Project Report

On

MONEY TRACKER

Submitted in partial fulfilment of the requirements for the award of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

(Artificial Intelligence & Machine Learning)

by

Ms. Yashaswiny Sripada - 22wh1a6607

Ms. R.Ishwarya – 22wh1a6609

Ms. S.Aishwarya– 22wh1a6644

Ms. N.Vaishnavi – 22wh1a6645

Under the esteemed guidance of

Ms. S Annapoorna

Assistant Professor, CSE(AI&ML)

BVRIT HYDERABAD College of Engineering for Women

(UGC Autonomous Institution | Approved by AICTE | Affiliated to JNTUH)

(NAAC Accredited - A Grade | NBA Accredited B.Tech. (EEE, ECE, CSE and IT)

Bachupally, Hyderabad – 500090

2024-25

Department of Computer Science & Engineering

(Artificial Intelligence & Machine Learning)

BVRIT HYDERABAD COLLEGE OF ENGINEERING FOR WOMEN

(Approved by AICTE, New Delhi and Affiliated to JNTUH, Hyderabad)

Accredited by NBA and NAAC with A Grade

Bachupally, Hyderabad – 500090

2023-24

CERTIFICATE

This is to certify that the major project entitled **"Money Tracker"** is a bonafide work carried out by **Ms. Yashaswiny Sripada (22WH1A6607), Ms. R. Ishwarya (22WH1A6609)**, **Ms. S. Aishwarya (22WH1A6644), Ms. N. Vaishnavi (22WH1A6645**) in partial fulfilment for the award of B. Tech degree in **Computer Science & Engineering (AI&ML)**, **BVRIT HYDERABAD College of Engineering for Women, Bachupally, Hyderabad**, affiliated to Jawaharlal Nehru Technological University Hyderabad, Hyderabad under my guidance and supervision. The results embodied in the project work have not been submitted to any other University or Institute for the award of any degree or diploma.

Supervisor	**Head of the Department**
Ms. S Annapoorna	**Dr. B. Lakshmi Praveena**
Assistant Professor	**HOD & Professor**
Dept of CSE(AI&ML)	**Dept of CSE(AI&ML)**

External Examiner

DECLARATION

We hereby declare that the work presented in this project entitled **"Money Tracker"** submitted towards completion of Project work in III Year of B.Tech of CSE(AI&ML) at **BVRIT HYDERABAD College of Engineering for Women,** Hyderabad is an authentic record of our original work carried out under the guidance of **Ms. S Annapoorna, Assistant Professor, Department of CSE(AI&ML).**

Sign with Date:
Yashaswiny Sripada
(22WH1A6607)

Sign with Date:
R. Ishwarya
(22WH1A6609)

Sign with Date:
S. Aishwarya
(22WH1A6644)

Sign with Date:
N. Vaishnavi
(22WH1A6645)

ACKNOWLEDGEMENT

We would like to express our sincere thanks to **Dr. K. V. N. Sunitha, Principal, BVRIT HYDERABAD College of Engineering for Women**, for her support by providing the working facilities in the college.

Our sincere thanks and gratitude to **Dr. B. Lakshmi Praveena, Head of the Department, Department of CSE(AI&ML), BVRIT HYDERABAD College of Engineering for Women,** for all timely support and valuable suggestions during the period of our project.

We are extremely thankful to our Internal Guide, **Ms. S Annpoorna, Assistant Professor, CSE(AI&ML), BVRIT HYDERABAD College of Engineering for Women,** for her constant guidance and encouragement throughout the project.

Finally, we would like to thank our Project Coordinator, all Faculty and Staff of CSE(AI&ML) department who helped us directly or indirectly. Last but not least, we wish to acknowledge our **Parents** and **Friends** for giving moral strength and constant encouragement.

Yashaswiny Sripada (22WH1A6607)
R. Ishwarya (22WH1A6609)
S. Aishwarya (22WH1A6644)
N. Vaishnavi (22WH1A6645)

ABSTRACT

Money tracker project focuses on designing a Money Tracker UI to monitor and visualize the monthly expenses of four accounts. The application, developed using Flutter, offers a streamlined interface for recording and categorizing transactions for each account. Users can input their income and expenditures, track account-specific activity, and analyze their financial habits with ease.

At the end of the month, the app will generate a comparative graphical representation displaying the expenses of all four accounts, providing clear insights into spending patterns and helping users identify areas of improvement. The design prioritizes simplicity, responsiveness, and accessibility, ensuring a smooth user experience across platforms. This project aims to enhance personal finance management by offering a clear and intuitive visualization of multi-account expenditures in one cohesive interface.

PROBLEM STATEMENT

Managing personal finances across multiple accounts can be a challenging and time-consuming task. Individuals often struggle to track and categorize their expenses effectively, leading to a lack of clarity about their spending habits. This issue is further compounded when trying to monitor and compare expenses across multiple accounts over a specific period, such as a month. Without a clear and organized system, users may find it difficult to identify patterns, set budgets, or make informed financial decisions.

There is a need for an intuitive and efficient solution that allows users to track their expenses across four accounts, categorize transactions, and visualize their spending through a comparative graph at the end of each month. Such a solution would empower users to take better control of their finances, ensure transparency across accounts, and simplify financial planning.

FILE STRUCTURE

Project_ Money tracker

```
├── .dart_tool
├── .idea
├── android
├── build
├── ios
├── lib
│   ├──  models
│   │   ├── expense.dart
│   ├── screens
│   │   ├── add_expense_screen.dart
│   │   ├── expense_input_screen.dart
│   │   ├── graph_screen.dart
│   │   ├── home_screen.dart
│   │   ├──signup_screen.dart
│   │   ├──summary_screen.dart
│   ├── widgets
│   │   ├── expense_chart.dart
│   │   ├── expense_form,dart
│   │   ├── expense_list.dart
├── main.dart
├── linux
│   ├── flutter
│   │   ├── gitignore
│   │   ├──  CMakeLists.txt
│   │   ├── main.cc
│   │   ├── my_application.cc
│   │   ├──  my_application.h
├── macos
├── test
│   ├── widget_test.dart
├── web
```

SOURCE CODE

```
  #models
#expenses.dart

class Expense {
  final String id;
  final String account;
  final String description;
  final double amount;
  final DateTime date;

  Expense({
    required this.id,
    required this.account,
    required this.description,
    required this.amount,
    required this.date,
  });
}
#screens
#add_expense_screen.dart
import 'package:flutter/material.dart';
class AddExpenseScreen extends StatefulWidget {
  final List<String> accounts;
 final Function(String account, Map<String, dynamic> expense)
```

```
onAddExpense;
const AddExpenseScreen({super.key, required this.accounts, required
this.onAddExpense});
  @override
_AddExpenseScreenState   createState()=>   _AddExpenseScreenState();
}
class _AddExpenseScreenState extends State<AddExpenseScreen> {
  final _formKey = GlobalKey<FormState>();
  String? _selectedAccount;
String _description = "";
double _amount = 0.0;
void _submitForm() {
    if  (_formKey.currentState!.validate())  {
      _formKey.currentState!.save();
      widget.onAddExpense(
        _selectedAccount!,
        {
          "description": _description,
          "amount": _amount,
          "date": DateTime.now(),
        },
      );
      Navigator.of(context).pop();
    }
  }
```

```
@override
 Widget build(BuildContext context) {
   return Scaffold(
     appBar:  AppBar(
       title: const Text("Add Expense"),
     ),
     body: Padding(
       padding: const EdgeInsets.all(16.0),
       child: Form(
         key: _formKey,
         child:  Column(
         children: [
             DropdownButtonFormField<String>(
               value: _selectedAccount,
               onChanged: (value) {
                 setState(() {
                   _selectedAccount = value;
                 });
               },
               items: widget.accounts
                   .map((account) => DropdownMenuItem(
                         value: account,
                         child: Text(account),
                       ))
                   .toList(),
```

```
            decoration: const InputDecoration(labelText: "Account"),
            validator: (value) =>
                value == null ? "Please select an account" : null,
          ),
          TextFormField(
            decoration: const InputDecoration(labelText: "Description"),
            onSaved: (value) {
              _description = value!;
            },
            validator: (value) =>value!.isEmpty ? "Please enter a description" : null,
          ),
          TextFormField(
            decoration: const InputDecoration(labelText: "Amount"),
            keyboardType: TextInputType.number,
            onSaved: (value) {
              _amount = double.parse(value!);
            },
            validator:(value)=>value!.isEmpty ? "Please enter an amount" : null,
          ),
          const SizedBox(height: 20),
          ElevatedButton(
            onPressed: _submitForm,
            child: const Text("Add Expense"),
```

```
            ),
          ],
        ),
      ),
    ),
  );
 }
}
#expense_input_screen.dart
import 'package:flutter/material.dart';
import 'graph_screen.dart';
class ExpenseInputScreen extends StatefulWidget {
 final String accountName;
const ExpenseInputScreen({super.key, required this.accountName});
@override
State<ExpenseInputScreen>createState()=>_ExpenseInputScreenSta()
;
}
class _ExpenseInputScreenState extends State<ExpenseInputScreen>
{
 finalTextEditingControllerdescriptionController =
TextEditingController();
 final amountController = TextEditingController();
List<Map<String, dynamic>> expenses = [];
```

```
staticMap<String,List<Map<String,dynamic>>>allAccountsExpense
= {};
@override
  void initState() {
    super.initState();
    if (!allAccountsExpenses.containsKey(widget.accountName)) {
      allAccountsExpenses[widget.accountName] = [];
    }
  }
 @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Expenses for ${widget.accountName}"),
        backgroundColor: Colors.teal,
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          children: [
            TextField(
              controller: descriptionController,
              decoration: const InputDecoration(
              labelText: "Expense Description",
              border: OutlineInputBorder(),
```

```
            ),
          ),
          const SizedBox(height: 16),
          TextField(
            controller: amountController,
            decoration: const InputDecoration(
            labelText: "Amount",
              border: OutlineInputBorder(),
            ),
            keyboardType:  TextInputType.number,
          ),
          const SizedBox(height: 16),
          ElevatedButton(
            style: ElevatedButton.styleFrom(
              backgroundColor: Colors.teal.shade400,
            ),
            onPressed: () {
              setState(() {
                final expense = {
                  "description": descriptionController.text,
                  "amount":  double.tryParse(amountController.text)  ??  0.0,
                };

allAccountsExpenses[widget.accountName]?.add(expense);
                descriptionController.clear();
```

```
                amountController.clear();
              });
            },
            child: const Text("Add Expense"),
          ),
          const SizedBox(height: 16),
          Expanded(
            child: ListView.builder(
              itemCount:
allAccountsExpenses[widget.accountName]?.length ?? 0,itemBuilder:
(context, index) {
                final                              expense                              =
allAccountsExpenses[widget.accountName]![index];
                return Card(
                  margin: const EdgeInsets.symmetric(vertical: 4),
                  child: ListTile(
                    title: Text(expense["description"]),
                    trailing: Text("\$${expense["amount"]}"),
                  ),
                );
              },
            ),
          ),
        ],
      ),
```

```
      ),
      floatingActionButton: FloatingActionButton(
        backgroundColor: Colors.teal,
        onPressed: () {
          Navigator.push(
          context,
          MaterialPageRoute(

builder:(context)=>GraphScreen(allExpenses:allAccountsExpenses),
            ),
          );
        },
        child: const Icon(Icons.bar_chart),
      ),
    );
  }
}
```

#graph_screen.dart

```
import 'package:fl_chart/fl_chart.dart'; import
'package:flutter/material.dart';
class GraphScreen extends StatelessWidget {
  final Map<String, List<Map<String, dynamic>>> allExpenses;
const GraphScreen({super.key, required this.allExpenses});
@override
```

```
  Widget build(BuildContext context) {
    final List<Color> barColors = [Colors.blue, Colors.green, Colors.orange,
Colors.red];
    final accountNames = allExpenses.keys.toList();
return Scaffold(
      appBar: AppBar(
        title: const Text("Expenses Graph"),
        backgroundColor: Colors.teal,
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0), child:
        Column(
          crossAxisAlignment: CrossAxisAlignment.stretch,
          children: [
            const Text(
              'Monthly Expense Comparison',
              textAlign: TextAlign.center,
              style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold),
            ),
            const SizedBox(height: 16), Expanded(
              child: BarChart(
                BarChartData(
                  borderData: FlBorderData(show: true),
                  gridData: FlGridData(show: true), titlesData:
                  FlTitlesData(
                    leftTitles: AxisTitles(
                      sideTitles: SideTitles(
                      showTitles: true,
```

```
            reservedSize: 40,
            getTitlesWidget: (value, meta) {
            return Text(
              value.toInt().toString(),
              style: const TextStyle(fontSize: 12),
            );
            },
          ),
        ),
        bottomTitles: AxisTitles( sideTitles:
          SideTitles( showTitles: true,
          getTitlesWidget: (value, meta) {
            int index = value.toInt();
            if (index >= 0 && index < accountNames.length) {
              return Text(
                accountNames[index],
                style: const TextStyle(fontSize: 12),
              );
            }
            return const SizedBox.shrink();
          },
        ),
      ),
    ),

barGroups:accountNames.asMap().entries.map<BarChartGroupData>((en try) {
          int index = entry.key;
```

```
                final account = entry.value;

                double total = allExpenses[account]!.fold<double>( 0.0,
                  (sum, expense) => sum + expense['amount'],);
                return BarChartGroupData(
                  x: index,
                  barRods: [
                    BarChartRodData(toY:total,color:barColors[index], width:20,
borderRadius: BorderRadius.circular(4),
                    ),
                  ],
                );
              }).toList(),
            ),
          ),
        ),
        const SizedBox(height: 16), Row(
          mainAxisAlignment:      MainAxisAlignment.spaceEvenly,
          children: accountNames.asMap().entries.map((entry) { int
          index = entry.key;
            String name = entry.value;
            return Row(
              children: [
                Container(width: 12, height: 12, color: barColors[index]),
                const SizedBox(width: 4),
                Text(name),
              ],
```

```
                );
              })
.toList(),
            ),
          ],
        ),
      ),
    );
  }
}
#home_screen.dart
import 'package:flutter/material.dart'; import
'expense_input_screen.dart';
class HomeScreen extends StatelessWidget {
  const HomeScreen({super.key});
final List<Map<String, dynamic>> accounts = const [
    {"name": "Aishwarya"},
    {"name": "Ishwarya"},
    {"name": "Vaishnavi"},
    {"name": "Yashaswiny"},
  ];
 @override
  Widget build(BuildContext context) { return
    Scaffold(
      appBar: AppBar(
        title: const Text("Select an Account"),
        backgroundColor: Colors.teal,
      ),
 body: Padding(
```

```
        padding: const EdgeInsets.all(16.0), child:
        GridView.builder(
          gridDelegate: const SliverGridDelegateWithFixedCrossAxisCount(
            crossAxisCount: 2, // Two cards in each row
            crossAxisSpacing: 12,
            mainAxisSpacing: 12,
            childAspectRatio: 3, // Adjusts the height of the cards
          ),
          itemCount: accounts.length,
          itemBuilder: (context, index) {
          final account = accounts[index];
          return GestureDetector(
              onTap: () {
                Navigator.push(
                context,
                MaterialPageRoute(

builder:(context)=>ExpenseInputScreen(accountName:account["name"]),
                  ),
                );
              },
              child: Card(
                elevation: 4,

shape:RoundedRectangleBorder(borderRadius:BorderRadius.circular(8)),
                color:Colors.primaries[index%Colors.primaries.length].shade200,
                child: Center(child: Text(account["name"],
textAlign: TextAlign.center,
                    style: const TextStyle(
```

```
                    fontSize: 18,
                    fontWeight: FontWeight.bold,
                    color: Colors.black87,
                  ),
                ),
              ),
            ),
          );
        },
      ),
    ),
  );
 }
}
```

#login_screen.dart

```
import 'package:flutter/material.dart'; import
'signup_screen.dart';
import 'home_screen.dart';

class LoginScreen extends StatelessWidget {
 const LoginScreen({super.key});

 @override
 Widget build(BuildContext context) { return
  Scaffold(
   body: Container(
    decoration: const BoxDecoration(
     gradient: LinearGradient(
```

```
      colors: [Colors.teal, Colors.tealAccent],
      begin: Alignment.topLeft,
      end: Alignment.bottomRight,
    ),
  ),
  child: Padding(
    padding: const EdgeInsets.all(16.0),
    child: Column(
      mainAxisAlignment: MainAxisAlignment.center,
      children: [
        const Text(
          'Money Tracker Login',
          style: TextStyle(
            color: Colors.white,
            fontSize: 30,
            fontWeight: FontWeight.bold,
          ),
        ),
        const SizedBox(height: 40),
        _buildInputField('Username', Icons.person), const
        SizedBox(height: 16),
        _buildInputField('Password', Icons.lock, obscureText: true), const
        SizedBox(height: 32),
        ElevatedButton(
          style:     ElevatedButton.styleFrom(
            backgroundColor:     Colors.white,
            foregroundColor: Colors.teal,
            padding:  const  EdgeInsets.symmetric(vertical:  12,  horizontal:24),
            shape: RoundedRectangleBorder(
```

```
            borderRadius: BorderRadius.circular(8),
          ),
        ),
        onPressed: () {
          Navigator.push(
          context,
          MaterialPageRoute(
              builder: (context) => const HomeScreen(),
            ),
          );
        },
        child: const Text('Login', style: TextStyle(fontSize: 18)),
      ),
      TextButton(
        onPressed: () {
        Navigator.push(
            context, MaterialPageRoute(
              builder: (context) => const SignupScreen(),
            ),
          );
        },
        child: const Text(
          "Don't have an account? Sign up",
          style: TextStyle(color: Colors.white),
        ),
      ),
    ],
  ),
```

```
        ),
      ),
    );
  }

  Widget _buildInputField(String hint, IconData icon, {bool obscureText =
false}) {
    return TextField( obscureText:
      obscureText, decoration:
      InputDecoration( hintText:
      hint,
        prefixIcon: Icon(icon, color: Colors.teal), filled:
        true,
        fillColor: Colors.white.withOpacity(0.8),
        border: OutlineInputBorder( borderRadius:
        BorderRadius.circular(8),
        ),
      ),
    );
  }
}
```

widgets

#expense_chart.dart

```
import 'package:fl_chart/fl_chart.dart'; import
'package:flutter/material.dart';

class ExpenseChart extends StatelessWidget {
  final List<Map<String, dynamic>> accounts;
```

```
  const ExpenseChart({super.key, required this.accounts});

  @override
  Widget build(BuildContext context) {
    final List<Color> barColors = [Colors.blue, Colors.green, Colors.orange,
Colors.red];

    return Padding(
      padding: const EdgeInsets.all(16.0), child:
      Column(
        crossAxisAlignment: CrossAxisAlignment.stretch,
        children: [
          const Text(
            'Monthly Expense Comparison',
            textAlign: TextAlign.center,
            style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold),
          ),
          const SizedBox(height: 16),
          Expanded(
            child: BarChart(
              BarChartData(
                borderData: FlBorderData(show: false),
                gridData: FlGridData(show: false),
                titlesData: FlTitlesData(
                  leftTitles: AxisTitles(
                    sideTitles: SideTitles(
                    showTitles: true,
                    reservedSize: 40,
```

```
                getTitlesWidget: (value, meta) {
                  return Text(
                  value.toInt().toString(),
                    style: const TextStyle(fontSize: 12),
                  );
                },
              ),
            ),
            bottomTitles: AxisTitles(
              sideTitles: SideTitles( showTitles:
              true, getTitlesWidget: (value,
              meta) {
                int index = value.toInt();
                if (index >= 0 && index < accounts.length) {
                  return Text(
                    accounts[index]['name'],
                    style: const TextStyle(fontSize: 12),
                  );
                }
                return const SizedBox.shrink();
              },
            ),
          ),
        ),
        barGroups:
accounts.asMap().entries.map<BarChartGroupData>((entry) {
          int index = entry.key;
          final account = entry.value;
```

```
        double total = account['expenses'].fold<double>(
          0.0,
          (sum, expense) => sum + expense['amount'],
        );

        return BarChartGroupData( x:
          index,
          barRods: [
            BarChartRodData(
            toY: total,
              color: barColors[index], width:
              20,
              borderRadius: BorderRadius.circular(4),
            ),
          ],
        );
      }).toList(),
    ),
  ),
),
const SizedBox(height: 16),
Row(
  mainAxisAlignment: MainAxisAlignment.spaceEvenly,
  children: accounts.asMap().entries.map((entry) {
    int index = entry.key;
    String name = entry.value['name'];
    return Row(
      children: [
        Container(width: 12, height: 12, color: barColors[index]),
```

```
                const SizedBox(width: 4),
                Text(name),
              ],
            );
          }).toList(),
        ),
      ],
    ),
  );
 }
}
```

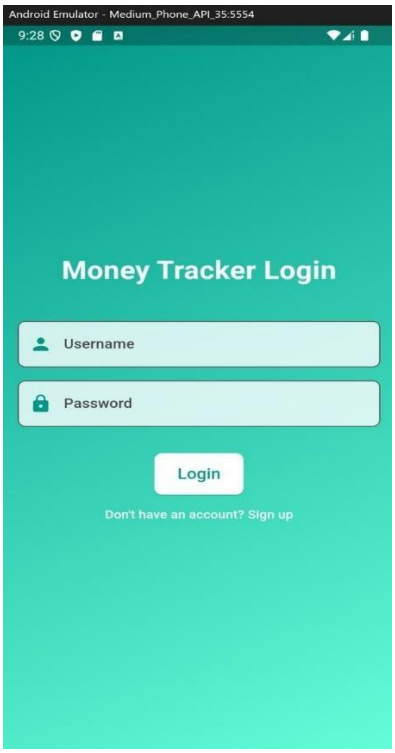
#expense_list.dart

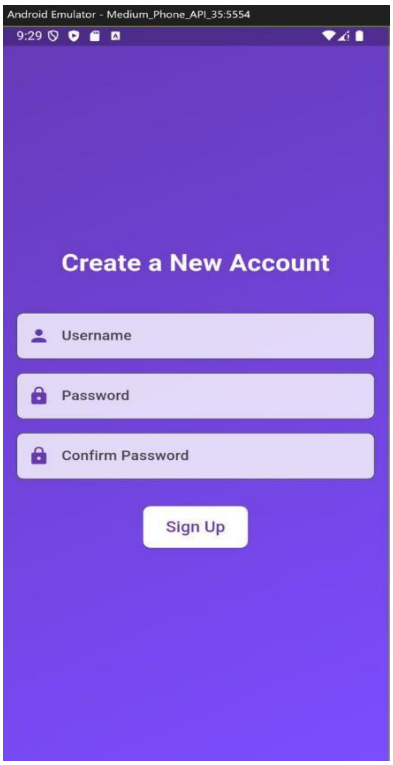
```
import 'package:flutter/material.dart'; class

ExpenseList extends StatelessWidget {
 final List<Map<String, dynamic>> accounts;

 const ExpenseList({super.key, required this.accounts});

 @override
 Widget build(BuildContext context) {
  return ListView.builder( itemCount:
  accounts.length, itemBuilder:
  (context, index) {
    final account = accounts[index];
    double totalExpenses = account["expenses"]
      .fold(0.0, (sum, expense) => sum + expense["amount"]);
```

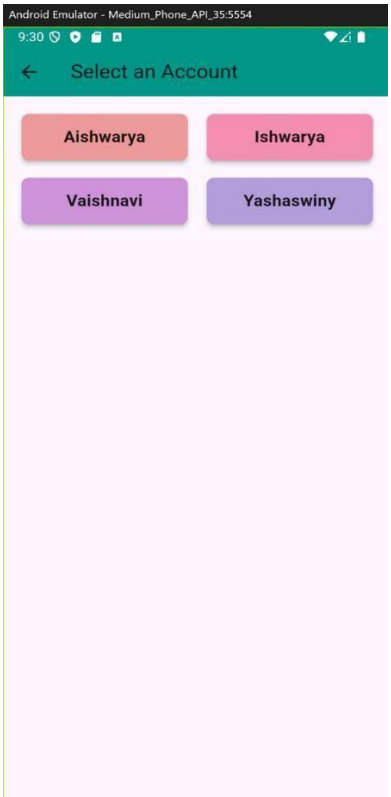
```
        return Card(
          margin: const EdgeInsets.symmetric(vertical: 8.0), child:
          ListTile(
            title: Text(
              account["name"],
              style: const TextStyle(fontWeight: FontWeight.bold),
            ),
            subtitle: Text(
              "Total Expenses: \$${totalExpenses.toStringAsFixed(2)}",
            ),
            trailing: const Icon(Icons.arrow_forward_ios, color: Colors.grey), onTap:
            () {
              // Navigate to account details (future feature)
            },
          ),
        );
      },
    );
  }
}
#main.dart
import 'package:flutter/material.dart'; import
'screens/login_screen.dart';

void main() {
  runApp(const MoneyTrackerApp());
}

class MoneyTrackerApp extends StatelessWidget {
```

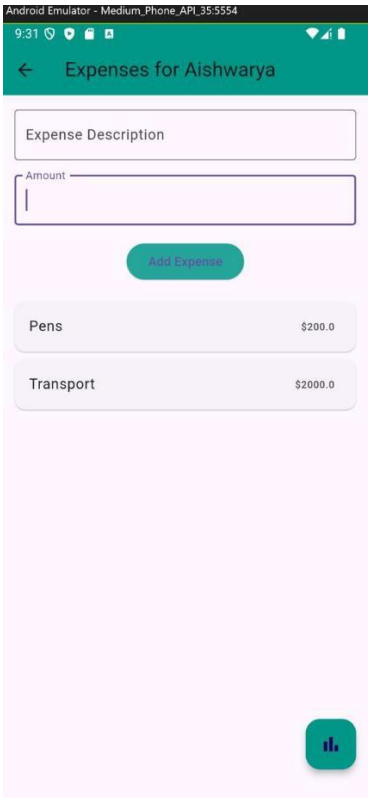
```
  const MoneyTrackerApp({super.key});

  @override
  Widget build(BuildContext context) { return
    MaterialApp(
    debugShowCheckedModeBanner: false,
    title: 'Money Tracker',
      theme: ThemeData( primarySwatch:
        Colors.teal,
      ),
      home: const LoginScreen(),
    );
  }
}
```

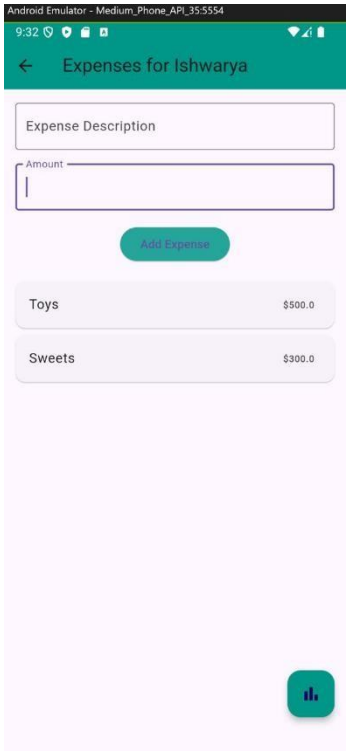
OUTPUT:

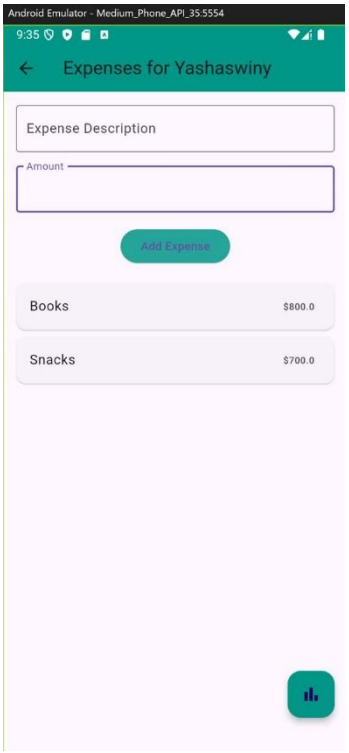

Android Emulator - Medium_Phone_API_35:5554
9:28
Money Tracker Login
Username
Password
Login
Don't have an account? Sign up

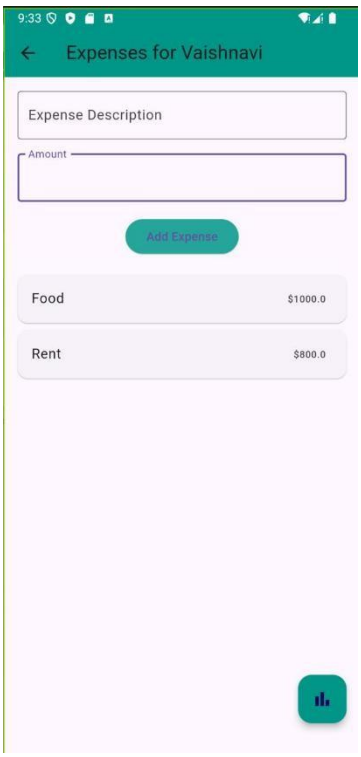

Android Emulator - Medium_Phone_API_35:5554
9:29
Create a New Account
Username
Password
Confirm Password
Sign Up

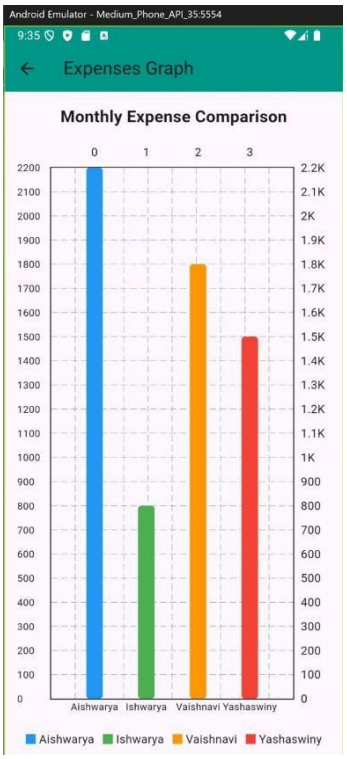

Android Emulator - Medium_Phone_API_35:5554
9:30
Select an Account
Aishwarya
Ishwarya
Vaishnavi
Yashaswiny


Android Emulator - Medium_Phone_API_35:5554
9:31
Expenses for Aishwarya
Expense Description
Amount
Add Expense
Pens
$200.0
Transport
$2000.0


Android Emulator - Medium_Phone_API_35:5554
9:32
Expenses for Ishwarya
Expense Description
Amount
Add Expense
Toys $500.0
Sweets $300.0


Android Emulator - Medium_Phone_API_35:5554
9:35
Expenses for Yashaswiny
Expense Description
Amount
Add Expense
Books $800.0
Snacks $700.0


9:33
Expenses for Vaishnavi
Expense Description
Amount
Add Expense
Food $1000.0
Rent $800.0


Android Emulator - Medium_Phone_API_35:5554
9:35
Expenses Graph
Monthly Expense Comparison
0 1 2 3
2200 2100 2000 1900 1800 1700 1600 1500 1400 1300 1200 1100 1000 900 800 700 600 500 400 300 200 100 0
2.2K 2.1K 2K 1.9K 1.8K 1.7K 1.6K 1.5K 1.4K 1.3K 1.2K 1.1K 1K 900 800 700 600 500 400 300 200 100 0
Aishwarya Ishwarya Vaishnavi Yashaswiny
Aishwarya Ishwarya Vaishnavi Yashaswiny