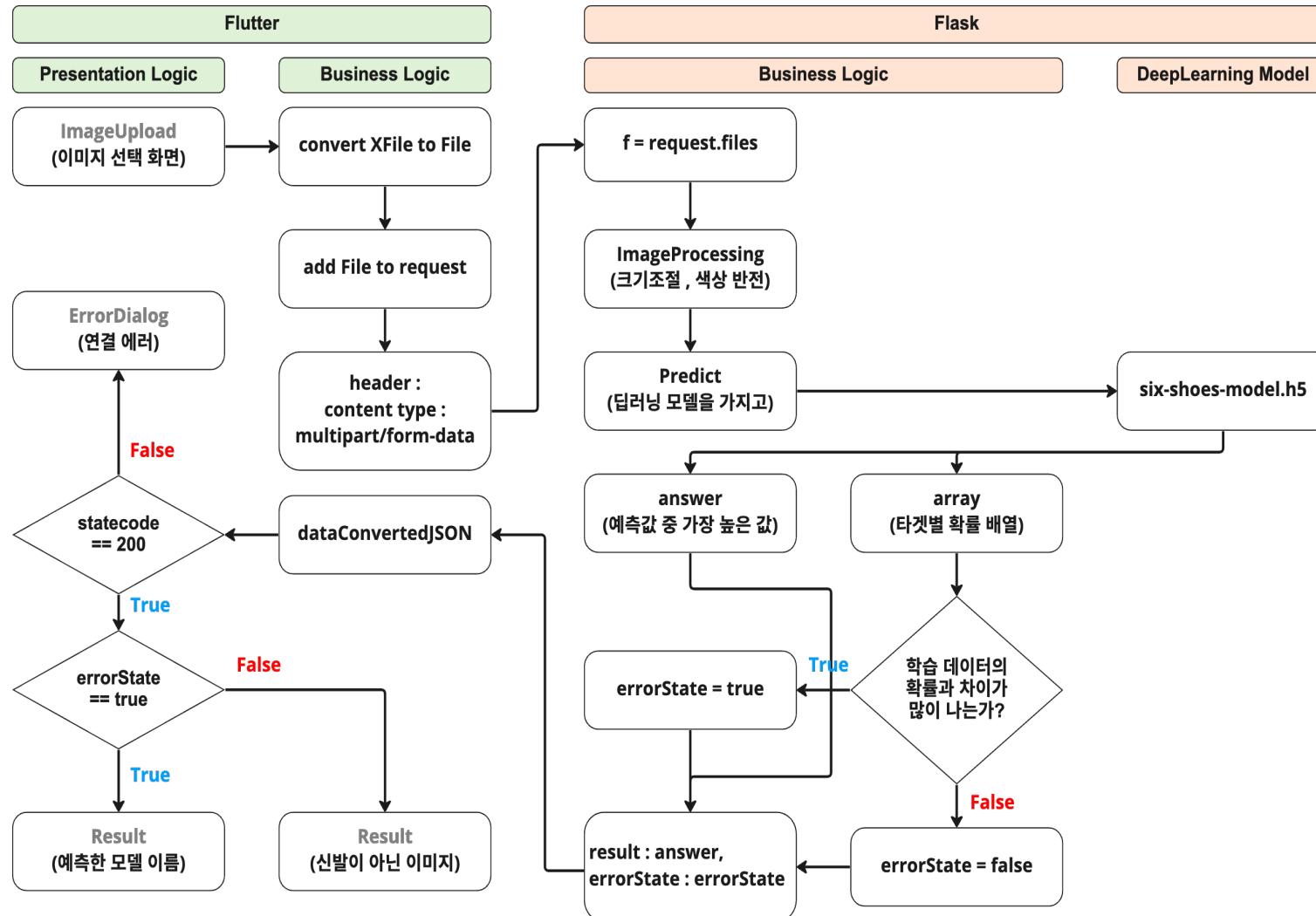


구현 화면 - 예측 화면

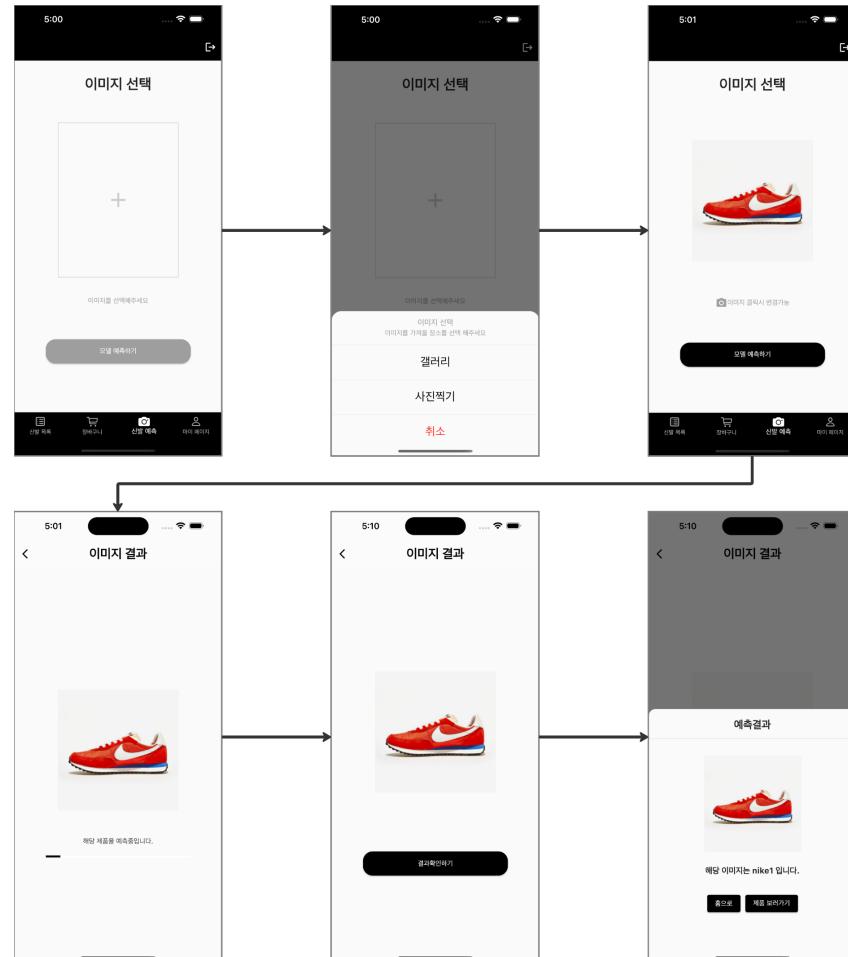


Flask + AI by.python(tensorflow - keras)

앱으로 신발 이미지를 받아서 어떠한 모델인지 맞추는 딥러닝 모델 구현



구현 화면 - 예측 화면



Open API

목 차 | 1 [서울 열린 데이터 광장](#)

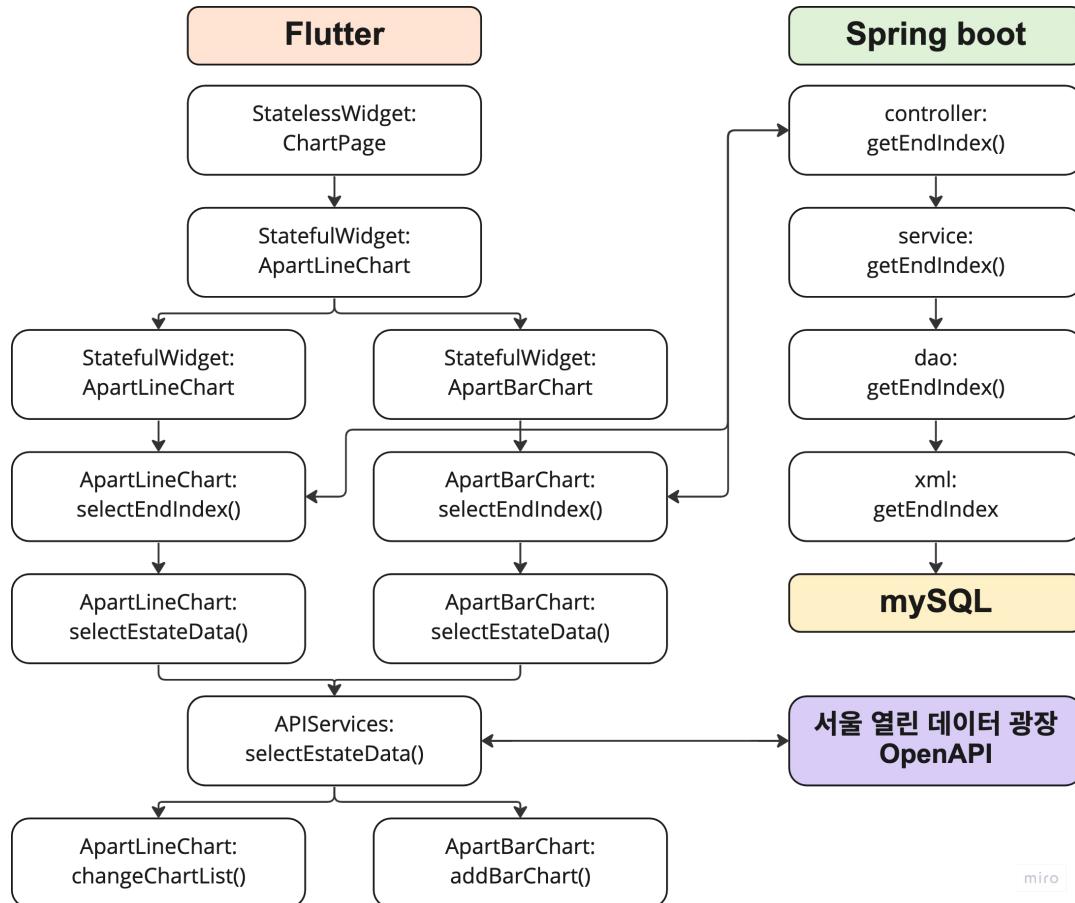
2 [Google Map API](#)

3 [Kakao Login API](#)

서울 열린 데이터 광장 - 서울시 부동산 전월세가 정보

서울 열린 데이터 광장 openAPI로 JSON 구조의 데이터를 받아서 fl-chart 패키지 이용하여 차트로 표현

Flow Diagram



구현 화면

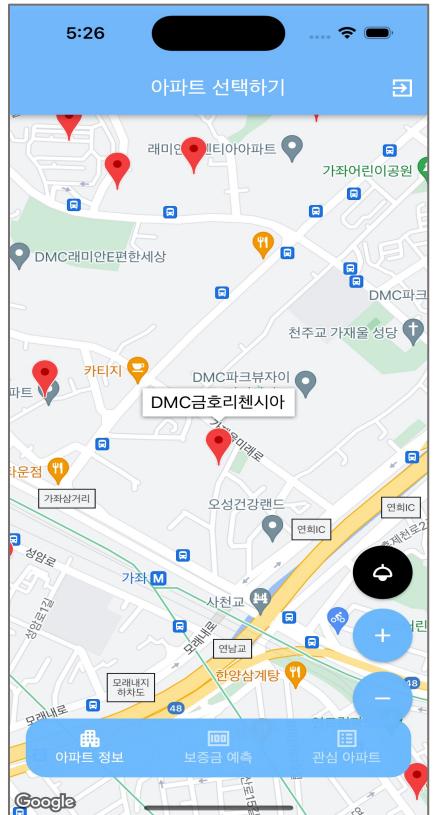


Google Map API

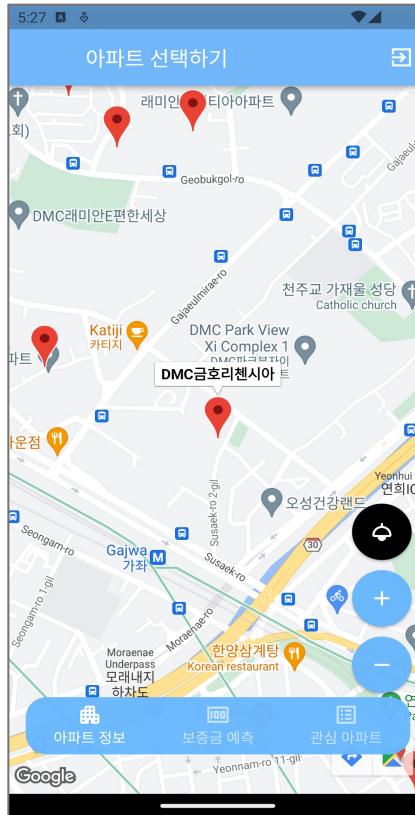
App에 위치 데이터를 마커 표시 위한 Maps SDK, 주소로 경도 위도를 찾기 위한 Geocoding API

Maps SDK for iOS , Android - App

iPhone

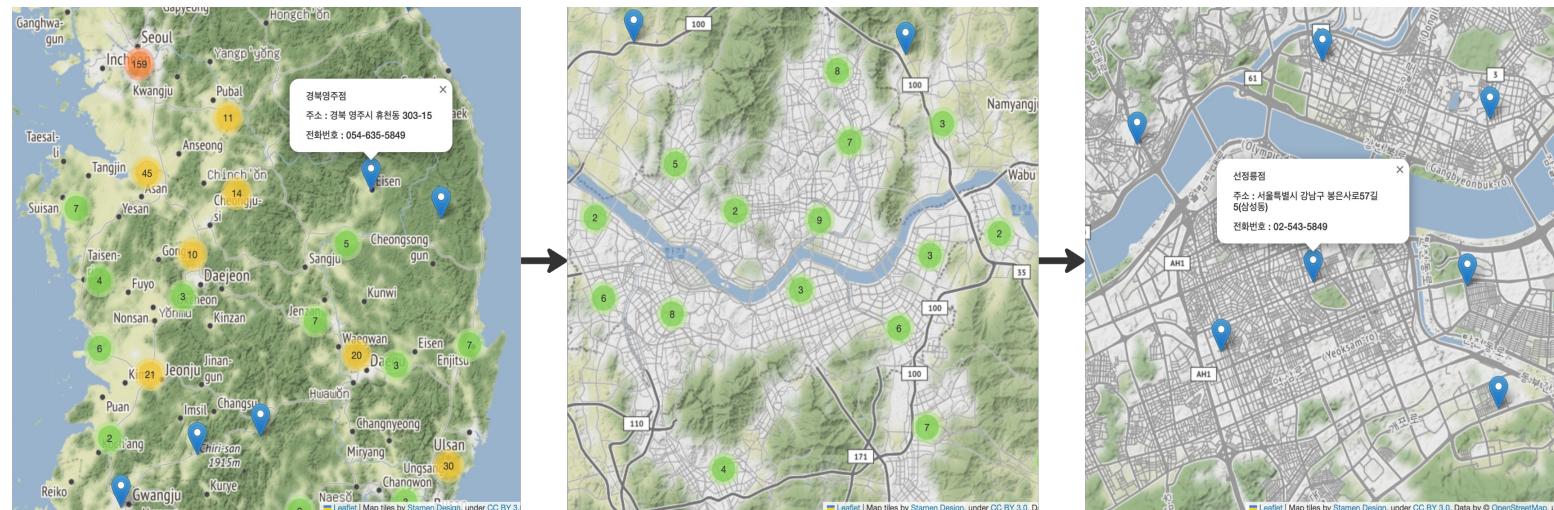


Android



Geocoding API - python

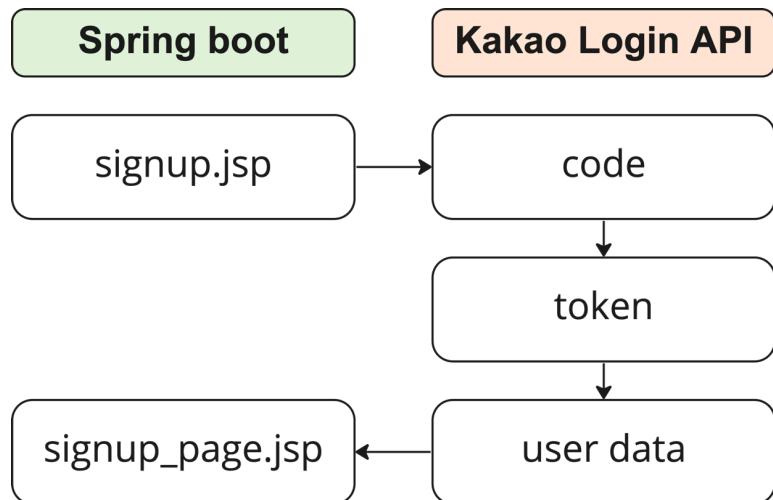
Scraping 한 데이터의 주소를 Geocoding API 이용하여 경도 위도를 찾아서 이를 python folium의 marker로 시각화하고 marker cluster를 이용하여 비슷한 범위의 마커들을 합쳐서 개수로 표시 클릭 시에 세부적인 영역으로 나뉘게 시각화 하였다.



Kakao Login API

Kakao Login API를 이용해서 회원가입 시 필요한 정보 가져오기

Flow Diagram



구현 화면

비밀번호 확인
한번 더 입력해 주세요

이름
이름을 입력해 주세요

닉네임
권순형

이메일 예시) petmily22@gmail.com 또는 petmily22@naver.com
ddilmam@naver.com

인증
인증번호를 입력해 주세요

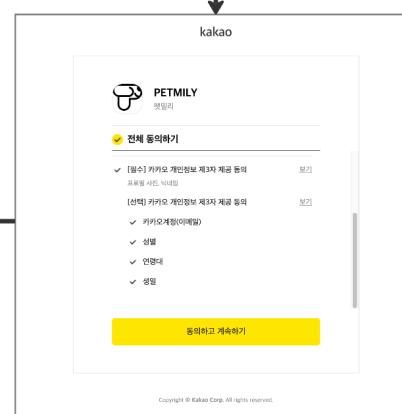
휴대폰 번호 예시) 010-1234-5678 하이픈 포함
번호를 입력해 주세요

주소
주소를 입력해 주세요

검색

상세 주소를 입력해 주세요

A registration form for a new user. It includes fields for password confirmation, name, nickname, email, verification number, phone number, and address. A yellow '인증' (Verification) button is prominent.



AI 모델링

목 차 | 1 강화학습을 이용한 딥러닝 모델링

주제 | 서울시 전세 금액 예측 모델 - python

2.1 [강화학습\(actor-critic\)](#)

2.2 [Modeling](#)

2 머신 러닝 지도학습 회귀 모델

주제 | 서울시 상권 분기당 매출 금액 예측 모델 - python

1.1 [EDA](#)

1.2 [Modeling](#)

3 머신 러닝 지도학습 분류 모델

주제 | 서울시 전세 금액 예측 모델 - R

2.1 [EDA](#)

2.2 [Modeling](#)

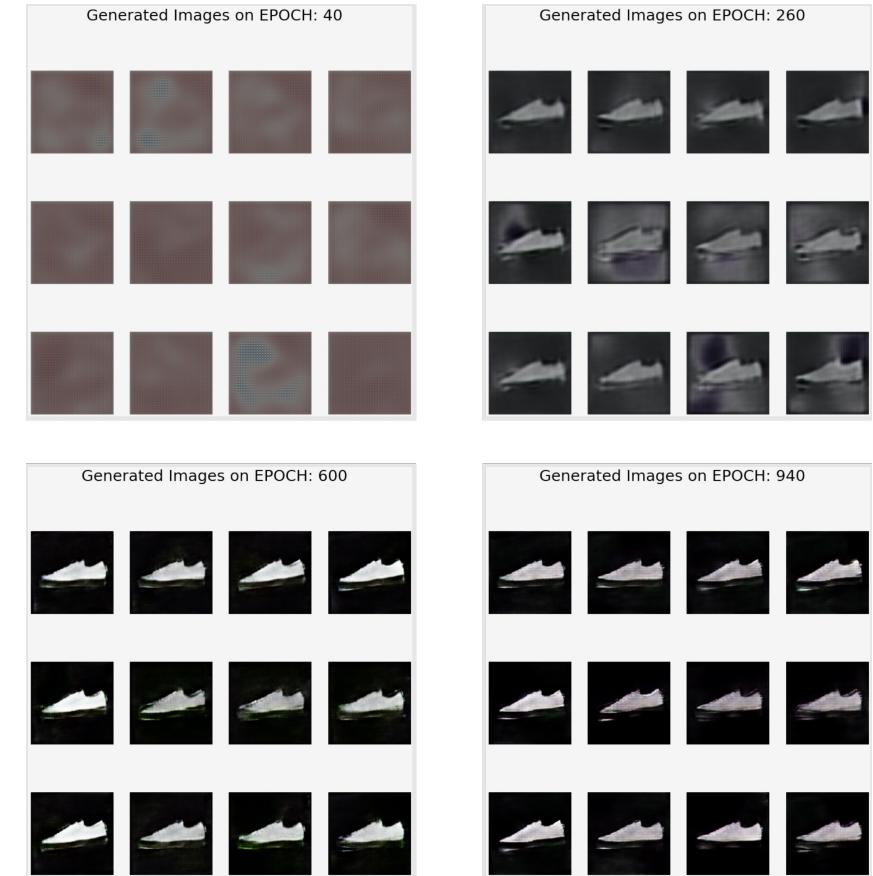
강화학습 actor-critic

원본 데이터를 이미지 반전 시키고 회전 시킨 후 강화학습을 시켜 딥러닝 모델 만들 시에 사용될 새로운 데이터들을 만든다.

이미지 프로세싱



강화학습



Deep Learning Modeling

이미지 데이터를 가지고 딥러닝 모델을 만들어 실제 사진을 가지고 예측 결과 보기

딥러닝 모델링

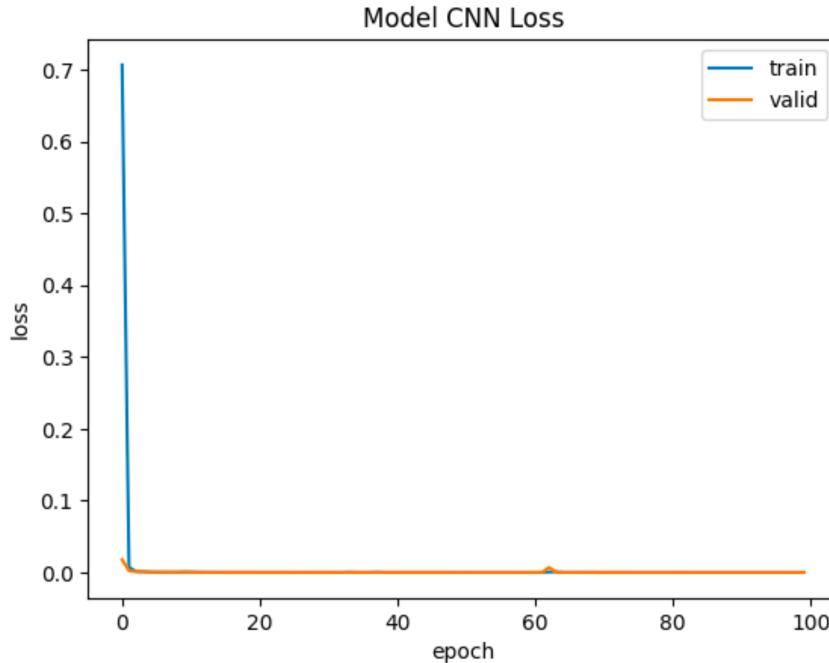
Model: "sequential_42"		
Layer (type)	Output Shape	Param #
conv2d_82 (Conv2D)	(None, 128, 128, 32)	896
max_pooling2d_81 (MaxPooling2D)	(None, 64, 64, 32)	0
flatten_41 (Flatten)	(None, 131072)	0
dense_81 (Dense)	(None, 100)	13107300
dropout_41 (Dropout)	(None, 100)	0
dense_82 (Dense)	(None, 6)	606
<hr/>		
Total params: 13,108,802		
Trainable params: 13,108,802		
Non-trainable params: 0		

딥러닝 예측률 확인하기

```

● model.evaluate(testData, testTarget)
✓ 0.3s
9/9 [=====] - 0s 21ms/step - loss: 4.5241e-07 - accuracy: 1.0000
[4.52405743089912e-07, 1.0]

```



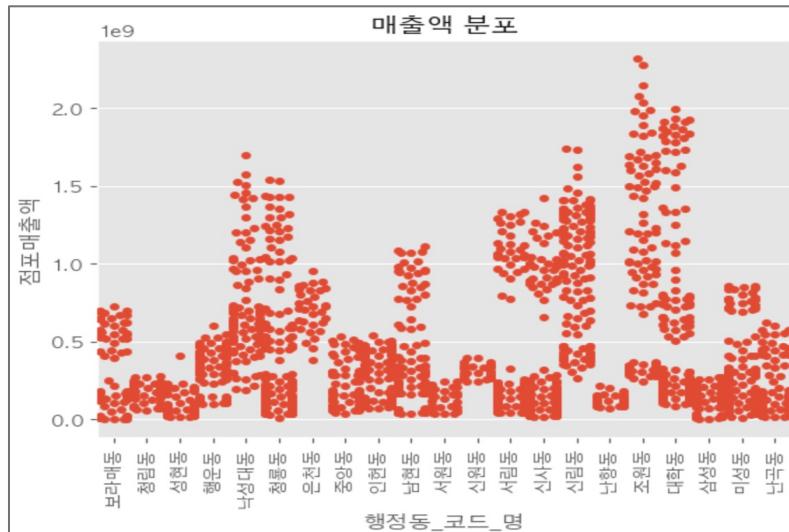
실제 데이터로 예측해보기



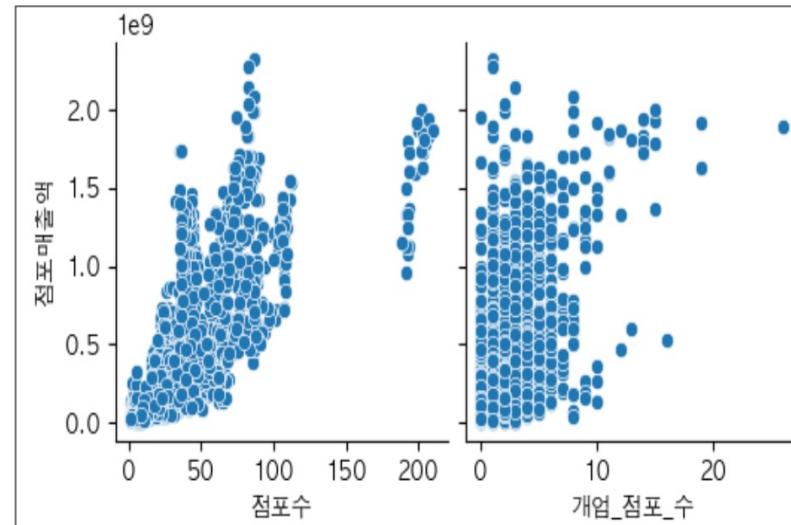
EDA

서울시 상권 데이터 중 머신 러닝 지도 학습의 회귀 모델의 feature로 사용 가능한 column 찾기

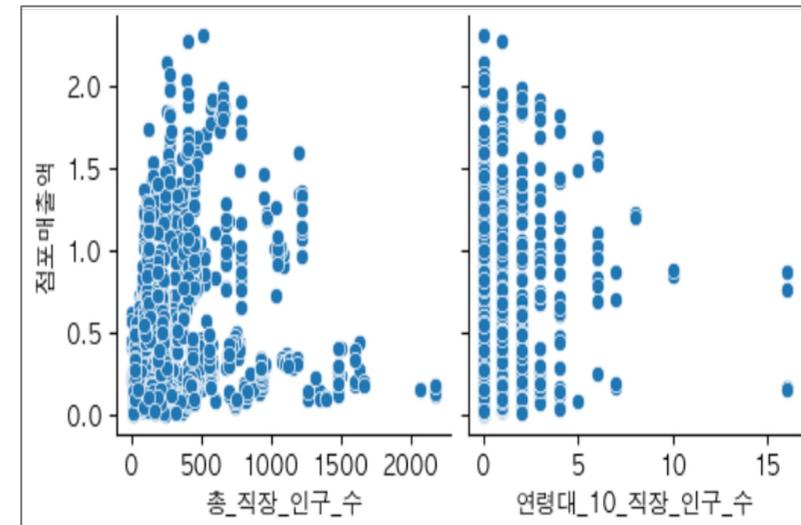
동 이름



점포 수 관련



직장 인구 수 관련



분석

동 별로 매출액의 분포가 다르다는 것을 확인할 수 있습니다.
점포 수, 개업 점포 수, 직장 인구 수, 10대 직장 인구 수가
매출액과 선형적인 관계를 가지는 것을 확인 할 수 있다.

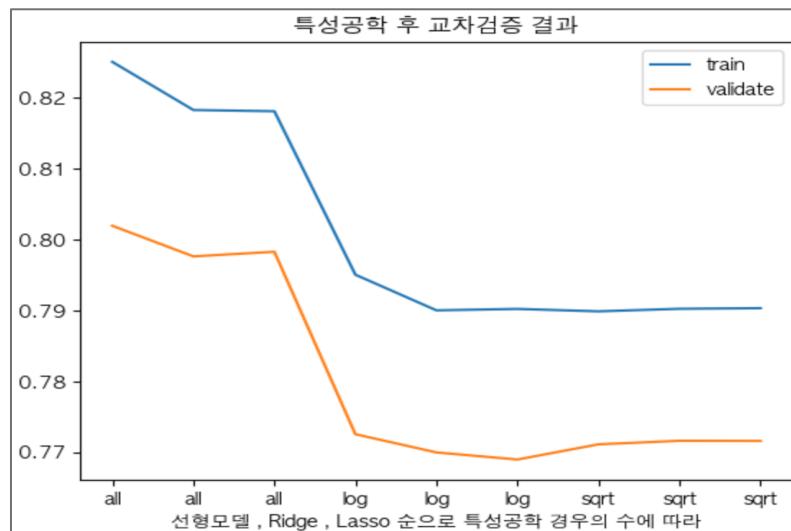
결론

선형적인 관계를 가지는 column별 매출액과의 분포가 다르고,
동 이름별로도 매출액의 분포가 다르기 때문에
위 5개의 column을 feature로 사용할 수 있다고 판단했습니다.

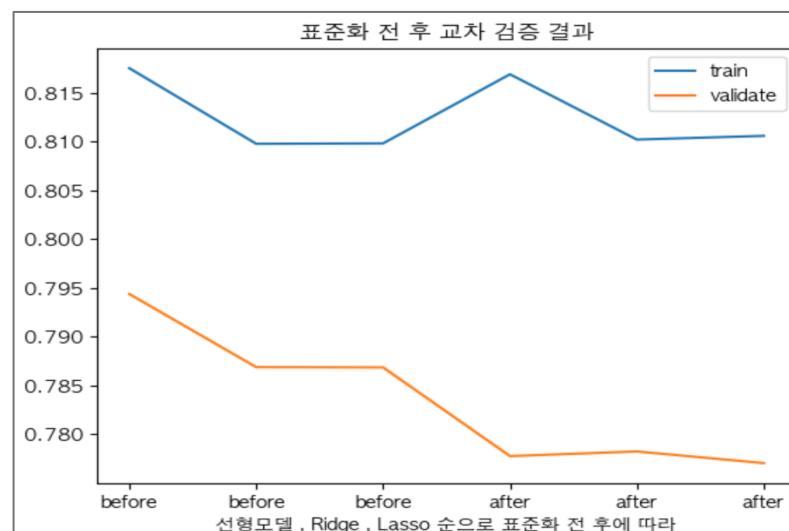
Modeling

표준화, 특성공학, 교차 검증을 통해 예측률을 높이고 grid search를 통해 일반화된 머신 러닝 모델 만들기

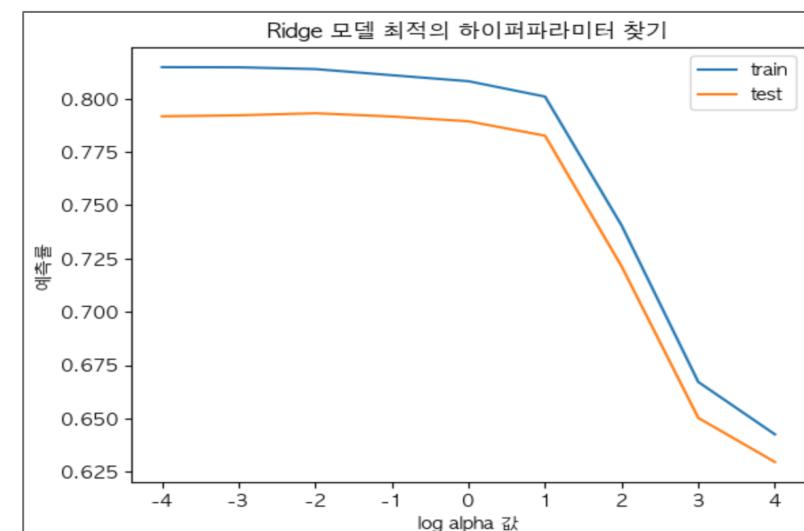
특성공학 후 교차 검증 결과



표준화 전 후 교차 검증 결과



최적의 hyper parameter값 찾기



분석

특성공학 한 것을 모두 사용했을 때 교차 검증결과가 가장 높습니다.
표준화 전후의 train 데이터의 예측률은 비슷하지만,
표준화 전의 validate 예측률이 더 높습니다.

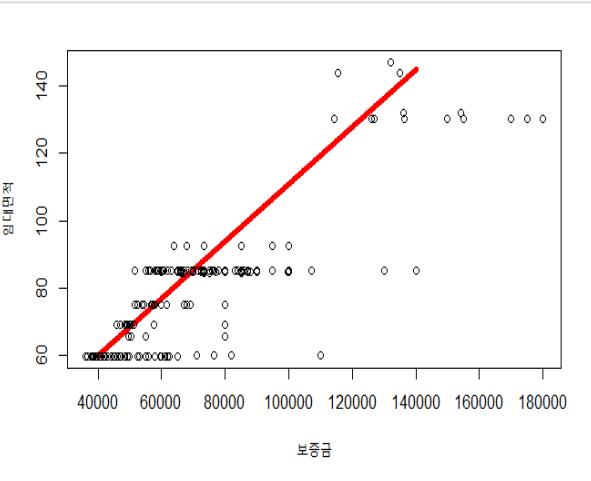
결론

특성공학 한 것을 모두 사용하고, 표준화 하지 않았을 때의 데이터를 가지고
Ridge 모델의 alpha 값을 10으로 했을 때가 예측률이 가장 높은 모델이고,
이를 매출 예상 머신 러닝 모델로 사용하기로 했습니다.

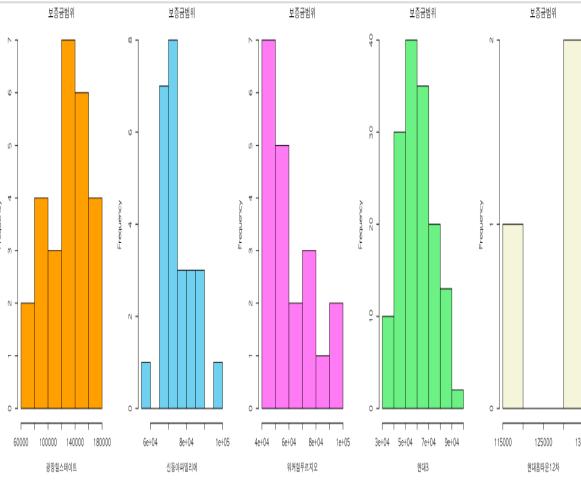
EDA

서울시 전세 데이터 중 머신 러닝 지도 학습의 분류 모델의 feature로 사용 가능한 column 찾기

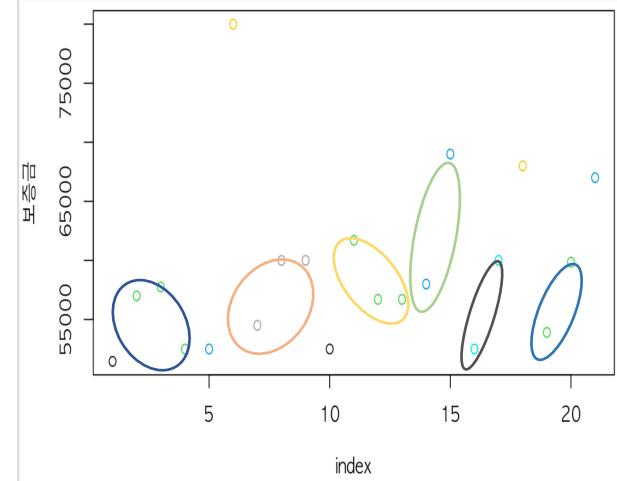
임대 면적



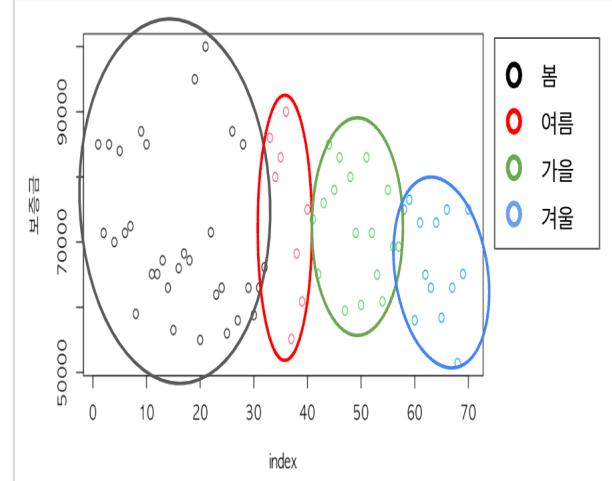
아파트 명



층 수



계약 계절



분석

보증금과 각 부제목과의 관계를 시각화 시킨 것입니다.

임대면적은 보증금과 선형적인 관계를 가진다는 것을 확인 할 수 있습니다.

나머지는 값들 별로 보증금의 분포가 다르다는 것을 확인 할 수 있습니다.

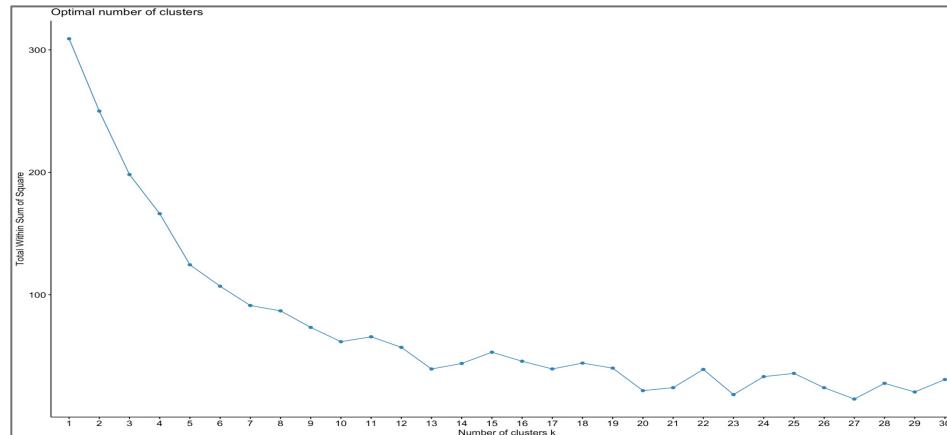
결론

위 4개의 column이 feature로서 적합하다고 판단했습니다.

Modeling

4개의 feature를 clustering 하여 label별 보증금 금액의 범위를 target으로 하여 모델링

Clustering



Modeling

예측률	정규화 전(%)	정규화 후(%)
인공신경망	47.6	98.9
랜덤포레스트	97.8	98.5
SVM	93	93.4

결론

elbow method를 통해 군집의 개수를 20으로 하기로 결정

결론

feature를 정규화 한 후

인공신경망 모델을 사용할 경우의 예측률이 제일 높으므로
해당 모델을 머신 러닝 모델로 채택하였습니다.

데이터 모델링

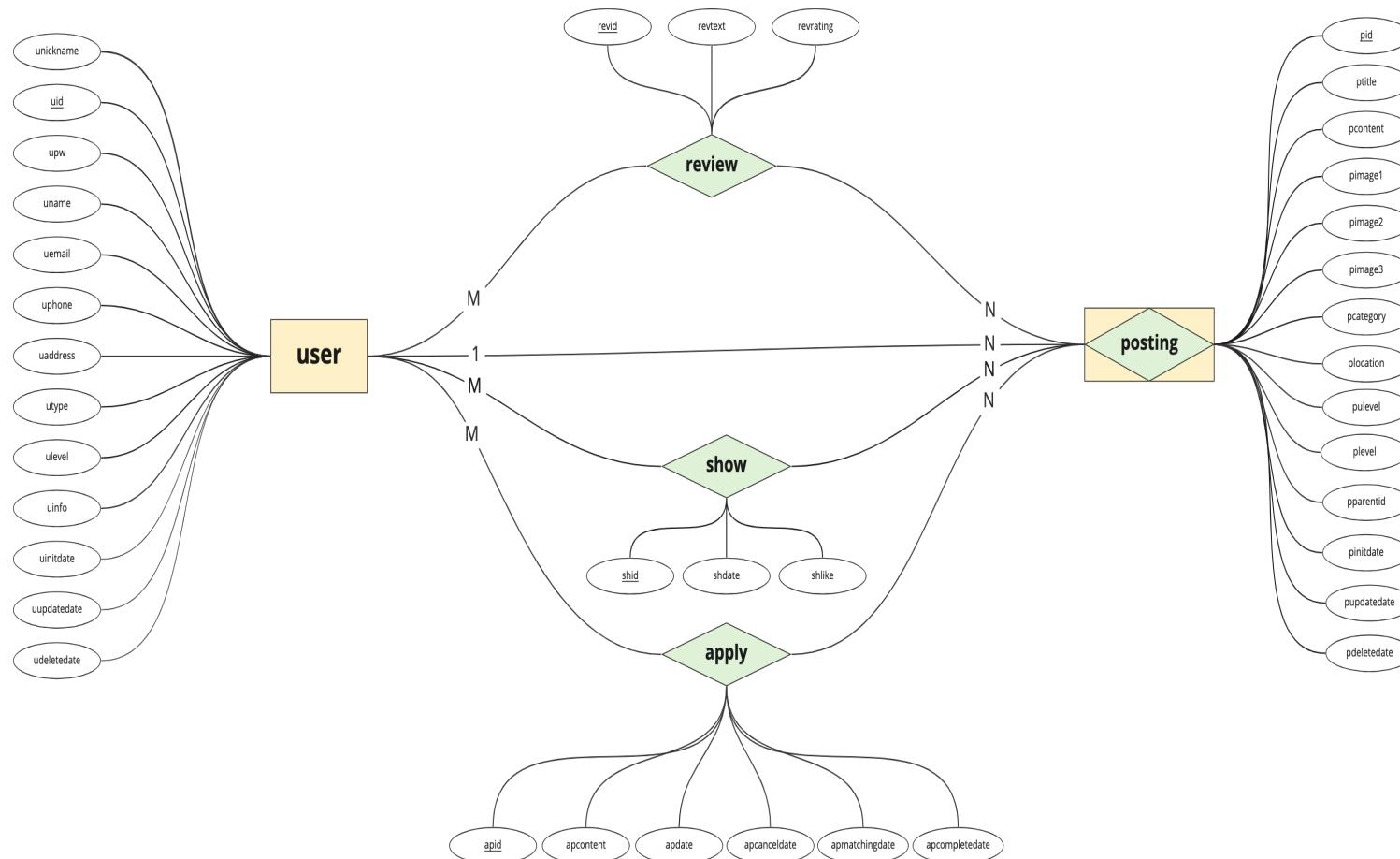
목 차 | 1 [데이터 모델링](#)

1.1 [ERD](#)

1.2 [Scraping 이용한 DB 구축](#)

RDBMS ERD

전체 ERD 중 Project의 핵심인 유저와 게시글 Entity 사이의 Relation Diagram



Entity

- **user** : 유저
- **posting** : 게시글

Relation

- **posting**
유저가 게시글을 작성한다는 관계를 가지기에 entity 이자 relation으로 표시
- **review / show / apply**
후기글과 평점 / 조회수와 좋아요 / 신청글은 유저와 게시글 사이의 관계

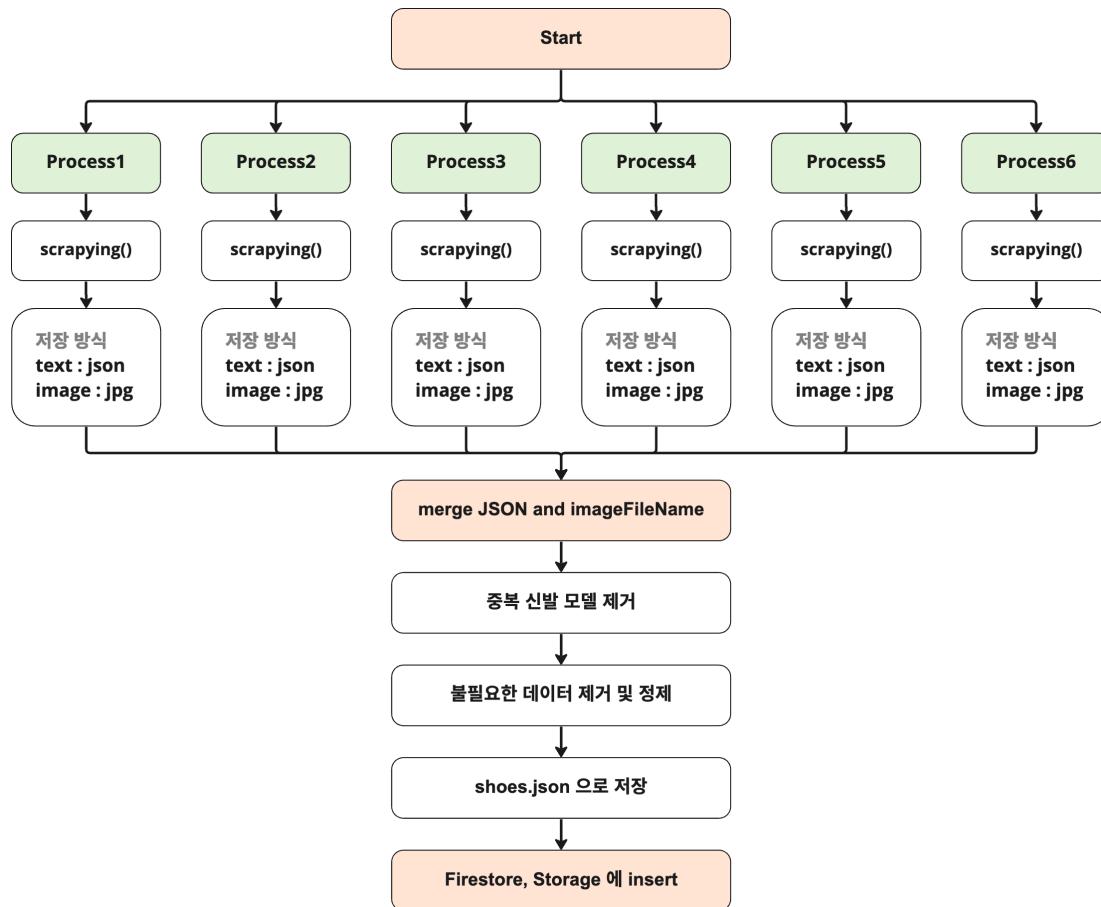
Attribute

- 게시글 삭제시 DELETE 문 사용하지 않고, UPDATE 문으로 delete date를 UPDATE한다.

Scraping 이용한 DB 구축

Selenium과 BeautifulSoup을 이용하여 Data Scraping하여 이를 Firebase Firestore와 Storage에 insert

MultiProcessing 이용하여 Firestore data insert 과정



Scraping 과정 설명

