```
package com.example.et5;

import android.app.DatePickerDialog;
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.ListView;
import android.widget.TextView;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
import androidx.lifecycle.ViewModelProvider;

import com.google.android.material.button.MaterialButton;
import com.google.android.material.snackbar.Snackbar;
import com.google.android.material.textfield.TextInputEditText;

import java.text.NumberFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;
import java.util.List;
```

```java
import java.util.Locale;

public class MainActivity extends AppCompatActivity {

    private TextInputEditText expenseName, expenseAmount;
    private MaterialButton addButton;
    private TextView totalText, expenseCountText, averageText;
    private ListView expenseList;
    private Button dateButton;
    private CardView inputCard, summaryCard, listCard;

    private ArrayList<String> expenses;
    private ArrayAdapter<String> adapter;
    private double total = 0.0;
    private int expenseCount = 0;
    private Calendar selectedDate = Calendar.getInstance();

    private ExpenseViewModel expenseViewModel;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);

        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets)
-> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
```

```
      v.setPadding(systemBars.left, systemBars.top, systemBars.right,
systemBars.bottom);
      return insets;
    });

    initializeViews();
    setupListView();
    setupViewModel();
    setClickListeners();
  }

  private void initializeViews() {
    expenseName = findViewById(R.id.expenseName);
    expenseAmount = findViewById(R.id.expenseAmount);
    addButton = findViewById(R.id.addButton);
    totalText = findViewById(R.id.totalText);
    expenseCountText = findViewById(R.id.expenseCountText);
    averageText = findViewById(R.id.averageText);
    expenseList = findViewById(R.id.expenseList);
    dateButton = findViewById(R.id.dateButton);
    inputCard = findViewById(R.id.inputCard);
    summaryCard = findViewById(R.id.summaryCard);
    listCard = findViewById(R.id.listCard);

    updateDateButtonText();
  }

  private void setupListView() {
    expenses = new ArrayList<>();
    adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1, expenses);
    expenseList.setAdapter(adapter);
  }

  private void setupViewModel() {
    expenseViewModel = new ViewModelProvider(this).get(ExpenseViewModel.class);

    expenseViewModel.getAllExpenses().observe(this, expenses -> {
      updateExpensesList(expenses);
    });

    expenseViewModel.getTotalAmount().observe(this, total -> {
      this.total = total != null ? total : 0.0;
      updateSummary();
    });

    expenseViewModel.getExpenseCount().observe(this, count -> {
      this.expenseCount = count != null ? count : 0;
      updateSummary();
    });
  }

  private void updateExpensesList(List<Expense> expenseList) {
    expenses.clear();
    NumberFormat format = NumberFormat.getCurrencyInstance(Locale.US);
    for (Expense expense : expenseList) {
      String dateStr = android.text.format.DateFormat.format("MMM dd", expense.getDate()).toString();
      String expenseEntry = String.format("%s: %s - %s",
              dateStr, expense.getName(), format.format(expense.getAmount()));
      expenses.add(expenseEntry);
    }

    adapter.notifyDataSetChanged();
  }

  private void setClickListeners() {
    addButton.setOnClickListener(v -> addExpense());
    dateButton.setOnClickListener(v -> showDatePicker());
    expenseList.setOnItemLongClickListener((parent, view, position, id) -> {
      removeExpense(position);
      return true;
    });
  }

  private void addExpense() {
    String name = expenseName.getText().toString().trim();
    String amountStr = expenseAmount.getText().toString().trim();

    if (name.isEmpty()) {
      showSnackbar("Please enter expense name");
      return;
    }
    if (amountStr.isEmpty()) {
      showSnackbar("Please enter amount");
      return;
    }

    try {
      double amount = Double.parseDouble(amountStr);
      Date date = selectedDate.getTime();

      Expense expense = new Expense(name, amount, date);
      expenseViewModel.insert(expense);

      clearInputs();

    } catch (NumberFormatException e) {
      showSnackbar("Invalid amount format");
    }
  }

  private void removeExpense(int position) {
    List<Expense> allExpenses = expenseViewModel.getAllExpenses().getValue();
    if (allExpenses != null && position < allExpenses.size()) {
      Expense expense = allExpenses.get(position);
      expenseViewModel.delete(expense);
      showSnackbar("Expense removed");
    } else {
      showSnackbar("Error removing expense");
    }
  }

  private void updateSummary() {
    NumberFormat format = NumberFormat.getCurrencyInstance(Locale.US);
    totalText.setText(String.format("Total: %s", format.format(total)));
    expenseCountText.setText(String.format("Expenses: %d", expenseCount));

    double average = expenseCount > 0 ? total / expenseCount : 0;
    averageText.setText(String.format("Average: %s", format.format(average)));
  }

  private void showDatePicker() {
    new DatePickerDialog(this, (view, year, month, day) -> {
      selectedDate.set(year, month, day);
      updateDateButtonText();
    },
            selectedDate.get(Calendar.YEAR),
            selectedDate.get(Calendar.MONTH),
            selectedDate.get(Calendar.DAY_OF_MONTH))
            .show();
  }

  private void updateDateButtonText() {
    dateButton.setText(android.text.format.DateFormat.format("MMM dd, yyyy", selectedDate));
  }
```

```
  private void clearInputs() {
    expenseName.setText("");
    expenseAmount.setText("");
    expenseName.requestFocus();
  }

  private void showSnackbar(String message) {
    Snackbar.make(findViewById(R.id.main), message, 
Snackbar.LENGTH_SHORT).show();
  }
}
```