ПРИЛОЖЕНИЕ Б. ТЕКСТ ПРОГРАММЫ

АННОТАЦИЯ

В данном программном документе приведен текст программы для поддержания работы мобильного приложения «Drivee».

В данном программном документе, в разделе «Тест программы», указана информация о информационной системе, области применения программы, модулей программы и код каждого программного модуля.

1. МОДУЛИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

В Таблице 1 представлено описание всех классов и модулей программы, количество строк верстки или кода и размер файла.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Название | Описание | Количество строк | Размер файла в Кб |
| 1 | 2 | 3 | 4 | 5 |
| 1 | AndroidManifest.xml | конфигурация приложения: разрешения, иконки, список активити | 54 | 2 Кб |
| 2 | AdminActivity.java | панель администратора, просмотр необработанных объявлений | 93 | 5 Кб |
| 3 | AdminBortjournalActivity.java | панель администратора для управления бортжурналами | 86 | 4 Кб |
| 4 | AdminBortjournalAdapter.java | адаптер для отображения бортжурналов в админке | 88 | 5 Кб |
| 5 | AdminCarAdapter.java | адаптер для списка машин в админке. | 110 | 6 Кб |
| 6 | AllActivity.java | главный экран со всеми объявлениями + фильтры и навигация. | 391 | 9 Кб |
| 7 | BortjournalActivity.java | экран бортжурнала пользователя | 62 | 2 Кб |
| 8 | BortjournalAdapter.java | адаптер для списка записей в бортжурнале | 123 | 3 Кб |
| 9 | BortjournalDetailActivity.java | экран подробностей по записи в бортжурнале | 30 | 2 Кб |
| 10 | BortjournalPost.java | модель (POJO) для записи в бортжурнал | 80 | 4 Кб |
| 11 | Car.java | модель машины | 47 | 2 Кб |
| 12 | CarAdapter.java | адаптер для отображения карточек машин | 198 | 3 Кб |
| 13 | CarAdapterUser.java | адаптер для отображения машин текущего пользователя | 86 | 3 Кб |
| 14 | CarForProfile.java | отдельная модель машины для профиля | 33 | 1 Кб |
| 15 | CenterActivity.java | экран карточки машины | 182 | 4 Кб |
| 16 | CreateListingActivity.java | создание нового объявления | 305 | 8 Кб |
| 17 | CreatePostActivity.java | создание записи в бортжурнал | 134 | 3 Кб |
| 18 | FavoritesActivity.java | избранные объявления пользователя | 106 | 2 Кб |
| № | Название | Описание | Количество строк | Размер файла в Кб |
| 1 | 2 | 3 | 4 | 5 |
| 19 | FilterActivity.java | расширенные фильтры поиска | 106 | 2 Кб |
| 20 | ImagePagerAdapter.java | адаптер для ViewPager с фото машины | 68 | 2 Кб |
| 21 | MainActivity.java | экран авторизации | 94 | 2 Кб |
| 22 | ManageModelsActivity.java | админка: управление марками и моделями | 97 | 2 Кб |
| 23 | RegActivity.java | экран регистрации | 114 | 3 Кб |
| 24 | ResetPasswordActivity.java | сброс пароля | 47 | 1 Кб |
| 25 | SelectBrandActivity.java | выбор марки авто. | 80 | 2 Кб |
| 26 | SelectModelActivity.java | выбор модели авто. | 95 | 2 Кб |
| 27 | SplashActivity.java | splash-экран при запуске | 22 | 1 Кб |
| 28 | User.java | модель пользователя | 30 | 1 Кб |
| 29 | UserCar.java | модель машины, привязанной к пользователю | 48 | 1 Кб |
| 30 | UserProfileActivity.java | профиль пользователя | 197 | 3 Кб |
| 31 | circle.xml | закруглённый стиль | 5 | 1 Кб |
| 32 | edittext\_background.xml | фон для полей ввода | 3 | 1 Кб |
| 33 | ic\_heart\_filled.xml | иконка «лайка» (избранное) | 9 | 1 Кб |
| 34 | ic\_heart\_outline.xml | иконка «лайка» — пустая | 9 | 1 Кб |
| 35 | ic\_launcher\_background.xml | стандартные ресурсы иконки | 75 | 2 Кб |
| 36 | ic\_launcher\_foreground.xml | стандартные ресурсы иконки | 30 | 1 Кб |
| 37 | spinner\_background.xml | фон для выпадающего списка (Spinner). | 6 | 1 Кб |
| 38 | splash\_background.xml | фон для splash-экрана. | 4 | 1 Кб |
| 39 | activity\_admin.xml | UI экрана AdminActivity | 45 | 1 Кб |
| 40 | activity\_admin\_bortjournal.xml | UI для AdminBortjournalActivity | 26 | 1 Кб |
| 41 | activity\_all.xml | UI для экрана со всеми объявлениями | 107 | 3 Кб |
| 42 | activity\_center.xml | UI карточки машины | 160 | 3 Кб |
| 43 | activity\_create\_listing.xml | UI создания нового объявления | 292 | 4 Кб |
| 44 | activity\_favorites.xml | UI избранного | 49 | 1 Кб |
| 45 | activity\_filter.xml | UI экрана фильтрации | 216 | 4 Кб |
| 46 | activity\_main.xml | UI авторизации | 69 | 1 Кб |
| 47 | activity\_manage\_models.xml | UI управления марками/моделями. | 68 | 1 Кб |
| 48 | activity\_reg.xml | UI регистрации | 91 | 2 Кб |
| № | Название | Описание | Количество строк | Размер файла в Кб |
| 1 | 2 | 3 | 4 | 5 |
| 49 | activity\_reset\_password.xml | UI сброса пароля | 41 | 1 Кб |
| 50 | activity\_select\_brand.xml | UI выбора марки | 33 | 1 Кб |
| 51 | activity\_select\_model.xml | UI выбора модели | 33 | 1 Кб |
| 52 | activity\_splash.xml | UI заставки | 14 | 1 Кб |
| 53 | activity\_user\_profile.xml | UI профиля пользователя | 113 | 2 Кб |
| 54 | bortjournal\_activity.xml | UI экрана с записями бортжурнала | 28 | 1 Кб |
| 55 | bortjournal\_detail\_activity.xml | UI экрана с деталями записи | 38 | 1 Кб |
| 56 | create\_post\_activity.xml | UI создания записи в бортжурнал | 60 | 2 Кб |
| 57 | item\_admin\_bortjournal.xml | элемент списка записей в админке | 61 | 2 Кб |
| 58 | item\_bortjournal.xml | элемент записи в бортжурнале | 70 | 2 Кб |
| 59 | item\_car.xml | карточка машины. | 86 | 2 Кб |
| 60 | item\_car\_admin.xml | карточка машины для админки | 63 | 2 Кб |
| 61 | item\_car\_image.xml | элемент изображения машины | 5 | 1 Кб |
| 62 | item\_car\_user.xml | карточка машины в профиле пользователя | 86 | 2 Кб |
| 63 | item\_image.xml | отдельный элемент изображения | 6 | 1 Кб |
| 64 | bottom\_nav\_menu.xml | нижняя навигация | 11 | 1 Кб |
| 65 | bottom\_navigation\_menu.xml | нижняя навигация | 19 | 1 Кб |
| 66 | colors.xml | определение цветов в приложении | 6 | 1 Кб |
| 67 | strings.xml | строковые ресурсы | 3 | 1 Кб |
| 68 | styles.xml | стили | 8 | 1 Кб |
| 69 | values.xml | общее хранилище ресурсов | 9 | 1 Кб |
| 70 | build.gradle | конфигурация сборки приложения | 50 | 1 Кб |
| 71 | libs.versions.toml | зависимости, версии библиотек | 29 | 1 Кб |

1. ИСХОДНЫЙ КОД ПРОГРАММЫ
2. AndroidManifest.xml

# <?xml version="1.0" encoding="utf-8"?>

# <manifest xmlns:android="http://schemas.android.com/apk/res/android"

# xmlns:tools="http://schemas.android.com/tools">

# <uses-feature

# android:name="android.hardware.telephony"

# android:required="false" />

# <uses-permission android:name="android.permission.CALL\_PHONE" />

# <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />

# <uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" />

# <application

# android:allowBackup="true"

# android:dataExtractionRules="@xml/data\_extraction\_rules"

# android:fullBackupContent="@xml/backup\_rules"

# android:icon="@mipmap/ic\_launcher"

# android:roundIcon="@mipmap/ic\_launcher\_round"

# android:label="@string/app\_name"

# android:supportsRtl="true"

# android:theme="@style/Theme.Drivee"

# tools:targetApi="31">

# <activity

# android:name=".MainActivity"

# android:exported="true">

# <intent-filter>

# <action android:name="android.intent.action.MAIN"/>

# <category android:name="android.intent.category.LAUNCHER"/>

# </intent-filter>

# </activity>

# <activity android:name=".RegActivity" />

# <activity android:name=".SplashActivity" />

# <activity android:name=".FavoritesActivity" />

# <activity android:name=".SelectModelActivity" />

# <activity android:name=".ManageModelsActivity" />

# <activity android:name=".AdminBortjournalActivity" />

# <activity android:name=".FilterActivity" />

# <activity android:name=".CreateListingActivity" />

# <activity android:name=".ResetPasswordActivity" />

# <activity android:name=".CenterActivity" />

# <activity android:name=".SelectBrandActivity" />

# <activity android:name=".AllActivity" />

# <activity android:name=".BortjournalDetailActivity" />

# <activity android:name=".BortjournalActivity" />

# <activity android:name=".CreatePostActivity" />

# <activity android:name=".UserProfileActivity" />

# <activity android:name=".AdminActivity" />

# </application>

# </manifest>

1. AdminActivity.java

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.util.Log;

# import androidx.appcompat.app.AppCompatActivity;

# import androidx.recyclerview.widget.LinearLayoutManager;

# import androidx.recyclerview.widget.RecyclerView;

# import com.google.android.material.bottomnavigation.BottomNavigationView;

# import com.google.firebase.firestore.FirebaseFirestore;

# import com.google.firebase.firestore.QueryDocumentSnapshot;

# import java.util.ArrayList;

# import java.util.List;

# public class AdminActivity extends AppCompatActivity {

# private RecyclerView recyclerView;

# private AdminCarAdapter adminCarAdapter;

# private List<Car> carList;

# private BottomNavigationView bottomNavigationView;

# private FirebaseFirestore db;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_admin);

# recyclerView = findViewById(R.id.recyclerView);

# recyclerView.setLayoutManager(new LinearLayoutManager(this));

# carList = new ArrayList<>();

# adminCarAdapter = new AdminCarAdapter(this, carList);

# recyclerView.setAdapter(adminCarAdapter);

# db = FirebaseFirestore.getInstance();

# bottomNavigationView = findViewById(R.id.bottomNavigationView);

# bottomNavigationView.setOnItemSelectedListener(item -> {

# if (item.getItemId() == R.id.nav\_manage\_models) {

# Intent intent = new Intent(AdminActivity.this, ManageModelsActivity.class);

# startActivity(intent);

# return true;

# }

# else if (item.getItemId() == R.id.nav\_manage\_bort) {

# Intent intent = new Intent(AdminActivity.this, AdminBortjournalActivity.class);

# startActivity(intent);

# return true;

# }

# return false;

# });

# loadPendingCars();

# }

# private void loadPendingCars() {

# db.collection("cars")

# .whereEqualTo("status", "pending")

# .get()

# .addOnCompleteListener(task -> {

# if (task.isSuccessful()) {

# carList.clear();

# for (QueryDocumentSnapshot document : task.getResult()) {

# String photoone = document.getString("photoone");

# if (photoone == null) photoone = "";

# long mileage = document.getLong("mileage") != null ? document.getLong("mileage") : 0;

# Car car = new Car(

# document.getId(),

# document.getString("brand"),

# document.getString("model"),

# photoone,

# mileage,

# document.getLong("price")

# );

# carList.add(car);

# }

# adminCarAdapter.notifyDataSetChanged();

# } else {

# Log.e("AdminActivity", "Ошибка загрузки машин", task.getException());

# }

# });

# }

# }

1. AdminBortjournalActivity.java

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.view.View;

# import android.widget.Toast;

# import androidx.appcompat.app.AppCompatActivity;

# import androidx.recyclerview.widget.LinearLayoutManager;

# import androidx.recyclerview.widget.RecyclerView;

# import com.google.firebase.firestore.FirebaseFirestore;

# import com.google.firebase.firestore.QueryDocumentSnapshot;

# import java.util.ArrayList;

# import java.util.List;

# public class AdminBortjournalActivity extends AppCompatActivity {

# private RecyclerView recyclerView;

# private FirebaseFirestore db;

# private List<BortjournalPost> postList = new ArrayList<>();

# private AdminBortjournalAdapter adapter;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_admin\_bortjournal);

# recyclerView = findViewById(R.id.adminRecyclerView);

# recyclerView.setLayoutManager(new LinearLayoutManager(this));

# db = FirebaseFirestore.getInstance();

# adapter = new AdminBortjournalAdapter(postList, this::openDetail, this::approvePost, this::deletePost);

# recyclerView.setAdapter(adapter);

# loadPendingPosts();

# }

# private void loadPendingPosts() {

# db.collection("bortjornal")

# .whereEqualTo("status", "pending")

# .get()

# .addOnSuccessListener(querySnapshot -> {

# postList.clear();

# for (QueryDocumentSnapshot doc : querySnapshot) {

# BortjournalPost post = doc.toObject(BortjournalPost.class);

# post.setId(doc.getId());

# postList.add(post);

# }

# adapter.notifyDataSetChanged();

# })

# .addOnFailureListener(e -> Toast.makeText(this, "Ошибка загрузки", Toast.LENGTH\_SHORT).show());

# }

# private void openDetail(BortjournalPost post) {

# Intent intent = new Intent(this, BortjournalDetailActivity.class);

# intent.putExtra("brand", post.getBrand());

# intent.putExtra("model", post.getModel());

# intent.putExtra("title", post.getTitle());

# intent.putExtra("content", post.getContent());

# startActivity(intent);

# }

# private void approvePost(BortjournalPost post) {

# db.collection("bortjornal").document(post.getId())

# .update("status", "approved")

# .addOnSuccessListener(unused -> {

# Toast.makeText(this, "Подтверждено", Toast.LENGTH\_SHORT).show();

# postList.remove(post);

# adapter.notifyDataSetChanged();

# });

# }

# private void deletePost(BortjournalPost post) {

# db.collection("bortjornal").document(post.getId())

# .delete()

# .addOnSuccessListener(unused -> {

# Toast.makeText(this, "Удалено", Toast.LENGTH\_SHORT).show();

# postList.remove(post);

# adapter.notifyDataSetChanged();

# });

# }

# }

1. AdminBortjournalAdapter.java

# package com.example.drivee;

# import android.view.LayoutInflater;

# import android.view.View;

# import android.view.ViewGroup;

# import android.widget.Button;

# import android.widget.TextView;

# import androidx.annotation.NonNull;

# import androidx.recyclerview.widget.RecyclerView;

# import java.util.List;

# public class AdminBortjournalAdapter extends RecyclerView.Adapter<AdminBortjournalAdapter.ViewHolder> {

# public interface OnPostOpenListener {

# void onOpen(BortjournalPost post);

# }

# public interface OnPostApproveListener {

# void onApprove(BortjournalPost post);

# }

# public interface OnPostDeleteListener {

# void onDelete(BortjournalPost post);

# }

# private List<BortjournalPost> posts;

# private OnPostOpenListener openListener;

# private OnPostApproveListener approveListener;

# private OnPostDeleteListener deleteListener;

# public AdminBortjournalAdapter(List<BortjournalPost> posts,

# OnPostOpenListener openListener,

# OnPostApproveListener approveListener,

# OnPostDeleteListener deleteListener) {

# this.posts = posts;

# this.openListener = openListener;

# this.approveListener = approveListener;

# this.deleteListener = deleteListener;

# }

# @NonNull

# @Override

# public AdminBortjournalAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

# View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.item\_admin\_bortjournal, parent, false);

# return new ViewHolder(view);

# }

# @Override

# public void onBindViewHolder(@NonNull AdminBortjournalAdapter.ViewHolder holder, int position) {

# BortjournalPost post = posts.get(position);

# holder.brandModelTextView.setText(post.getBrand() + " " + post.getModel());

# holder.titleTextView.setText(post.getTitle());

# holder.itemView.setOnClickListener(v -> {

# if (openListener != null) openListener.onOpen(post);

# });

# holder.approveButton.setOnClickListener(v -> {

# if (approveListener != null) approveListener.onApprove(post);

# });

# holder.deleteButton.setOnClickListener(v -> {

# if (deleteListener != null) deleteListener.onDelete(post);

# });

# }

# @Override

# public int getItemCount() {

# return posts.size();

# }

# public static class ViewHolder extends RecyclerView.ViewHolder {

# TextView titleTextView, brandModelTextView;

# Button approveButton, deleteButton;

# public ViewHolder(@NonNull View itemView) {

# super(itemView);

# brandModelTextView = itemView.findViewById(R.id.adminBrandModel);

# titleTextView = itemView.findViewById(R.id.adminTitle);

# approveButton = itemView.findViewById(R.id.approveButton);

# deleteButton = itemView.findViewById(R.id.deleteButton);

# }

# }

# }

1. AdminCarAdapter.java

# package com.example.drivee;

# import android.content.Context;

# import android.content.Intent;

# import android.view.LayoutInflater;

# import android.view.View;

# import android.view.ViewGroup;

# import android.widget.Button;

# import android.widget.TextView;

# import android.widget.Toast;

# import androidx.annotation.NonNull;

# import androidx.recyclerview.widget.RecyclerView;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.util.List;

# public class AdminCarAdapter extends RecyclerView.Adapter<AdminCarAdapter.CarViewHolder> {

# private Context context;

# private List<Car> carList;

# public AdminCarAdapter(Context context, List<Car> carList) {

# this.context = context;

# this.carList = carList;

# }

# @NonNull

# @Override

# public CarViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

# View itemView = LayoutInflater.from(context).inflate(R.layout.item\_car\_admin, parent, false);

# return new CarViewHolder(itemView);

# }

# @Override

# public void onBindViewHolder(@NonNull CarViewHolder holder, int position) {

# Car car = carList.get(position);

# holder.tvBrand.setText(car.getBrand());

# holder.tvModel.setText(car.getModel());

# holder.tvPrice.setText(String.format("%d руб.", car.getPrice()));

# holder.btnAccept.setOnClickListener(v -> {

# holder.btnAccept.setEnabled(false); // анти-спам

# updateCarStatus(car.getId(), "approved", position);

# });

# holder.btnDelete.setOnClickListener(v -> {

# holder.btnDelete.setEnabled(false);

# deleteCar(car.getId(), position);

# });

# holder.itemView.setOnClickListener(v -> {

# Intent intent = new Intent(context, CenterActivity.class);

# intent.putExtra("carId", car.getId());

# context.startActivity(intent);

# });

# }

# @Override

# public int getItemCount() {

# return carList.size();

# }

# public class CarViewHolder extends RecyclerView.ViewHolder {

# TextView tvBrand, tvModel, tvPrice;

# Button btnAccept, btnDelete;

# public CarViewHolder(@NonNull View itemView) {

# super(itemView);

# tvBrand = itemView.findViewById(R.id.tvBrand);

# tvModel = itemView.findViewById(R.id.tvModel);

# tvPrice = itemView.findViewById(R.id.tvPrice);

# btnAccept = itemView.findViewById(R.id.btnAccept);

# btnDelete = itemView.findViewById(R.id.btnDelete);

# }

# }

# private void updateCarStatus(String carId, String status, int position) {

# FirebaseFirestore db = FirebaseFirestore.getInstance();

# db.collection("cars").document(carId)

# .update("status", status)

# .addOnSuccessListener(aVoid -> {

# Toast.makeText(context, "Объявление утверждено!", Toast.LENGTH\_SHORT).show();

# carList.remove(position);

# notifyItemRemoved(position);

# })

# .addOnFailureListener(e -> {

# Toast.makeText(context, "Ошибка: " + e.getMessage(), Toast.LENGTH\_SHORT).show();

# });

# }

# private void deleteCar(String carId, int position) {

# FirebaseFirestore db = FirebaseFirestore.getInstance();

# db.collection("cars").document(carId)

# .delete()

# .addOnSuccessListener(aVoid -> {

# Toast.makeText(context, "Объявление удалено!", Toast.LENGTH\_SHORT).show();

# carList.remove(position);

# notifyItemRemoved(position);

# })

# .addOnFailureListener(e -> {

# Toast.makeText(context, "Ошибка: " + e.getMessage(), Toast.LENGTH\_SHORT).show();

# });

# }

# }

1. AllActivity.java

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.util.Log;

# import android.view.MenuItem;

# import android.view.View;

# import android.widget.ArrayAdapter;

# import android.widget.Button;

# import android.widget.ImageButton;

# import android.widget.Spinner;

# import android.widget.TextView;

# import android.widget.Toast;

# import androidx.annotation.NonNull;

# import androidx.appcompat.app.AppCompatActivity;

# import androidx.recyclerview.widget.LinearLayoutManager;

# import androidx.recyclerview.widget.RecyclerView;

# import com.google.android.material.bottomnavigation.BottomNavigationView;

# import com.google.firebase.auth.FirebaseAuth;

# import com.google.firebase.firestore.FirebaseFirestore;

# import com.google.firebase.firestore.QueryDocumentSnapshot;

# import java.util.ArrayList;

# import java.util.HashSet;

# import java.util.List;

# import java.util.Map;

# import java.util.Set;

# public class AllActivity extends AppCompatActivity {

# private RecyclerView recyclerView;

# private CarAdapter carAdapter;

# private Button resetFiltersButton, paramsButton;

# private List<Car> carList;

# private Spinner brandSpinner, modelSpinner;

# private List<String> brandList = new ArrayList<>();

# private List<String> modelList = new ArrayList<>();

# private String selectedBrand = null;

# private String selectedModel = null;

# private ImageButton filterButton;

# private FirebaseFirestore db;

# private Set<String> favoriteCarIds = new HashSet<>();

# private FirebaseAuth auth;

# private TextView adsCountTextView;

# private final BottomNavigationView.OnNavigationItemSelectedListener navListener =

# new BottomNavigationView.OnNavigationItemSelectedListener() {

# @Override

# public boolean onNavigationItemSelected(@NonNull MenuItem item) {

# int id = item.getItemId();

# if (id == R.id.nav\_home) {

# startActivity(new Intent(AllActivity.this, AllActivity.class));

# return true;

# } else if (id == R.id.nav\_saved) {

# startActivity(new Intent(AllActivity.this, FavoritesActivity.class));

# return true;

# } else if (id == R.id.nav\_create) {

# startActivity(new Intent(AllActivity.this, SelectBrandActivity.class));

# return true;

# } else if (id == R.id.nav\_profile) {

# startActivity(new Intent(AllActivity.this, UserProfileActivity.class));

# return true;

# } else if (id == R.id.nav\_garage) {

# startActivity(new Intent(AllActivity.this, BortjournalActivity.class));

# return true;

# }

# return false;

# }

# };

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_all);

# db = FirebaseFirestore.getInstance();

# auth = FirebaseAuth.getInstance();

# carList = new ArrayList<>();

# carAdapter = new CarAdapter(this, carList);

# recyclerView = findViewById(R.id.recyclerView);

# recyclerView.setLayoutManager(new LinearLayoutManager(this));

# recyclerView.setAdapter(carAdapter);

# adsCountTextView = findViewById(R.id.adsCountTextView);

# filterButton = findViewById(R.id.filterButton);

# brandSpinner = findViewById(R.id.brandSpinner);

# modelSpinner = findViewById(R.id.modelSpinner);

# resetFiltersButton = findViewById(R.id.resetFiltersButton);

# paramsButton = findViewById(R.id.paramsButton);

# paramsButton.setOnClickListener(v -> {

# Intent intent = new Intent(this, FilterActivity.class);

# startActivityForResult(intent, 101);

# });

# filterButton.setOnClickListener(v -> showFilterMenu());

# resetFiltersButton.setOnClickListener(v -> {

# brandSpinner.setSelection(0);

# modelSpinner.setSelection(0);

# selectedBrand = null;

# selectedModel = null;

# loadCars();

# });

# loadFavoriteCars();

# loadCars();

# loadBrands();

# BottomNavigationView bottomNavigationView = findViewById(R.id.bottomNavigation);

# bottomNavigationView.setOnNavigationItemSelectedListener(navListener);

# }private void applyAdvancedFilters(Intent data) {

# db.collection("cars")

# .whereEqualTo("status", "approved")

# .get()

# .addOnSuccessListener(queryDocumentSnapshots -> {

# carList.clear();

# for (QueryDocumentSnapshot doc : queryDocumentSnapshots) {

# boolean isValid = true;

# Long year = doc.getLong("year");

# Double volume = doc.getDouble("volume");

# Long horsepower = doc.getLong("horsepower");

# Long owners = doc.getLong("owners");

# Long price = doc.getLong("price");

# String color = doc.getString("color");

# String body = doc.getString("body");

# String drive = doc.getString("drive");

# String fuel = doc.getString("fuel");

# String transmission = doc.getString("transmission");

# String city = doc.getString("city");

# isValid &= checkRange(year, data.getStringExtra("yearFrom"), data.getStringExtra("yearTo"));

# isValid &= checkRange(volume, data.getStringExtra("volumeFrom"), data.getStringExtra("volumeTo"));

# isValid &= checkRange(horsepower, data.getStringExtra("powerFrom"), data.getStringExtra("powerTo"));

# isValid &= checkRange(owners, null, data.getStringExtra("ownersTo"));

# isValid &= checkRange(price, data.getStringExtra("priceFrom"), data.getStringExtra("priceTo"));

# isValid &= checkString(color, data.getStringExtra("color"));

# isValid &= checkString(body, data.getStringExtra("body"));

# isValid &= checkString(drive, data.getStringExtra("drive"));

# isValid &= checkString(fuel, data.getStringExtra("fuel"));

# isValid &= checkString(transmission, data.getStringExtra("transmission"));

# isValid &= checkString(city, data.getStringExtra("city"));

# if (isValid) {

# String photoone = doc.getString("photoone");

# if (photoone == null) photoone = "";

# long mileage = doc.getLong("mileage") != null ? doc.getLong("mileage") : 0;

# Car car = new Car(

# doc.getId(),

# doc.getString("brand"),

# doc.getString("model"),

# photoone,

# mileage,

# doc.getLong("price")

# );

# car.setFavorite(favoriteCarIds.contains(car.getId()));

# carList.add(car);

# }

# }

# adsCountTextView.setText("Найдено объявлений: " + carList.size());

# carAdapter.notifyDataSetChanged();

# });

# }

# private boolean checkRange(Number value, String fromStr, String toStr) {

# if (value == null) return false;

# try {

# double doubleValue = value.doubleValue();

# if (fromStr != null && doubleValue < Double.parseDouble(fromStr)) return false;

# if (toStr != null && doubleValue > Double.parseDouble(toStr)) return false;

# } catch (NumberFormatException e) {

# return false;

# }

# return true;

# }

# private boolean checkString(String fieldValue, String expected) {

# if (expected == null) return true; // Не фильтруем

# return expected.equalsIgnoreCase(fieldValue);

# }

# @Override

# protected void onActivityResult(int requestCode, int resultCode, Intent data) {

# super.onActivityResult(requestCode, resultCode, data);

# if (requestCode == 101 && resultCode == RESULT\_OK && data != null) {

# applyAdvancedFilters(data);

# }

# }

# private void loadFavoriteCars() {

# String userId = auth.getCurrentUser().getUid();

# db.collection("users").document(userId).get()

# .addOnSuccessListener(documentSnapshot -> {

# if (documentSnapshot.exists() && documentSnapshot.contains("favorites")) {

# List<String> favoriteCarIdsList = (List<String>) documentSnapshot.get("favorites");

# if (favoriteCarIdsList != null) {

# favoriteCarIds.addAll(favoriteCarIdsList);

# }

# }

# })

# .addOnFailureListener(e -> Log.e("AllActivity", "Ошибка загрузки избранных машин", e));

# }

# private void showFilterMenu() {

# android.widget.PopupMenu popupMenu = new android.widget.PopupMenu(this, filterButton);

# popupMenu.getMenu().add("Цена ↑");

# popupMenu.getMenu().add("Цена ↓");

# popupMenu.getMenu().add("Новые");

# popupMenu.setOnMenuItemClickListener(item -> {

# String selected = item.getTitle().toString();

# switch (selected) {

# case "Цена ↑":

# carList.sort((a, b) -> Long.compare(a.getPrice(), b.getPrice()));

# break;

# case "Цена ↓":

# carList.sort((a, b) -> Long.compare(b.getPrice(), a.getPrice()));

# break;

# case "Новые":

# carList.sort((a, b) -> b.getId().compareTo(a.getId()));

# break;

# }

# carAdapter.notifyDataSetChanged();

# return true;

# });

# popupMenu.show();

# }

# private void loadBrands() {

# db.collection("dfd").document("models")

# .get()

# .addOnSuccessListener(documentSnapshot -> {

# if (documentSnapshot.exists()) {

# Map<String, Object> data = documentSnapshot.getData();

# if (data != null) {

# brandList.clear();

# brandList.add("Марка");

# brandList.addAll(data.keySet());

# ArrayAdapter<String> brandAdapter = new ArrayAdapter<>(this,

# android.R.layout.simple\_spinner\_item, brandList);

# brandAdapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

# brandSpinner.setAdapter(brandAdapter);

# brandSpinner.setOnItemSelectedListener(new android.widget.AdapterView.OnItemSelectedListener() {

# @Override

# public void onItemSelected(android.widget.AdapterView<?> parent, View view, int position, long id) {

# if (position == 0) return;

# selectedBrand = brandList.get(position);

# loadModels(selectedBrand, data);

# }

# @Override

# public void onNothingSelected(android.widget.AdapterView<?> parent) {}

# });

# }

# }

# });

# }

# private void loadModels(String brand, Map<String, Object> data) {

# Object rawModels = data.get(brand);

# if (rawModels instanceof List) {

# List<String> models = (List<String>) rawModels;

# modelList.clear();

# modelList.add("Модель");

# modelList.addAll(models);

# ArrayAdapter<String> modelAdapter = new ArrayAdapter<>(this,

# android.R.layout.simple\_spinner\_item, modelList);

# modelAdapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

# modelSpinner.setAdapter(modelAdapter);

# modelSpinner.setOnItemSelectedListener(new android.widget.AdapterView.OnItemSelectedListener() {

# @Override

# public void onItemSelected(android.widget.AdapterView<?> parent, View view, int position, long id) {

# if (position == 0) return;

# selectedModel = modelList.get(position);

# filterCars();

# }

# @Override

# public void onNothingSelected(android.widget.AdapterView<?> parent) {}

# });

# }

# }

# private void filterCars() {

# db.collection("cars")

# .whereEqualTo("status", "approved")

# .whereEqualTo("brand", selectedBrand)

# .whereEqualTo("model", selectedModel)

# .get()

# .addOnSuccessListener(queryDocumentSnapshots -> {

# carList.clear();

# for (QueryDocumentSnapshot document : queryDocumentSnapshots) {

# String photoone = document.getString("photoone");

# if (photoone == null) photoone = "";

# long mileage = document.getLong("mileage") != null ? document.getLong("mileage") : 0;

# Car car = new Car(

# document.getId(),

# document.getString("brand"),

# document.getString("model"),

# photoone,

# mileage,

# document.getLong("price")

# );

# car.setFavorite(favoriteCarIds.contains(car.getId()));

# carList.add(car);

# }

# adsCountTextView.setText("Найдено объявлений: " + carList.size());

# carAdapter.notifyDataSetChanged();

# });

# }

# private long getMileage(Car car) {

# try {

# java.lang.reflect.Field mileageField = car.getClass().getDeclaredField("mileage");

# mileageField.setAccessible(true);

# Object value = mileageField.get(car);

# return value instanceof Long ? (Long) value : 0L;

# } catch (Exception e) {

# return 0L;

# }

# }

# private void loadCars() {

# db.collection("cars")

# .whereEqualTo("status", "approved")

# .get()

# .addOnCompleteListener(task -> {

# if (task.isSuccessful()) {

# carList.clear();

# for (QueryDocumentSnapshot document : task.getResult()) {

# String photoone = document.getString("photoone");

# if (photoone == null) {

# photoone = "";

# }

# long mileage = document.getLong("mileage") != null ? document.getLong("mileage") : 0;

# Car car = new Car(

# document.getId(),

# document.getString("brand"),

# document.getString("model"),

# photoone,

# mileage,

# document.getLong("price")

# );

# car.setFavorite(favoriteCarIds.contains(car.getId()));

# carList.add(car);

# }

# adsCountTextView.setText(carList.size() + " предложений");

# carAdapter.notifyDataSetChanged();

# } else {

# Log.e("AllActivity", "Ошибка загрузки машин", task.getException());

# }

# });

# }

# }

1. BortjournalActivity.java;

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.widget.Button;

# import androidx.appcompat.app.AppCompatActivity;

# import androidx.recyclerview.widget.LinearLayoutManager;

# import androidx.recyclerview.widget.RecyclerView;

# import com.google.firebase.firestore.FirebaseFirestore;

# import com.google.firebase.firestore.Query;

# import com.google.firebase.firestore.QueryDocumentSnapshot;

# import java.util.ArrayList;

# import java.util.List;

# public class BortjournalActivity extends AppCompatActivity {

# private RecyclerView postsRecyclerView;

# private Button createPostButton;

# private FirebaseFirestore db;

# private List<BortjournalPost> posts = new ArrayList<>();

# private BortjournalAdapter adapter;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.bortjournal\_activity);

# postsRecyclerView = findViewById(R.id.postsRecyclerView);

# createPostButton = findViewById(R.id.createPostButton);

# db = FirebaseFirestore.getInstance();

# adapter = new BortjournalAdapter(this, posts);

# postsRecyclerView.setLayoutManager(new LinearLayoutManager(this));

# postsRecyclerView.setAdapter(adapter);

# createPostButton.setOnClickListener(v -> {

# startActivity(new Intent(this, CreatePostActivity.class));

# });

# loadPosts();

# }

# private void loadPosts() {

# db.collection("bortjournal")

# .whereEqualTo("status", "approved")

# .orderBy("timestamp", Query.Direction.DESCENDING)

# .get()

# .addOnSuccessListener(queryDocumentSnapshots -> {

# posts.clear();

# for (QueryDocumentSnapshot doc : queryDocumentSnapshots) {

# BortjournalPost post = doc.toObject(BortjournalPost.class);

# posts.add(post);

# }

# adapter.notifyDataSetChanged();

# });

# }

# }

1. BortjournalAdapter.java;

# package com.example.drivee;

# import android.content.Context;

# import android.content.Intent;

# import android.graphics.Bitmap;

# import android.graphics.BitmapFactory;

# import android.graphics.Canvas;

# import android.graphics.Paint;

# import android.graphics.PorterDuffXfermode;

# import android.graphics.Rect;

# import android.graphics.RectF;

# import android.graphics.PorterDuff;

# import android.util.Base64;

# import android.view.LayoutInflater;

# import android.view.View;

# import android.view.ViewGroup;

# import android.widget.ImageView;

# import android.widget.TextView;

# import androidx.annotation.NonNull;

# import androidx.recyclerview.widget.RecyclerView;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.util.List;

# public class BortjournalAdapter extends RecyclerView.Adapter<BortjournalAdapter.PostViewHolder> {

# private Context context;

# private List<BortjournalPost> postList;

# private FirebaseFirestore db = FirebaseFirestore.getInstance();

# public BortjournalAdapter(Context context, List<BortjournalPost> postList) {

# this.context = context;

# this.postList = postList;

# }

# @NonNull

# @Override

# public PostViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

# View view = LayoutInflater.from(context).inflate(R.layout.item\_bortjournal, parent, false);

# return new PostViewHolder(view);

# }

# @Override

# public void onBindViewHolder(@NonNull PostViewHolder holder, int position) {

# BortjournalPost post = postList.get(position);

# holder.brandModelTextView.setText(post.getBrand() + " " + post.getModel());

# holder.titleTextView.setText(post.getTitle());

# holder.mileageTextView.setText("Пробег: " + post.getMileage() + " км");

# // Загружаем имя и аватар пользователя

# db.collection("users").document(post.getUserId())

# .get()

# .addOnSuccessListener(doc -> {

# String name = doc.getString("name");

# String imageBase64 = doc.getString("profile\_image");

# if (name != null) {

# holder.userNameTextView.setText(name);

# }

# if (imageBase64 != null && !imageBase64.isEmpty()) {

# try {

# byte[] decoded = Base64.decode(imageBase64, Base64.DEFAULT);

# Bitmap bitmap = BitmapFactory.decodeByteArray(decoded, 0, decoded.length);

# holder.profileImageView.setImageBitmap(getCircularBitmap(bitmap));

# } catch (Exception e) {

# e.printStackTrace();

# }

# } else {

# holder.profileImageView.setImageResource(R.drawable.logo);

# }

# });

# holder.itemView.setOnClickListener(v -> {

# Intent intent = new Intent(context, BortjournalDetailActivity.class);

# intent.putExtra("brand", post.getBrand());

# intent.putExtra("model", post.getModel());

# intent.putExtra("title", post.getTitle());

# intent.putExtra("content", post.getContent());

# context.startActivity(intent);

# });

# }

# @Override

# public int getItemCount() {

# return postList.size();

# }

# public static class PostViewHolder extends RecyclerView.ViewHolder {

# TextView brandModelTextView, titleTextView, mileageTextView, userNameTextView;

# ImageView profileImageView;

# public PostViewHolder(@NonNull View itemView) {

# super(itemView);

# brandModelTextView = itemView.findViewById(R.id.brandModelTextView);

# titleTextView = itemView.findViewById(R.id.titleTextView);

# mileageTextView = itemView.findViewById(R.id.mileageTextView);

# userNameTextView = itemView.findViewById(R.id.userNameTextView);

# profileImageView = itemView.findViewById(R.id.profileImageView);

# }

# }

# private Bitmap getCircularBitmap(Bitmap bitmap) {

# int size = Math.min(bitmap.getWidth(), bitmap.getHeight());

# Bitmap output = Bitmap.createBitmap(size, size, Bitmap.Config.ARGB\_8888);

# Canvas canvas = new Canvas(output);

# final Paint paint = new Paint();

# final Rect rect = new Rect(0, 0, size, size);

# final RectF rectF = new RectF(rect);

# paint.setAntiAlias(true);

# canvas.drawCircle(size / 2f, size / 2f, size / 2f, paint);

# paint.setXfermode(new PorterDuffXfermode(PorterDuff.Mode.SRC\_IN));

# canvas.drawBitmap(bitmap, null, rect, paint);

# return output;

# }

# }

1. BortjournalDetailActivity.java;

# package com.example.drivee;

# import android.os.Bundle;

# import android.widget.TextView;

# import androidx.appcompat.app.AppCompatActivity;

# public class BortjournalDetailActivity extends AppCompatActivity {

# private TextView detailBrandModel, detailTitle, detailContent;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.bortjournal\_detail\_activity);

# detailBrandModel = findViewById(R.id.detailBrandModel);

# detailTitle = findViewById(R.id.detailTitle);

# detailContent = findViewById(R.id.detailContent);

# String brand = getIntent().getStringExtra("brand");

# String model = getIntent().getStringExtra("model");

# String title = getIntent().getStringExtra("title");

# String content = getIntent().getStringExtra("content");

# detailBrandModel.setText(brand + " " + model);

# detailTitle.setText(title);

# detailContent.setText(content);

# }

# }

1. BortjournalPost.java;

# package com.example.drivee;

# public class BortjournalPost {

# private String id;

# private String brand;

# private String model;

# private String title;

# private String content;

# private String userId;

# private long mileage;

# private String status;

# public BortjournalPost() {

# }

# public String getId() {

# return id;

# }

# public void setId(String id) {

# this.id = id;

# }

# public String getBrand() {

# return brand;

# }

# public void setBrand(String brand) {

# this.brand = brand;

# }

# public String getModel() {

# return model;

# }

# public void setModel(String model) {

# this.model = model;

# }

# public String getTitle() {

# return title;

# }

# public void setTitle(String title) {

# this.title = title;

# }

# public String getContent() {

# return content;

# }

# public void setContent(String content) {

# this.content = content;

# }

# public String getUserId() {

# return userId;

# }

# public void setUserId(String userId) {

# this.userId = userId;

# }

# public long getMileage() {

# return mileage;

# }

# public void setMileage(long mileage) {

# this.mileage = mileage;

# }

# public String getStatus() {

# return status;

# }

# public void setStatus(String status) {

# this.status = status;

# }

# }

1. Car.java;

# package com.example.drivee;

# import android.util.Log;

# import com.google.android.gms.common.api.internal.IStatusCallback;

# public class Car {

# private String id;

# private String brand;

# private String model;

# private String photoBase64;

# private long mileage;

# private long price;

# private boolean isFavorite;

# private String status;

# // Конструктор

# public Car(String id, String brand, String model, String photoBase64, long mileage, long price) {

# this.id = id;

# this.brand = brand;

# this.model = model;

# this.photoBase64 = photoBase64;

# this.mileage = mileage;

# this.price = price;

# }

# public String getId() { return id; }

# public String getBrand() { return brand; }

# public String getModel() { return model; }

# public String getPhotoBase64() { return photoBase64; }

# public long getMileage() { return mileage; }

# public long getPrice() { return price; }

# public boolean isFavorite() { return isFavorite; }

# public void setFavorite(boolean favorite) { isFavorite = favorite; }

# public String getStatus() { return status; }

# public void setStatus(String status) { this.status = status; }

# }

1. CarAdapter.java;

# package com.example.drivee;

# import android.content.Context;

# import android.content.Intent;

# import android.graphics.Bitmap;

# import android.graphics.BitmapFactory;

# import android.util.Base64;

# import android.view.LayoutInflater;

# import android.view.View;

# import android.view.ViewGroup;

# import android.widget.ImageButton;

# import android.widget.ImageView;

# import android.widget.TextView;

# import android.widget.Toast;

# import androidx.annotation.NonNull;

# import androidx.recyclerview.widget.RecyclerView;

# import com.google.firebase.auth.FirebaseAuth;

# import com.google.firebase.firestore.FieldValue;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.text.DecimalFormat;

# import java.util.List;

# public class CarAdapter extends RecyclerView.Adapter<CarAdapter.CarViewHolder> {

# private Context context;

# private List<Car> carList;

# private boolean showStatus = true;

# private boolean showFavorite = true;

# private FirebaseAuth auth = FirebaseAuth.getInstance();

# private FirebaseFirestore db = FirebaseFirestore.getInstance();

# public CarAdapter(Context context, List<Car> carList) {

# this(context, carList, true, false);

# }

# public CarAdapter(Context context, List<Car> carList, boolean showFavorite, boolean showStatus) {

# this.context = context;

# this.carList = carList;

# this.showFavorite = showFavorite;

# this.showStatus = showStatus;

# }

# @NonNull

# @Override

# public CarViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

# View view = LayoutInflater.from(context).inflate(R.layout.item\_car, parent, false);

# return new CarViewHolder(view);

# }

# @Override

# public void onBindViewHolder(@NonNull CarViewHolder holder, int position) {

# Car car = carList.get(position);

# holder.carTitleTextView.setText(car.getBrand() + " " + car.getModel());

# DecimalFormat formatter = new DecimalFormat("#,###");

# String formattedPrice = formatter.format(car.getPrice()).replace(",", " ");

# holder.carPriceTextView.setText(formattedPrice + " ₽");

# // 🟡 Обработка статуса (с переводом)

# if (showStatus && holder.carStatusTextView != null && car.getStatus() != null) {

# holder.carStatusTextView.setVisibility(View.VISIBLE);

# String statusTranslated;

# switch (car.getStatus()) {

# case "approved":

# statusTranslated = "Подтверждено";

# break;

# case "pending":

# statusTranslated = "На проверке";

# break;

# case "rejected":

# statusTranslated = "Отклонено";

# break;

# default:

# statusTranslated = car.getStatus();

# break;

# }

# holder.carStatusTextView.setText("Статус: " + statusTranslated);

# } else if (holder.carStatusTextView != null) {

# holder.carStatusTextView.setVisibility(View.GONE);

# }

# // Фото из base64

# String base64 = car.getPhotoBase64();

# if (base64 != null && !base64.isEmpty()) {

# try {

# byte[] decodedBytes = Base64.decode(base64, Base64.DEFAULT);

# Bitmap bitmap = BitmapFactory.decodeByteArray(decodedBytes, 0, decodedBytes.length);

# holder.carImageView.setImageBitmap(bitmap);

# } catch (Exception e) {

# e.printStackTrace();

# holder.carImageView.setImageResource(R.drawable.logo);

# }

# } else {

# holder.carImageView.setImageResource(R.drawable.logo);

# }

# // Кнопка "избранное"

# if (showFavorite) {

# holder.favoriteButton.setVisibility(View.VISIBLE);

# updateFavoriteIcon(holder.favoriteButton, car.isFavorite());

# holder.favoriteButton.setOnClickListener(v -> {

# boolean isNowFavorite = !car.isFavorite();

# car.setFavorite(isNowFavorite);

# updateFavoriteIcon(holder.favoriteButton, isNowFavorite);

# updateFavoriteInFirebase(car.getId(), isNowFavorite);

# });

# } else {

# holder.favoriteButton.setVisibility(View.GONE); // скрываем в профиле

# }

# if (showStatus && holder.deleteButton != null) {

# holder.deleteButton.setVisibility(View.VISIBLE);

# holder.deleteButton.setOnClickListener(v -> {

# String carId = car.getId();

# db.collection("cars").document(carId)

# .delete()

# .addOnSuccessListener(unused -> {

# carList.remove(position);

# notifyItemRemoved(position);

# Toast.makeText(context, "Машина удалена", Toast.LENGTH\_SHORT).show();

# })

# .addOnFailureListener(e -> Toast.makeText(context, "Ошибка при удалении", Toast.LENGTH\_SHORT).show());

# });

# } else if (holder.deleteButton != null) {

# holder.deleteButton.setVisibility(View.GONE);

# }

# long mileage = 0;

# try {

# java.lang.reflect.Field mileageField = car.getClass().getDeclaredField("mileage");

# mileageField.setAccessible(true);

# Object value = mileageField.get(car);

# if (value instanceof Long) mileage = (Long) value;

# } catch (Exception e) {

# e.printStackTrace();

# }

# holder.carMileageTextView.setText("Пробег: " + mileage + " км");

# holder.itemView.setOnClickListener(v -> {

# Intent intent = new Intent(context, CenterActivity.class);

# intent.putExtra("carId", car.getId());

# context.startActivity(intent);

# });

# }

# @Override

# public int getItemCount() {

# return carList.size();

# }

# public static class CarViewHolder extends RecyclerView.ViewHolder {

# ImageView carImageView;

# TextView carTitleTextView, carPriceTextView, carStatusTextView, deleteButton;

# TextView carMileageTextView;

# ImageButton favoriteButton;

# public CarViewHolder(@NonNull View itemView) {

# super(itemView);

# carImageView = itemView.findViewById(R.id.carImageView);

# carTitleTextView = itemView.findViewById(R.id.carTitleTextView);

# carPriceTextView = itemView.findViewById(R.id.carPriceTextView);

# carMileageTextView = itemView.findViewById(R.id.carMileageTextView);

# carStatusTextView = itemView.findViewById(R.id.carStatusTextView);

# favoriteButton = itemView.findViewById(R.id.favoriteButton);

# deleteButton = itemView.findViewById(R.id.deleteCarButton); // Добавили

# }

# }

# private void updateFavoriteIcon(ImageButton button, boolean isFavorite) {

# if (isFavorite) {

# button.setImageResource(R.drawable.ic\_heart\_filled);

# } else {

# button.setImageResource(R.drawable.ic\_heart\_outline);

# }

# }

# private void updateFavoriteInFirebase(String carId, boolean add) {

# String userId = auth.getCurrentUser().getUid();

# if (add) {

# db.collection("users").document(userId)

# .update("favorites", FieldValue.arrayUnion(carId));

# } else {

# db.collection("users").document(userId)

# .update("favorites", FieldValue.arrayRemove(carId));

# }

# }

# }

1. CarAdapterUser.java;

# package com.example.drivee;

# import android.content.Context;

# import android.view.LayoutInflater;

# import android.view.View;

# import android.view.ViewGroup;

# import android.widget.ImageView;

# import android.widget.TextView;

# import android.widget.Toast;

# import androidx.annotation.NonNull;

# import androidx.recyclerview.widget.RecyclerView;

# import com.bumptech.glide.Glide;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.util.List;

# public class CarAdapterUser extends RecyclerView.Adapter<CarAdapterUser.CarViewHolder> {

# private Context context;

# private List<UserCar> carList;

# private FirebaseFirestore db = FirebaseFirestore.getInstance();

# public CarAdapterUser(Context context, List<UserCar> carList) {

# this.context = context;

# this.carList = carList;

# }

# @NonNull

# @Override

# public CarViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

# View view = LayoutInflater.from(context).inflate(R.layout.item\_car\_user, parent, false);

# return new CarViewHolder(view);

# }

# @Override

# public void onBindViewHolder(@NonNull CarViewHolder holder, int position) {

# UserCar car = carList.get(position);

# holder.title.setText(car.getBrand() + " " + car.getModel());

# holder.price.setText(car.getPrice() + " ₽");

# holder.status.setText("Статус: " + car.getStatus());

# // Загрузка фото

# if (car.getPhotoUrl() != null && !car.getPhotoUrl().isEmpty()) {

# Glide.with(context)

# .load(car.getPhotoUrl())

# .into(holder.imageView);

# } else {

# holder.imageView.setImageResource(R.drawable.logo);

# }

# holder.deleteButton.setOnClickListener(v -> {

# db.collection("cars").document(car.getId())

# .delete()

# .addOnSuccessListener(unused -> {

# carList.remove(position);

# notifyItemRemoved(position);

# notifyItemRangeChanged(position, carList.size());

# Toast.makeText(context, "Объявление удалено", Toast.LENGTH\_SHORT).show();

# })

# .addOnFailureListener(e -> {

# Toast.makeText(context, "Ошибка удаления", Toast.LENGTH\_SHORT).show();

# });

# });

# }

# @Override

# public int getItemCount() {

# return carList.size();

# }

# public static class CarViewHolder extends RecyclerView.ViewHolder {

# ImageView imageView;

# TextView title, price, status, deleteButton;

# public CarViewHolder(@NonNull View itemView) {

# super(itemView);

# imageView = itemView.findViewById(R.id.carImageView);

# title = itemView.findViewById(R.id.carTitleTextView);

# price = itemView.findViewById(R.id.carPriceTextView);

# status = itemView.findViewById(R.id.carStatusTextView);

# deleteButton = itemView.findViewById(R.id.deleteCarButton);

# }

# }

# }

1. CarForProfile.java;

# package com.example.drivee;

# public class CarForProfile {

# private String id;

# private String brand;

# private String model;

# private long price;

# private String status;

# public CarForProfile(String id, String brand, String model, long price, String status) {

# this.id = id;

# this.brand = brand;

# this.model = model;

# this.price = price;

# this.status = status;

# }

# public String getId() { return id; }

# public void setId(String id) { this.id = id; }

# public String getBrand() { return brand; }

# public void setBrand(String brand) { this.brand = brand; }

# public String getModel() { return model; }

# public void setModel(String model) { this.model = model; }

# public long getPrice() { return price; }

# public void setPrice(long price) { this.price = price; }

# public String getStatus() { return status; }

# public void setStatus(String status) { this.status = status; }

# }

1. CenterActivity.java;

# package com.example.drivee;

# import android.content.Intent;

# import android.content.pm.PackageManager;

# import android.graphics.Bitmap;

# import android.graphics.BitmapFactory;

# import android.net.Uri;

# import android.os.Bundle;

# import android.util.Base64;

# import android.util.Log;

# import android.widget.ImageView;

# import android.widget.TextView;

# import android.widget.Toast;

# import androidx.appcompat.app.AppCompatActivity;

# import androidx.viewpager2.widget.ViewPager2;

# import com.google.firebase.Timestamp;

# import com.google.firebase.firestore.DocumentReference;

# import com.google.firebase.firestore.DocumentSnapshot;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.text.SimpleDateFormat;

# import java.util.ArrayList;

# import java.util.Date;

# import java.util.List;

# import java.util.Locale;

# public class CenterActivity extends AppCompatActivity {

# private static final String TAG = "CenterActivity";

# private ViewPager2 carImagePager;

# private ImageView userAvatar;

# private TextView userName, userPhone;

# private TextView vinTextView, dataPublicTextView, bodyTextView, brandTextView, colorTextView, descriptionTextView;

# private TextView driveTextView, fuelTextView, horsepowerTextView, mileageTextView, modelTextView;

# private TextView ownersTextView, priceTextView, transmissionTextView, volumeTextView, yearTextView;

# private FirebaseFirestore db;

# private String carId;

# private ImagePagerAdapter imageAdapter;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_center);

# carId = getIntent().getStringExtra("carId");

# if (carId == null || carId.isEmpty()) {

# Log.e(TAG, "Ошибка: carId не передан!");

# finish();

# return;

# }

# vinTextView = findViewById(R.id.vinTextView);

# userAvatar = findViewById(R.id.userAvatar);

# userName = findViewById(R.id.userName);

# userPhone = findViewById(R.id.userPhone);

# bodyTextView = findViewById(R.id.bodyTextView);

# brandTextView = findViewById(R.id.brandTextView);

# colorTextView = findViewById(R.id.colorTextView);

# descriptionTextView = findViewById(R.id.descriptionTextView);

# driveTextView = findViewById(R.id.driveTextView);

# fuelTextView = findViewById(R.id.fuelTextView);

# horsepowerTextView = findViewById(R.id.horsepowerTextView);

# mileageTextView = findViewById(R.id.mileageTextView);

# modelTextView = findViewById(R.id.modelTextView);

# carImagePager = findViewById(R.id.carImagePager);

# dataPublicTextView = findViewById(R.id.dataPublicTextView);

# ownersTextView = findViewById(R.id.ownersTextView);

# priceTextView = findViewById(R.id.priceTextView);

# transmissionTextView = findViewById(R.id.transmissionTextView);

# volumeTextView = findViewById(R.id.volumeTextView);

# yearTextView = findViewById(R.id.yearTextView);

# db = FirebaseFirestore.getInstance();

# List<String> initialPhotoUrls = new ArrayList<>();

# imageAdapter = new ImagePagerAdapter(this, initialPhotoUrls);

# carImagePager.setAdapter(imageAdapter);

# loadCarDetails();

# TextView phoneTextView = findViewById(R.id.userPhone);

# phoneTextView.setOnClickListener(v -> {

# String phoneNumber = phoneTextView.getText().toString().trim();

# Intent callIntent = new Intent(Intent.ACTION\_CALL);

# callIntent.setData(Uri.parse("tel:" + phoneNumber));

# if (checkSelfPermission(android.Manifest.permission.CALL\_PHONE) == PackageManager.PERMISSION\_GRANTED) {

# startActivity(callIntent);

# } else {

# requestPermissions(new String[]{android.Manifest.permission.CALL\_PHONE}, 1);

# }

# });

# }

# @Override

# public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {

# super.onRequestPermissionsResult(requestCode, permissions, grantResults);

# if (requestCode == 1) {

# if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED) {

# Toast.makeText(this, "Разрешение получено, нажмите еще раз для звонка", Toast.LENGTH\_SHORT).show();

# } else {

# Toast.makeText(this, "Разрешение на звонок не предоставлено", Toast.LENGTH\_SHORT).show();

# }

# }

# }

# private void loadCarDetails() {

# DocumentReference docRef = db.collection("cars").document(carId);

# docRef.get().addOnCompleteListener(task -> {

# if (task.isSuccessful()) {

# DocumentSnapshot document = task.getResult();

# if (document.exists()) {

# vinTextView.setText("VIN: " + document.getString("vin"));

# bodyTextView.setText("Кузов: " + document.getString("body"));

# brandTextView.setText(document.getString("brand"));

# colorTextView.setText("Цвет: " + document.getString("color"));

# descriptionTextView.setText(document.getString("description"));

# driveTextView.setText("Привод: " + document.getString("drive"));

# Timestamp timestamp = document.getTimestamp("DataPublic");

# if (timestamp != null) {

# Date date = timestamp.toDate();

# SimpleDateFormat sdf = new SimpleDateFormat("dd MMMM yyyy", Locale.getDefault());

# dataPublicTextView.setText(sdf.format(date));

# }

# fuelTextView.setText("Топливо: " + document.getString("fuel"));

# horsepowerTextView.setText("Мощность: " + document.getLong("horsepower") + " л.с.");

# mileageTextView.setText("Пробег: " + document.getLong("mileage") + " км");

# modelTextView.setText(document.getString("model"));

# ownersTextView.setText("Количество владельцев: " + document.getLong("owners"));

# priceTextView.setText(document.getLong("price") + " ₽");

# transmissionTextView.setText("Трансмиссия: " + document.getString("transmission"));

# volumeTextView.setText("Объем двигателя: " + document.get("volume") + " л");

# yearTextView.setText(String.valueOf(document.getLong("year")));

# List<String> photoBase64List = new ArrayList<>();

# String[] photoFields = {"photoone", "phototwo", "photothree", "photofour", "photofive"};

# for (String field : photoFields) {

# String base64 = document.getString(field);

# if (base64 != null && !base64.isEmpty()) {

# photoBase64List.add(base64);

# }

# }

# String ownerId = document.getString("ownerId");

# if (ownerId != null) {

# db.collection("users").document(ownerId).get()

# .addOnSuccessListener(userDoc -> {

# if (userDoc.exists()) {

# userName.setText(userDoc.getString("name"));

# userPhone.setText(userDoc.getString("phone"));

# String profileBase64 = userDoc.getString("profile\_image");

# if (profileBase64 != null && !profileBase64.isEmpty()) {

# byte[] decodedBytes = Base64.decode(profileBase64, Base64.DEFAULT);

# Bitmap bitmap = BitmapFactory.decodeByteArray(decodedBytes, 0, decodedBytes.length);

# userAvatar.setImageBitmap(bitmap);

# } else {

# userAvatar.setImageResource(R.drawable.user\_prof\_1);

# }

# }

# })

# .addOnFailureListener(e -> Log.e(TAG, "Ошибка загрузки данных пользователя", e));

# }

# imageAdapter.updateImages(photoBase64List);

# } else {

# Log.e(TAG, "Документ не найден!");

# }

# } else {

# Log.e(TAG, "Ошибка при загрузке данных", task.getException());

# }

# });

# }

# }

1. CreateListingActivity.java;

# package com.example.drivee;

# import android.Manifest;

# import android.content.pm.PackageManager;

# import androidx.core.app.ActivityCompat;

# import androidx.core.content.ContextCompat;

# import com.google.android.gms.location.FusedLocationProviderClient;

# import com.google.android.gms.location.LocationServices;

# import android.location.Address;

# import android.location.Geocoder;

# import java.util.Locale;

# import android.app.Activity;

# import android.content.Intent;

# import android.content.pm.PackageManager;

# import android.graphics.Bitmap;

# import android.graphics.BitmapFactory;

# import android.location.Address;

# import android.location.Geocoder;

# import android.net.Uri;

# import android.os.Bundle;

# import android.util.Base64;

# import android.view.View;

# import android.widget.ArrayAdapter;

# import android.widget.Button;

# import android.widget.EditText;

# import android.widget.ImageView;

# import android.widget.Spinner;

# import android.widget.Toast;

# import androidx.annotation.NonNull;

# import androidx.annotation.Nullable;

# import androidx.appcompat.app.AppCompatActivity;

# import androidx.core.app.ActivityCompat;

# import androidx.core.content.ContextCompat;

# import com.google.firebase.auth.FirebaseAuth;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.io.ByteArrayOutputStream;

# import java.io.InputStream;

# import java.util.ArrayList;

# import java.util.HashMap;

# import java.util.List;

# import java.util.Locale;

# import java.util.Map;

# public class CreateListingActivity extends AppCompatActivity {

# private List<ImageView> imagePreviews = new ArrayList<>();

# private EditText editVin, editBrand, editModel, editYear, editColor, editVolume, editHorsepower,

# editMileage, editOwners, editPrice, editDescription, editCity;

# private Spinner spinnerDrive, spinnerTransmission, spinnerFuel, spinnerBody;

# private Button btnCreateListing, btnUploadPhoto, btnResetPhotos;

# private final List<Uri> selectedImages = new ArrayList<>();

# private static final int MAX\_PHOTOS = 5;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_create\_listing);

# if (ContextCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_FINE\_LOCATION)

# != PackageManager.PERMISSION\_GRANTED) {

# ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.ACCESS\_FINE\_LOCATION}, 1);

# } else {

# getCityFromLocation();

# }

# imagePreviews.add(findViewById(R.id.imagePreview1));

# imagePreviews.add(findViewById(R.id.imagePreview2));

# imagePreviews.add(findViewById(R.id.imagePreview3));

# imagePreviews.add(findViewById(R.id.imagePreview4));

# imagePreviews.add(findViewById(R.id.imagePreview5));

# editVin = findViewById(R.id.editVin);

# editBrand = findViewById(R.id.editBrand);

# editModel = findViewById(R.id.editModel);

# editYear = findViewById(R.id.editYear);

# editColor = findViewById(R.id.editColor);

# editVolume = findViewById(R.id.editVolume);

# editHorsepower = findViewById(R.id.editHorsepower);

# editMileage = findViewById(R.id.editMileage);

# editOwners = findViewById(R.id.editOwners);

# editPrice = findViewById(R.id.editPrice);

# editDescription = findViewById(R.id.editDescription);

# editCity = findViewById(R.id.editCity);

# spinnerDrive = findViewById(R.id.spinnerDrive);

# spinnerTransmission = findViewById(R.id.spinnerTransmission);

# spinnerFuel = findViewById(R.id.spinnerFuel);

# spinnerBody = findViewById(R.id.spinnerBody);

# btnCreateListing = findViewById(R.id.btnCreateListing);

# btnUploadPhoto = findViewById(R.id.btnUploadPhoto);

# btnResetPhotos = findViewById(R.id.btnRemovePhoto5);

# String selectedBrand = getIntent().getStringExtra("selectedBrand");

# String selectedModel = getIntent().getStringExtra("selectedModel");

# if (selectedBrand != null && !selectedBrand.isEmpty()) {

# editBrand.setText(selectedBrand);

# editBrand.setEnabled(false);

# }

# if (selectedModel != null && !selectedModel.isEmpty()) {

# editModel.setText(selectedModel);

# editModel.setEnabled(false);

# }

# setupSpinners();

# btnUploadPhoto.setOnClickListener(v -> openGallery());

# btnCreateListing.setOnClickListener(v -> saveListing());

# btnResetPhotos.setOnClickListener(v -> resetPhotos());

# }

# private void getCityFromLocation() {

# if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED) {

# return;

# }

# FusedLocationProviderClient fusedLocationClient = LocationServices.getFusedLocationProviderClient(this);

# Geocoder geocoder = new Geocoder(this, Locale.getDefault());

# fusedLocationClient.getLastLocation().addOnSuccessListener(location -> {

# if (location != null) {

# try {

# List<Address> addresses = geocoder.getFromLocation(location.getLatitude(), location.getLongitude(), 1);

# if (addresses != null && !addresses.isEmpty()) {

# String city = addresses.get(0).getLocality();

# if (city != null) {

# editCity.setText(city);

# }

# }

# } catch (Exception e) {

# e.printStackTrace();

# }

# }

# });

# }

# @Override

# public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {

# super.onRequestPermissionsResult(requestCode, permissions, grantResults);

# if (requestCode == 1 && grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED) {

# getCityFromLocation();

# }

# }

# private void setupSpinners() {

# String[] driveOptions = {"Привод", "Полный", "Передний", "Задний"};

# String[] transmissionOptions = {"Коробка", "Автоматическая", "Механика", "Вариатор", "Робот"};

# String[] fuelOptions = {"Топливо", "Бензин", "Дизель", "Гибрид", "Электро", "ГБО", "Метан"};

# String[] bodyOptions = {"Тип кузова", "Купе", "Седан", "Хэтчбек", "Кроссовер", "Внедорожник"};

# spinnerDrive.setAdapter(new ArrayAdapter<>(this, android.R.layout.simple\_spinner\_dropdown\_item, driveOptions));

# spinnerTransmission.setAdapter(new ArrayAdapter<>(this, android.R.layout.simple\_spinner\_dropdown\_item, transmissionOptions));

# spinnerFuel.setAdapter(new ArrayAdapter<>(this, android.R.layout.simple\_spinner\_dropdown\_item, fuelOptions));

# spinnerBody.setAdapter(new ArrayAdapter<>(this, android.R.layout.simple\_spinner\_dropdown\_item, bodyOptions));

# }

# private void openGallery() {

# if (selectedImages.size() < MAX\_PHOTOS) {

# Intent intent = new Intent(Intent.ACTION\_PICK);

# intent.setType("image/\*");

# intent.putExtra(Intent.EXTRA\_ALLOW\_MULTIPLE, true);

# startActivityForResult(intent, 1);

# }

# }

# @Override

# protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {

# super.onActivityResult(requestCode, resultCode, data);

# if (requestCode == 1 && resultCode == Activity.RESULT\_OK && data != null) {

# if (data.getClipData() != null) {

# int count = data.getClipData().getItemCount();

# for (int i = 0; i < count && selectedImages.size() < MAX\_PHOTOS; i++) {

# Uri imageUri = data.getClipData().getItemAt(i).getUri();

# selectedImages.add(imageUri);

# imagePreviews.get(i).setImageURI(imageUri);

# }

# } else if (data.getData() != null && selectedImages.size() < MAX\_PHOTOS) {

# Uri imageUri = data.getData();

# selectedImages.add(imageUri);

# imagePreviews.get(0).setImageURI(imageUri);

# }

# }

# }

# private void resetPhotos() {

# selectedImages.clear();

# for (ImageView imageView : imagePreviews) {

# imageView.setImageResource(android.R.color.darker\_gray);

# }

# }

# private void saveListing() {

# String userId = FirebaseAuth.getInstance().getCurrentUser().getUid();

# if (spinnerDrive.getSelectedItemPosition() == 0) {

# Toast.makeText(this, "Выберите тип привода", Toast.LENGTH\_SHORT).show();

# return;

# }

# if (spinnerTransmission.getSelectedItemPosition() == 0) {

# Toast.makeText(this, "Выберите тип коробки", Toast.LENGTH\_SHORT).show();

# return;

# }

# if (spinnerFuel.getSelectedItemPosition() == 0) {

# Toast.makeText(this, "Выберите тип топлива", Toast.LENGTH\_SHORT).show();

# return;

# }

# if (spinnerBody.getSelectedItemPosition() == 0) {

# Toast.makeText(this, "Выберите тип кузова", Toast.LENGTH\_SHORT).show();

# return;

# }

# String vin = editVin.getText().toString().trim();

# if (vin.length() != 17) {

# Toast.makeText(this, "VIN должен содержать 17 символов", Toast.LENGTH\_SHORT).show();

# return;

# }

# int year = parseIntSafe(editYear.getText().toString());

# int currentYear = java.util.Calendar.getInstance().get(java.util.Calendar.YEAR);

# if (year < 1850 || year > currentYear) {

# Toast.makeText(this, "Год выпуска должен быть между 1850 и " + currentYear, Toast.LENGTH\_SHORT).show();

# return;

# }

# String color = editColor.getText().toString().trim();

# if (!color.matches("^[А-Яа-яA-Za-z\\s]+$")) {

# Toast.makeText(this, "Цвет должен содержать только буквы", Toast.LENGTH\_SHORT).show();

# return;

# }

# FirebaseFirestore db = FirebaseFirestore.getInstance();

# Map<String, Object> carData = new HashMap<>();

# carData.put("vin", vin);

# carData.put("brand", editBrand.getText().toString().trim());

# carData.put("model", editModel.getText().toString().trim());

# carData.put("year", year);

# carData.put("ownerId", userId);

# carData.put("color", editColor.getText().toString().trim());

# carData.put("volume", parseIntSafe(editVolume.getText().toString()));

# carData.put("horsepower", parseIntSafe(editHorsepower.getText().toString()));

# carData.put("mileage", parseIntSafe(editMileage.getText().toString()));

# carData.put("owners", parseIntSafe(editOwners.getText().toString()));

# carData.put("price", parseIntSafe(editPrice.getText().toString()));

# carData.put("description", editDescription.getText().toString().trim());

# carData.put("city", editCity.getText().toString().trim());

# carData.put("drive", spinnerDrive.getSelectedItem().toString());

# carData.put("transmission", spinnerTransmission.getSelectedItem().toString());

# carData.put("fuel", spinnerFuel.getSelectedItem().toString());

# carData.put("body", spinnerBody.getSelectedItem().toString());

# carData.put("status", "pending");

# for (int i = 0; i < 5; i++) {

# String keyName = "photo" + (i == 0 ? "one" :

# i == 1 ? "two" :

# i == 2 ? "three" :

# i == 3 ? "four" : "five");

# if (i < selectedImages.size()) {

# try {

# InputStream inputStream = getContentResolver().openInputStream(selectedImages.get(i));

# Bitmap bitmap = BitmapFactory.decodeStream(inputStream);

# ByteArrayOutputStream outputStream = new ByteArrayOutputStream();

# bitmap.compress(Bitmap.CompressFormat.JPEG, 70, outputStream);

# byte[] imageBytes = outputStream.toByteArray();

# String base64 = Base64.encodeToString(imageBytes, Base64.DEFAULT); // Кодируем изображение в Base64

# carData.put(keyName, base64); // Добавляем в коллекцию Firestore

# } catch (Exception e) {

# e.printStackTrace();

# carData.put(keyName, "");

# }

# } else {

# carData.put(keyName, "");

# }

# }

# db.collection("cars")

# .add(carData)

# .addOnSuccessListener(documentReference -> {

# Toast.makeText(this, "Объявление добавлено и ожидает модерации!", Toast.LENGTH\_SHORT).show();

# Intent intent = new Intent(CreateListingActivity.this, AllActivity.class);

# startActivity(intent);

# finish();

# })

# .addOnFailureListener(e -> {

# Toast.makeText(this, "Ошибка: " + e.getMessage(), Toast.LENGTH\_SHORT).show();

# });

# }

# private int parseIntSafe(String value) {

# try {

# return Integer.parseInt(value);

# } catch (NumberFormatException e) {

# return 0;

# }

# }

# }

1. CreatePostActivity.java;

# package com.example.drivee;

# import android.os.Bundle;

# import android.text.TextUtils;

# import android.widget.ArrayAdapter;

# import android.widget.Button;

# import android.widget.EditText;

# import android.widget.Spinner;

# import android.widget.Toast;

# import androidx.annotation.Nullable;

# import androidx.appcompat.app.AppCompatActivity;

# import com.google.firebase.auth.FirebaseAuth;

# import com.google.firebase.firestore.FieldValue;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.util.ArrayList;

# import java.util.HashMap;

# import java.util.List;

# import java.util.Map;

# public class CreatePostActivity extends AppCompatActivity {

# private Spinner brandSpinner, modelSpinner;

# private EditText titleEditText, contentEditText, mileageEditText;

# private Button publishButton;

# private FirebaseFirestore db;

# private FirebaseAuth auth;

# private Map<String, List<String>> brandModelMap = new HashMap<>();

# @Override

# protected void onCreate(@Nullable Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.create\_post\_activity);

# brandSpinner = findViewById(R.id.brandSpinner);

# modelSpinner = findViewById(R.id.modelSpinner);

# titleEditText = findViewById(R.id.titleEditText);

# contentEditText = findViewById(R.id.contentEditText);

# mileageEditText = findViewById(R.id.mileageEditText);

# publishButton = findViewById(R.id.publishButton);

# db = FirebaseFirestore.getInstance();

# auth = FirebaseAuth.getInstance();

# loadBrands();

# brandSpinner.setOnItemSelectedListener(new android.widget.AdapterView.OnItemSelectedListener() {

# @Override public void onItemSelected(android.widget.AdapterView<?> parent, android.view.View view, int pos, long id) {

# String selectedBrand = brandSpinner.getSelectedItem().toString();

# loadModelsForBrand(selectedBrand);

# }

# @Override public void onNothingSelected(android.widget.AdapterView<?> parent) {}

# });

# publishButton.setOnClickListener(v -> publishPost());

# }

# private void loadBrands() {

# db.collection("dfd").document("models").get().addOnSuccessListener(documentSnapshot -> {

# if (documentSnapshot.exists()) {

# Map<String, Object> data = documentSnapshot.getData();

# if (data != null) {

# List<String> brands = new ArrayList<>();

# for (String key : data.keySet()) {

# brands.add(key);

# brandModelMap.put(key, (List<String>) data.get(key));

# }

# ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple\_spinner\_item, brands);

# adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

# brandSpinner.setAdapter(adapter);

# }

# }

# });

# }

# private void loadModelsForBrand(String brand) {

# List<String> models = brandModelMap.get(brand);

# if (models != null) {

# ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple\_spinner\_item, models);

# adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

# modelSpinner.setAdapter(adapter);

# }

# }

# private void publishPost() {

# String brand = brandSpinner.getSelectedItem().toString();

# String model = modelSpinner.getSelectedItem().toString();

# String title = titleEditText.getText().toString().trim();

# String content = contentEditText.getText().toString().trim();

# String mileageStr = mileageEditText.getText().toString().trim();

# if (TextUtils.isEmpty(brand) || TextUtils.isEmpty(model) || TextUtils.isEmpty(title)

# || TextUtils.isEmpty(content) || TextUtils.isEmpty(mileageStr)) {

# Toast.makeText(this, "Пожалуйста, заполните все поля", Toast.LENGTH\_SHORT).show();

# return;

# }

# long mileage;

# try {

# mileage = Long.parseLong(mileageStr);

# } catch (NumberFormatException e) {

# Toast.makeText(this, "Некорректный пробег", Toast.LENGTH\_SHORT).show();

# return;

# }

# String userId = auth.getCurrentUser().getUid();

# Map<String, Object> postMap = new HashMap<>();

# postMap.put("brand", brand);

# postMap.put("model", model);

# postMap.put("title", title);

# postMap.put("content", content);

# postMap.put("mileage", mileage);

# postMap.put("userId", userId);

# postMap.put("status", "pending");

# postMap.put("timestamp", FieldValue.serverTimestamp());

# db.collection("bortjornal")

# .add(postMap)

# .addOnSuccessListener(documentReference -> {

# Toast.makeText(this, "Запись отправлена на модерацию", Toast.LENGTH\_SHORT).show();

# finish();

# })

# .addOnFailureListener(e -> {

# Toast.makeText(this, "Ошибка при сохранении", Toast.LENGTH\_SHORT).show();

# });

# }

# }

1. FavoritesActivity.java;

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.util.Log;

# import android.view.View;

# import android.widget.TextView;

# import androidx.appcompat.app.AppCompatActivity;

# import androidx.recyclerview.widget.LinearLayoutManager;

# import androidx.recyclerview.widget.RecyclerView;

# import com.google.firebase.auth.FirebaseAuth;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.util.ArrayList;

# import java.util.List;

# public class FavoritesActivity extends AppCompatActivity {

# private RecyclerView recyclerView;

# private CarAdapter carAdapter;

# private List<Car> favoriteCars;

# private FirebaseFirestore db;

# private FirebaseAuth auth;

# private View progressBar;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_favorites);

# recyclerView = findViewById(R.id.recyclerViewFavorites);

# progressBar = findViewById(R.id.progressBar);

# recyclerView.setLayoutManager(new LinearLayoutManager(this));

# favoriteCars = new ArrayList<>();

# carAdapter = new CarAdapter(this, favoriteCars);

# recyclerView.setAdapter(carAdapter);

# db = FirebaseFirestore.getInstance();

# auth = FirebaseAuth.getInstance();

# TextView backButton = findViewById(R.id.backButton);

# backButton.setOnClickListener(v -> {

# startActivity(new Intent(FavoritesActivity.this, AllActivity.class));

# finish();

# });

# loadFavoriteCars();

# }

# private void loadFavoriteCars() {

# progressBar.setVisibility(View.VISIBLE);

# String userId = auth.getCurrentUser().getUid();

# db.collection("users").document(userId).get()

# .addOnSuccessListener(documentSnapshot -> {

# if (documentSnapshot.exists() && documentSnapshot.contains("favorites")) {

# List<String> favoriteCarIds = (List<String>) documentSnapshot.get("favorites");

# if (favoriteCarIds != null && !favoriteCarIds.isEmpty()) {

# fetchCarsOneByOne(favoriteCarIds);

# } else {

# progressBar.setVisibility(View.GONE);

# }

# } else {

# progressBar.setVisibility(View.GONE);

# }

# })

# .addOnFailureListener(e -> {

# Log.e("FavoritesActivity", "Ошибка загрузки избранного", e);

# progressBar.setVisibility(View.GONE);

# });

# }

# private void fetchCarsOneByOne(List<String> carIds) {

# favoriteCars.clear();

# for (String carId : carIds) {

# db.collection("cars").document(carId).get()

# .addOnSuccessListener(documentSnapshot -> {

# if (documentSnapshot.exists()) {

# long mileage = documentSnapshot.getLong("mileage") != null ? documentSnapshot.getLong("mileage") : 0;

# Car car = new Car(

# documentSnapshot.getId(),

# documentSnapshot.getString("brand"),

# documentSnapshot.getString("model"),

# documentSnapshot.getString("photoone"),

# mileage,

# documentSnapshot.getLong("price")

# );

# car.setFavorite(true);

# favoriteCars.add(car);

# carAdapter.notifyDataSetChanged();

# }

# progressBar.setVisibility(View.GONE);

# })

# .addOnFailureListener(e -> {

# Log.e("FavoritesActivity", "Ошибка загрузки машины", e);

# progressBar.setVisibility(View.GONE);

# });

# }

# }

# }

1. FilterActivity.java;

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.text.TextUtils;

# import android.widget.ArrayAdapter;

# import android.widget.Button;

# import android.widget.EditText;

# import android.widget.Spinner;

# import androidx.appcompat.app.AppCompatActivity;

# public class FilterActivity extends AppCompatActivity {

# private EditText yearFrom, yearTo, volumeFrom, volumeTo, powerFrom, powerTo, ownersTo, priceFrom, priceTo, cityInput;

# private Spinner colorSpinner, bodySpinner, driveSpinner, fuelSpinner, transmissionSpinner;

# private Button applyButton, cancelButton;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_filter);

# yearFrom = findViewById(R.id.yearFrom);

# yearTo = findViewById(R.id.yearTo);

# volumeFrom = findViewById(R.id.volumeFrom);

# volumeTo = findViewById(R.id.volumeTo);

# powerFrom = findViewById(R.id.powerFrom);

# powerTo = findViewById(R.id.powerTo);

# ownersTo = findViewById(R.id.ownersTo);

# priceFrom = findViewById(R.id.priceFrom);

# priceTo = findViewById(R.id.priceTo);

# cityInput = findViewById(R.id.cityInput);

# colorSpinner = findViewById(R.id.colorSpinner);

# bodySpinner = findViewById(R.id.bodySpinner);

# driveSpinner = findViewById(R.id.driveSpinner);

# fuelSpinner = findViewById(R.id.fuelSpinner);

# transmissionSpinner = findViewById(R.id.transmissionSpinner);

# applyButton = findViewById(R.id.applyButton);

# cancelButton = findViewById(R.id.cancelButton);

# setupSpinners();

# applyButton.setOnClickListener(v -> returnFilters());

# cancelButton.setOnClickListener(v -> finish());

# }

# private void setupSpinners() {

# String[] fuels = {"Топливо", "Бензин", "Дизель", "Гибрид", "Электро", "ГБО", "Метан"};

# String[] bodies = {"Тип кузова", "Купе", "Седан", "Хэтчбек", "Кроссовер", "Внедорожник"};

# String[] drives = {"Привод", "Полный", "Передний", "Задний"};

# String[] transmissions = {"Коробка", "Автоматическая", "Механика", "Вариатор", "Робот"};

# String[] colors = {"Цвет", "Красный", "Черный", "Белый", "Синий", "Зеленый", "Желтый", "Серый"};

# setupSpinner(colorSpinner, colors);

# setupSpinner(bodySpinner, bodies);

# setupSpinner(driveSpinner, drives);

# setupSpinner(fuelSpinner, fuels);

# setupSpinner(transmissionSpinner, transmissions);

# }

# private void setupSpinner(Spinner spinner, String[] values) {

# ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple\_spinner\_item, values);

# adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

# spinner.setAdapter(adapter);

# }

# private void returnFilters() {

# Intent result = new Intent();

# result.putExtra("yearFrom", getTextOrNull(yearFrom));

# result.putExtra("yearTo", getTextOrNull(yearTo));

# result.putExtra("volumeFrom", getTextOrNull(volumeFrom));

# result.putExtra("volumeTo", getTextOrNull(volumeTo));

# result.putExtra("powerFrom", getTextOrNull(powerFrom));

# result.putExtra("powerTo", getTextOrNull(powerTo));

# result.putExtra("ownersTo", getTextOrNull(ownersTo));

# result.putExtra("priceFrom", getTextOrNull(priceFrom));

# result.putExtra("priceTo", getTextOrNull(priceTo));

# result.putExtra("city", getTextOrNull(cityInput));

# result.putExtra("color", getSpinnerValue(colorSpinner));

# result.putExtra("body", getSpinnerValue(bodySpinner));

# result.putExtra("drive", getSpinnerValue(driveSpinner));

# result.putExtra("fuel", getSpinnerValue(fuelSpinner));

# result.putExtra("transmission", getSpinnerValue(transmissionSpinner));

# setResult(RESULT\_OK, result);

# finish();

# }

# private String getTextOrNull(EditText editText) {

# String text = editText.getText().toString().trim();

# return TextUtils.isEmpty(text) ? null : text;

# }

# private String getSpinnerValue(Spinner spinner) {

# if (spinner.getSelectedItemPosition() == 0) {

# return null;

# }

# return spinner.getSelectedItem().toString();

# }

# }

1. ImagePagerAdapter.java;

# package com.example.drivee;

# import android.content.Context;

# import android.graphics.Bitmap;

# import android.graphics.BitmapFactory;

# import android.util.Base64;

# import android.view.LayoutInflater;

# import android.view.View;

# import android.view.ViewGroup;

# import android.widget.ImageView;

# import androidx.annotation.NonNull;

# import androidx.viewpager2.widget.ViewPager2;

# import androidx.recyclerview.widget.RecyclerView;

# import java.util.List;

# public class ImagePagerAdapter extends RecyclerView.Adapter<ImagePagerAdapter.ImageViewHolder> {

# private final Context context;

# private List<String> photoBase64List;

# public ImagePagerAdapter(Context context, List<String> photoBase64List) {

# this.context = context;

# this.photoBase64List = photoBase64List;

# }

# @NonNull

# @Override

# public ImageViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

# View view = LayoutInflater.from(context).inflate(R.layout.item\_car\_image, parent, false);

# return new ImageViewHolder(view);

# }

# @Override

# public void onBindViewHolder(@NonNull ImageViewHolder holder, int position) {

# String base64Image = photoBase64List.get(position);

# if (!base64Image.isEmpty()) {

# byte[] decodedString = Base64.decode(base64Image, Base64.DEFAULT);

# Bitmap decodedByte = BitmapFactory.decodeByteArray(decodedString, 0, decodedString.length);

# holder.imageView.setImageBitmap(decodedByte);

# } else {

# holder.imageView.setImageResource(R.drawable.logo);

# }

# }

# @Override

# public int getItemCount() {

# return photoBase64List.size();

# }

# public void updateImages(List<String> newPhotoBase64List) {

# this.photoBase64List = newPhotoBase64List;

# notifyDataSetChanged();

# }

# public static class ImageViewHolder extends RecyclerView.ViewHolder {

# public ImageView imageView;

# public ImageViewHolder(View itemView) {

# super(itemView);

# imageView = itemView.findViewById(R.id.imageView);

# }

# }

# }

1. MainActivity.java;

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.view.View;

# import android.widget.EditText;

# import android.widget.Toast;

# import android.widget.Button;

# import androidx.appcompat.app.AppCompatActivity;

# import com.google.firebase.auth.FirebaseAuth;

# import com.google.firebase.auth.FirebaseUser;

# import com.google.firebase.firestore.DocumentSnapshot;

# import com.google.firebase.firestore.FirebaseFirestore;

# public class MainActivity extends AppCompatActivity {

# private EditText emailEditText, passwordEditText;

# private Button loginButton, registerButton;

# private FirebaseAuth mAuth;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_main);

# Button resetPasswordButton = findViewById(R.id.btnResetPassword);

# resetPasswordButton.setOnClickListener(v -> {

# startActivity(new Intent(MainActivity.this, ResetPasswordActivity.class));

# });

# mAuth = FirebaseAuth.getInstance();

# emailEditText = findViewById(R.id.email);

# passwordEditText = findViewById(R.id.passwordtxt);

# loginButton = findViewById(R.id.btnVhod);

# registerButton = findViewById(R.id.btnReg);

# loginButton.setOnClickListener(new View.OnClickListener() {

# @Override

# public void onClick(View v) {

# loginUser();

# }

# });

# registerButton.setOnClickListener(new View.OnClickListener() {

# @Override

# public void onClick(View v) {

# Intent intent = new Intent(MainActivity.this, RegActivity.class);

# startActivity(intent);

# }

# });

# }

# private void loginUser() {

# String email = emailEditText.getText().toString().trim();

# String password = passwordEditText.getText().toString().trim();

# if (email.isEmpty() || password.isEmpty()) {

# Toast.makeText(MainActivity.this, "Заполните все поля", Toast.LENGTH\_SHORT).show();

# return;

# }

# mAuth.signInWithEmailAndPassword(email, password)

# .addOnCompleteListener(this, task -> {

# if (task.isSuccessful()) {

# FirebaseUser user = mAuth.getCurrentUser();

# if (user != null) {

# FirebaseFirestore db = FirebaseFirestore.getInstance();

# db.collection("users").document(user.getUid()).get()

# .addOnCompleteListener(roleTask -> {

# if (roleTask.isSuccessful() && roleTask.getResult().exists()) {

# DocumentSnapshot document = roleTask.getResult();

# String role = document.getString("role");

# if ("admin".equals(role)) {

# startActivity(new Intent(MainActivity.this, AdminActivity.class));

# } else {

# startActivity(new Intent(MainActivity.this, AllActivity.class));

# }

# finish();

# } else {

# Toast.makeText(MainActivity.this, "Ошибка загрузки роли", Toast.LENGTH\_SHORT).show();

# }

# });

# }

# } else {

# Toast.makeText(MainActivity.this, "Ошибка авторизации: " + task.getException().getMessage(), Toast.LENGTH\_LONG).show();

# }

# });

# }

# }

1. ManageModelsActivity.java;

# package com.example.drivee;

# import android.os.Bundle;

# import android.util.Log;

# import android.view.View;

# import android.widget.\*;

# import androidx.appcompat.app.AppCompatActivity;

# import com.google.firebase.firestore.\*;

# import java.util.\*;

# public class ManageModelsActivity extends AppCompatActivity {

# EditText brandInput, modelInput;

# Button addBrandBtn, addModelBtn;

# Spinner brandSpinner;

# FirebaseFirestore db;

# ArrayAdapter<String> brandAdapter;

# List<String> brandList = new ArrayList<>();

# Map<String, List<String>> allData = new HashMap<>();

# DocumentReference docRef;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_manage\_models);

# brandInput = findViewById(R.id.brandInput);

# modelInput = findViewById(R.id.modelInput);

# addBrandBtn = findViewById(R.id.addBrandBtn);

# addModelBtn = findViewById(R.id.addModelBtn);

# brandSpinner = findViewById(R.id.brandSpinner);

# db = FirebaseFirestore.getInstance();

# docRef = db.collection("dfd").document("models");

# brandAdapter = new ArrayAdapter<>(this, android.R.layout.simple\_spinner\_item, brandList);

# brandAdapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

# brandSpinner.setAdapter(brandAdapter);

# loadBrands();

# addBrandBtn.setOnClickListener(v -> {

# String brand = brandInput.getText().toString().trim();

# if (!brand.isEmpty()) {

# docRef.update(brand, new ArrayList<String>())

# .addOnSuccessListener(unused -> {

# Toast.makeText(this, "Марка добавлена", Toast.LENGTH\_SHORT).show();

# loadBrands();

# })

# .addOnFailureListener(e -> {

# Map<String, Object> newBrand = new HashMap<>();

# newBrand.put(brand, new ArrayList<String>());

# docRef.set(newBrand, SetOptions.merge());

# Toast.makeText(this, "Марка создана", Toast.LENGTH\_SHORT).show();

# loadBrands();

# });

# }

# });

# addModelBtn.setOnClickListener(v -> {

# String model = modelInput.getText().toString().trim();

# String selectedBrand = brandSpinner.getSelectedItem().toString();

# if (!model.isEmpty() && selectedBrand != null) {

# docRef.get().addOnSuccessListener(snapshot -> {

# List<String> models = (List<String>) snapshot.get(selectedBrand);

# if (models == null) models = new ArrayList<>();

# if (!models.contains(model)) {

# models.add(model);

# docRef.update(selectedBrand, models);

# Toast.makeText(this, "Модель добавлена", Toast.LENGTH\_SHORT).show();

# } else {

# Toast.makeText(this, "Модель уже существует", Toast.LENGTH\_SHORT).show();

# }

# });

# }

# });

# }

# private void loadBrands() {

# docRef.get().addOnSuccessListener(documentSnapshot -> {

# if (documentSnapshot.exists()) {

# brandList.clear();

# allData.clear();

# Map<String, Object> data = documentSnapshot.getData();

# for (String brand : data.keySet()) {

# brandList.add(brand);

# allData.put(brand, (List<String>) data.get(brand));

# }

# brandAdapter.notifyDataSetChanged();

# }

# });

# }

# }

1. RegActivity.java;

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.view.View;

# import android.widget.Button;

# import android.widget.EditText;

# import android.widget.Toast;

# import androidx.appcompat.app.AppCompatActivity;

# import com.google.firebase.auth.FirebaseAuth;

# import com.google.firebase.auth.FirebaseUser;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.util.HashMap;

# import java.util.Map;

# public class RegActivity extends AppCompatActivity {

# private EditText emailEditText, passwordEditText, confirmPasswordEditText, nameEditText, phoneEditText, dobEditText;

# private Button registerButton, backToLoginButton;

# private FirebaseAuth mAuth;

# private FirebaseFirestore db;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_reg);

# mAuth = FirebaseAuth.getInstance();

# db = FirebaseFirestore.getInstance();

# emailEditText = findViewById(R.id.email);

# passwordEditText = findViewById(R.id.passwordtxt);

# confirmPasswordEditText = findViewById(R.id.confirmPassword);

# nameEditText = findViewById(R.id.nameEditText);

# phoneEditText = findViewById(R.id.phoneEditText);

# dobEditText = findViewById(R.id.dobEditText);

# registerButton = findViewById(R.id.btnRegister);

# backToLoginButton = findViewById(R.id.btnBackToLogin);

# registerButton.setOnClickListener(new View.OnClickListener() {

# @Override

# public void onClick(View v) {

# registerUser();

# }

# });

# backToLoginButton.setOnClickListener(new View.OnClickListener() {

# @Override

# public void onClick(View v) {

# startActivity(new Intent(RegActivity.this, MainActivity.class));

# finish();

# }

# });

# }

# private void registerUser() {

# String email = emailEditText.getText().toString().trim();

# String password = passwordEditText.getText().toString().trim();

# String confirmPassword = confirmPasswordEditText.getText().toString().trim();

# String name = nameEditText.getText().toString().trim();

# String phone = phoneEditText.getText().toString().trim();

# String dob = dobEditText.getText().toString().trim();

# if (email.isEmpty() || password.isEmpty() || confirmPassword.isEmpty() || name.isEmpty() || phone.isEmpty() || dob.isEmpty()) {

# Toast.makeText(RegActivity.this, "Заполните все поля", Toast.LENGTH\_SHORT).show();

# return;

# }

# if (!password.equals(confirmPassword)) {

# Toast.makeText(RegActivity.this, "Пароли не совпадают", Toast.LENGTH\_SHORT).show();

# return;

# }

# mAuth.createUserWithEmailAndPassword(email, password)

# .addOnCompleteListener(this, task -> {

# if (task.isSuccessful()) {

# FirebaseUser user = mAuth.getCurrentUser();

# saveUserData(user.getUid(), name, email, phone, dob);

# Toast.makeText(RegActivity.this, "Добро пожаловать, " + user.getEmail(), Toast.LENGTH\_SHORT).show();

# startActivity(new Intent(RegActivity.this, CenterActivity.class));

# finish();

# } else {

# Toast.makeText(RegActivity.this, "Ошибка регистрации: " + task.getException().getMessage(), Toast.LENGTH\_LONG).show();

# }

# });

# }

# private void saveUserData(String userId, String name, String email, String phone, String dob) {

# Map<String, Object> user = new HashMap<>();

# user.put("name", name);

# user.put("email", email);

# user.put("phone", phone);

# user.put("role", "user");

# user.put("date\_of\_birth", dob);

# user.put("account\_creation\_date", System.currentTimeMillis());

# db.collection("users")

# .document(userId)

# .set(user)

# .addOnSuccessListener(aVoid -> {

# Toast.makeText(RegActivity.this, "Пользователь зарегистрирован", Toast.LENGTH\_SHORT).show();

# })

# .addOnFailureListener(e -> {

# Toast.makeText(RegActivity.this, "Ошибка сохранения данных", Toast.LENGTH\_SHORT).show();

# });

# }

# }

1. ResetPasswordActivity.java;

# package com.example.drivee;

# import android.os.Bundle;

# import android.view.View;

# import android.widget.Button;

# import android.widget.EditText;

# import android.widget.Toast;

# import androidx.appcompat.app.AppCompatActivity;

# import com.google.firebase.auth.FirebaseAuth;

# public class ResetPasswordActivity extends AppCompatActivity {

# private EditText emailEditText;

# private Button resetPasswordButton;

# private FirebaseAuth mAuth;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_reset\_password);

# emailEditText = findViewById(R.id.emailEditText);

# resetPasswordButton = findViewById(R.id.btnResetPassword);

# mAuth = FirebaseAuth.getInstance();

# resetPasswordButton.setOnClickListener(v -> {

# String email = emailEditText.getText().toString().trim();

# if (email.isEmpty()) {

# Toast.makeText(ResetPasswordActivity.this, "Введите ваш Email", Toast.LENGTH\_SHORT).show();

# return;

# }

# mAuth.sendPasswordResetEmail(email)

# .addOnCompleteListener(task -> {

# if (task.isSuccessful()) {

# Toast.makeText(ResetPasswordActivity.this, "Инструкция по сбросу пароля отправлена на почту", Toast.LENGTH\_LONG).show();

# } else {

# Toast.makeText(ResetPasswordActivity.this, "Ошибка: " + task.getException().getMessage(), Toast.LENGTH\_LONG).show();

# }

# });

# });

# }

# }

1. SelectBrandActivity.java;

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.util.Log;

# import android.view.View;

# import android.widget.ArrayAdapter;

# import android.widget.Button;

# import android.widget.Spinner;

# import android.widget.Toast;

# import androidx.appcompat.app.AppCompatActivity;

# import com.google.firebase.firestore.DocumentSnapshot;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.util.ArrayList;

# import java.util.List;

# import java.util.Map;

# public class SelectBrandActivity extends AppCompatActivity {

# private Spinner brandSpinner;

# private Button nextButton;

# private FirebaseFirestore db;

# private List<String> brandList;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_select\_brand);

# brandSpinner = findViewById(R.id.brandSpinner);

# nextButton = findViewById(R.id.btnNext);

# db = FirebaseFirestore.getInstance();

# brandList = new ArrayList<>();

# loadBrands();

# nextButton.setOnClickListener(v -> {

# String selectedBrand = brandSpinner.getSelectedItem().toString();

# if (selectedBrand.equals("Выберите марку")) {

# Toast.makeText(this, "Пожалуйста, выберите марку", Toast.LENGTH\_SHORT).show();

# } else {

# Intent intent = new Intent(SelectBrandActivity.this, SelectModelActivity.class);

# intent.putExtra("selectedBrand", selectedBrand);

# startActivity(intent);

# }

# });

# }

# private void loadBrands() {

# db.collection("dfd").document("models")

# .get()

# .addOnCompleteListener(task -> {

# if (task.isSuccessful()) {

# DocumentSnapshot document = task.getResult();

# if (document.exists()) {

# Map<String, Object> data = document.getData();

# if (data != null) {

# brandList.add("Выберите марку");

# for (Object value : data.keySet()) { // Используем keySet(), так как марки — это ключи

# brandList.add(value.toString());

# }

# ArrayAdapter<String> adapter = new ArrayAdapter<>(this,

# android.R.layout.simple\_spinner\_item, brandList);

# adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

# brandSpinner.setAdapter(adapter);

# }

# } else {

# Log.e("SelectBrandActivity", "Документ models не найден");

# Toast.makeText(this, "Марки не найдены", Toast.LENGTH\_SHORT).show();

# }

# } else {

# Log.e("SelectBrandActivity", "Ошибка загрузки: ", task.getException());

# Toast.makeText(this, "Ошибка загрузки марок", Toast.LENGTH\_SHORT).show();

# }

# });

# }

# }

1. SelectModelActivity.java;

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.util.Log;

# import android.view.View;

# import android.widget.ArrayAdapter;

# import android.widget.Button;

# import android.widget.Spinner;

# import android.widget.Toast;

# import androidx.appcompat.app.AppCompatActivity;

# import com.google.firebase.firestore.DocumentSnapshot;

# import com.google.firebase.firestore.FirebaseFirestore;

# import java.util.ArrayList;

# import java.util.List;

# import java.util.Map;

# public class SelectModelActivity extends AppCompatActivity {

# private Spinner modelSpinner;

# private Button nextButton;

# private FirebaseFirestore db;

# private List<String> modelList;

# private String selectedBrand;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_select\_model);

# modelSpinner = findViewById(R.id.modelSpinner);

# nextButton = findViewById(R.id.btnNext);

# db = FirebaseFirestore.getInstance();

# modelList = new ArrayList<>();

# selectedBrand = getIntent().getStringExtra("selectedBrand");

# if (selectedBrand == null || selectedBrand.isEmpty()) {

# Toast.makeText(this, "Ошибка: марка не выбрана", Toast.LENGTH\_SHORT).show();

# finish();

# return;

# }

# loadModels();

# nextButton.setOnClickListener(v -> {

# String selectedModel = modelSpinner.getSelectedItem().toString();

# if (selectedModel.equals("Выберите модель")) {

# Toast.makeText(this, "Пожалуйста, выберите модель", Toast.LENGTH\_SHORT).show();

# } else {

# Intent intent = new Intent(SelectModelActivity.this, CreateListingActivity.class);

# intent.putExtra("selectedBrand", selectedBrand);

# intent.putExtra("selectedModel", selectedModel);

# startActivity(intent);

# }

# });

# }

# private void loadModels() {

# db.collection("dfd").document("models")

# .get()

# .addOnCompleteListener(task -> {

# if (task.isSuccessful()) {

# DocumentSnapshot document = task.getResult();

# if (document.exists()) {

# Map<String, Object> data = document.getData();

# if (data != null && data.containsKey(selectedBrand)) {

# List<String> models = (List<String>) data.get(selectedBrand);

# if (models != null && !models.isEmpty()) {

# modelList.add("Выберите модель");

# modelList.addAll(models);

# ArrayAdapter<String> adapter = new ArrayAdapter<>(this,

# android.R.layout.simple\_spinner\_item, modelList);

# adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

# modelSpinner.setAdapter(adapter);

# } else {

# Toast.makeText(this, "Модели для " + selectedBrand + " не найдены", Toast.LENGTH\_SHORT).show();

# }

# } else {

# Log.e("SelectModelActivity", "Марка " + selectedBrand + " не найдена");

# Toast.makeText(this, "Марка не найдена", Toast.LENGTH\_SHORT).show();

# }

# } else {

# Log.e("SelectModelActivity", "Документ models не найден");

# Toast.makeText(this, "Модели не найдены", Toast.LENGTH\_SHORT).show();

# }

# } else {

# Log.e("SelectModelActivity", "Ошибка загрузки: ", task.getException());

# Toast.makeText(this, "Ошибка загрузки моделей", Toast.LENGTH\_SHORT).show();

# }

# });

# }

# }

1. SplashActivity.java;

# package com.example.drivee;

# import android.content.Intent;

# import android.os.Bundle;

# import android.os.Handler;

# import androidx.appcompat.app.AppCompatActivity;

# public class SplashActivity extends AppCompatActivity {

# private static final int SPLASH\_TIME\_OUT = 3000;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_splash);

# new Handler().postDelayed(() -> {

# Intent intent = new Intent(SplashActivity.this, MainActivity.class);

# startActivity(intent);

# finish();

# }, SPLASH\_TIME\_OUT);

# }

# }

1. User.java;

# package com.example.drivee;

# public class User {

# private String name;

# private String email;

# private String phone;

# public User() {

# }

# public User(String name, String email, String phone) {

# this.name = name;

# this.email = email;

# this.phone = phone;

# }

# public String getName() {

# return name;

# }

# public String getEmail() {

# return email;

# }

# public String getPhone() {

# return phone;

# }

# }

1. UserCar.java;

# package com.example.drivee;

# public class UserCar {

# private String id;

# private String brand;

# private String model;

# private String photoUrl;

# private long price;

# private String status;

# public UserCar(String id, String brand, String model, String photoUrl, long price, String status) {

# this.id = id;

# this.brand = brand;

# this.model = model;

# this.photoUrl = photoUrl;

# this.price = price;

# this.status = status;

# }

# public String getId() {

# return id;

# }

# public String getBrand() {

# return brand;

# }

# public String getModel() {

# return model;

# }

# public String getPhotoUrl() {

# return photoUrl;

# }

# public long getPrice() {

# return price;

# }

# public String getStatus() {

# return status;

# }

# public void setStatus(String status) {

# this.status = status;

# }

# }

1. UserProfileActivity.java;

# package com.example.drivee;

# import android.content.Intent;

# import android.graphics.Bitmap;

# import android.graphics.BitmapFactory;

# import android.net.Uri;

# import android.os.Bundle;

# import android.util.Base64;

# import android.util.Log;

# import android.widget.Button;

# import android.widget.ImageView;

# import android.widget.TextView;

# import android.widget.Toast;

# import androidx.appcompat.app.AppCompatActivity;

# import androidx.recyclerview.widget.LinearLayoutManager;

# import androidx.recyclerview.widget.RecyclerView;

# import com.google.firebase.auth.FirebaseAuth;

# import com.google.firebase.firestore.FirebaseFirestore;

# import com.google.firebase.firestore.QueryDocumentSnapshot;

# import java.io.ByteArrayOutputStream;

# import java.io.FileNotFoundException;

# import java.io.InputStream;

# import java.util.ArrayList;

# import java.util.List;

# public class UserProfileActivity extends AppCompatActivity {

# private static final int PICK\_IMAGE\_REQUEST = 1;

# private ImageView profileImage;

# private Button selectImageBtn;

# private TextView userName, userPhone, userEmail;

# private RecyclerView userCarsRecyclerView;

# private List<Car> userCars = new ArrayList<>();

# private CarAdapter carAdapter;

# private FirebaseFirestore db;

# private FirebaseAuth mAuth;

# private Uri imageUri;

# private String userId;

# @Override

# protected void onCreate(Bundle savedInstanceState) {

# super.onCreate(savedInstanceState);

# setContentView(R.layout.activity\_user\_profile);

# profileImage = findViewById(R.id.profileImage);

# selectImageBtn = findViewById(R.id.selectImageBtn);

# userName = findViewById(R.id.userName);

# userPhone = findViewById(R.id.userPhone);

# userEmail = findViewById(R.id.userEmail);

# userCarsRecyclerView = findViewById(R.id.userCarsRecyclerView);

# mAuth = FirebaseAuth.getInstance();

# db = FirebaseFirestore.getInstance();

# userId = mAuth.getCurrentUser().getUid();

# carAdapter = new CarAdapter(this, userCars, false, true);

# userCarsRecyclerView.setLayoutManager(new LinearLayoutManager(this));

# userCarsRecyclerView.setAdapter(carAdapter);

# selectImageBtn.setOnClickListener(v -> openFileChooser());

# Button logoutButton = findViewById(R.id.logoutButton);

# logoutButton.setOnClickListener(v -> {

# FirebaseAuth.getInstance().signOut();

# Intent intent = new Intent(UserProfileActivity.this, MainActivity.class);

# intent.setFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK | Intent.FLAG\_ACTIVITY\_CLEAR\_TASK); // очищаем back stack

# startActivity(intent);

# });

# loadUserProfile();

# loadUserCars();

# }

# private void loadUserProfile() {

# db.collection("users").document(userId)

# .get()

# .addOnSuccessListener(documentSnapshot -> {

# if (documentSnapshot.exists()) {

# userName.setText(documentSnapshot.getString("name"));

# userPhone.setText(documentSnapshot.getString("phone"));

# userEmail.setText(documentSnapshot.getString("email"));

# String profileImageBase64 = documentSnapshot.getString("profile\_image");

# if (profileImageBase64 != null && !profileImageBase64.isEmpty()) {

# Bitmap decodedImage = decodeBase64(profileImageBase64);

# profileImage.setImageBitmap(decodedImage);

# } else {

# profileImage.setImageResource(R.drawable.logo);

# }

# }

# })

# .addOnFailureListener(e -> {

# Toast.makeText(this, "Ошибка загрузки данных", Toast.LENGTH\_SHORT).show();

# });

# }

# private void loadUserCars() {

# db.collection("cars")

# .get()

# .addOnSuccessListener(queryDocumentSnapshots -> {

# userCars.clear();

# for (QueryDocumentSnapshot document : queryDocumentSnapshots) {

# String ownerId = document.getString("ownerId");

# if (ownerId != null && ownerId.equals(userId)) {

# String id = document.getId();

# String brand = document.getString("brand");

# String model = document.getString("model");

# String photo = document.getString("photoone");

# if (photo == null) photo = "";

# long price = document.getLong("price") != null ? document.getLong("price") : 0;

# String status = document.getString("status") != null ? document.getString("status") : "Неизвестно";

# long mileage = document.getLong("mileage") != null ? document.getLong("mileage") : 0;

# Car car = new Car(id, brand, model, photo, mileage, price);

# try {

# java.lang.reflect.Field field = car.getClass().getDeclaredField("status");

# field.setAccessible(true);

# field.set(car, status);

# } catch (Exception e) {

# e.printStackTrace();

# }

# userCars.add(car);

# }

# }

# carAdapter.notifyDataSetChanged();

# })

# .addOnFailureListener(e -> {

# Toast.makeText(this, "Ошибка при загрузке машин", Toast.LENGTH\_SHORT).show();

# });

# }

# private void openFileChooser() {

# Intent intent = new Intent(Intent.ACTION\_PICK);

# intent.setType("image/\*");

# startActivityForResult(intent, PICK\_IMAGE\_REQUEST);

# }

# @Override

# protected void onActivityResult(int requestCode, int resultCode, Intent data) {

# super.onActivityResult(requestCode, resultCode, data);

# if (requestCode == PICK\_IMAGE\_REQUEST && resultCode == RESULT\_OK && data != null && data.getData() != null) {

# imageUri = data.getData();

# try {

# InputStream imageStream = getContentResolver().openInputStream(imageUri);

# Bitmap selectedImage = BitmapFactory.decodeStream(imageStream);

# profileImage.setImageBitmap(selectedImage);

# String imageBase64 = encodeImageToBase64(selectedImage);

# saveImageToFirestore(imageBase64);

# } catch (FileNotFoundException e) {

# e.printStackTrace();

# Toast.makeText(this, "Ошибка при загрузке изображения", Toast.LENGTH\_SHORT).show();

# }

# }

# }

# private String encodeImageToBase64(Bitmap bitmap) {

# ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();

# bitmap.compress(Bitmap.CompressFormat.PNG, 100, byteArrayOutputStream);

# byte[] byteArray = byteArrayOutputStream.toByteArray();

# return Base64.encodeToString(byteArray, Base64.DEFAULT);

# }

# private Bitmap decodeBase64(String base64String) {

# byte[] decodedBytes = Base64.decode(base64String, Base64.DEFAULT);

# return BitmapFactory.decodeByteArray(decodedBytes, 0, decodedBytes.length);

# }

# private void saveImageToFirestore(String imageBase64) {

# db.collection("users").document(userId)

# .update("profile\_image", imageBase64)

# .addOnSuccessListener(aVoid -> {

# Toast.makeText(this, "Изображение обновлено", Toast.LENGTH\_SHORT).show();

# })

# .addOnFailureListener(e -> {

# Toast.makeText(this, "Ошибка при обновлении изображения", Toast.LENGTH\_SHORT).show();

# });

# }

# }

1. circle.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <shape xmlns:android="http://schemas.android.com/apk/res/android"

# android:shape="oval">

# <solid android:color="#DDDDDD" /> <!-- можно заменить на белый или прозрачный -->

# </shape>

1. edittext\_background.xml;

# <shape xmlns:android="http://schemas.android.com/apk/res/android">

# <solid android:color="@android:color/white"/>

# <corners android:radius="8dp"/>

# <stroke

# android:width="1dp"

# android:color="#CCCCCC"/>

# <padding

# android:left="12dp"

# android:top="8dp"

# android:right="12dp"

# android:bottom="8dp"/>

# </shape>

1. ic\_heart\_filled.xml;

# <vector xmlns:android="http://schemas.android.com/apk/res/android"

# android:width="24dp"

# android:height="24dp"

# android:viewportWidth="24"

# android:viewportHeight="24">

# <path

# android:fillColor="#FF0000"

# android:pathData="M12,21.35l-1.45,-1.32C5.4,15.36 2,12.28 2,8.5C2,5.42 4.42,3 7.5,3C9.24,3 10.91,3.81 12,5.09C13.09,3.81 14.76,3 16.5,3C19.58,3 22,5.42 22,8.5C22,12.28 18.6,15.36 13.45,19.68L12,21.35z"/>

# </vector>

1. ic\_heart\_outline.xml;

# <vector xmlns:android="http://schemas.android.com/apk/res/android"

# android:width="24dp"

# android:height="24dp"

# android:viewportWidth="24"

# android:viewportHeight="24">

# <path

# android:fillColor="#808080"

# android:pathData="M12,21.35l-1.45,-1.32C5.4,15.36 2,12.28 2,8.5C2,5.42 4.42,3 7.5,3C9.24,3 10.91,3.81 12,5.09C13.09,3.81 14.76,3 16.5,3C19.58,3 22,5.42 22,8.5C22,12.28 18.6,15.36 13.45,19.68L12,21.35z"/>

# </vector>

1. ic\_launcher\_background.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <vector

# android:height="108dp"

# android:width="108dp"

# android:viewportHeight="108"

# android:viewportWidth="108"

# xmlns:android="http://schemas.android.com/apk/res/android">

# <path android:fillColor="#3DDC84"

# android:pathData="M0,0h108v108h-108z"/>

# <path android:fillColor="#00000000" android:pathData="M9,0L9,108"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M19,0L19,108"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M29,0L29,108"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M39,0L39,108"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M49,0L49,108"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M59,0L59,108"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M69,0L69,108"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M79,0L79,108"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M89,0L89,108"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M99,0L99,108"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M0,9L108,9"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M0,19L108,19"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M0,29L108,29"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M0,39L108,39"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M0,49L108,49"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M0,59L108,59"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M0,69L108,69"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M0,79L108,79"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M0,89L108,89"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M0,99L108,99"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M19,29L89,29"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M19,39L89,39"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M19,49L89,49"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M19,59L89,59"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M19,69L89,69"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M19,79L89,79"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M29,19L29,89"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M39,19L39,89"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M49,19L49,89"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M59,19L59,89"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M69,19L69,89"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# <path android:fillColor="#00000000" android:pathData="M79,19L79,89"

# android:strokeColor="#33FFFFFF" android:strokeWidth="0.8"/>

# </vector>

1. ic\_launcher\_foreground.xml;

# <vector xmlns:android="http://schemas.android.com/apk/res/android"

# xmlns:aapt="http://schemas.android.com/aapt"

# android:width="108dp"

# android:height="108dp"

# android:viewportWidth="108"

# android:viewportHeight="108">

# <path android:pathData="M31,63.928c0,0 6.4,-11 12.1,-13.1c7.2,-2.6 26,-1.4 26,-1.4l38.1,38.1L107,108.928l-32,-1L31,63.928z">

# <aapt:attr name="android:fillColor">

# <gradient

# android:endX="85.84757"

# android:endY="92.4963"

# android:startX="42.9492"

# android:startY="49.59793"

# android:type="linear">

# <item

# android:color="#44000000"

# android:offset="0.0" />

# <item

# android:color="#00000000"

# android:offset="1.0" />

# </gradient>

# </aapt:attr>

# </path>

# <path

# android:fillColor="#FFFFFF"

# android:fillType="nonZero"

# android:pathData="M65.3,45.828l3.8,-6.6c0.2,-0.4 0.1,-0.9 -0.3,-1.1c-0.4,-0.2 -0.9,-0.1 -1.1,0.3l-3.9,6.7c-6.3,-2.8 -13.4,-2.8 -19.7,0l-3.9,-6.7c-0.2,-0.4 -0.7,-0.5 -1.1,-0.3C38.8,38.328 38.7,38.828 38.9,39.228l3.8,6.6C36.2,49.428 31.7,56.028 31,63.928h46C76.3,56.028 71.8,49.428 65.3,45.828zM43.4,57.328c-0.8,0 -1.5,-0.5 -1.8,-1.2c-0.3,-0.7 -0.1,-1.5 0.4,-2.1c0.5,-0.5 1.4,-0.7 2.1,-0.4c0.7,0.3 1.2,1 1.2,1.8C45.3,56.528 44.5,57.328 43.4,57.328L43.4,57.328zM64.6,57.328c-0.8,0 -1.5,-0.5 -1.8,-1.2s-0.1,-1.5 0.4,-2.1c0.5,-0.5 1.4,-0.7 2.1,-0.4c0.7,0.3 1.2,1 1.2,1.8C66.5,56.528 65.6,57.328 64.6,57.328L64.6,57.328z"

# android:strokeWidth="1"

# android:strokeColor="#00000000" />

# </vector>

1. spinner\_background.xml;

# <shape xmlns:android="http://schemas.android.com/apk/res/android"

# android:shape="rectangle">

# <solid android:color="@android:color/white"/>

# <corners android:radius="12dp"/>

# <stroke android:width="1dp" android:color="#DDDDDD"/>

# </shape>

1. splash\_background.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <layer-list xmlns:android="http://schemas.android.com/apk/res/android">

# <item android:drawable="@color/white" />

# </layer-list>

1. activity\_admin.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <androidx.constraintlayout.widget.ConstraintLayout

# xmlns:android="http://schemas.android.com/apk/res/android"

# xmlns:app="http://schemas.android.com/apk/res-auto"

# xmlns:tools="http://schemas.android.com/tools"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# tools:context=".AdminActivity">

# <TextView

# android:id="@+id/adminTitleText"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Администрирование"

# android:textSize="22sp"

# android:textStyle="bold"

# android:textColor="#000000"

# android:layout\_marginTop="16dp"

# app:layout\_constraintTop\_toTopOf="parent"

# app:layout\_constraintStart\_toStartOf="parent"

# app:layout\_constraintEnd\_toEndOf="parent"

# android:gravity="center" />

# <androidx.recyclerview.widget.RecyclerView

# android:id="@+id/recyclerView"

# android:layout\_width="0dp"

# android:layout\_height="0dp"

# android:layout\_marginTop="12dp"

# app:layout\_constraintTop\_toBottomOf="@+id/adminTitleText"

# app:layout\_constraintBottom\_toTopOf="@+id/bottomNavigationView"

# app:layout\_constraintStart\_toStartOf="parent"

# app:layout\_constraintEnd\_toEndOf="parent" />

# <com.google.android.material.bottomnavigation.BottomNavigationView

# android:id="@+id/bottomNavigationView"

# android:layout\_width="0dp"

# android:layout\_height="wrap\_content"

# android:background="?android:attr/windowBackground"

# app:menu="@menu/bottom\_nav\_menu"

# app:layout\_constraintBottom\_toBottomOf="parent"

# app:layout\_constraintStart\_toStartOf="parent"

# app:layout\_constraintEnd\_toEndOf="parent" />

# </androidx.constraintlayout.widget.ConstraintLayout>

1. activity\_admin\_bortjournal.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:orientation="vertical"

# android:padding="16dp"

# android:background="#F5F5F5">

# <TextView

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Модерация записей"

# android:textSize="20sp"

# android:textStyle="bold"

# android:textColor="#000"

# android:gravity="center"

# android:paddingBottom="12dp" />

# <androidx.recyclerview.widget.RecyclerView

# android:id="@+id/adminRecyclerView"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:clipToPadding="false"

# android:paddingBottom="16dp" />

# </LinearLayout>

1. activity\_all.xml;

# <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

# xmlns:app="http://schemas.android.com/apk/res-auto"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:orientation="vertical"

# android:background="#F5F5F5">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="50dp"

# android:background="#FFFFFF"

# android:orientation="horizontal"

# android:gravity="center\_vertical"

# android:paddingStart="16dp"

# android:paddingEnd="16dp"

# android:elevation="2dp">

# <TextView

# android:id="@+id/adsCountTextView"

# android:layout\_width="0dp"

# android:layout\_height="wrap\_content"

# android:layout\_weight="1"

# android:text="0 объявлений"

# android:textSize="18sp"

# android:textStyle="bold"

# android:gravity="center"

# android:textColor="#000000" />

# <ImageButton

# android:layout\_marginTop="4dp"

# android:id="@+id/filterButton"

# android:layout\_width="40dp"

# android:layout\_height="40dp"

# android:background="?attr/selectableItemBackgroundBorderless"

# android:contentDescription="Фильтр"

# android:src="@drawable/sor"

# android:scaleType="fitCenter"

# android:padding="8dp" />

# </LinearLayout>

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="43dp"

# android:gravity="center\_vertical"

# android:orientation="horizontal"

# android:padding="2dp">

# <Spinner

# android:id="@+id/brandSpinner"

# android:layout\_width="0dp"

# android:layout\_height="wrap\_content"

# android:layout\_weight="1" />

# <View

# android:layout\_width="12dp"

# android:layout\_height="wrap\_content" />

# <Spinner

# android:id="@+id/modelSpinner"

# android:layout\_width="0dp"

# android:layout\_height="wrap\_content"

# android:layout\_weight="1" />

# </LinearLayout>

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:gravity="center"

# android:paddingHorizontal="16dp"

# android:layout\_marginBottom="8dp">

# <Button

# android:id="@+id/resetFiltersButton"

# android:layout\_width="0dp"

# android:layout\_height="wrap\_content"

# android:layout\_weight="1"

# android:text="Сбросить"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"

# android:layout\_marginEnd="8dp"/>

# <Button

# android:id="@+id/paramsButton"

# android:layout\_width="0dp"

# android:layout\_height="wrap\_content"

# android:layout\_weight="1"

# android:text="Параметры"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"/>

# </LinearLayout>

# <androidx.recyclerview.widget.RecyclerView

# android:id="@+id/recyclerView"

# android:layout\_width="match\_parent"

# android:layout\_height="0dp"

# android:layout\_weight="1"

# android:padding="16dp" />

# <com.google.android.material.bottomnavigation.BottomNavigationView

# android:id="@+id/bottomNavigation"

# android:layout\_width="match\_parent"

# android:layout\_height="90dp"

# android:background="@color/white"

# app:menu="@menu/bottom\_navigation\_menu" />

# </LinearLayout>

1. activity\_center.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <android.widget.ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# xmlns:app="http://schemas.android.com/apk/res-auto"

# android:fillViewport="true">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="vertical"

# android:padding="16dp"

# android:background="#F5F5F5">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:layout\_marginTop="50dp"

# android:gravity="left">

# <TextView

# android:id="@+id/brandTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="25sp"

# android:textColor="#000"/>

# <TextView

# android:id="@+id/modelTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="25sp"

# android:layout\_marginStart="8dp"

# android:textColor="#000"/>

# <TextView

# android:id="@+id/editVolume"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="25sp"

# android:layout\_marginStart="8dp"

# android:textColor="#000"/>

# <TextView

# android:id="@+id/yearTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="25sp"

# android:layout\_marginStart="8dp"

# android:textColor="#000"/>

# </LinearLayout>

# <TextView

# android:id="@+id/priceTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="30sp"

# android:textStyle="bold"

# android:layout\_marginTop="4dp"/>

# <androidx.viewpager2.widget.ViewPager2

# android:id="@+id/carImagePager"

# android:layout\_width="match\_parent"

# android:layout\_height="250dp"

# android:layout\_marginTop="12dp"/>

# <TextView

# android:id="@+id/dataPublicTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="12sp"

# android:textColor="#777"

# android:layout\_marginTop="4dp"

# android:layout\_gravity="end"/>

# <TextView

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Характеристики"

# android:textSize="21sp"

# android:textStyle="bold"

# android:textColor="#333"

# android:layout\_marginTop="12dp"/>

# <TextView android:id="@+id/bodyTextView" style="@style/CarSpecItem"/>

# <TextView android:id="@+id/colorTextView" style="@style/CarSpecItem"/>

# <TextView android:id="@+id/driveTextView" style="@style/CarSpecItem"/>

# <TextView android:id="@+id/fuelTextView" style="@style/CarSpecItem"/>

# <TextView android:id="@+id/horsepowerTextView" style="@style/CarSpecItem"/>

# <TextView android:id="@+id/mileageTextView" style="@style/CarSpecItem"/>

# <TextView android:id="@+id/ownersTextView" style="@style/CarSpecItem"/>

# <TextView android:id="@+id/transmissionTextView" style="@style/CarSpecItem"/>

# <TextView android:id="@+id/volumeTextView" style="@style/CarSpecItem"/>

# <TextView android:id="@+id/vinTextView" style="@style/CarSpecItem"/>

# <TextView

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Описание"

# android:textSize="21sp"

# android:textStyle="bold"

# android:textColor="#333"

# android:layout\_marginTop="12dp"/>

# <TextView

# android:id="@+id/descriptionTextView"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:textSize="14sp"

# android:textColor="#555"

# android:layout\_marginTop="4dp"/>

# <LinearLayout

# android:id="@+id/userInfoBlock"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:layout\_marginTop="20dp"

# android:padding="12dp"

# android:background="@android:color/white"

# android:gravity="center\_vertical"

# android:elevation="2dp">

# <com.google.android.material.imageview.ShapeableImageView

# android:id="@+id/userAvatar"

# android:layout\_width="64dp"

# android:layout\_height="64dp"

# android:layout\_marginEnd="16dp"

# android:scaleType="centerCrop"

# android:src="@drawable/user\_prof\_1"

# app:shapeAppearanceOverlay="@style/CircleImage" />

# <LinearLayout

# android:layout\_width="0dp"

# android:layout\_height="wrap\_content"

# android:layout\_weight="1"

# android:orientation="vertical">

# <TextView

# android:id="@+id/userName"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textStyle="bold"

# android:textSize="18sp"

# android:textColor="#000" />

# <TextView

# android:id="@+id/userPhone"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="16sp"

# android:textColor="#555" />

# </LinearLayout>

# </LinearLayout>

# </LinearLayout>

# </android.widget.ScrollView>

1. activity\_create\_listing.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:fillViewport="true"

# android:background="#F5F5F5">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="vertical"

# android:padding="20dp">

# <HorizontalScrollView

# android:layout\_width="match\_parent"

# android:layout\_height="200dp"

# android:layout\_marginBottom="10dp">

# <LinearLayout

# android:orientation="horizontal"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content">

# <LinearLayout

# android:orientation="vertical"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:gravity="center">

# <ImageView

# android:id="@+id/imagePreview1"

# android:layout\_width="200dp"

# android:layout\_height="200dp"

# android:scaleType="centerCrop"

# android:src="@drawable/pusto"

# android:background="@android:color/darker\_gray"

# android:layout\_marginEnd="10dp"/>

# <Button

# android:id="@+id/btnRemovePhoto1"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Сбросить"

# android:layout\_marginTop="5dp"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"/>

# </LinearLayout>

# <LinearLayout

# android:orientation="vertical"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:gravity="center">

# <ImageView

# android:id="@+id/imagePreview2"

# android:layout\_width="200dp"

# android:layout\_height="200dp"

# android:scaleType="centerCrop"

# android:src="@drawable/pusto"

# android:background="@android:color/darker\_gray"

# android:layout\_marginEnd="10dp"/>

# <Button

# android:id="@+id/btnRemovePhoto2"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Сбросить"

# android:layout\_marginTop="5dp"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"/>

# </LinearLayout>

# <LinearLayout

# android:orientation="vertical"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:gravity="center">

# <ImageView

# android:id="@+id/imagePreview3"

# android:layout\_width="200dp"

# android:layout\_height="200dp"

# android:scaleType="centerCrop"

# android:src="@drawable/pusto"

# android:background="@android:color/darker\_gray"

# android:layout\_marginEnd="10dp"/>

# <Button

# android:id="@+id/btnRemovePhoto3"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Сбросить"

# android:layout\_marginTop="5dp"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"/>

# </LinearLayout>

# <LinearLayout

# android:orientation="vertical"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:gravity="center">

# <ImageView

# android:id="@+id/imagePreview4"

# android:layout\_width="200dp"

# android:layout\_height="200dp"

# android:scaleType="centerCrop"

# android:src="@drawable/pusto"

# android:background="@android:color/darker\_gray"

# android:layout\_marginEnd="10dp"/>

# <Button

# android:id="@+id/btnRemovePhoto4"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Сбросить"

# android:layout\_marginTop="5dp"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"/>

# </LinearLayout>

# <LinearLayout

# android:orientation="vertical"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:gravity="center">

# <ImageView

# android:id="@+id/imagePreview5"

# android:layout\_width="200dp"

# android:layout\_height="200dp"

# android:scaleType="centerCrop"

# android:src="@drawable/pusto"

# android:background="@android:color/darker\_gray"

# android:layout\_marginEnd="10dp"/>

# </LinearLayout>

# </LinearLayout>

# </HorizontalScrollView>

# <Button

# android:id="@+id/btnUploadPhoto"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Загрузить фото"

# android:backgroundTint="@color/black"

# android:textColor="@android:color/white"/>

# <Button

# android:id="@+id/btnRemovePhoto5"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Сбросить"

# android:layout\_marginTop="5dp"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"/>

# <EditText

# android:id="@+id/editVin"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="VIN"

# android:inputType="text"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/editBrand"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Марка"

# android:inputType="text"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/editModel"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Модель"

# android:inputType="text"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/editYear"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Год выпуска"

# android:inputType="number"

# android:layout\_marginTop="10dp"/>

# <Spinner

# android:id="@+id/spinnerFuel"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginTop="10dp" />

# <EditText

# android:id="@+id/editColor"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Цвет"

# android:inputType="text"

# android:layout\_marginTop="10dp"/>

# <Spinner

# android:id="@+id/spinnerBody"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginTop="10dp"/>

# <Spinner

# android:id="@+id/spinnerDrive"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginTop="10dp"/>

# <Spinner

# android:id="@+id/spinnerTransmission"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/editVolume"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Объем двигателя (л)"

# android:inputType="numberDecimal"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/editHorsepower"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Мощность (л.с.)"

# android:inputType="number"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/editMileage"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Пробег (км)"

# android:inputType="number"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/editOwners"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Количество владельцев"

# android:inputType="number"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/editPrice"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Цена (₽)"

# android:inputType="number"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/editDescription"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Описание"

# android:inputType="textMultiLine"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/editCity"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Город"

# android:inputType="text"

# android:layout\_marginTop="10dp"/>

# <Button

# android:id="@+id/btnCreateListing"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Создать объявление"

# android:layout\_marginTop="20dp"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"/>

# </LinearLayout>

# </ScrollView>

1. activity\_favorites.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:orientation="vertical"

# android:padding="16dp"

# android:background="#F5F5F5">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:gravity="center\_vertical">

# <TextView

# android:id="@+id/backButton"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="←"

# android:textSize="24sp"

# android:textColor="@android:color/black"

# android:paddingEnd="12dp" />

# <TextView

# android:id="@+id/titleText"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Избранное"

# android:textStyle="bold"

# android:textSize="20sp"

# android:textColor="#000000" />

# </LinearLayout>

# <androidx.recyclerview.widget.RecyclerView

# android:id="@+id/recyclerViewFavorites"

# android:layout\_width="match\_parent"

# android:layout\_height="0dp"

# android:layout\_weight="1"

# android:layout\_marginTop="16dp" />

# <ProgressBar

# android:id="@+id/progressBar"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:layout\_gravity="center"

# android:visibility="gone" />

# </LinearLayout>

1. activity\_filter.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:fillViewport="true"

# android:background="#F5F5F5">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:padding="20dp"

# android:orientation="vertical">

# <TextView

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Параметры"

# android:textSize="24sp"

# android:textStyle="bold"

# android:textColor="#000000"

# android:gravity="center"

# android:layout\_marginBottom="20dp"/>

# <LinearLayout android:orientation="horizontal"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginBottom="12dp">

# <EditText

# android:id="@+id/yearFrom"

# android:layout\_width="0dp"

# android:layout\_weight="1"

# android:layout\_height="48dp"

# android:hint="Год от"

# android:inputType="number"

# android:background="@drawable/edittext\_background"

# android:padding="12dp"/>

# <View android:layout\_width="12dp" android:layout\_height="match\_parent"/>

# <EditText

# android:id="@+id/yearTo"

# android:layout\_width="0dp"

# android:layout\_weight="1"

# android:layout\_height="48dp"

# android:hint="Год до"

# android:inputType="number"

# android:background="@drawable/edittext\_background"

# android:padding="12dp"/>

# </LinearLayout>

# <Spinner

# android:id="@+id/colorSpinner"

# android:layout\_width="match\_parent"

# android:layout\_height="48dp"

# android:layout\_marginBottom="12dp"

# android:background="@drawable/spinner\_background"

# android:paddingStart="12dp"

# android:paddingEnd="12dp" />

# <Spinner

# android:id="@+id/bodySpinner"

# android:layout\_width="match\_parent"

# android:layout\_height="48dp"

# android:layout\_marginBottom="12dp"

# android:background="@drawable/spinner\_background"

# android:paddingStart="12dp"

# android:paddingEnd="12dp" />

# <Spinner

# android:id="@+id/driveSpinner"

# android:layout\_width="match\_parent"

# android:layout\_height="48dp"

# android:layout\_marginBottom="12dp"

# android:background="@drawable/spinner\_background"

# android:paddingStart="12dp"

# android:paddingEnd="12dp" />

# <Spinner

# android:id="@+id/fuelSpinner"

# android:layout\_width="match\_parent"

# android:layout\_height="48dp"

# android:layout\_marginBottom="12dp"

# android:background="@drawable/spinner\_background"

# android:paddingStart="12dp"

# android:paddingEnd="12dp" />

# <Spinner

# android:id="@+id/transmissionSpinner"

# android:layout\_width="match\_parent"

# android:layout\_height="48dp"

# android:layout\_marginBottom="12dp"

# android:background="@drawable/spinner\_background"

# android:paddingStart="12dp"

# android:paddingEnd="12dp" />

# <LinearLayout android:orientation="horizontal"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginBottom="12dp">

# <EditText

# android:id="@+id/volumeFrom"

# android:hint="Объем от"

# android:inputType="numberDecimal"

# android:layout\_weight="1"

# android:layout\_width="0dp"

# android:layout\_height="48dp"

# android:background="@drawable/edittext\_background"

# android:padding="12dp"/>

# <View android:layout\_width="12dp" android:layout\_height="match\_parent"/>

# <EditText

# android:id="@+id/volumeTo"

# android:hint="Объем до"

# android:inputType="numberDecimal"

# android:layout\_weight="1"

# android:layout\_width="0dp"

# android:layout\_height="48dp"

# android:background="@drawable/edittext\_background"

# android:padding="12dp"/>

# </LinearLayout>

# <LinearLayout android:orientation="horizontal"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginBottom="12dp">

# <EditText

# android:id="@+id/powerFrom"

# android:hint="Мощность от"

# android:inputType="number"

# android:layout\_weight="1"

# android:layout\_width="0dp"

# android:layout\_height="48dp"

# android:background="@drawable/edittext\_background"

# android:padding="12dp"/>

# <View android:layout\_width="12dp" android:layout\_height="match\_parent"/>

# <EditText

# android:id="@+id/powerTo"

# android:hint="до"

# android:inputType="number"

# android:layout\_weight="1"

# android:layout\_width="0dp"

# android:layout\_height="48dp"

# android:background="@drawable/edittext\_background"

# android:padding="12dp"/>

# </LinearLayout>

# <EditText

# android:id="@+id/ownersTo"

# android:layout\_width="match\_parent"

# android:layout\_height="48dp"

# android:hint="Владельцев до"

# android:inputType="number"

# android:background="@drawable/edittext\_background"

# android:padding="12dp"

# android:layout\_marginBottom="12dp"/>

# <LinearLayout android:orientation="horizontal"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginBottom="12dp">

# <EditText

# android:id="@+id/priceFrom"

# android:hint="Цена от"

# android:inputType="number"

# android:layout\_weight="1"

# android:layout\_width="0dp"

# android:layout\_height="48dp"

# android:background="@drawable/edittext\_background"

# android:padding="12dp"/>

# <View android:layout\_width="12dp" android:layout\_height="match\_parent"/>

# <EditText

# android:id="@+id/priceTo"

# android:hint="до"

# android:inputType="number"

# android:layout\_weight="1"

# android:layout\_width="0dp"

# android:layout\_height="48dp"

# android:background="@drawable/edittext\_background"

# android:padding="12dp"/>

# </LinearLayout>

# <EditText

# android:id="@+id/cityInput"

# android:layout\_width="match\_parent"

# android:layout\_height="48dp"

# android:hint="Город"

# android:inputType="text"

# android:background="@drawable/edittext\_background"

# android:padding="12dp"

# android:layout\_marginBottom="12dp"/>

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:gravity="center">

# <Button

# android:id="@+id/applyButton"

# android:layout\_width="0dp"

# android:layout\_weight="1"

# android:layout\_height="wrap\_content"

# android:text="Применить"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"/>

# <View android:layout\_width="12dp" android:layout\_height="match\_parent"/>

# <Button

# android:id="@+id/cancelButton"

# android:layout\_width="0dp"

# android:layout\_weight="1"

# android:layout\_height="wrap\_content"

# android:text="Отмена"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"/>

# </LinearLayout>

# </LinearLayout>

# </ScrollView>

1. activity\_main.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <LinearLayout

# xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:orientation="vertical"

# android:gravity="center"

# android:padding="20dp"

# android:background="@color/white">

# <ImageView

# android:layout\_width="300dp"

# android:layout\_height="100dp"

# android:contentDescription="Логотип компании"

# android:layout\_marginBottom="60dp"

# android:src="@drawable/logo" />

# <TextView

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Вход"

# android:textSize="24sp"

# android:textStyle="bold"

# android:layout\_marginTop="20dp"/>

# <EditText

# android:id="@+id/email"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Email"

# android:inputType="textEmailAddress"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/passwordtxt"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Пароль"

# android:inputType="textPassword"

# android:layout\_marginTop="10dp"/>

# <Button

# android:id="@+id/btnVhod"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Войти"

# android:height="50dp"

# android:layout\_marginTop="50dp"

# android:backgroundTint="@color/red"/>

# <Button

# android:id="@+id/btnReg"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Зарегистрироваться"

# android:height="50dp"

# android:layout\_marginTop="10dp"

# android:layout\_marginBottom="30dp"

# android:backgroundTint="@color/red"/>

# <Button

# android:id="@+id/btnResetPassword"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Сбросить пароль"

# android:height="50dp"

# android:textColor="@color/black"

# android:backgroundTint="@color/white"/>

# </LinearLayout>

1. activity\_manage\_models.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:fillViewport="true"

# android:padding="24dp">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:orientation="vertical"

# android:gravity="center\_horizontal|center\_vertical">

# <TextView

# android:id="@+id/titleText"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Марки и модели"

# android:textSize="24sp"

# android:textStyle="bold"

# android:layout\_marginBottom="32dp" />

# <EditText

# android:id="@+id/brandInput"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Новая марка"

# android:padding="12dp"

# android:background="@drawable/edittext\_background" />

# <Button

# android:id="@+id/addBrandBtn"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginTop="12dp"

# android:text="Добавить марку"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white" />

# <View

# android:layout\_width="match\_parent"

# android:layout\_height="24dp" />

# <Spinner

# android:id="@+id/brandSpinner"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content" />

# <EditText

# android:id="@+id/modelInput"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginTop="16dp"

# android:hint="Новая модель"

# android:padding="12dp"

# android:background="@drawable/edittext\_background" />

# <Button

# android:id="@+id/addModelBtn"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginTop="12dp"

# android:text="Добавить модель"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white" />

# </LinearLayout>

# </ScrollView>

1. activity\_reg.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <LinearLayout

# xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:orientation="vertical"

# android:gravity="center"

# android:padding="20dp"

# android:background="@color/white">

# <ImageView

# android:layout\_width="300dp"

# android:layout\_height="100dp"

# android:contentDescription="Логотип компании"

# android:layout\_marginBottom="20dp"

# android:src="@drawable/logo" />

# <TextView

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Регистрация"

# android:textSize="24sp"

# android:textStyle="bold"

# android:layout\_marginTop="0dp"/>

# <EditText

# android:id="@+id/nameEditText"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Имя" />

# <EditText

# android:id="@+id/dobEditText"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Дата рождения (дд/мм/гггг)"

# android:inputType="date"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/phoneEditText"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Номер телефона"

# android:inputType="phone" />

# <EditText

# android:id="@+id/email"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Email"

# android:inputType="textEmailAddress"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/passwordtxt"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Пароль"

# android:inputType="textPassword"

# android:layout\_marginTop="10dp"/>

# <EditText

# android:id="@+id/confirmPassword"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Подтвердите пароль"

# android:inputType="textPassword"

# android:layout\_marginTop="10dp"/>

# <Button

# android:id="@+id/btnRegister"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Зарегистрироваться"

# android:height="50dp"

# android:layout\_marginTop="40dp"

# android:backgroundTint="@color/red"/>

# <Button

# android:id="@+id/btnBackToLogin"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Уже есть аккаунт? Войти"

# android:height="50dp"

# android:layout\_marginTop="10dp"

# android:layout\_marginBottom="30dp"

# android:backgroundTint="@color/red"/>

# </LinearLayout>

1. activity\_reset\_password.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:orientation="vertical"

# android:gravity="center"

# android:padding="20dp"

# android:background="@color/white">

# <TextView

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Сброс пароля"

# android:textSize="24sp"

# android:textStyle="bold"

# android:layout\_marginTop="20dp"

# android:layout\_marginBottom="40dp"/>

# <EditText

# android:id="@+id/emailEditText"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Введите ваш Email"

# android:inputType="textEmailAddress"

# android:padding="10dp"

# android:layout\_marginBottom="20dp"/>

# <Button

# android:id="@+id/btnResetPassword"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Сбросить пароль"

# android:textColor="@color/white"

# android:backgroundTint="@color/red"

# android:height="50dp"

# android:layout\_marginTop="50dp"/>

# <Button

# android:id="@+id/btnBackToLogin"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Помните пароль? Войти"

# android:height="50dp"

# android:layout\_marginTop="10dp"

# android:backgroundTint="@color/red"/>

# </LinearLayout>

1. activity\_select\_brand.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:orientation="vertical"

# android:gravity="center"

# android:padding="20dp"

# android:background="@color/white">

# <TextView

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Выберите марку автомобиля"

# android:textSize="24sp"

# android:textStyle="bold"

# android:textColor="@android:color/black"

# android:layout\_marginBottom="20dp"/>

# <Spinner

# android:id="@+id/brandSpinner"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginBottom="20dp"/>

# <Button

# android:id="@+id/btnNext"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Далее"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"

# android:height="50dp"/>

# </LinearLayout>

1. activity\_select\_model.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:orientation="vertical"

# android:gravity="center"

# android:padding="20dp"

# android:background="@color/white">

# <TextView

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Выберите модель автомобиля"

# android:textSize="24sp"

# android:textStyle="bold"

# android:textColor="@android:color/black"

# android:layout\_marginBottom="20dp"/>

# <Spinner

# android:id="@+id/modelSpinner"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginBottom="20dp"/>

# <Button

# android:id="@+id/btnNext"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Далее"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"

# android:height="50dp"/>

# </LinearLayout>

1. activity\_splash.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:gravity="center"

# android:orientation="vertical"

# android:background="@color/white">

# <ImageView

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:contentDescription="App Logo"

# android:src="@drawable/logo" />

# </LinearLayout>

1. activity\_user\_profile.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <LinearLayout

# xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:orientation="vertical"

# android:gravity="center"

# android:padding="20dp"

# android:background="@color/white">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:gravity="center\_vertical"

# android:layout\_marginTop="30dp"

# android:layout\_marginBottom="20dp">

# <ImageView

# android:id="@+id/profileImage"

# android:layout\_width="100dp"

# android:layout\_height="100dp"

# android:layout\_marginEnd="16dp"

# android:scaleType="centerCrop"

# android:src="@drawable/user\_prof\_1"

# android:background="@drawable/circle"

# android:clipToOutline="true" />

# <LinearLayout

# android:layout\_width="0dp"

# android:layout\_height="wrap\_content"

# android:orientation="vertical"

# android:layout\_weight="1">

# <TextView

# android:id="@+id/userName"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Имя"

# android:textSize="20sp"

# android:textStyle="bold"

# android:layout\_marginBottom="8dp" />

# <TextView

# android:id="@+id/userPhone"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Телефон"

# android:textSize="18sp"

# android:layout\_marginBottom="8dp" />

# <TextView

# android:id="@+id/userEmail"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Email"

# android:textSize="18sp"

# android:layout\_marginBottom="20dp" />

# </LinearLayout>

# </LinearLayout>

# <!-- Раздел для кнопок -->

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="vertical"

# android:gravity="center"

# android:layout\_marginBottom="20dp">

# <Button

# android:id="@+id/selectImageBtn"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginBottom="1dp"

# android:backgroundTint="@color/red"

# android:text="Выбрать фото"

# android:textSize="14sp"

# android:minHeight="45dp" />

# <Button

# android:id="@+id/logoutButton"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_marginTop="10dp"

# android:backgroundTint="@color/black"

# android:text="Выйти из аккаунта"

# android:textColor="@color/white"

# android:textSize="14sp"

# android:minHeight="45dp" />

# </LinearLayout>

# <TextView

# android:id="@+id/userCarsTitle"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Мои машины"

# android:textSize="18sp"

# android:textStyle="bold"

# android:layout\_marginTop="1dp"

# android:layout\_marginBottom="10dp"/>

# <androidx.recyclerview.widget.RecyclerView

# android:id="@+id/userCarsRecyclerView"

# android:layout\_width="match\_parent"

# android:layout\_height="0dp"

# android:layout\_marginTop="10dp"

# android:layout\_marginBottom="20dp"

# android:layout\_weight="1" />

# </LinearLayout>

1. bortjournal\_activity.xml;

# <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

# android:orientation="vertical" android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:padding="16dp"

# android:background="#F5F5F5">

# <TextView

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Бортовой журнал"

# android:textSize="24sp"

# android:textStyle="bold"

# android:textColor="#000000"

# android:gravity="center"

# android:layout\_marginBottom="20dp"/>

# <Button

# android:id="@+id/createPostButton"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Создать запись"

# android:backgroundTint="@color/red"

# android:textColor="@android:color/white"

# android:layout\_marginBottom="12dp"/>

# <androidx.recyclerview.widget.RecyclerView

# android:id="@+id/postsRecyclerView"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent" />

# </LinearLayout>

1. bortjournal\_detail\_activity.xml;

# <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent" android:layout\_height="match\_parent"

# android:padding="16dp">

# <LinearLayout

# android:orientation="vertical"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content">

# <TextView

# android:id="@+id/detailBrandModel"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Toyota Camry"

# android:textSize="16sp"

# android:textStyle="bold"

# android:textColor="#555555" />

# <TextView

# android:id="@+id/detailTitle"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Заголовок"

# android:textSize="20sp"

# android:textStyle="bold"

# android:textColor="#000000"

# android:layout\_marginTop="8dp" />

# <TextView

# android:id="@+id/detailContent"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:text="Текст"

# android:textSize="16sp"

# android:layout\_marginTop="12dp"

# android:textColor="#333333" />

# </LinearLayout>

# </ScrollView>

1. create\_post\_activity.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:padding="16dp">

# <LinearLayout

# android:orientation="vertical"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content">

# <Spinner

# android:id="@+id/brandSpinner"

# android:layout\_width="match\_parent"

# android:layout\_height="45dp"

# android:layout\_marginBottom="16dp"

# android:padding="12dp"

# android:textAlignment="center"

# android:spinnerMode="dropdown" />

# <Spinner

# android:id="@+id/modelSpinner"

# android:layout\_width="match\_parent"

# android:layout\_height="45dp"

# android:layout\_marginBottom="16dp"

# android:padding="12dp"

# android:textAlignment="center"

# android:spinnerMode="dropdown" />

# <EditText

# android:id="@+id/titleEditText"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Заголовок"

# android:layout\_marginBottom="12dp" />

# <EditText

# android:id="@+id/contentEditText"

# android:layout\_width="match\_parent"

# android:layout\_height="150dp"

# android:gravity="top"

# android:hint="Текст записи"

# android:layout\_marginBottom="12dp" />

# <EditText

# android:id="@+id/mileageEditText"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:hint="Пробег (в км)"

# android:inputType="number"

# android:layout\_marginBottom="20dp" />

# <Button

# android:id="@+id/publishButton"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:backgroundTint="@color/red"

# android:text="Опубликовать" />

# </LinearLayout>

# </ScrollView>

1. item\_admin\_bortjournal.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <androidx.cardview.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"

# xmlns:app="http://schemas.android.com/apk/res-auto"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_margin="12dp"

# app:cardCornerRadius="16dp"

# app:cardElevation="6dp"

# android:backgroundTint="#FFFFFF">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="vertical"

# android:padding="16dp">

# <TextView

# android:id="@+id/adminTitle"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Заголовок поста"

# android:textStyle="bold"

# android:textSize="18sp"

# android:textColor="#222222" />

# <TextView

# android:id="@+id/adminBrandModel"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:layout\_marginTop="4dp"

# android:text="Марка Модель"

# android:textSize="16sp"

# android:textColor="#666666" />

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:layout\_marginTop="12dp"

# android:gravity="end">

# <Button

# android:id="@+id/approveButton"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:backgroundTint="@color/red"

# android:text="Подтвердить"

# android:textColor="#FFFFFF"

# android:layout\_marginEnd="8dp"/>

# <Button

# android:id="@+id/deleteButton"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:backgroundTint="@color/red"

# android:text="Удалить"

# android:textColor="#FFFFFF" />

# </LinearLayout>

# </LinearLayout>

# </androidx.cardview.widget.CardView>

1. item\_bortjournal.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <androidx.cardview.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"

# xmlns:card\_view="http://schemas.android.com/apk/res-auto"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_margin="12dp"

# android:elevation="6dp"

# card\_view:cardCornerRadius="16dp"

# card\_view:cardUseCompatPadding="true"

# android:background="@android:color/white">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="vertical"

# android:padding="16dp">

# <TextView

# android:id="@+id/brandModelTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textStyle="bold"

# android:textSize="16sp"

# android:textColor="#111111"

# android:layout\_marginBottom="4dp" />

# <TextView

# android:id="@+id/titleTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textStyle="bold"

# android:textSize="18sp"

# android:textColor="#222222"

# android:layout\_marginBottom="6dp" />

# <TextView

# android:id="@+id/mileageTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textColor="#666666"

# android:textSize="14sp"

# android:layout\_marginBottom="10dp" />

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:gravity="center\_vertical">

# <ImageView

# android:id="@+id/profileImageView"

# android:layout\_width="42dp"

# android:layout\_height="42dp"

# android:layout\_marginEnd="12dp"

# android:scaleType="centerCrop"

# android:background="@drawable/circle"

# android:src="@drawable/user" />

# <TextView

# android:id="@+id/userNameTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="16sp"

# android:textColor="#444444"

# android:text="Имя пользователя" />

# </LinearLayout>

# </LinearLayout>

# </androidx.cardview.widget.CardView>

1. item\_car.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <androidx.cardview.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"

# xmlns:app="http://schemas.android.com/apk/res-auto"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_margin="8dp"

# app:cardCornerRadius="12dp"

# app:cardElevation="4dp">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="vertical"

# android:padding="8dp">

# <ImageView

# android:id="@+id/carImageView"

# android:layout\_width="match\_parent"

# android:layout\_height="150dp"

# android:scaleType="centerCrop"

# android:background="@android:color/darker\_gray" />

# <TextView

# android:id="@+id/carTitleTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="18sp"

# android:textStyle="bold"

# android:layout\_marginTop="8dp"

# android:textColor="#000" />

# <TextView

# android:id="@+id/carPriceTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="16sp"

# android:textColor="#333" />

# <TextView

# android:id="@+id/carMileageTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Пробег: 0 км"

# android:textSize="14sp"

# android:textColor="#666"

# android:layout\_marginTop="4dp" />

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:layout\_marginTop="4dp"

# android:gravity="center\_vertical">

# <TextView

# android:id="@+id/carStatusTextView"

# android:layout\_width="0dp"

# android:layout\_height="wrap\_content"

# android:layout\_weight="1"

# android:text="Статус:"

# android:textSize="14sp"

# android:textColor="#888" />

# <TextView

# android:id="@+id/deleteCarButton"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Удалить"

# android:textColor="#B00020"

# android:textSize="14sp"

# android:paddingStart="12dp"

# android:background="@android:color/transparent" />

# </LinearLayout>

# <ImageButton

# android:id="@+id/favoriteButton"

# android:layout\_width="40dp"

# android:layout\_height="40dp"

# android:layout\_gravity="end"

# android:background="?attr/selectableItemBackgroundBorderless"

# android:padding="8dp"

# android:src="@drawable/ic\_heart\_outline"

# android:contentDescription="Добавить в избранное" />

# </LinearLayout>

# </androidx.cardview.widget.CardView>

1. item\_car\_admin.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <androidx.cardview.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_margin="8dp"

# android:elevation="4dp">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="vertical"

# android:padding="16dp">

# <TextView

# android:id="@+id/tvBrand"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Brand"

# android:textSize="18sp"

# android:textStyle="bold" />

# <TextView

# android:id="@+id/tvModel"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Model"

# android:textSize="16sp" />

# <TextView

# android:id="@+id/tvPrice"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Price"

# android:textSize="16sp"

# android:textColor="@android:color/holo\_green\_dark" />

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:gravity="end"

# android:layout\_marginTop="8dp">

# <Button

# android:id="@+id/btnAccept"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Принять"

# android:backgroundTint="@color/red"

# android:layout\_marginEnd="8dp" />

# <Button

# android:id="@+id/btnDelete"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:backgroundTint="@color/red"

# android:text="Удалить" />

# </LinearLayout>

# </LinearLayout>

# </androidx.cardview.widget.CardView>

1. item\_car\_image.xml;

# <ImageView xmlns:android="http://schemas.android.com/apk/res/android"

# android:id="@+id/imageView"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:scaleType="centerCrop"/>

1. item\_car\_user.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <androidx.cardview.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"

# xmlns:app="http://schemas.android.com/apk/res-auto"

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:layout\_margin="8dp"

# app:cardCornerRadius="12dp"

# app:cardElevation="4dp">

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="vertical"

# android:padding="8dp">

# <ImageView

# android:id="@+id/carImageView"

# android:layout\_width="match\_parent"

# android:layout\_height="150dp"

# android:scaleType="centerCrop"

# android:background="@android:color/darker\_gray" />

# <TextView

# android:id="@+id/carTitleTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="18sp"

# android:textStyle="bold"

# android:layout\_marginTop="8dp"

# android:textColor="#000" />

# <TextView

# android:id="@+id/carPriceTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:textSize="16sp"

# android:textColor="#333" />

# <TextView

# android:id="@+id/carMileageTextView"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Пробег: 0 км"

# android:textSize="14sp"

# android:textColor="#666"

# android:layout\_marginTop="4dp" />

# <LinearLayout

# android:layout\_width="match\_parent"

# android:layout\_height="wrap\_content"

# android:orientation="horizontal"

# android:layout\_marginTop="4dp"

# android:gravity="center\_vertical">

# <TextView

# android:id="@+id/carStatusTextView"

# android:layout\_width="0dp"

# android:layout\_height="wrap\_content"

# android:layout\_weight="1"

# android:text="Статус:"

# android:textSize="14sp"

# android:textColor="#888" />

# <TextView

# android:id="@+id/deleteCarButton"

# android:layout\_width="wrap\_content"

# android:layout\_height="wrap\_content"

# android:text="Удалить"

# android:textColor="#B00020"

# android:textSize="14sp"

# android:paddingStart="12dp"

# android:background="@android:color/transparent" />

# </LinearLayout>

# <ImageButton

# android:id="@+id/favoriteButton"

# android:layout\_width="40dp"

# android:layout\_height="40dp"

# android:layout\_gravity="end"

# android:background="?attr/selectableItemBackgroundBorderless"

# android:padding="8dp"

# android:src="@drawable/ic\_heart\_outline"

# android:contentDescription="Добавить в избранное" />

# </LinearLayout>

# </androidx.cardview.widget.CardView>

1. item\_image.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <ImageView xmlns:android="http://schemas.android.com/apk/res/android"

# android:id="@+id/imageView"

# android:layout\_width="match\_parent"

# android:layout\_height="match\_parent"

# android:scaleType="centerCrop"/>

1. bottom\_nav\_menu.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <menu xmlns:android="http://schemas.android.com/apk/res/android">

# <item

# android:id="@+id/nav\_manage\_models"

# android:icon="@drawable/pluss"

# android:title="Марки и модели" />

# <item

# android:id="@+id/nav\_manage\_bort"

# android:icon="@drawable/garage"

# android:title="Бортжурнал" />

# </menu>

1. bottom\_navigation\_menu.xml;

# <!-- res/menu/bottom\_navigation\_menu.xml -->

# <menu xmlns:android="http://schemas.android.com/apk/res/android">

# <item

# android:id="@+id/nav\_home"

# android:icon="@drawable/home" />

# <item

# android:id="@+id/nav\_saved"

# android:icon="@drawable/star" />

# <item

# android:id="@+id/nav\_create"

# android:icon="@drawable/pluss" />

# <item

# android:id="@+id/nav\_garage"

# android:icon="@drawable/garage" />

# <item

# android:id="@+id/nav\_profile"

# android:icon="@drawable/user" />

# </menu>

1. colors.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <resources>

# <color name="black">#FF000000</color>

# <color name="white">#FFFFFFFF</color>

# <color name="red">#ba0f1d</color>

# </resources>

1. strings.xml;

# <resources>

# <string name="app\_name">Drivee</string>

# </resources>

1. styles.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <resources>

# <style name="CircleImage" parent="">

# <item name="cornerFamily">rounded</item>

# <item name="cornerSize">50%</item>

# </style>

# </resources>

1. values.xml;

# <?xml version="1.0" encoding="utf-8"?>

# <resources>

# <string name="gcm\_defaultSenderId" translatable="false">1034230179468</string>

# <string name="google\_api\_key" translatable="false">AIzaSyBBQRNebcXKiFtSWDjlTZE56yuXzamgpGA</string>

# <string name="google\_app\_id" translatable="false">1:1034230179468:android:36f2445a4d7d74f680bda9</string>

# <string name="google\_crash\_reporting\_api\_key" translatable="false">AIzaSyBBQRNebcXKiFtSWDjlTZE56yuXzamgpGA</string>

# <string name="google\_storage\_bucket" translatable="false">drivee-42a66.firebasestorage.app</string>

# <string name="project\_id" translatable="false">drivee-42a66</string>

# </resources>

1. build.gradle;

# plugins {

# alias(libs.plugins.android.application)

# alias(libs.plugins.google.gms.google.services)

# }

# android {

# namespace 'com.example.drivee'

# compileSdk 35

# defaultConfig {

# applicationId "com.example.drivee"

# minSdk 24

# targetSdk 35

# versionCode 1

# versionName "1.0"

# testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"

# }

# buildTypes {

# release {

# minifyEnabled false

# proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'

# }

# }

# compileOptions {

# sourceCompatibility JavaVersion.VERSION\_11

# targetCompatibility JavaVersion.VERSION\_11

# }

# }

# dependencies {

# implementation 'com.github.bumptech.glide:glide:4.12.0'

# implementation libs.firebase.firestore

# implementation libs.play.services.location

# annotationProcessor 'com.github.bumptech.glide:compiler:4.12.0'

# implementation libs.appcompat

# implementation libs.material

# implementation 'com.squareup.retrofit2:retrofit:2.9.0'

# implementation 'com.squareup.retrofit2:converter-gson:2.9.0'

# implementation 'com.squareup.okhttp3:okhttp:4.9.3'

# implementation libs.activity

# implementation libs.constraintlayout

# implementation 'com.squareup.picasso:picasso:2.71828'

# implementation libs.firebase.auth

# testImplementation libs.junit

# androidTestImplementation libs.ext.junit

# androidTestImplementation libs.espresso.core

# implementation 'androidx.viewpager2:viewpager2:1.0.0'

# }

1. libs.versions.toml;

# [versions]

# agp = "8.8.0"

# junit = "4.13.2"

# junitVersion = "1.2.1"

# espressoCore = "3.6.1"

# appcompat = "1.7.0"

# material = "1.12.0"

# activity = "1.10.1"

# constraintlayout = "2.2.1"

# firebaseAuth = "23.2.0"

# googleGmsGoogleServices = "4.4.2"

# firebaseFirestore = "25.1.3"

# playServicesLocation = "21.3.0"

# [libraries]

# junit = { group = "junit", name = "junit", version.ref = "junit" }

# ext-junit = { group = "androidx.test.ext", name = "junit", version.ref = "junitVersion" }

# espresso-core = { group = "androidx.test.espresso", name = "espresso-core", version.ref = "espressoCore" }

# appcompat = { group = "androidx.appcompat", name = "appcompat", version.ref = "appcompat" }

# material = { group = "com.google.android.material", name = "material", version.ref = "material" }

# activity = { group = "androidx.activity", name = "activity", version.ref = "activity" }

# constraintlayout = { group = "androidx.constraintlayout", name = "constraintlayout", version.ref = "constraintlayout" }

# firebase-auth = { group = "com.google.firebase", name = "firebase-auth", version.ref = "firebaseAuth" }

# firebase-firestore = { group = "com.google.firebase", name = "firebase-firestore", version.ref = "firebaseFirestore" }

# play-services-location = { group = "com.google.android.gms", name = "play-services-location", version.ref = "playServicesLocation" }

# [plugins]

# android-application = { id = "com.android.application", version.ref = "agp" }

# google-gms-google-services = { id = "com.google.gms.google-services", version.ref = "googleGmsGoogleServices" }