
머신러닝을 이용한 중고차 시세예측 앱



작성일 : 2022. 01. 12 (목)

작성조 : 차사조

이름 : 김영혁, 이예진,
김진형 박태권



목차

프로젝트 개요	1	Page Flow Diagram	16
일정	2	Flutter 기능구현_스플래쉬	17
Git Log	3	Flutter 기능구현_로그인&회원가입	18-20
사용TOOL	4	Flutter 기능구현_예측값 조회	21-23
분석 모델 컬럼 소개	5	Flutter 기능구현_게시판	24-26
Data 정제 과정	6-11	Flutter 기능구현_채팅	27
K means의 K 값 / 군집분포	12	Flutter 기능구현_다크모드	28
브랜드 모델 별 예측모델 표	13	Spring RESTful API	29
ERD	14	향후계획	31
본	15	부록	32



프로젝트 개요

• 개요

어느 도시를 가도 차가 없는 곳은 볼 수 없습니다. 자동차는 이제 우리 생활에 있어서 꼭 필요한 이동수단입니다. 돈을 벌고 차를 사는 젊은 친구들도 많이 볼 수 있습니다.

하지만 소득에 따라 신차, 중고차를 선택하여 구매를 하게 되는데, 중고차를 구매하게 될 때 허위딜러로 인한 사기 및 중고차에 대한 지식부족으로 많은 사람들이 손해를 보고 있습니다. 그렇다고 중고차 시세조회를 쳐서 중고차 시세를 보아도 딜러가 올린 임의 값일 뿐 나의 차량정보를 입력하게 되면 예측값 대신 해당 사이트 판매내역 최고값을 보여주며, 자세한 내용은 딜러를 연결시켜줍니다.

모든 사람이 딜러와 컨택하고 싶어하지 않고, 가볍게 알아보시는 분들에겐 부담일 수 있습니다.

코로나 19가 발병한 이후 신차의 출고는 짧게는 수개월 길게는 1년이 넘어가는 차 종들도 생기며 중고차 시장은 더욱 활발해지고 있습니다. 중고차 시장이 활발해 진 이유는 반도체 수급이 원활하지 않아 신차의 출고가 늦어지기 때문입니다. 수급이 원활하지 않은 이유는

첫째, IT업계의 호황으로 반도체 공급이 늘어나고 IT업계의 반도체가 차량 반도체보다 고부가가치 이므로 반도체 생산공장에서는 이윤을 남기기 위하여 IT업계에 반도체를 우선으로 공급합니다.

둘째, 단일공급사 선정(싱글벤더)입니다. 자동차 반도체의 품질은 안전과 연관성이 있어 생산업체의 안전성과 신뢰성을 보게됩니다. 그에 따른 과정으로 검증 및 안정성 테스트는 비용과 직결됩니다. 한번 공급사 선택을 하면 변경이 어렵기 때문입니다.

셋째, 러시아와 우크라이나의 전쟁으로 인한 반도체 원료 수급 문제입니다. 우크라이나는 반도체의 주요 물질인 네온가스, 크립톤을 공급하는 국가입니다. 하지만 러시아와의 전쟁으로 인하여 문제가 생겼고, 대부분의 원료를 수입하는 우리나라에선 생산에는 차질이 없으나, 재료수급이 어렵기 때문에 제조가 늦어집니다.

이러한 이유로 신차의 출고가 늦어지기 때문에 중고차를 구매하고자 하는 고객들이 늘어나고 있습니다.

이렇게 활발해진 중고차 시장은 모든 고객에게 친절하지는 않습니다. 신차를 출고하려면 오래걸리고 그래서 중고차를 구매하는데 허위딜러 및 지식부족으로 사기를 당하는 고객들도 빈번하게 볼 수 있습니다. 그래서 우리는 '중고차의 간단한 시세조회를 통하여 소비자에게 지식을 전달하는 것은 어떨까?'라는 의견에 도달했습니다. 위에서 말했듯이 '중고차 시세조회 사이트는 딜러를 연결시켜주는 사이트지 대략적인 시세를 볼 수 없다.'라고 판단하여 중고차 판매데이터를 통하여 소비자에게 가격의 범위를 대략적으로 전달할 필요성을 느끼고, 고객에게 보다 현명한 선택을 지원해주고 싶어 중고차 예측모델을 생성하기로 결정했습니다.



개발 일정

분류	항목	D-10	D-9	D-8	D-7	D-6	D-5	D-4	D-3	D-2	D-1	D-day
기획	컨텐츠 기획											
	개발 기능 정의											
	R&R 확립											
예측모델생성	데이터 수집											
	데이터 정제											
	예측모델 생성 및 테스트											
DB설계	로직 및 서비스 정책 구현											
	데이터베이스 구현											
	단위테스트											
디자인	화면 디자인 구축											
	세부 디자인 확립											
	단위테스트											
개발	R&D											
	기능 설계 및 구현											
	단위테스트											
테스트 및 발표	통합 테스트 및 안정화											
	PPT 자료준비											
	발표 및 시연											



Git Log



Jinhyeong	480ee6b	Today 2:25 PM
Bhan Kim	ddb5d61	Today 2:05 PM
jinhyeong	e220f99	Today 2:24 PM
Bhan	fd5672a	Yesterday 7:52 PM
jinhyeong	83942f0	Today 2:23 PM
=	b552bfa	Today 12:51 PM
jinhyeong	f432c4e	Today 12:36 PM
Bhan Kim	104882a	Today 10:46 AM
Bhan Kim	301192c	Today 10:38 AM
jinhyeong	5d1eabd	Today 12:35 PM
=	57a1df3	Today 12:50 PM
=	bd54369	Today 10:40 AM
Bhan Kim	0929f67	Yesterday 8:34 PM
Bhan Kim	1224d36	Yesterday 7:58 PM
javanyam	c84f086	Yesterday 7:57 PM
javanyam	31a8cb0	Yesterday 7:56 PM
Bhan	a161ecf	Yesterday 7:53 PM
Bhan Kim	fd9e977	Yesterday 7:58 PM
=	65ce4c2	Today 10:38 AM
=	f87cb13	Yesterday 7:48 PM
javanyam	ca8908c	Yesterday 6:41 PM
Bhan Kim	cce5f38	Yesterday 6:29 PM
javanyam	ac68473	Yesterday 6:41 PM
=	7047a14	Yesterday 7:47 PM
=	afec1bc	Yesterday 4:25 PM
javanyam	605f2e5	Yesterday 3:40 PM
Bhan Kim	2821961	Yesterday 3:38 PM
Bhan Kim	eedb45c	Yesterday 3:32 PM
jinhyeong	b6696e8	Yesterday 12:48 PM
jinhyeong	31e7780	Yesterday 3:43 AM
jinhyeong	646dc8	Yesterday 3:40 AM
Bhan Kim	f4f2857	Yesterday 3:31 PM
javanyam	56481f5	Yesterday 3:40 PM
javanyam	2c9ae65	Yesterday 1:55 AM
jinhyeong	94624d7	Yesterday 1:51 AM
javanyam	e860b87	Yesterday 1:50 AM
javanyam	3db272e	Yesterday 12:00 AM
javanyam	5d69996	Jan 11, 2023 6:21 PM
javanyam	27f2f50	Jan 11, 2023 6:20 PM
javanyam	09850c5	Jan 11, 2023 4:03 PM
javanyam	09f903a	Jan 11, 2023 4:02 PM
jinhyeong	f45c6a1	Yesterday 1:51 AM
=	3b46bd5	Yesterday 4:24 PM
=	6788a16	Yesterday 12:44 AM
Bhan Kim	d7e6f78	Yesterday 12:41 AM
=	efc8559	Yesterday 12:44 AM

fix dark mode button	Bhan Kim	1b4e90e	Jan 11, 2023 11:58 PM
make darkmode switch button	Bhan Kim	6c939fc	Jan 11, 2023 11:42 PM
Merge remote-tracking branch 'origin/jinhyeong' into yhKim	Bhan Kim	34ccc6d	Jan 11, 2023 11:21 PM
carsearchlist 수정	jinhyeong	7f75161	Jan 11, 2023 11:20 PM
Merge remote-tracking branch 'origin/parktekwon' into jinhyeong	jinhyeong	fc267c2	Jan 11, 2023 6:20 PM
commit carlist CRUD	=	b351095	Jan 11, 2023 4:13 PM
signup modify	jinhyeong	fe97d16	Jan 11, 2023 6:20 PM
Merge remote-tracking branch 'origin/yhKim' into jinhyeong	jinhyeong	91a508c	Jan 11, 2023 4:02 PM
detailpage_detail_comments modify	jinhyeong	94f94ae	Jan 11, 2023 4:01 PM
change spring source	Bhan Kim	78dd117	Jan 11, 2023 11:20 PM
Merge remote-tracking branch 'origin/parktekwon' into yhKim	Bhan Kim	041963f	Jan 11, 2023 2:34 PM
Merge remote-tracking branch 'origin/yejin' into parktekwon	jinhyeong	b76db5a	Jan 11, 2023 12:39 PM
Merge remote-tracking branch 'origin/jinhyeong' into yejin	jinhyeong	309dad0	Jan 11, 2023 12:05 PM
boardPage Error Modify	jinhyeong	336b28e	Jan 11, 2023 12:02 PM
error modify	jinhyeong	db5b866	Jan 11, 2023 12:04 PM
detail page, update page	jinhyeong	e85de34	Jan 11, 2023 10:33 PM
Merge branch 'yejin'	jinhyeong	8af95b9	Jan 10, 2023 9:41 PM
fix board	jinhyeong	b8af4eb	Jan 10, 2023 9:40 PM
Merge branch 'main' into yejin	jinhyeong	090e28e	Jan 10, 2023 5:09 PM
commit carlist page	=	f72537f	Jan 11, 2023 12:38 PM
Merge remote-tracking branch 'origin/jinhyeong' into parktekwon	=	98bcd08	Jan 10, 2023 5:38 PM
Merge remote-tracking branch 'origin/jinhyeong' into parktekwon	=	c53904e	Jan 10, 2023 5:20 PM
fix design login signup page	Bhan Kim	2a78332	Jan 11, 2023 2:29 PM
로그인 회원가입 체팅 게시판 취합작업	jinhyeong	303774e	Jan 10, 2023 7:54 PM
Add Login	jinhyeong	5a33a3c	Jan 10, 2023 5:38 PM
Merge remote-tracking branch 'origin/yejin' into jinhyeong	jinhyeong	f89b2a2	Jan 10, 2023 5:09 PM
make for board contents	jinhyeong	4f0d1aa	Jan 10, 2023 5:07 PM
Merge remote-tracking branch 'origin/main' into jinhyeong	jinhyeong	4fd6b60	Jan 10, 2023 5:09 PM
change MaterialApp to GetMaterialApp	Bhan Kim	e9b2968	Jan 10, 2023 5:04 PM
commit	jinhyeong	5943ccb	Jan 10, 2023 5:09 PM
Add model	jinhyeong	47a0f3b	Jan 9, 2023 9:34 PM
Add Spring	jinhyeong	1f2969b	Jan 9, 2023 9:00 PM
Add Chatting	jinhyeong	7c785fc	Jan 9, 2023 8:57 PM
flutter dependencies add getx,chart,firebase_auth	jinhyeong	a1b36a8	Jan 8, 2023 1:13 AM
modify	jinhyeong	0482aa5	Jan 7, 2023 1:40 AM
flutter setting	jinhyeong	0a0f287	Jan 7, 2023 1:30 AM
test	jinhyeong	3334732	Jan 6, 2023 8:49 PM
test	jinhyeong	80e310d	Jan 6, 2023 8:42 PM
git ignore	jinhyeong	7ff0815	Jan 6, 2023 8:38 PM
Initial commit	wlsgud0452	335c1ab	Jan 6, 2023 8:29 PM



사용 TOOL

사용툴



Ver 3.3



Ver 17



Ver 4.25.0



Ver 4.2.2



Apache Tomcat

Ver 9.0



Dart
Ver 2.18.5



Ver 1.74.2



Ver 3.0.1



Ver 8.0ce



협업툴



GitHub



Notion



miro





R DATA 컬럼소개 & 모델 선정 방법

• 컬럼소개

- model : 모델
- year : 연식
- price : 가격
- transmission : 자동, 수동, 세미오토
- mileage : 주행거리
- fuelType : 연료타입 디젤, 휘발유, 기타, 하이브리드, 전기
- tax : 세금
- mpg : 마일퍼 갤런, 연비
- engineSize : 엔진 사이즈 cc

모델 선정

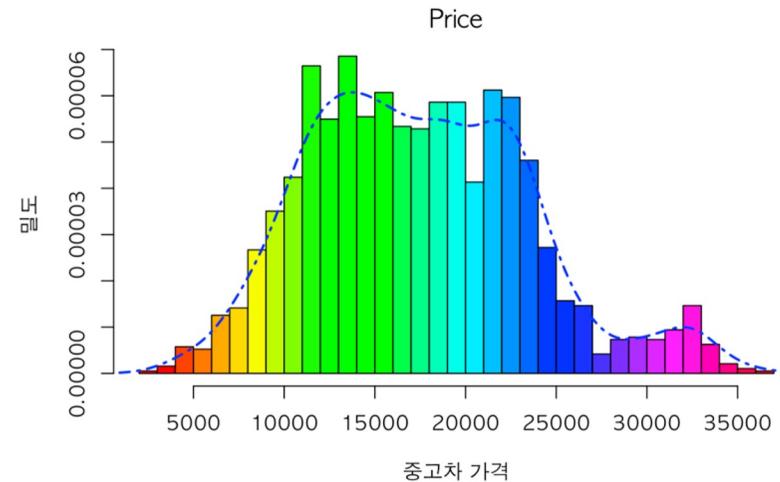
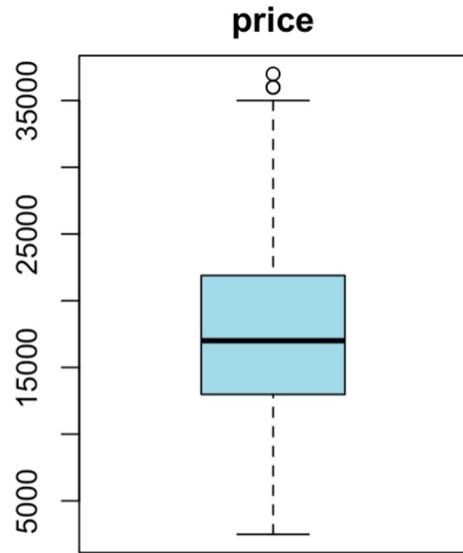
Var1 <fctr>	Freq <int>
A3	1929
Q3	1417
A4	1381
A1	1347
A5	882
Q5	877
Q2	822
A6	748
Q7	397
TT	336
A7	122
A8	118
Q8	69
RS6	39
RS3	33
RS4	31
RS5	29
R8	28
S3	18
SQ5	16
S4	12
SQ7	8
S8	4
S5	3
A2	1

- Data.frame에 등록된 모델 중 Row 수가 가장 많은 모델 3가지를 선정하여 데이터 정제 및 예측모델을 만들기로 결정



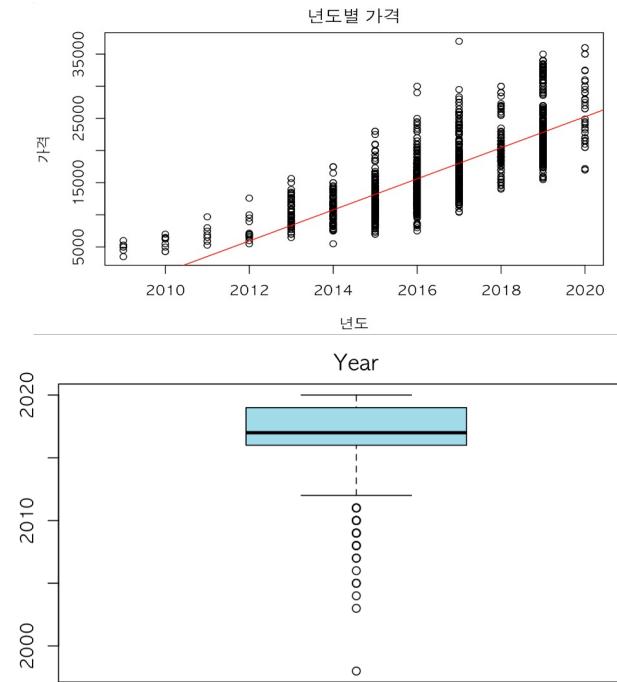
Price & Year Column 정제

Price Column



- Target column인 price는 이상치가 확인되지 않고 정규분포를 그리고 있어서 이 모델의 경우 정제를 하지 않았지만, 특이치가 나온 모델은 특이치 제거를 하여 사용함

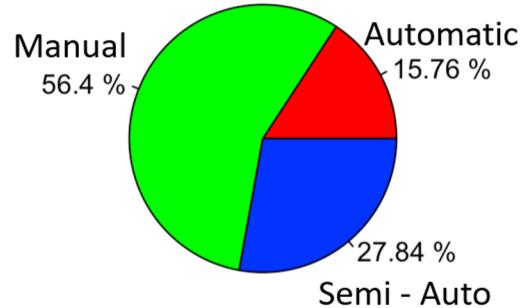
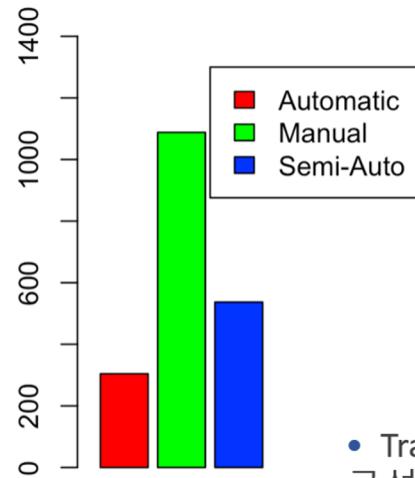
Year Column



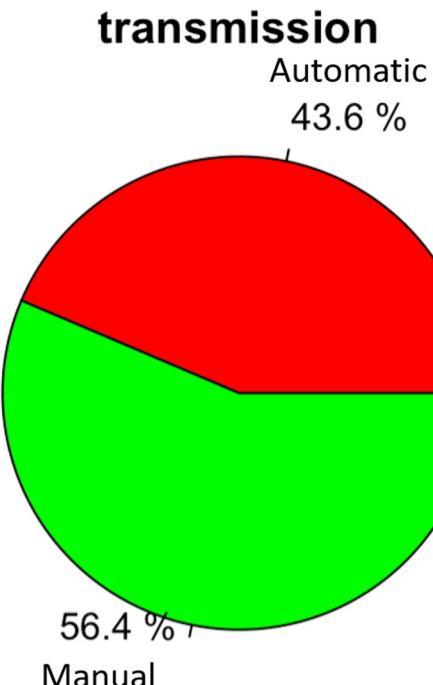
- 근거 :
 - 연도별 가격 데이터를 회귀분석한 결과 회귀선이 시작되기 이전의 year Data는 삭제하기로 결정
- 결과 : 2010년도 이후 연도부터 사용하기로 결정



Transmission Column 정제



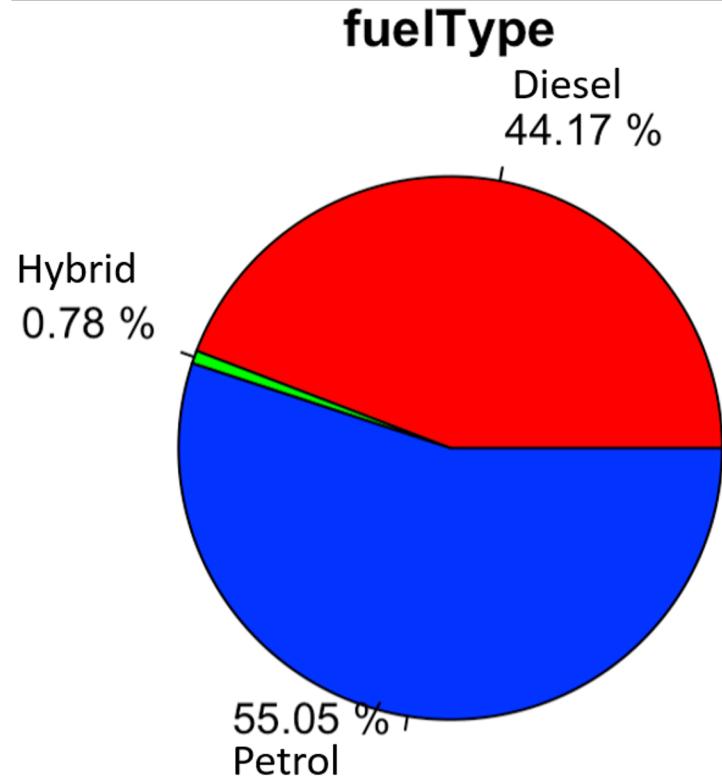
- Transmission Column은 3가지로 구성되어 있고, 셋의 비율이 피처 컬럼으로 사용되기에 적절하다고 판단



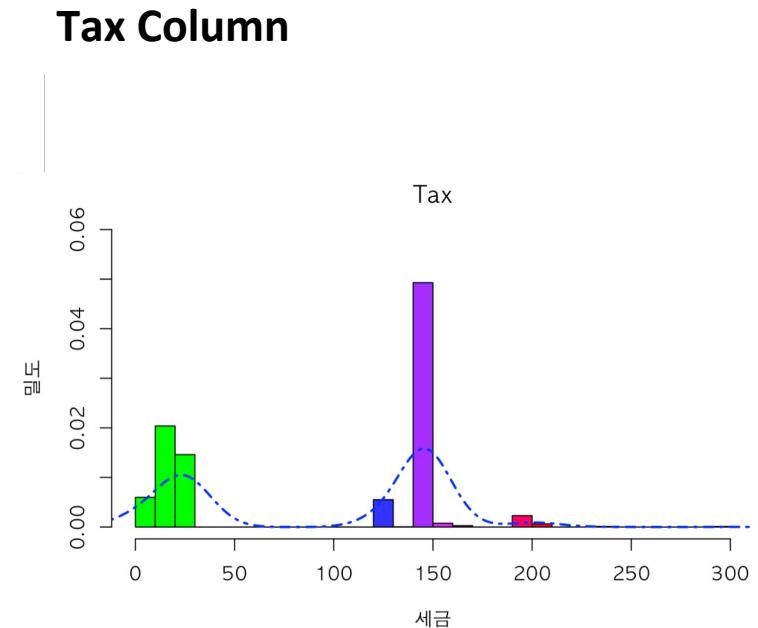
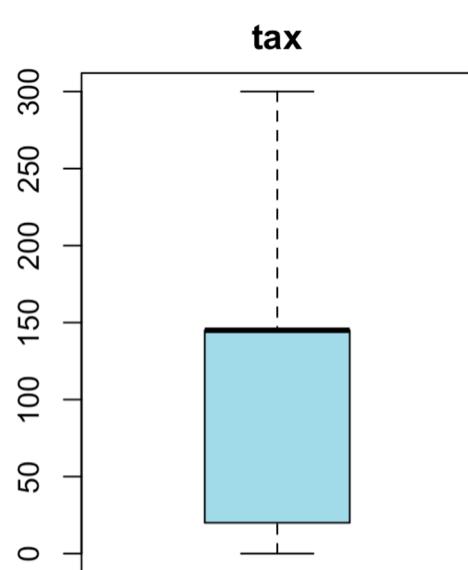
- Automatic과 Seme-Auto 컬럼을 합치고, Manual 컬럼과 비교 시 비율의 차이가 있어 피처컬럼으로 사용하기로 함.
- 합친 이유는 Semi-Auto는 Automatic과의 방식이 거의 일치하며, 클러치가 없는 점, 영국의 중고차사이트([부록 참조](#))를 통하여 합치게 됨.



Fuel Type & Tax Column 정제



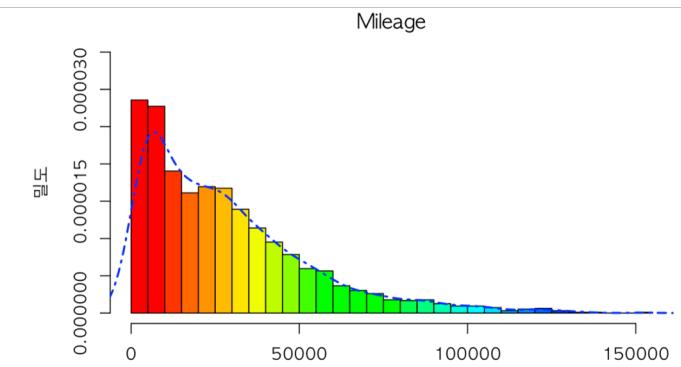
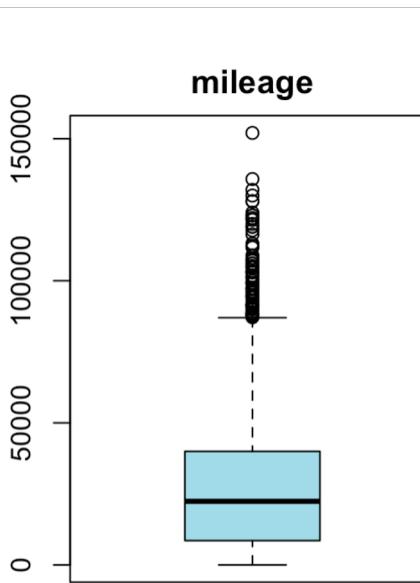
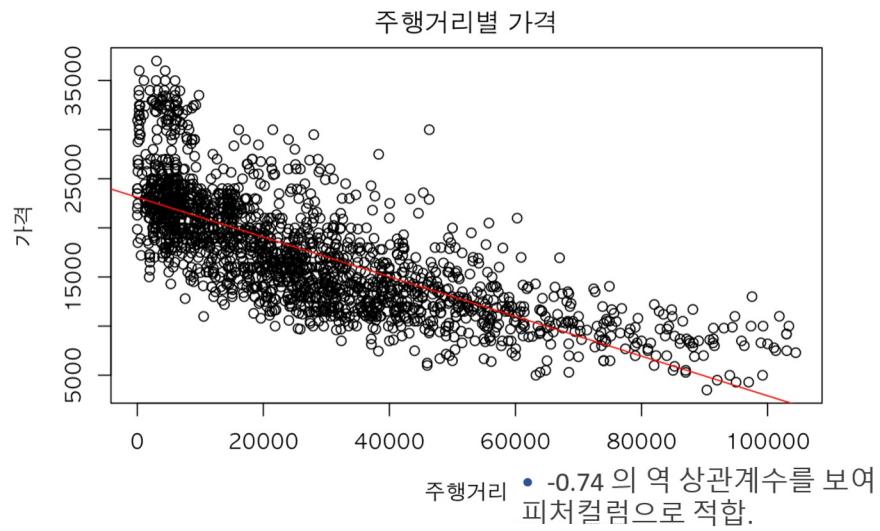
- Fuel Type 컬럼은 Petrol, Diesel, Hybrid, 3가지로 구성되어 있음 Hybrid Column은 0.78%로 데이터가 너무 적어 삭제하고 Petrol과 Diesel Column만을 사용하기로 결정



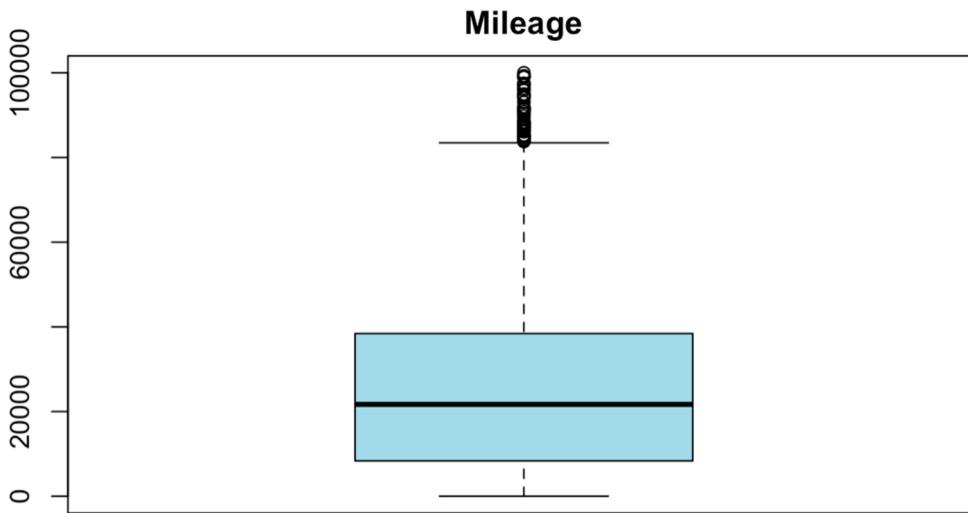
- Tax은 상관계수는 0.66 으로 높은 관계를 나타내지만, 0 값이 132개로 많은 비중을 차지하고 있어, 중위 값이 3사 분위 값은 같아서 제거하기로 결정



Mileage Column 정제



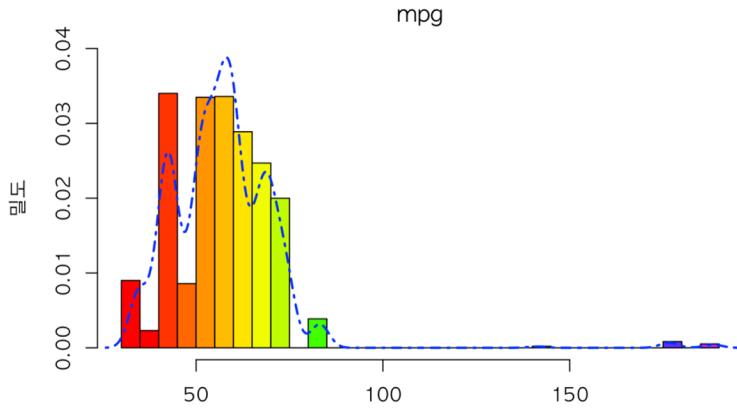
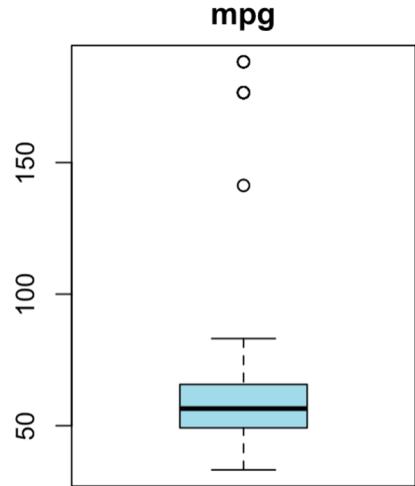
- Histogram과 boxplot에서 Data의 분포도가 일정구간에 집중되어 있지만, 특이치의 값들은 집중된 구간과의 거리가 먼 것을 알 수 있고, 데이터가 없다는 것을 알 수 있음. 이러한 결과로 특이치를 확인하고, 이를 제거하여 사용하기로 결정함



- 정제가 완료된 mileage Column

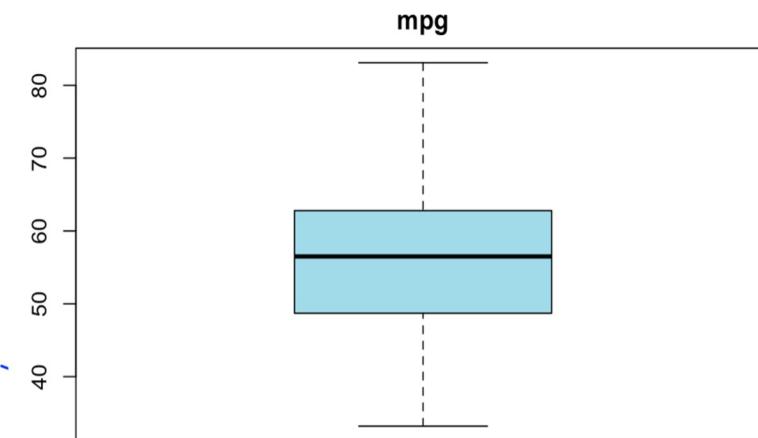
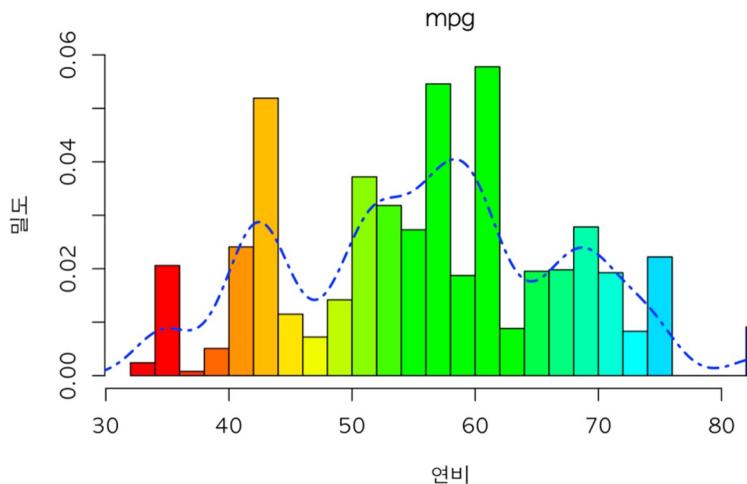


Mpg Column 정제



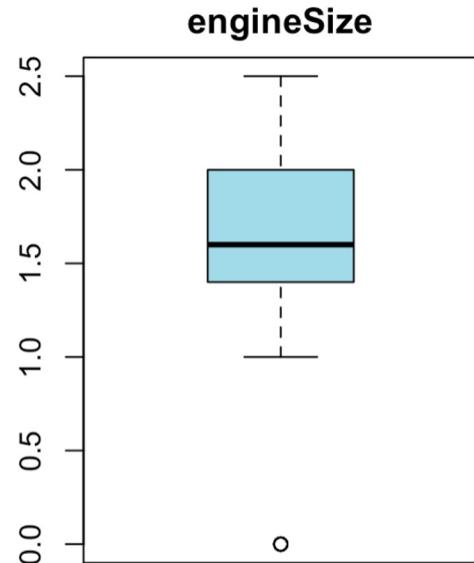
- Histogram과 boxplot에서 Data의 분포도가 일정구간에 집중되어 있지만, 특이치의 값들은 집중된 구간과의 거리가 먼 것을 알 수 있고, 데이터가 없다는 것을 알 수 있음. 이러한 결과로 특이치를 확인하고, 이를 제거하여 사용하기로 결정함

- 정제가 완료된 mpg Histogram과 boxplot





Engine Size 정제



model	fuelType	engineSize
<chr>	<chr>	<dbl>
A3	Petrol	0

- EngineSize 값 0인 것을 확인해본 결과 연료타입이 Petrol 인 것을 확인했다. 배기량이 0값이 될 수가 없으므로 제거하기로 결정

```
Call:
lm(formula = price ~ ., data = Audi_engine)

Residuals:
    Min      1Q  Median      3Q     Max 
-8093.0 -1463.7   -87.2  1386.9 13262.2 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -2.535e+06  8.512e+04 -29.78 <2e-16 ***
year         1.269e+03  4.214e+01  30.11 <2e-16 ***
mileage      -6.119e-02  3.894e-03 -15.71 <2e-16 ***
mpg          -2.246e+02  5.865e+00 -38.30 <2e-16 ***
engineSize   4.449e+03  1.794e+02  24.79 <2e-16 ***

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

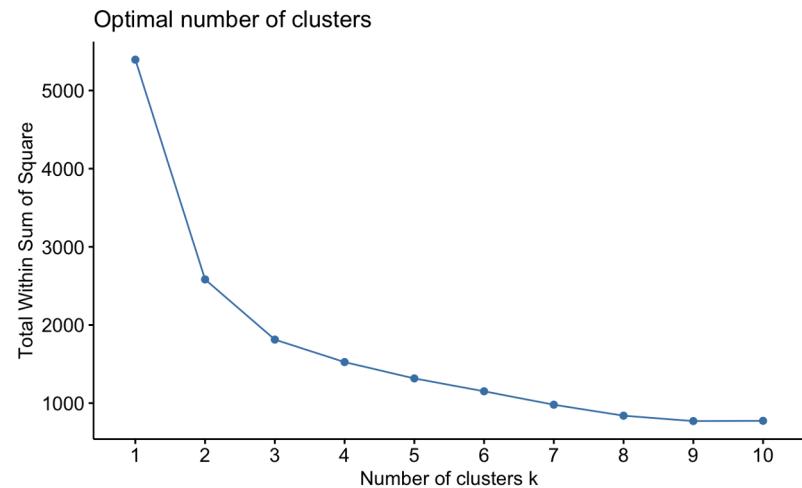
Residual standard error: 2278 on 1864 degrees of freedom
Multiple R-squared:  0.8564,    Adjusted R-squared:  0.8561 
F-statistic: 2779 on 4 and 1864 DF,  p-value: < 2.2e-16
```

- Engine Size 상관계수는 0.2로 관련이 없다고 나오지만, 다중회귀분석 결과 연관성이 '***'로 높다고 나와, 사용하기로 결정



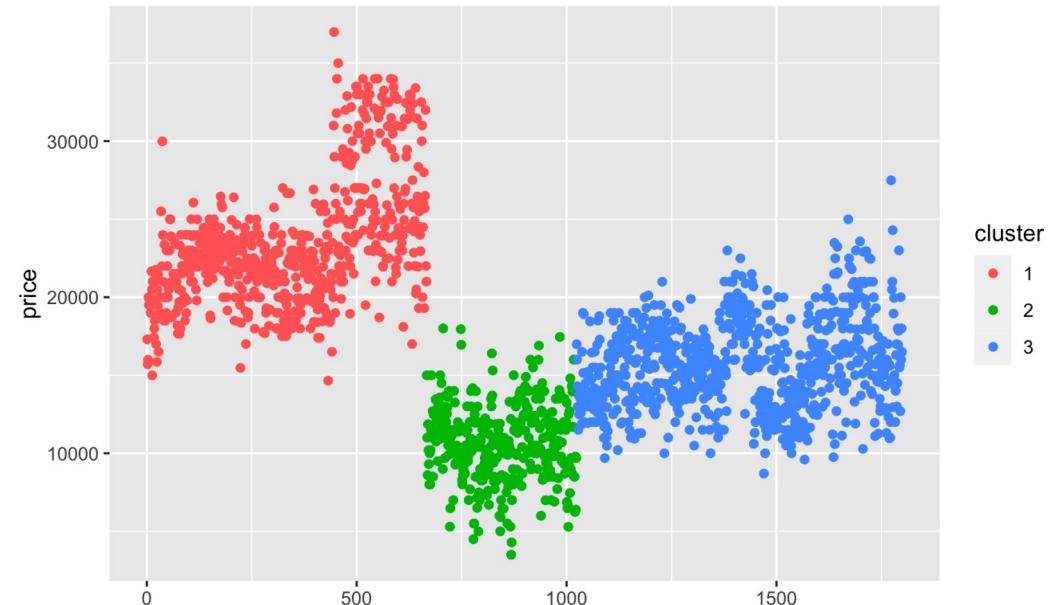
K means & 군집분포

K means



- library(factoextra) 안에 있는 fviz_nbclust 함수를 사용하여 Elbow 값을 확인하여 Kmeans center값으로 사용.

군집분포



- 클러스터를 분포도로 표현하여 군집이 잘 형성되어있는 것을 확인하여 kmeans가 잘 되었는지 확인을 한 후 target column인 price의 범위로 지정하여 사용



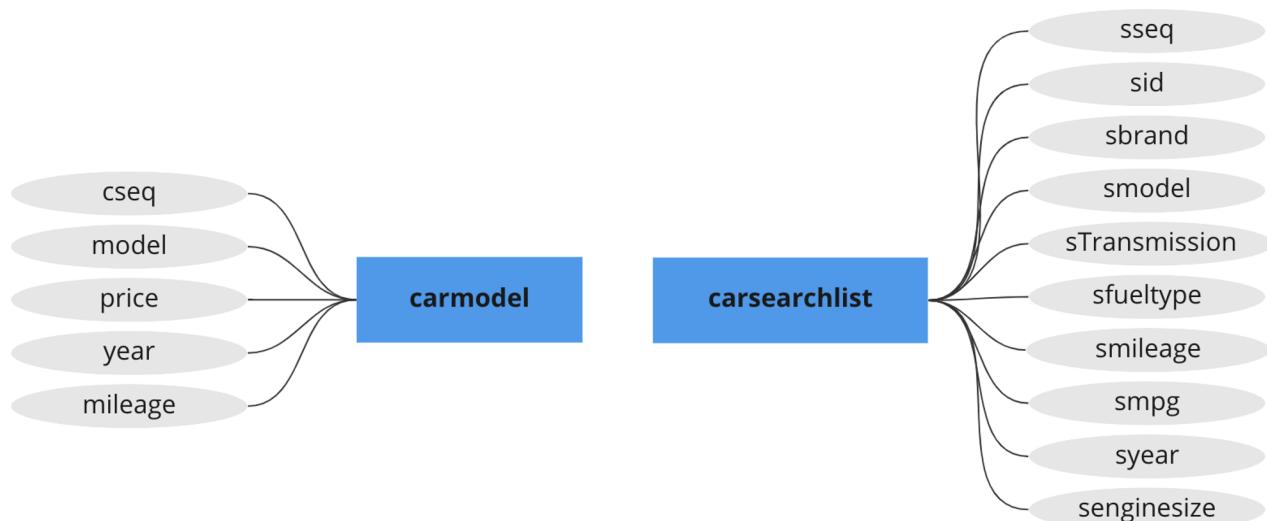
브랜드 모델 별 예측모델 표



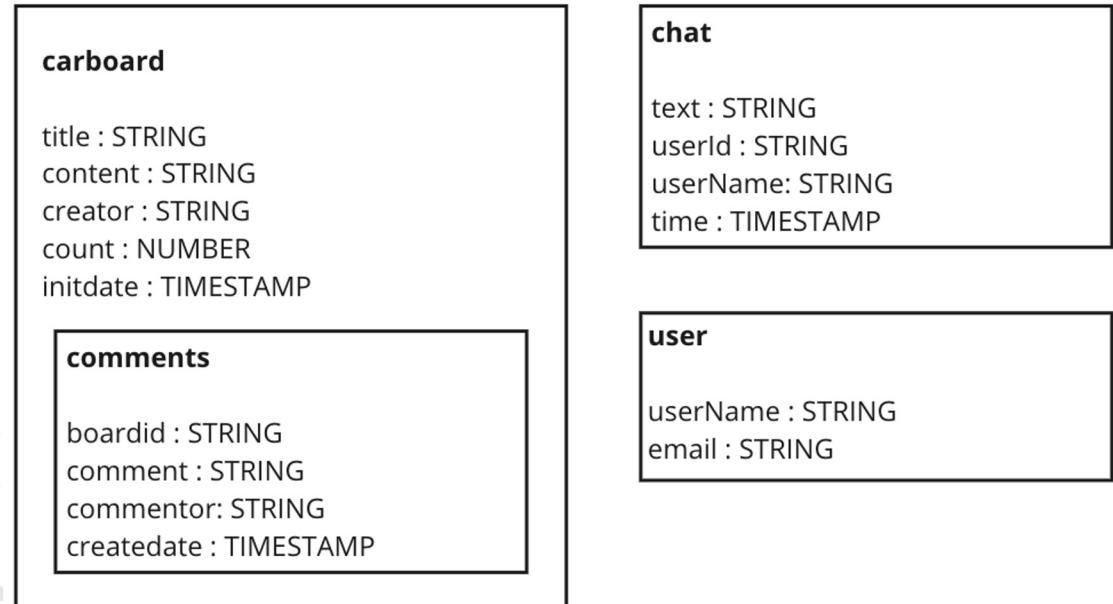
모델이름/예측모델	Random Forest	SVM	인공신경망	의사결정나무
A - Class	0.9684639	0.9777213	0.2875339	0.960529
C - Class	0.9684639	0.9877213	0.4922686	0.960529
E - Class	0.9511737	0.9816887	0.4286385	0.6044601
1 Series	0.9469428	0.9605523	0.2706114	0.9364892
3 Series	0.9142066	0.954428	0.3404059	0.9396679
5 Series	0.9365741	0.95	0.3166667	0.9097222
A3	0.9632432	0.9565766	0.2926126	0.9302703
A4	0.99631	0.9745387	0.5121771	0.9704797
Q3	0.9587121	0.9643939	0.3049242	0.9291667
Golf	0.9755906	0.9750895	0.3717251	0.9605583
Polo	0.9898075	0.9908267	0.3761042	0.9833522
Tiguan	0.9762745	0.9676471	0.3472549	0.9498039
Fiesta	0.9916471	0.9807059	0.3632941	0.9828235
Focus	0.9809015	0.9851031	0.3748663	0.9787624
Kuga	0.9695853	0.9794163	0.2666667	0.9466974



MY SQL ERD



FIRE BASE ERD





Spring : 21본

Project Structure:

- chasajo_Spring [boot] [cha_sa_jo]
- src/main/java
 - com.javalec.base
 - com.javalec.base.controller
 - carController.java
 - rdsController.java
 - com.javalec.base.mapper
 - modelMapper.java
 - com.javalec.base.model
 - carcnt.java
 - carModel.java
 - carsearchlist.java
- src/main/resources
 - webapp
 - application.properties
- rds
 - Audi_A3.rds
 - Audi_A4.rds
 - Audi_Q3.rds
 - Benz_A_Class.rds
 - Benz_C_Class.rds
 - Benz_E_Class.rds
 - BMW_1_Series.rds
 - BMW_3_Series.rds
 - BMW_5_Series.rds
 - Ford_Fiesta.rds
 - Ford_Focus.rds
 - Ford_Kuga.rds
 - VW_Golf.rds
 - VW_Polo.rds
 - VW_Tiguan.rds

총본수 : 75본

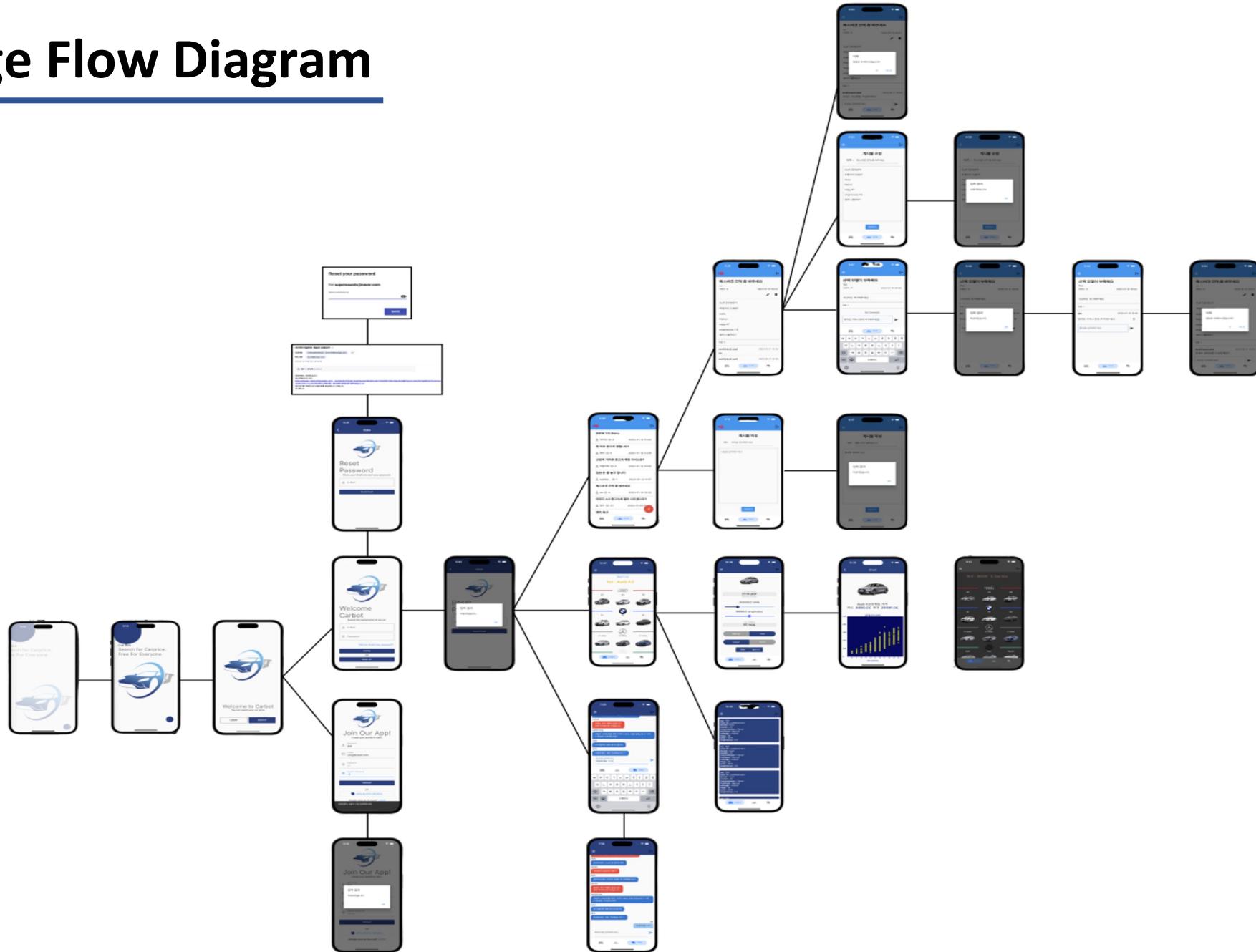
Project Structure:

- lib
 - constants
 - colors.dart
 - hello.dart
 - image_strings.dart
 - sizes.dart
 - text_strings.dart
 - controller
 - splash_screen_controller.dart
 - model
 - boardtype.dart
 - comment.dart
 - fade_in_animation_model.dart
 - hi.dart
 - tabbar.dart
 - view
 - board
 - boardpage.dart
 - detailpage.dart
 - updatepage.dart
 - writepage.dart
 - chart
 - chartscreen.dart
 - linechart.dart
 - chatting
 - chat_screen.dart
 - chatting.dart
 - widget
 - board_detail_comments.dart
 - board_detail.dart
 - board_stream.dart
 - update_widget.dart
 - write_widget.dart
 - carlist
 - Collection.dart
 - list.dart
 - MessageCol.dart
 - showcar.dart
- Home
 - Home.dart
 - splash_screen.dart
 - welcome.dart
- list
 - C.select.dart
 - listinsert.dart
- login
 - login_forget.dart
 - login_screen.dart
 - login.dart
 - main_screen.dart
- signup
 - signup_screen.dart
 - carList.dart
- chatting
 - chat_bubble.dart
 - chat_list.dart
 - message.dart
 - new_message.dart
- home / fade_in_animation
 - animation_design.dart
- login
 - login_footer_widget.dart
 - login_forget_widget.dart
 - login_form_widget.dart
 - login_header_widget.dart
- signup
 - form_header_widget.dart
 - signup_footer_widget.dart
 - signup_form_widget.dart
- firebase_options.dart
- main.dart

Flutter : 54본

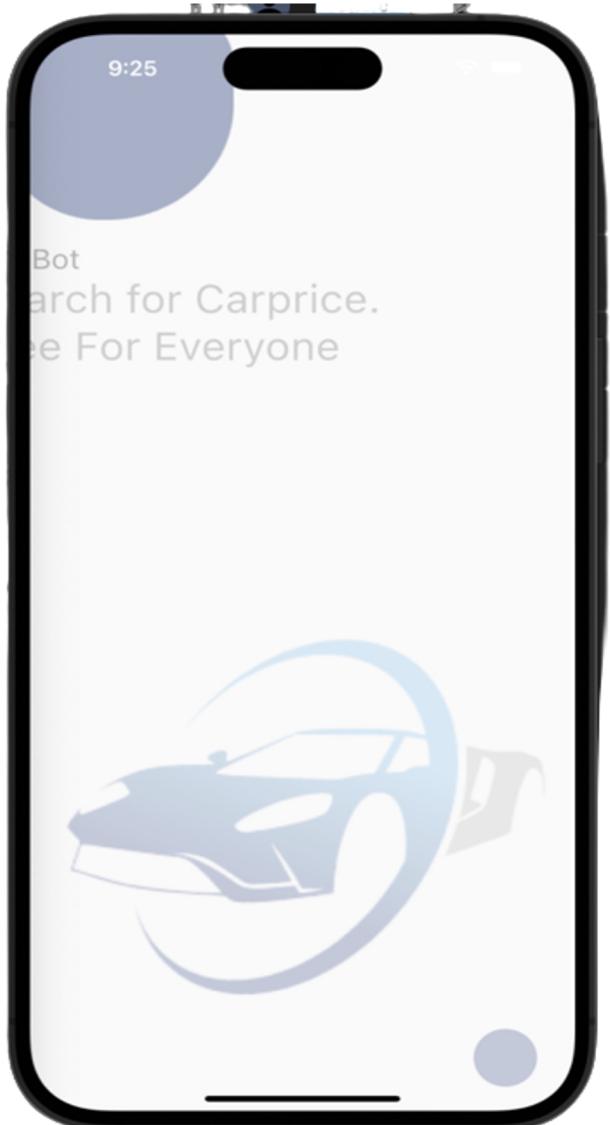


Page Flow Diagram

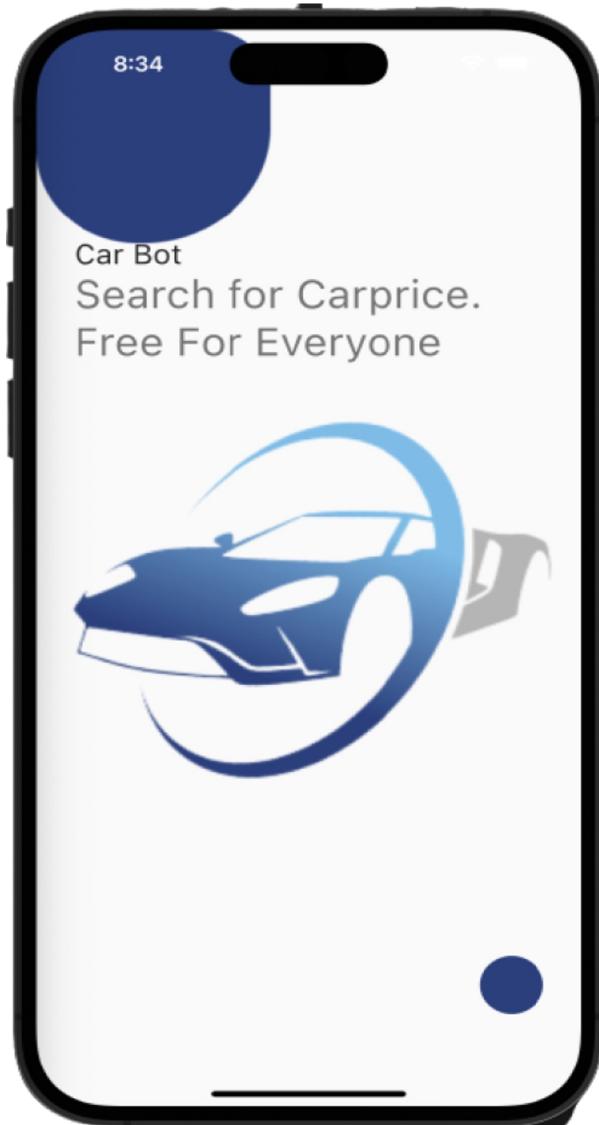




FLUTTER 기능구현_스플래쉬

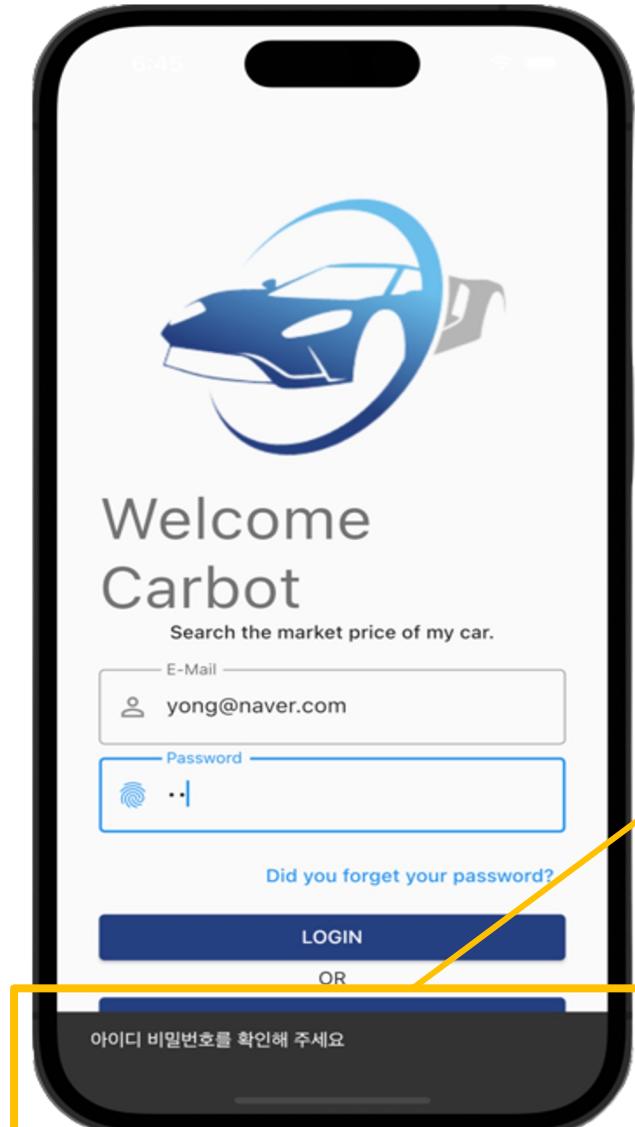
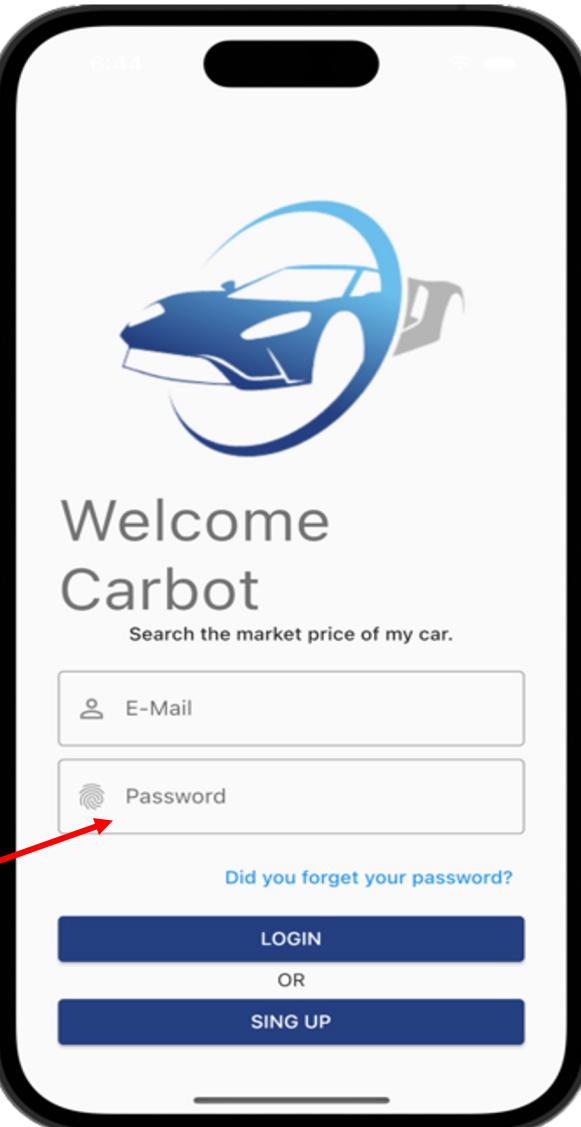


Splash 효과로 화면에 아이콘과 그림이 투명으로 시작하여 위치를 잡아주며 명확해지는 효과이다.





FLUTTER 기능구현_로그인



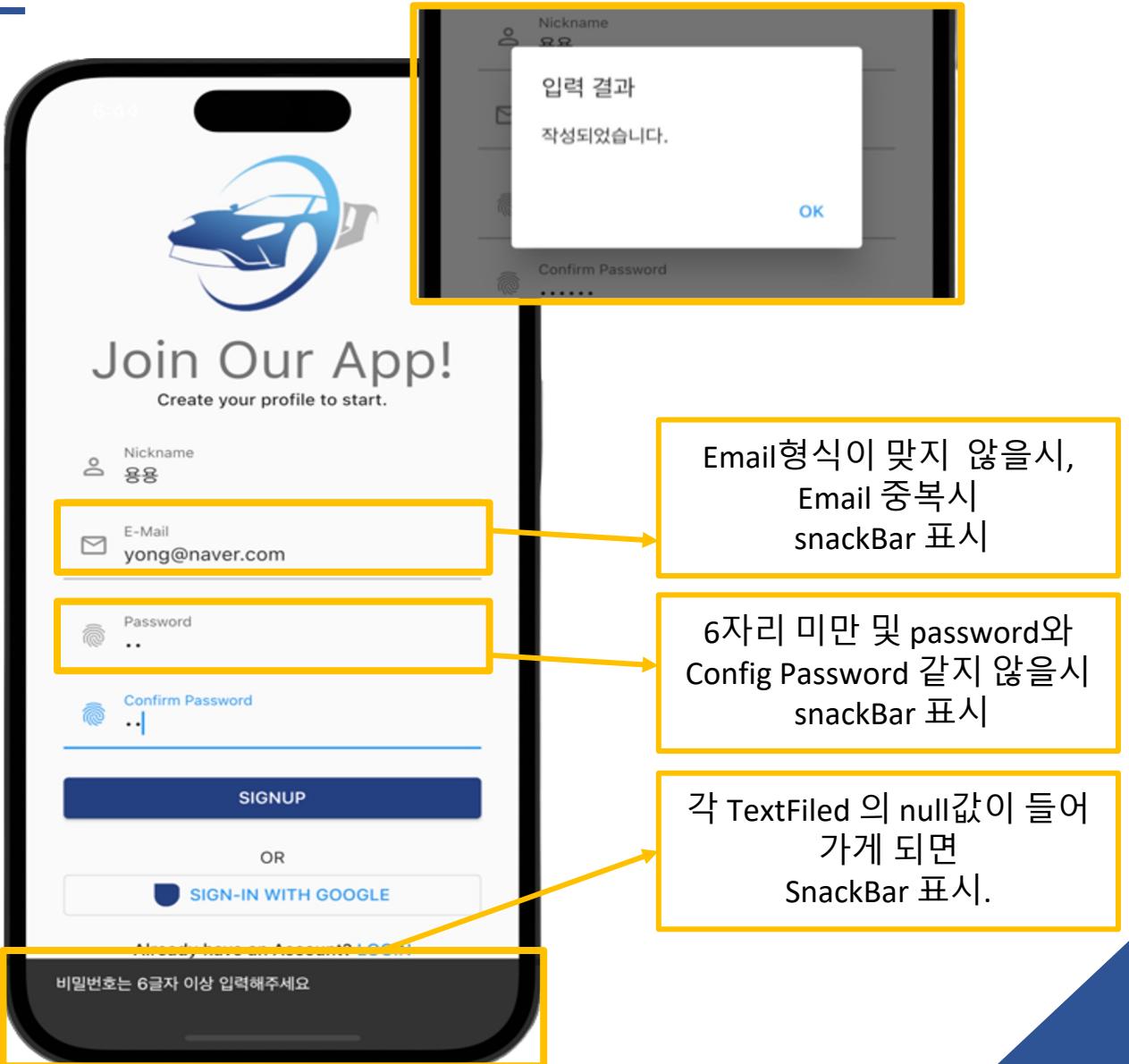
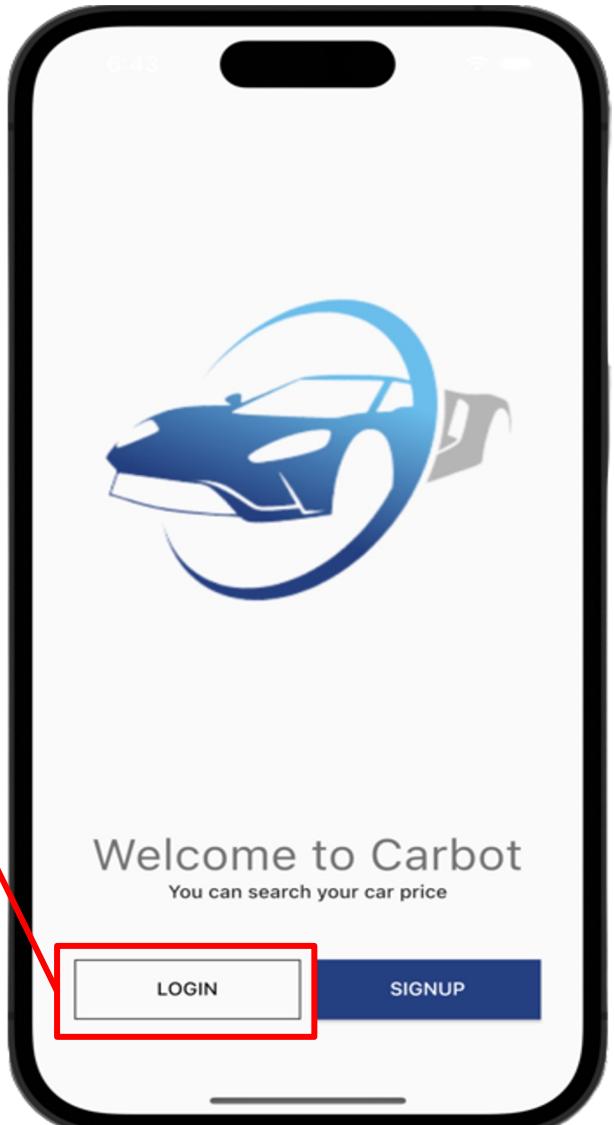
LOGIN 버튼 클릭시
ID(E-Mail), Password 입력창으
로 이동한다.

ID 또는 Password가 일치하지
않으면 Snackbar를 띄워준다.



FLUTTER 기능구현_회원가입

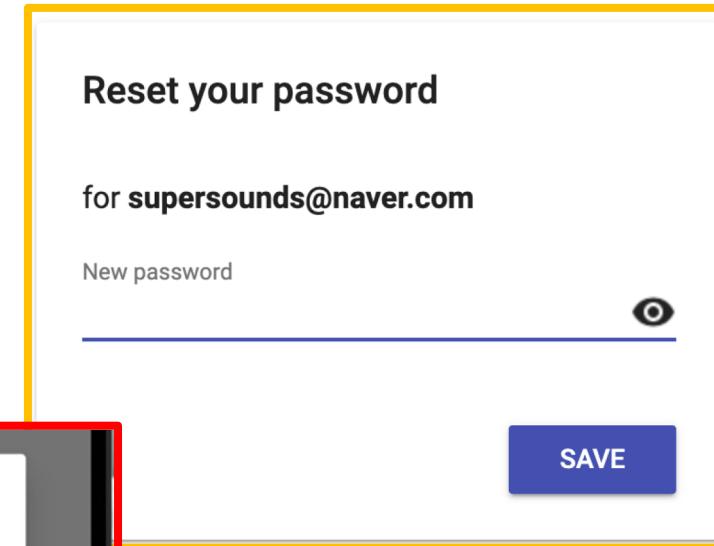
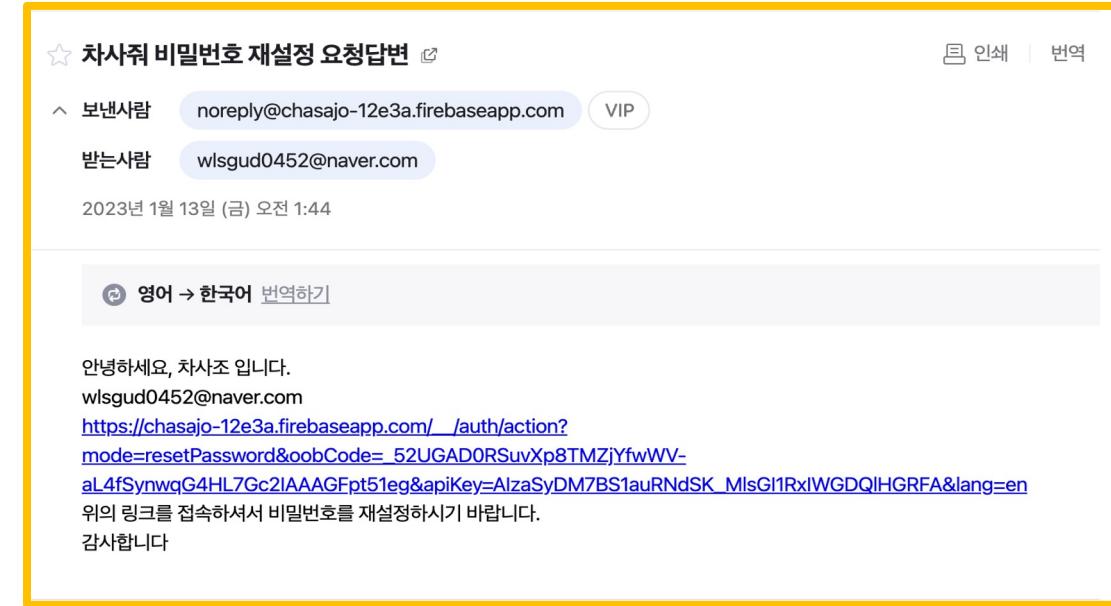
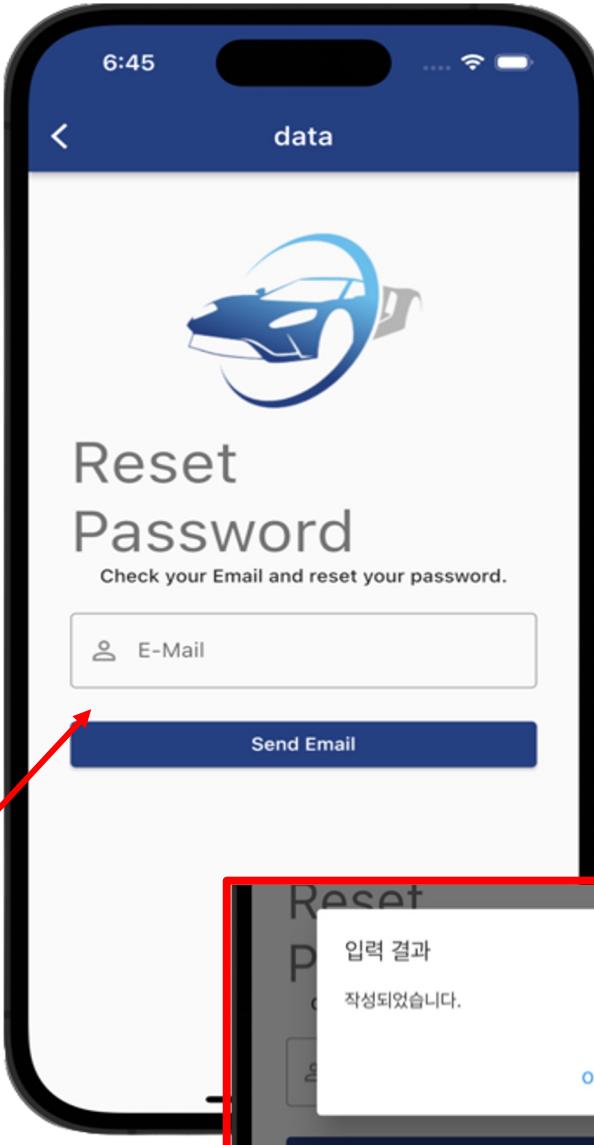
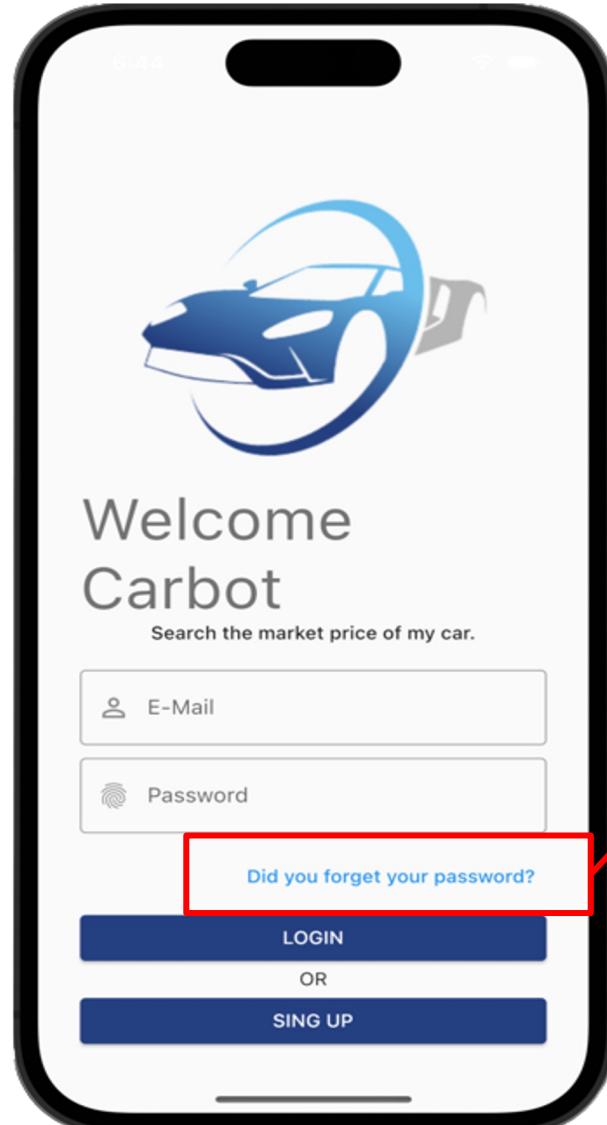
SIGNUP 버튼 클릭시
회원가입 화면으로 이동한다.





FLUTTER 기능구현_비밀번호 찾기

인쇄 | 번역



비밀번호 찾기 버튼을 누르면
Firebase에서 해당 유저
이메일로 비밀번호 변경 링크를 전달한다.



FLUTTER 기능구현_중고차 예측값 조회

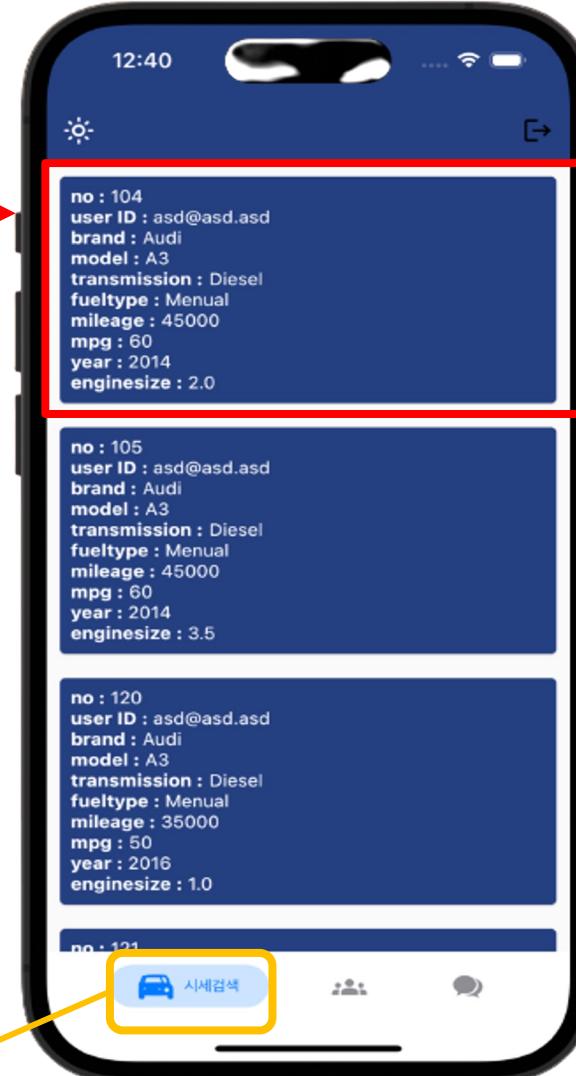


클릭시 Listview
화면으로 이동한
다.

예측 값을 조회한 모
델을 count 하여 가
장 많은 3개의 모델
을 animation text로
표시함.

클릭 시 해당 모델
의 예측치 입력 화
면으로 이동한다.

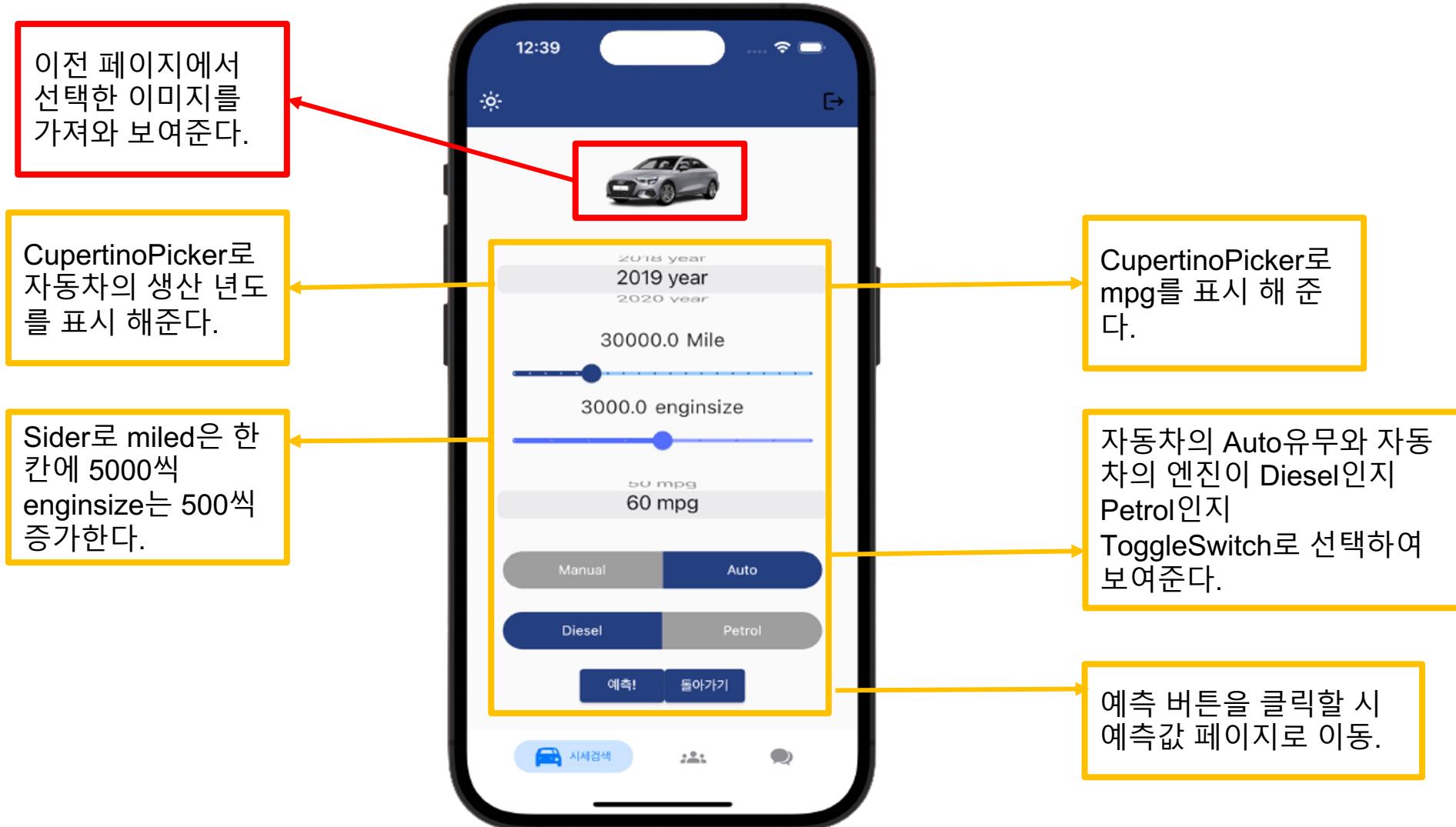
클릭 시 colection
페이지로 이동.



그동안 조회한 목록을
MYSQL에서 select하여
Listview로 보여준다.
Dismissible로 옆으로
밀어서 삭제 할 수 있다.



FLUTTER 기능구현_중고차 예측값 조회





FLUTTER 기능구현_중고차 예측값 조회





FLUTTER 기능구현_게시판

Tab Bar 두번째 버튼을 누르면 게시판으로 이동한다.

게시판에 작성된 게시물들이 최근 날짜순으로 정렬되어 ListView로 보여진다.



게시글 제목, 작성자, 조회수, 작성날짜가 표시되며 작성자 외 다른 사람이 게시글 클릭시 조회수가 올라간다.



“+” Floating Button을 누르면 게시물을 작성할 수 있는 페이지로 이동한다.

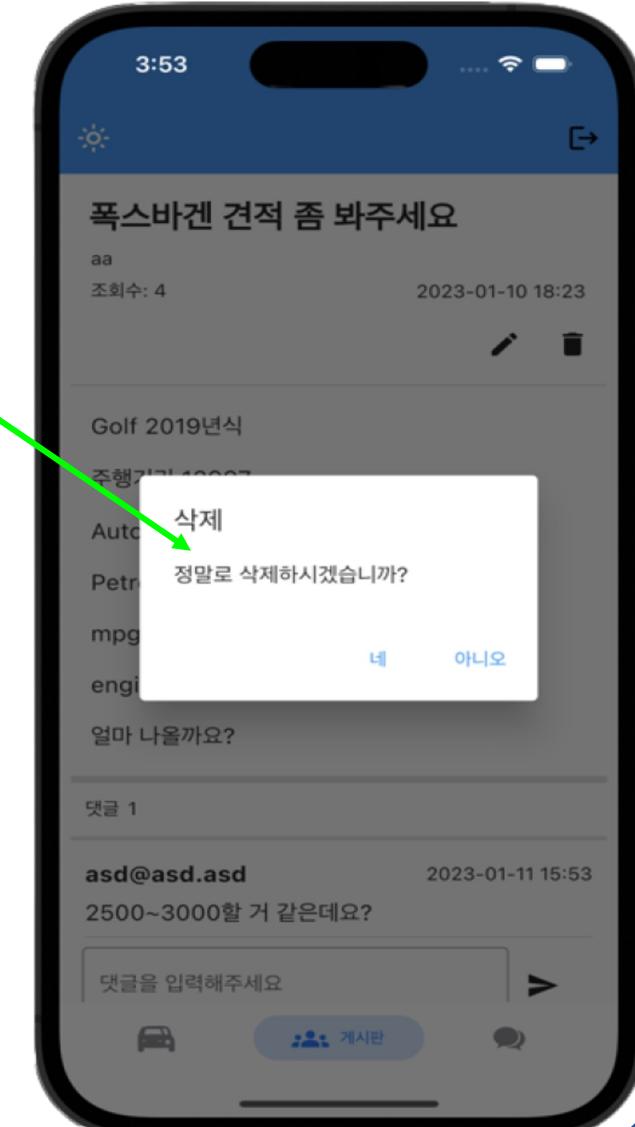
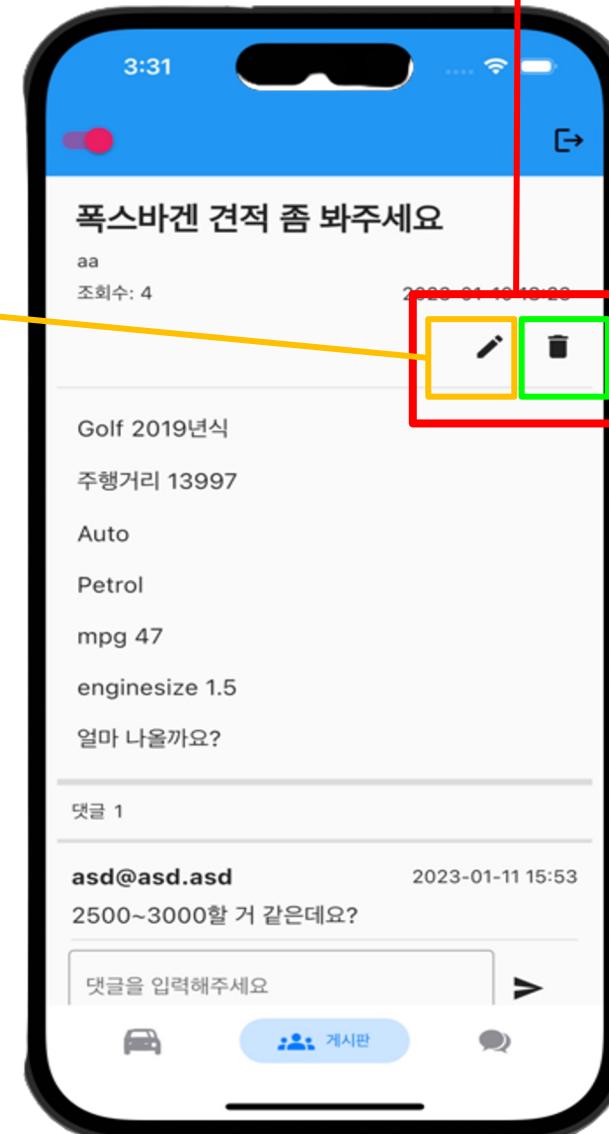
“작성하기” 버튼을 누르면 Firebase DB에 저장되고 ListView가 새롭게 구성되어 출력된다



FLUTTER 기능구현_게시판

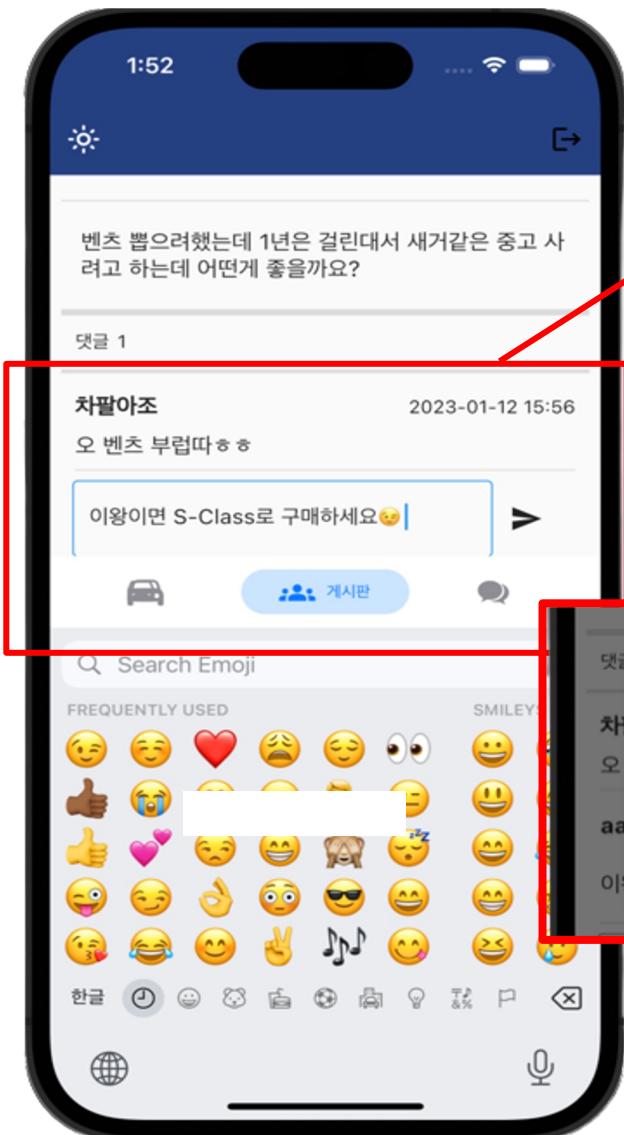
본인이 작성한 게시글이라면
편집 버튼과 삭제버튼이 표시된다.

삭제 버튼 클릭시
삭제 여부 Alert창 띄워준다.



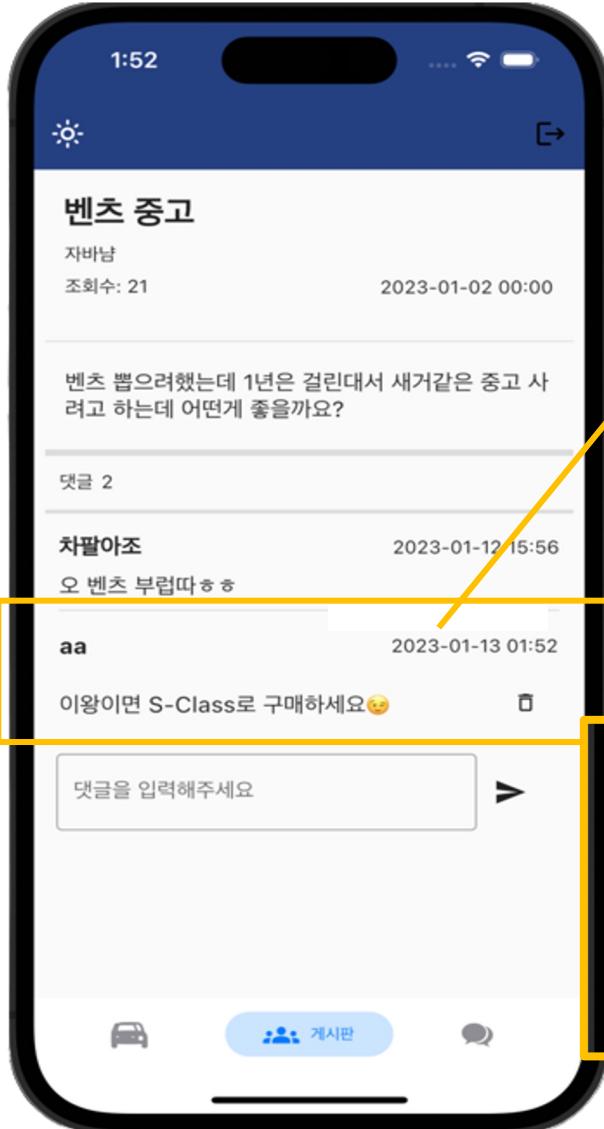
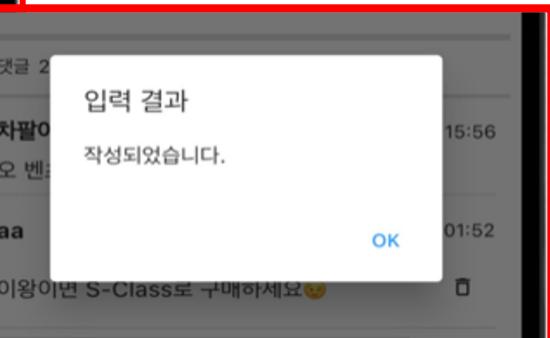


FLUTTER 기능구현_게시판



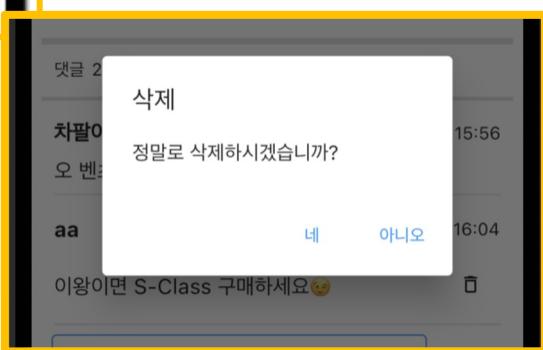
게시물의 상세페이지 하단에는 현재 게시물의 댓글 수와 댓글 내용을 볼 수 있다.

댓글 입력창을 통해 새 댓글을 작성하면 Alert창으로 입력 결과를 알려주고, Firebase DB에 입력된다.



본인이 작성한 댓글이라면 삭제 버튼이 표시된다.

삭제버튼을 누르면 삭제 여부 Alert창을 띄워준다.

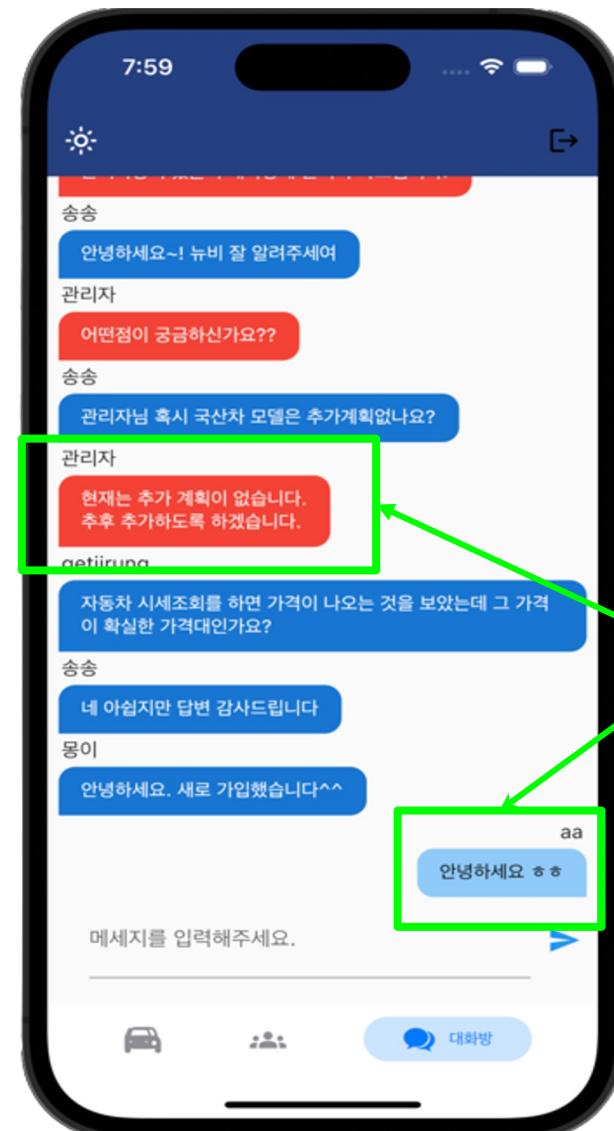
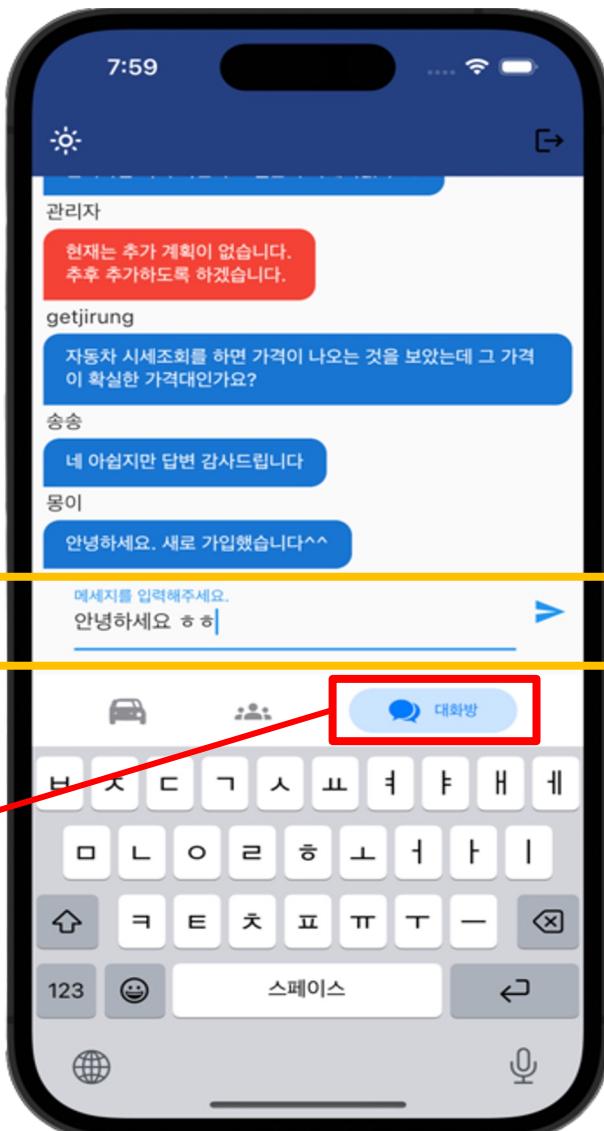




FLUTTER 기능구현_채팅

입력창을 통해 메세지를 작성하면 FirebaseDatabase에 입력되고 Future Builder로 실시간으로 재구성 되어 보여진다.

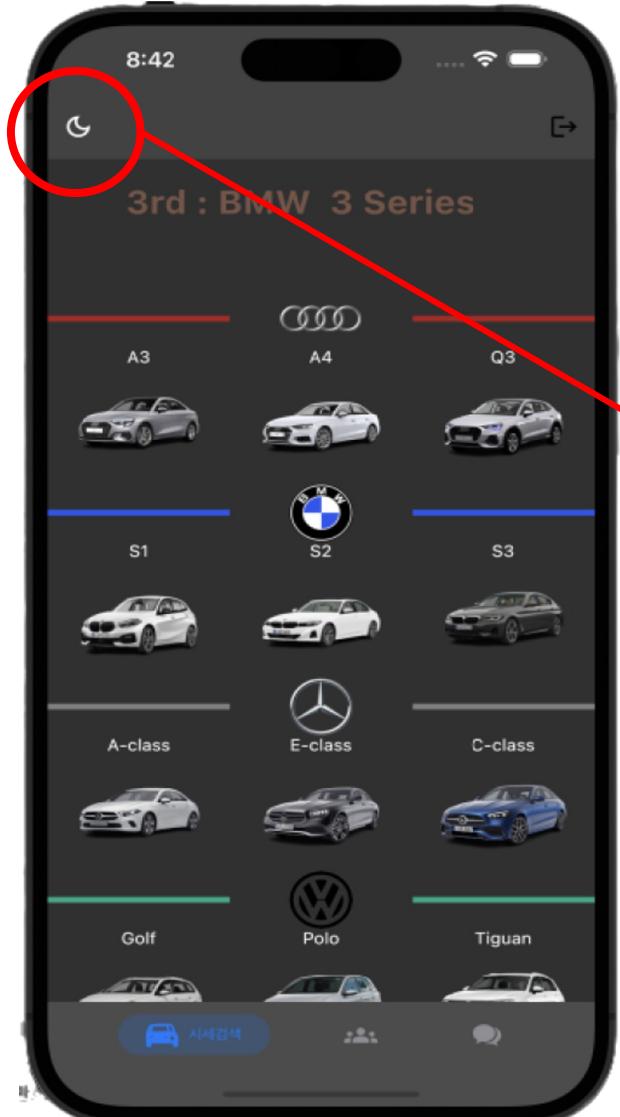
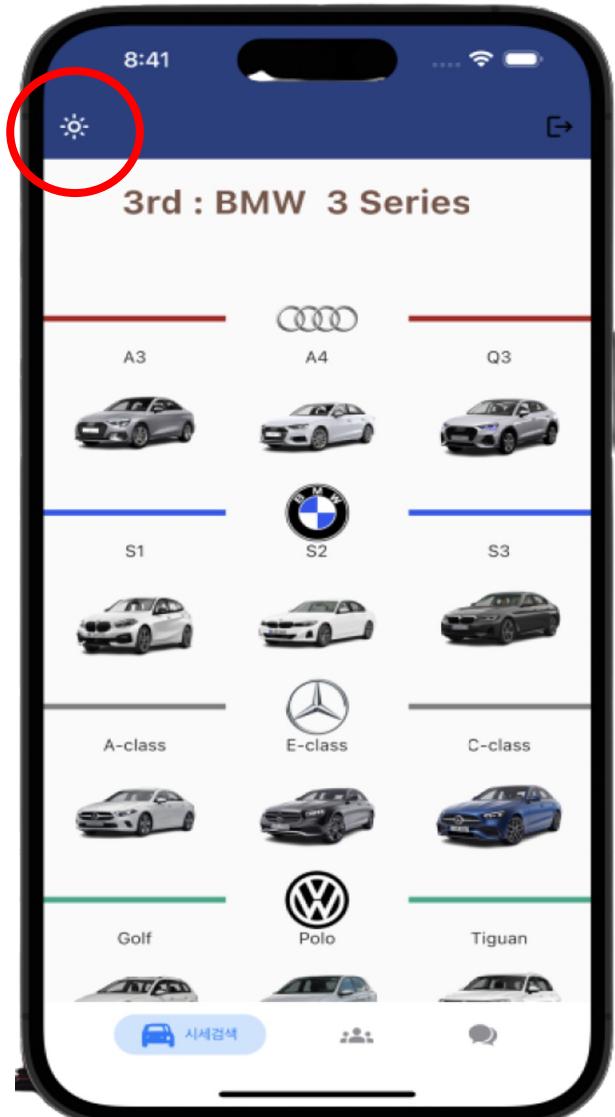
Tab Bar 세번째 버튼을 누르면 채팅 화면으로 이동한다.



관리자는 빨간색,
작성자 본인은 하늘색,
타유저는 파란색으로
구분지어 보여진다.



FLUTTER 기능구현_다크모드



GETX를 이용한 다크모드 구현
Custom을 하기 위해 Theme class를
만들어주어 색상을 변경함.



Spring_RESTful API

```
@RestController
public class carController {

    private modelMapper mapper;

    public carController(modelMapper mapper) {
        this.mapper = mapper;
    }
    // FOR CHART
    @GetMapping("/carmodel/{cseq}")
    public carModel getCarModelInformation(@PathVariable("cseq")String id) {
        return mapper.getCarModelInformation(id);
    }
    // FOR CHART LIST USE THIS
    @GetMapping("/carmodel/list/{model}")
    public List<carModel> getCarAudiModelList(@PathVariable("model")String model){
        return mapper.getCarAudiModelList(model);
    }

    // FOR CHART LIST USE THIS
    @GetMapping("/carmodel/VWlist/{model}")
    public List<carModel2> getCarVWModelList(@PathVariable("model")String model){
        return mapper.getCarVWModelList(model);
    }
}
```

Spring Rest api를 통해 flutter에서 작동하는 Query문을 스프링에서 처리한다.

```
@Mapper
public interface modelMapper {
    // 특정 ..... 값 내용 선택
    @Select("SELECT * FROM audi WHERE cseq =#{cseq}")
    carModel getCarModelInformation(@Param("cseq")String id);

    @Select("SELECT * FROM audi WHERE model=#{model}")
    List<carModel> getCarAudiModelList(@Param("model")String model);

    @Select("SELECT * FROM vw_tiguan WHERE model=#{model}")
    List<carModel2> getCarVWModelList(@Param("model")String model);
```

Controller를 통해 입력받은 값을 Mapper에서 DB에 연결하여 원하는 Query문을 실행한 후,
Flutter를 View 페이지로 사용하여 결과를 보여준다.



Spring_RESTful API

```
@RestController
public class rdsController {

    @RequestMapping(value ="/urlcar", produces = "application/json; charset=utf8", method = RequestMethod.GET)
    public JSONObject Aboutpp(HttpServletRequest request) throws Exception{
        request.setCharacterEncoding("utf-8");
        double year = Double.parseDouble(request.getParameter("year"));
        double mileage = Double.parseDouble(request.getParameter("mileage"));
        double engineSize = Double.parseDouble(request.getParameter("engineSize"));
        double mpg = Double.parseDouble(request.getParameter("mpg"));
        String Manual = request.getParameter("Manual");
        String fuelType_D = request.getParameter("fuelType_D");
        String fuelType_p = request.getParameter("fuelType_p");
        String fileName = request.getParameter("fileName");
        RConnection conn1 = new RConnection();

        conn1 voidEval("library(randomForest)");
        conn1 voidEval("library(e1071)");
        String rdsRoot = System.getProperty("user.dir", "UTF-8") + "/src/main/resources/webapp/rds/";

        conn1 voidEval("rfs <- readRDS(\"" + rdsRoot+ fileName + "\", 'rb')");
        conn1 voidEval("result <- as.character(predict(rfs, (list(year=" + year + ", mileage=" + mileage + ","
        + "engineSize=" + engineSize + ", mpg=" + mpg + ", Manual=" + Manual +
        ", fuelType_D=" + fuelType_D + ", fuelType_P=" + fuelType_p + "))))");
        }

        String result2 = conn1.eval("result").asString();

        Map<String, Object> map = new HashMap<>();
        map.put("result", result2);

        JSONObject jsonObject2 = new JSONObject(map);
        conn1.close();
        return jsonObject2;
    }
}
```

Rserve Api를 이용해 스프링과 R을 연결하여 Flutter에서 값을 입력할 시 연결된 Rds 파일을 통해 예측 값을 리턴해 준다.



향후 계획

- 향후 계획

[R]

- 머신러닝 모델을 늘려 차종별로 예측율을 더욱 다양하게 추출하는 것
- 한국의 중고차 시세들이 나와있는 시장을 조사하여 한국의 데이터로 구현해보는 것
- Kmeans와 다른 방법을 물색하여 더 나은 방법을 찾아보는 것.

[FLUTTER]

- 게시판 리스트에 댓글 갯수 표시 및 댓글 수정기능 추가
- 관리자 로그인 시 채팅방 리스트 추가
- 차트 종류를 선택하는 기능 추가
- 아이디 및 비밀번호 저장기능 추가



부록

kaggle

- 참조 Data Site Kaggle (<https://www.kaggle.com/datasets/adityadesai13/used-car-dataset-ford-and-mercedes?select=merc.csv>)
- 참고자료
 1. 오피니언 뉴스(<http://www.opinionnews.co.kr/news/articleView.html?idxno=45670>)
 2. Techworld news(<https://www.epnc.co.kr/news/articleView.html?idxno=221825>)
 3. IT조선(https://it.chosun.com/site/data/html_dir/2022/02/27/2022022700695.html)
 4. Stoneacre([https://www.stoneacre.co.uk/search?makes\[\]bmw&ranges\[\]bmw-5-series&price_type=cash](https://www.stoneacre.co.uk/search?makes[]bmw&ranges[]bmw-5-series&price_type=cash))



감사합니다