package com.example.recipes;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.text.TextUtils;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ProgressBar;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import com.google.firebase.database.DataSnapshot;  
import com.google.firebase.database.DatabaseError;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
import com.google.firebase.database.ValueEventListener;  
  
import android.content.SharedPreferences;  
  
public class LoginActivity extends AppCompatActivity {  
  
 private EditText inputEmail, inputPassword;  
 private Button btnLogin;  
 private ProgressBar progressBar;  
 private DatabaseReference databaseReference;  
 private Button backButton;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_login);  
  
 inputEmail = findViewById(R.id.email);  
 inputPassword = findViewById(R.id.password);  
 btnLogin = findViewById(R.id.login\_button);  
 progressBar = findViewById(R.id.progressBar);  
  
 backButton = findViewById(R.id.back\_button);  
  
 databaseReference = FirebaseDatabase.getInstance().getReference("Users");  
  
 btnLogin.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 final String email = inputEmail.getText().toString().trim();  
 final String password = inputPassword.getText().toString().trim();  
  
 if (TextUtils.isEmpty(email)) {  
 Toast.makeText(getApplicationContext(), "Введите адрес электронной почты!", Toast.LENGTH\_SHORT).show();  
 return;  
 }  
  
 if (TextUtils.isEmpty(password)) {  
 Toast.makeText(getApplicationContext(), "Введите пароль!", Toast.LENGTH\_SHORT).show();  
 return;  
 }  
  
 progressBar.setVisibility(View.VISIBLE);  
  
 databaseReference.orderByChild("email").equalTo(email).addListenerForSingleValueEvent(new ValueEventListener() {  
 @Override  
 public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  
 progressBar.setVisibility(View.GONE);  
 if (dataSnapshot.exists()) {  
 for (DataSnapshot userSnapshot : dataSnapshot.getChildren()) {  
 User user = userSnapshot.getValue(User.class);  
 // При успешном входе пользователя сохраняем его данные в SharedPreferences  
 if (user != null && user.getPassword().equals(password)) {  
 // Сохраняем данные пользователя в SharedPreferences  
 SharedPreferences.Editor editor = getSharedPreferences("UserData", MODE\_PRIVATE).edit();  
 editor.putString("userName", user.getName());  
 editor.putString("userEmail", user.getEmail());  
 editor.apply();  
  
 // Переходим на MenuActivity  
 Intent intent = new Intent(LoginActivity.this, MenuActivity.class);  
 startActivity(intent);  
 finish();  
 return;  
 }  
 }  
 }  
 Toast.makeText(LoginActivity.this, "Неверный email или пароль", Toast.LENGTH\_SHORT).show();  
 }  
  
 @Override  
 public void onCancelled(@NonNull DatabaseError databaseError) {  
 progressBar.setVisibility(View.GONE);  
 Toast.makeText(LoginActivity.this, "Ошибка при входе", Toast.LENGTH\_SHORT).show();  
 }  
 });  
 }  
 });  
 backButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 finish();  
 }  
 });  
 }  
}

package com.example.recipes;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.text.TextUtils;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ProgressBar;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.Task;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
  
public class RegisterActivity extends AppCompatActivity {  
  
 private EditText inputEmail, inputPassword, inputName;  
 private Button btnRegister;  
 private ProgressBar progressBar;  
 private DatabaseReference databaseReference;  
 private Button backButton;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_register);  
  
 databaseReference = FirebaseDatabase.getInstance().getReference("Users");  
  
 inputEmail = findViewById(R.id.email);  
 inputPassword = findViewById(R.id.password);  
 inputName = findViewById(R.id.name);  
 btnRegister = findViewById(R.id.register\_button);  
 progressBar = findViewById(R.id.progressBar);  
 backButton = findViewById(R.id.back\_button);  
  
  
 btnRegister.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 final String email = inputEmail.getText().toString().trim();  
 final String password = inputPassword.getText().toString().trim();  
 final String name = inputName.getText().toString().trim();  
  
 if (TextUtils.isEmpty(email)) {  
 Toast.makeText(getApplicationContext(), "Введите адрес электронной почты!", Toast.LENGTH\_SHORT).show();  
 return;  
 }  
  
 if (TextUtils.isEmpty(password)) {  
 Toast.makeText(getApplicationContext(), "Введите пароль!", Toast.LENGTH\_SHORT).show();  
 return;  
 }  
  
 if (TextUtils.isEmpty(name)) {  
 Toast.makeText(getApplicationContext(), "Введите имя!", Toast.LENGTH\_SHORT).show();  
 return;  
 }  
  
 if (password.length() < 6) {  
 Toast.makeText(getApplicationContext(), "Пароль слишком короткий, введите минимум 6 символов!", Toast.LENGTH\_SHORT).show();  
 return;  
 }  
  
 progressBar.setVisibility(View.VISIBLE);  
  
 // Сохранение данных пользователя в базе данных  
 String userId = databaseReference.push().getKey();  
 User user = new User(userId, name, email, password);  
 databaseReference.child(userId).setValue(user).addOnCompleteListener(new OnCompleteListener<Void>() {  
 @Override  
 public void onComplete(@NonNull Task<Void> task) {  
 progressBar.setVisibility(View.GONE);  
 if (task.isSuccessful()) {  
 Toast.makeText(RegisterActivity.this, "Пользователь зарегистрирован", Toast.LENGTH\_SHORT).show();  
 Intent intent = new Intent(RegisterActivity.this, MenuActivity.class);  
 intent.putExtra("userName", name);  
 intent.putExtra("userEmail", email);  
 startActivity(intent);  
 finish();  
 } else {  
 Toast.makeText(RegisterActivity.this, "Ошибка при регистрации", Toast.LENGTH\_SHORT).show();  
 }  
 }  
 });  
 }  
 });  
 backButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 finish();  
 }  
 });  
 }  
}

избранные список

package com.example.recipes;  
  
import android.content.SharedPreferences;  
import android.os.Bundle;  
import android.widget.ArrayAdapter;  
import android.widget.ListView;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import com.google.firebase.database.DataSnapshot;  
import com.google.firebase.database.DatabaseError;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
import com.google.firebase.database.ValueEventListener;  
  
import java.util.ArrayList;  
  
public class FavoriteRecipesActivity extends AppCompatActivity {  
  
 private ListView favoriteRecipesListView;  
 private ArrayAdapter<String> adapter;  
 private ArrayList<String> favoriteRecipesList;  
  
 private DatabaseReference databaseReference;  
 private String userId;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_favorite\_recipes);  
  
 favoriteRecipesListView = findViewById(R.id.favorite\_recipes\_list\_view);  
 favoriteRecipesList = new ArrayList<>();  
 adapter = new ArrayAdapter<>(this, android.R.layout.simple\_list\_item\_1, favoriteRecipesList);  
 favoriteRecipesListView.setAdapter(adapter);  
  
 SharedPreferences sharedPreferences = getSharedPreferences("UserData", MODE\_PRIVATE);  
 userId = sharedPreferences.getString("userID", null);  
  
 if (userId != null) {  
 databaseReference = FirebaseDatabase.getInstance().getReference("Favorites").child(userId);  
  
 databaseReference.addValueEventListener(new ValueEventListener() {  
 @Override  
 public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  
 favoriteRecipesList.clear();  
 for (DataSnapshot snapshot : dataSnapshot.getChildren()) {  
 String recipeId = snapshot.getKey();  
 if (recipeId != null) {  
 loadRecipeDetails(recipeId);  
 }  
 }  
 }  
  
 @Override  
 public void onCancelled(@NonNull DatabaseError databaseError) {  
 Toast.makeText(FavoriteRecipesActivity.this, "Ошибка при загрузке данных", Toast.LENGTH\_SHORT).show();  
 }  
 });  
 } else {  
 Toast.makeText(this, "Пользователь не авторизован", Toast.LENGTH\_SHORT).show();  
 finish();  
 }  
 }  
  
 private void loadRecipeDetails(String recipeId) {  
 DatabaseReference recipeReference = FirebaseDatabase.getInstance().getReference("Recipes").child(recipeId);  
 recipeReference.addListenerForSingleValueEvent(new ValueEventListener() {  
 @Override  
 public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  
 String recipeTitle = dataSnapshot.child("title").getValue(String.class);  
 if (recipeTitle != null) {  
 favoriteRecipesList.add(recipeTitle);  
 adapter.notifyDataSetChanged();  
 }  
 }  
  
 @Override  
 public void onCancelled(@NonNull DatabaseError databaseError) {  
 Toast.makeText(FavoriteRecipesActivity.this, "Ошибка при загрузке данных рецепта", Toast.LENGTH\_SHORT).show();  
 }  
 });  
 }  
  
}

избранные с просмотром каждого

package com.example.recipes;  
  
import android.content.Intent;  
import android.content.SharedPreferences;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.ArrayAdapter;  
import android.widget.ListView;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import com.google.firebase.database.DataSnapshot;  
import com.google.firebase.database.DatabaseError;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
import com.google.firebase.database.ValueEventListener;  
  
import java.util.ArrayList;  
  
public class FavoriteRecipesActivity extends AppCompatActivity {  
  
 private ListView favoriteRecipesListView;  
 private ArrayAdapter<String> adapter;  
 private ArrayList<String> favoriteRecipesList;  
 private ArrayList<Recipe> recipeList; // Список для хранения полных данных рецептов  
  
 private DatabaseReference databaseReference;  
 private String userId;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_favorite\_recipes);  
  
 favoriteRecipesListView = findViewById(R.id.favorite\_recipes\_list\_view);  
 favoriteRecipesList = new ArrayList<>();  
 recipeList = new ArrayList<>();  
 adapter = new ArrayAdapter<>(this, android.R.layout.simple\_list\_item\_1, favoriteRecipesList);  
 favoriteRecipesListView.setAdapter(adapter);  
  
 SharedPreferences sharedPreferences = getSharedPreferences("UserData", MODE\_PRIVATE);  
 userId = sharedPreferences.getString("userID", null);  
  
 if (userId != null) {  
 databaseReference = FirebaseDatabase.getInstance().getReference("Favorites").child(userId);  
  
 databaseReference.addValueEventListener(new ValueEventListener() {  
 @Override  
 public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  
 favoriteRecipesList.clear();  
 recipeList.clear();  
 for (DataSnapshot snapshot : dataSnapshot.getChildren()) {  
 String recipeId = snapshot.getKey();  
 if (recipeId != null) {  
 loadRecipeDetails(recipeId);  
 }  
 }  
 }  
  
 @Override  
 public void onCancelled(@NonNull DatabaseError databaseError) {  
 Toast.makeText(FavoriteRecipesActivity.this, "Ошибка при загрузке данных", Toast.LENGTH\_SHORT).show();  
 }  
 });  
 } else {  
 Toast.makeText(this, "Пользователь не авторизован", Toast.LENGTH\_SHORT).show();  
 finish();  
 }  
  
 favoriteRecipesListView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
 // Получаем данные о выбранном рецепте  
 Recipe selectedRecipe = recipeList.get(position);  
  
 // Создаем интент для перехода к RecipeDetailsActivity и передачи информации о рецепте  
 Intent intent = new Intent(FavoriteRecipesActivity.this, RecipeDetailsActivity.class);  
 intent.putExtra("id", selectedRecipe.getId());  
 intent.putExtra("recipe\_title", selectedRecipe.getTitle());  
 intent.putExtra("ingredients", selectedRecipe.getIngredients());  
 intent.putExtra("instructions", selectedRecipe.getInstructions());  
 intent.putExtra("author", selectedRecipe.getAuthor());  
 startActivity(intent);  
 }  
 });  
 }  
  
 private void loadRecipeDetails(String recipeId) {  
 DatabaseReference recipeReference = FirebaseDatabase.getInstance().getReference("Recipes").child(recipeId);  
 recipeReference.addListenerForSingleValueEvent(new ValueEventListener() {  
 @Override  
 public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  
 String recipeTitle = dataSnapshot.child("title").getValue(String.class);  
 String recipeIngredients = dataSnapshot.child("ingredients").getValue(String.class);  
 String recipeInstructions = dataSnapshot.child("instructions").getValue(String.class);  
 String recipeAuthor = dataSnapshot.child("author").getValue(String.class);  
 Boolean recipeFavorite = dataSnapshot.child("favorite").getValue(Boolean.class);  
  
 if (recipeTitle != null && recipeIngredients != null && recipeInstructions != null && recipeAuthor != null) {  
 boolean isFavorite = recipeFavorite != null ? recipeFavorite : false; // Проверка на null  
 favoriteRecipesList.add(recipeTitle);  
 recipeList.add(new Recipe(recipeId, recipeTitle, recipeIngredients, recipeInstructions, recipeAuthor, isFavorite));  
 adapter.notifyDataSetChanged();  
 }  
 }  
  
 @Override  
 public void onCancelled(@NonNull DatabaseError databaseError) {  
 Toast.makeText(FavoriteRecipesActivity.this, "Ошибка при загрузке данных рецепта", Toast.LENGTH\_SHORT).show();  
 }  
 });  
 }  
}

**RecipeDetailsActivity**

package com.example.recipes;  
  
import android.content.Intent;  
import android.content.SharedPreferences;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.annotation.Nullable;  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.Task;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
  
public class RecipeDetailsActivity extends AppCompatActivity {  
  
 private static final String TAG = "RecipeDetailsActivity";  
 private String recipeId;  
 private String authorEmail;  
 private Button backButton;  
 private Button favoriteButtonAdd;  
 private Button favoriteButtonRemove;  
  
 @Override  
 protected void onCreate(@Nullable Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.recipe\_details);  
  
 // Находим кнопки для редактирования и удаления рецепта  
 Button editButton = findViewById(R.id.button\_edit);  
 Button deleteButton = findViewById(R.id.button\_delete);  
 backButton = findViewById(R.id.back\_button);  
 favoriteButtonAdd = findViewById(R.id.button\_favorite1);  
 favoriteButtonRemove = findViewById(R.id.button\_favorite2);  
  
 // Получаем данные о рецепте из предыдущей активности  
 Bundle extras = getIntent().getExtras();  
 if (extras != null) {  
 String recipeTitle = extras.getString("recipe\_title");  
 String ingredients = extras.getString("ingredients");  
 String instructions = extras.getString("instructions");  
 String author = extras.getString("author"); // Получение данных об авторе  
 recipeId = extras.getString("id"); // Получение идентификатора рецепта  
 authorEmail = extras.getString("author"); // Получение email автора  
  
 // Находим элементы представления  
 TextView titleTextView = findViewById(R.id.text\_recipe\_title);  
 TextView ingredientsTextView = findViewById(R.id.text\_ingredients);  
 TextView instructionsTextView = findViewById(R.id.text\_instructions);  
 TextView authorTextView = findViewById(R.id.text\_author);  
  
 backButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 finish();  
 }  
 });  
  
 // Устанавливаем данные о рецепте в соответствующие элементы представления  
 titleTextView.setText(recipeTitle);  
 ingredientsTextView.setText("Ингредиенты: " + ingredients);  
 instructionsTextView.setText("Инструкции: " + instructions);  
 authorTextView.setText("Автор: " + author); // Установка автора рецепта  
  
 // Получаем данные пользователя из SharedPreferences  
 SharedPreferences sharedPreferences = getSharedPreferences("UserData", MODE\_PRIVATE);  
 String currentUserEmail = sharedPreferences.getString("userEmail", null);  
 String currentUserID = sharedPreferences.getString("userID", null);  
 Log.d(TAG, "userID = " + currentUserID + ", userEmail = " + currentUserEmail);  
  
 // Проверка, является ли текущий пользователь автором рецепта  
 if (currentUserEmail != null && currentUserEmail.equals(authorEmail)) {  
 Log.d(TAG, "Current user is the author, showing buttons.");  
 // Делаем кнопки видимыми для автора рецепта  
 editButton.setVisibility(View.VISIBLE);  
 deleteButton.setVisibility(View.VISIBLE);  
  
 // Обработчик нажатия на кнопку "Редактировать рецепт"  
 editButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 // Переходим к активности редактирования рецепта  
 Intent intent = new Intent(RecipeDetailsActivity.this, EditRecipeActivity.class);  
 intent.putExtra("recipe\_id", recipeId);  
 intent.putExtra("recipe\_title", recipeTitle);  
 intent.putExtra("ingredients", ingredients);  
 intent.putExtra("instructions", instructions);  
 intent.putExtra("author", author);  
 startActivity(intent);  
 }  
 });  
  
 // Обработчик нажатия на кнопку "Удалить рецепт"  
 deleteButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 // Подтверждение удаления  
 new AlertDialog.Builder(RecipeDetailsActivity.this)  
 .setTitle("Удаление рецепта")  
 .setMessage("Вы уверены, что хотите удалить этот рецепт?")  
 .setPositiveButton(android.R.string.yes, (dialog, which) -> deleteRecipe(recipeId))  
 .setNegativeButton(android.R.string.no, null)  
 .show();  
 }  
 });  
 } else {  
 Log.d(TAG, "Current user is not the author, hiding buttons.");  
 // Скрываем кнопки редактирования и удаления для неавтора  
 editButton.setVisibility(View.GONE);  
 deleteButton.setVisibility(View.GONE);  
 }  
  
 // Добавляем обработчики для добавления и удаления рецептов из избранного  
 if (currentUserID != null) {  
 favoriteButtonAdd.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Log.d(TAG, "Adding to favorites: UserID = " + currentUserID + ", RecipeID = " + recipeId);  
 addRecipeToFavorites(currentUserID, recipeId);  
 }  
 });  
  
 favoriteButtonRemove.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Log.d(TAG, "Removing from favorites: UserID = " + currentUserID + ", RecipeID = " + recipeId);  
 removeRecipeFromFavorites(currentUserID, recipeId);  
 }  
 });  
 }  
 }  
 }  
  
 // Метод для удаления рецепта  
 private void deleteRecipe(String recipeId) {  
 DatabaseReference databaseReference = FirebaseDatabase.getInstance().getReference().child("Recipes").child(recipeId);  
 databaseReference.removeValue().addOnCompleteListener(new OnCompleteListener<Void>() {  
 @Override  
 public void onComplete(@NonNull Task<Void> task) {  
 if (task.isSuccessful()) {  
 Toast.makeText(RecipeDetailsActivity.this, "Рецепт успешно удален", Toast.LENGTH\_SHORT).show();  
 finish(); // Закрываем активность после удаления рецепта  
 } else {  
 Toast.makeText(RecipeDetailsActivity.this, "Ошибка при удалении рецепта", Toast.LENGTH\_SHORT).show();  
 }  
 }  
 });  
 }  
  
 private void addRecipeToFavorites(String currentUserID, String recipeId) {  
 DatabaseReference favoritesReference = FirebaseDatabase.getInstance().getReference().child("Favorites").child(currentUserID).child(recipeId);  
 favoritesReference.setValue(true).addOnCompleteListener(new OnCompleteListener<Void>() {  
 @Override  
 public void onComplete(@NonNull Task<Void> task) {  
 if (task.isSuccessful()) {  
 Log.d(TAG, "Recipe added to favorites");  
 Toast.makeText(RecipeDetailsActivity.this, "Рецепт добавлен в избранное", Toast.LENGTH\_SHORT).show();  
 favoriteButtonAdd.setVisibility(View.GONE);  
 favoriteButtonRemove.setVisibility(View.VISIBLE);  
 } else {  
 Log.d(TAG, "Error adding recipe to favorites: " + task.getException().getMessage());  
 Toast.makeText(RecipeDetailsActivity.this, "Ошибка при добавлении в избранное", Toast.LENGTH\_SHORT).show();  
 }  
 }  
 });  
 }  
  
 private void removeRecipeFromFavorites(String currentUserID, String recipeId) {  
 DatabaseReference favoritesReference = FirebaseDatabase.getInstance().getReference().child("Favorites").child(currentUserID).child(recipeId);  
 favoritesReference.removeValue().addOnCompleteListener(new OnCompleteListener<Void>() {  
 @Override  
 public void onComplete(@NonNull Task<Void> task) {  
 if (task.isSuccessful()) {  
 Log.d(TAG, "Recipe removed from favorites");  
 Toast.makeText(RecipeDetailsActivity.this, "Рецепт удален из избранного", Toast.LENGTH\_SHORT).show();  
 favoriteButtonAdd.setVisibility(View.VISIBLE);  
 favoriteButtonRemove.setVisibility(View.GONE);  
 } else {  
 Log.d(TAG, "Error removing recipe from favorites: " + task.getException().getMessage());  
 Toast.makeText(RecipeDetailsActivity.this, "Ошибка при удалении из избранного", Toast.LENGTH\_SHORT).show();  
 }  
 }  
 });  
 }  
}