package com.shruti.lofo.ui.Lost;  
  
import android.content.Context;  
import com.bumptech.glide.Glide;  
  
import android.content.Intent;  
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.firebase.ui.firestore.FirestoreRecyclerAdapter;  
import com.firebase.ui.firestore.FirestoreRecyclerOptions;  
import com.shruti.lofo.R;  
import com.shruti.lofo.Utility;  
  
public class LostItemsAdapter extends FirestoreRecyclerAdapter<LostItems, LostItemsAdapter.ItemViewHolder> {  
  
 Context context;  
 private String category;  
  
 private boolean showDeleteButton;  
  
  
 public LostItemsAdapter(@NonNull FirestoreRecyclerOptions<LostItems> options, Context context, String category, boolean showDeleteButton) {  
 super(options);  
 this.context = context;  
 this.category = category;  
 this.showDeleteButton=showDeleteButton;  
 }  
  
 public void setCategory(String category) {  
 this.category = category;  
 }  
  
 public String getCategory() {  
 return category;  
  
 }  
  
 @NonNull  
 @Override  
 public ItemViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  
 View view = LayoutInflater.*from*(parent.getContext()).inflate(R.layout.*lost\_item\_card*, parent, false);  
 return new ItemViewHolder(view);  
 }  
  
 @Override  
 public void onBindViewHolder(@NonNull ItemViewHolder holder, int position, @NonNull LostItems item) {  
 if(category.isEmpty() || item.getCategory().equals(category)) {  
 if (item.getImageURI() != null && !item.getImageURI().isEmpty()) {  
 Glide.*with*(context)  
 .load(item.getImageURI())  
 .placeholder(R.drawable.*placeholder\_image*) // Add a placeholder image while the actual image is loading  
 .error(R.drawable.*baseline\_image\_search\_24*) // Add an error image if the image fails to load  
 .into(holder.itemImageView);  
 }  
 holder.itemNameTextView.setText(item.getItemName());  
 holder.ownerNameTextView.setText(item.getOwnerName());  
 holder.description.setText(item.getDescription());  
 holder.location.setText(item.getLocation());  
 holder.date.setText(item.getDateLost());  
  
 // Set an onClickListener for the card view  
 holder.itemView.setOnClickListener(v -> {  
 // Create an Intent to start the LostDetails activity  
 Intent intent = new Intent(context, LostDetails.class);  
  
 // Pass the itemId as an extra to the intent  
 intent.putExtra("itemId", item.getItemName());  
  
 // Start the LostDetails activity  
 context.startActivity(intent);  
 });  
 if (showDeleteButton && (category.isEmpty() || item.getCategory().equals(category))) {  
 // Additional logic for the delete button  
 holder.deleteButton.setVisibility(View.*VISIBLE*);  
  
  
 holder.deleteButton.setOnClickListener(v -> {  
 String documentId = getSnapshots().getSnapshot(position).getId();  
 Utility.*getCollectionReferrenceForItems2*().document(documentId).delete()  
 .addOnSuccessListener(aVoid -> {  
 // Item deleted successfully, update the UI or perform other tasks if needed  
 })  
 .addOnFailureListener(e -> {  
 // An error occurred, handle the error appropriately  
 Toast.*makeText*(v.getContext(), "Couldn't delete!", Toast.*LENGTH\_SHORT*).show();  
 });  
 });  
 }  
 }  
 }  
  
 public static class ItemViewHolder extends RecyclerView.ViewHolder {  
 ImageView itemImageView;  
 TextView itemNameTextView;  
 TextView ownerNameTextView;  
 TextView description;  
 TextView location;  
 TextView date;  
 ImageButton deleteButton;  
  
  
  
  
 public ItemViewHolder(@NonNull View itemView) {  
 super(itemView);  
 itemImageView = itemView.findViewById(R.id.*itemImageView*);  
 itemNameTextView = itemView.findViewById(R.id.*itemNameTextView*);  
 ownerNameTextView = itemView.findViewById(R.id.*ownerNameTextView*);  
 description= itemView.findViewById(R.id.*item\_description*);  
 location = itemView.findViewById((R.id.*location*));  
 date = itemView.findViewById(R.id.*dateLost*);  
 deleteButton= itemView.findViewById(R.id.*deleteButton*);  
  
 }  
 }  
}