Personal Finance Tracker

Project Overview

Group 1: Ashish, Edward, Justin, Nicole, Sudip

Project Charter

Project Purpose:

- **Develop software application**
  - To track personal financial activity
  - Manage personal finances
  - Analyze collected data
  - Help user achieve financial goals
  - Promote financial literacy and planning

Project Timeline and Budget:

- → Within 6 months of project initiation
- → Up to $ 500

Key Participants & Roles:

- Ashish Acharya
  - Quality Assurance
- Edward Goodall
  - Lead Developer
- Justin Hill
  - Financial Advisor/Consultant
- Nicole Taylor
  - Project manager
- Sudip Aryal
  - Front End Developer

System Vision Document

Scope

Our app prioritizes secure access, straightforward expense tracking, and uncomplicated financial visualization. Users effortlessly categorize transactions and plan monthly budgets. The app stands out with clear and insightful financial graphs, providing users with a visual representation of their financial trends. Additionally, it offers personalized financial insights, potential notifications, and enhanced security features. It's a practical and user-friendly solution for streamlined financial management.

List of functions the app can do

- View Financial Report Graphs:
  - Yearly Line Graph
  - Spending Habits Pie Chart
- Track Monthly Transactions and Categorize Them
- Edit and Plan Monthly Budgets
- Financial Insights and Recommendations
- Secure Login Screen and Account Management Tools

System Capabilities

User Authentication: Securely manage accounts using Java Spring Security or alternatives like OAuth and Firebase.

Expense Tracking: Expense and income input, considering native or web development (React).

Transaction Categorization: Automatically categorize transactions using tagging or advanced methods like machine learning.

Data Visualization: Generate interactive financial charts using D3.js or Chart.js.

Budget Management: Utilize MySQL and PHP to enable users to set and monitor budgets in-app.

Advanced system capabilities

Financial Insights: Provide personalized financial insights and predictions using data analytics and machine learning for trend analysis.

Notification and Alerts: Send notifications, including budget limit alerts, using machine learning and push notifications via Firebase Cloud Messaging or Apple Push Notification Service.

Data Security and Privacy: Ensure user data security and privacy through SSL/TLS for data transmission, encryption algorithms, and secure cloud storage solutions.

Possible External Technologies

Machine Learning: Powers advanced features like transaction categorization, financial insights, and predictions, enhancing user experience.

MySQL and PHP: Manages storage and retrieval of budget data, supporting efficient budget management within the financial system.

Firebase Cloud Messaging and Apple Push Notification Service: Enables timely notifications and alerts to users for important financial events, enhancing user engagement.

SSL/TLS: Ensures secure data transmission, safeguarding user information during interactions with the financial management system.

Secure Cloud Storage Solutions: Safeguards user data by exploring secure cloud storage options, maintaining privacy and security standards

Possible Third-Party Libraries

Java Spring Security: Ensures secure user authentication and profile management within the financial management system.

OAuth: Alternative Secure Authentication Method for User Access.

Firebase Authentication: User-Friendly and Secure Authentication Option.

React: Enhances the user interface of the financial management system for a dynamic and responsive experience.

D3.js and Chart.js: Creates Interactive Data Visualizations for Clear Representation of Financial Data.

Project Schedule

Benefits of the Project Schedule:

- Efficient Resource Allocation: Ensures each team member focuses on their specialized tasks.
- Timely Feedback Loops: Regular check-ins and feedback sessions promote collaboration and quick issue resolution.
- Mitigation of Risks: Identifying and addressing potential issues early in the project timeline.
- Clear Milestones: Defined deadlines help in tracking progress and meeting project goals.

Tasks Descriptions

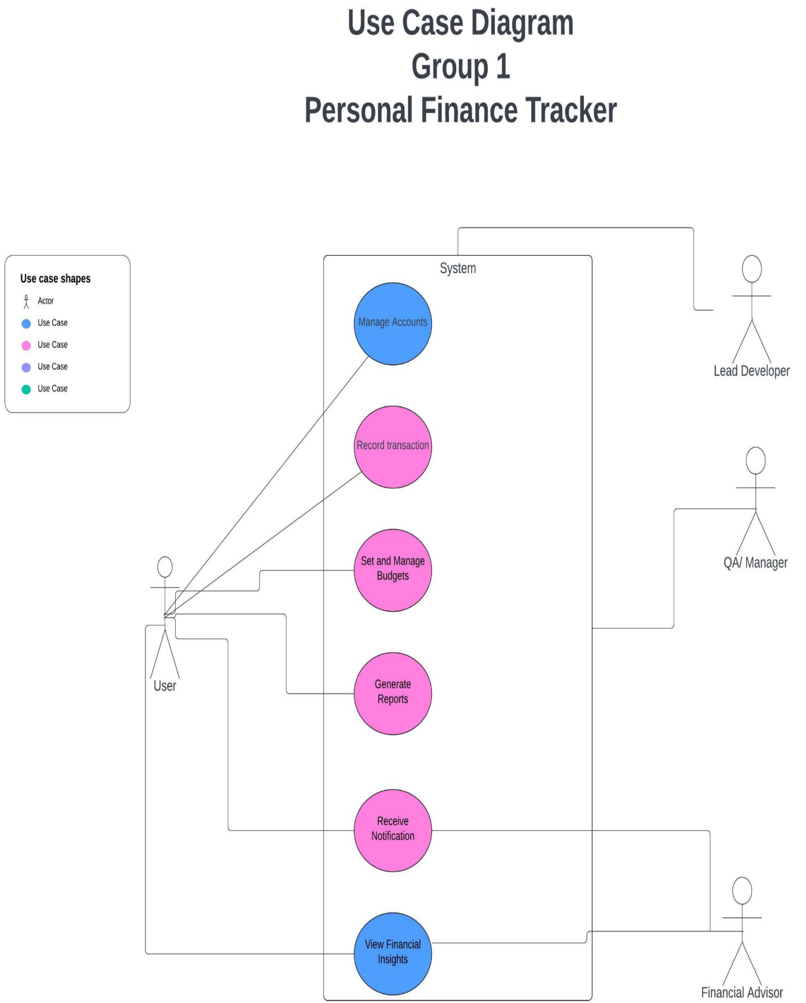
The following are the tasks overview definitions that describe why a task should be completed.

- Goal - Define and communicate the overarching project objectives, ensuring alignment with stakeholder expectations.
- Milestone: Identify key project milestones, breaking down the project into manageable phases with measurable achievements and established success criteria.
- Low Risk:Execute tasks with minimal potential for complications, ensuring steady progress and maintaining project momentum.

Tasks Descriptions Cont'd.

- Medium Risk:Address and manage tasks with moderate complexity, utilizing expertise to proactively mitigate potential challenges and deviations.
- High Risk:Oversee critical tasks with high-risk factors, employing specialized skills to navigate complexities and ensure successful outcomes.
- On Track:Monitor overall project progress, regularly assess individual and team performance, and implement corrective actions to maintain alignment with goals.

Object-oriented UML models



Brief Description

1.Manage Accounts:
- User-driven actions;Financial Advisors review accounts for tailored advice

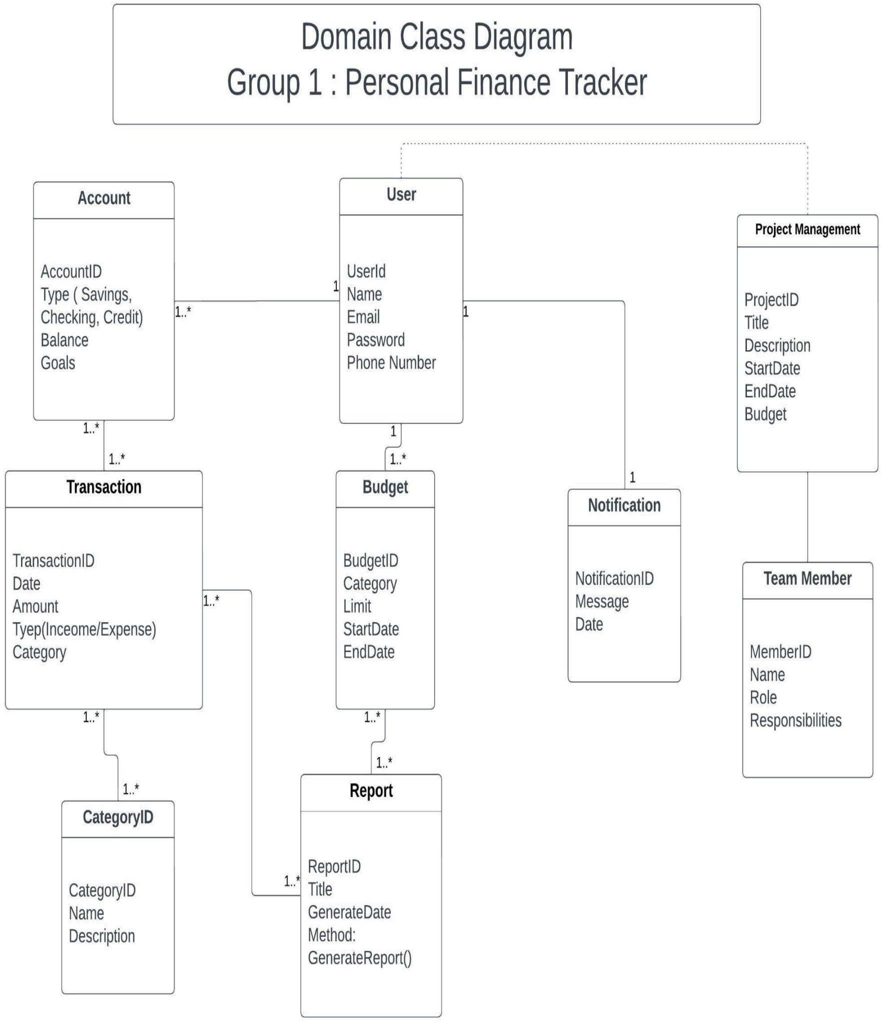
2.Record Transaction
- Quality Assurance ensures accurate recording and validation

3.Set and Manage Budgets
- Users set budget limits for expense categories
- Financial Advisors assist in establishing realistic budgets

4.Generate Reports
- Lead Developer ensures efficient and accurate report generation

5. Receive Notification
- Users receive notifications, alerts, or financial tips
- QA verifies accuracy and relevance of information

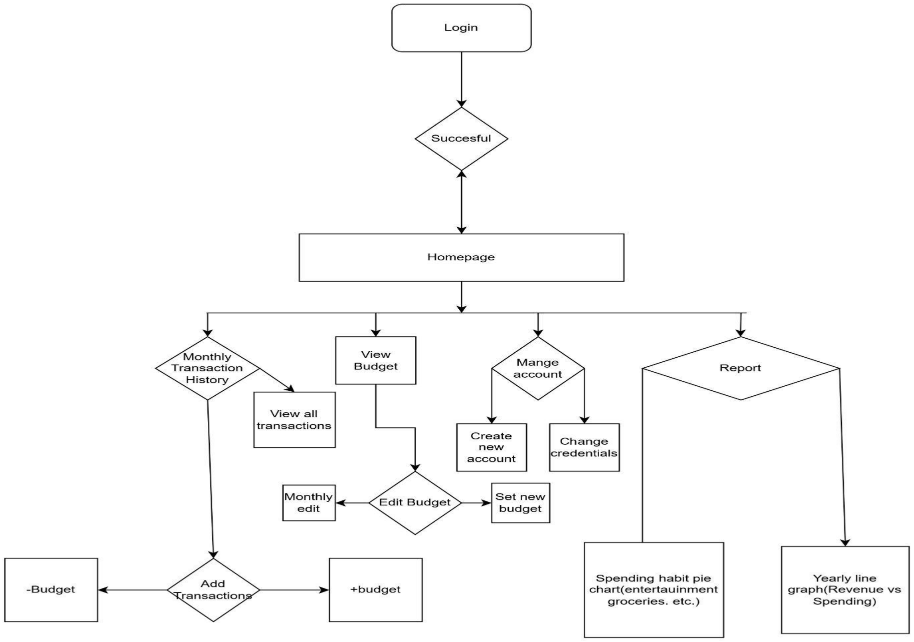
6.View Financial Insights:
- Users access insights into spending patterns and financial health
- Financial Advisors utilize insights for personalized recommendations



Brief Description

- User Class:
    1. Attributes: UserID, Name, Email, Password, Phone Number.
    2. Represents system users.
- Account Class:
    1. Attributes: AccountID, Type (Savings, Checking, Credit), Balance, Goals.
    2. Represents user accounts with financial details.
- Transaction Class:
    1. Attributes: TransactionID, Date, Amount, Type (Income/Expense), Category.
    2. Represents financial transactions recorded in user accounts.
- Budget Class:
    1. Attributes: BudgetID , Category, Limit, StartDate, EndDate.
    2. Represents user-defined budgets for expense categories.
- Report Class:
    1. Attributes: ReportID, CategoryID, Name, Description.
    2. Represents financial reports generated based on user data.
- Notification Class:
    1. Attributes: NotificationID, Message, Date.
    2. Represents notifications sent to users regarding account and system updates.
- Project Management Class:
    1. Attributes: ProjectID, Title, Description, StartDate, EndDate.
    2. Represents projects within the finance system.
- Team Member Class:
    1. Attributes: MemberID, Name, Role, Responsibilities.
    2. Represents team members associated with projects.
- Category Class:
    1. Attributes: CategoryID, Name.
    2. Represents transaction categories.
- Report Class:
    1. Attributes: ReportID, Title, GenerateDate , Method, GenerateReport()
    2. Represents a method to generate financial reports based on transaction and budgets

Activity Diagram
Group 1
Personal Finance Tracker



Brief Description

1.Login: User logs in.

2.Homepage: Monthly History, View Transactions, View Budget

3.Monthly Transaction History:Displays monthly transactions.

4.View Transactions:Shows all user transactions.

5.View Budget:Displays current budget.

6. Manage Account:Create New Account, Edit Budget, Monthly Edit.

7.Create New Account:Guides user through account creation.

8.Report:Options for generating financial reports.

9.Change Credentials:Allows user to update login information.

10.Edit Budget:Options for modifying budget details.

11.Set New Budget:Guides user in setting a new budget.

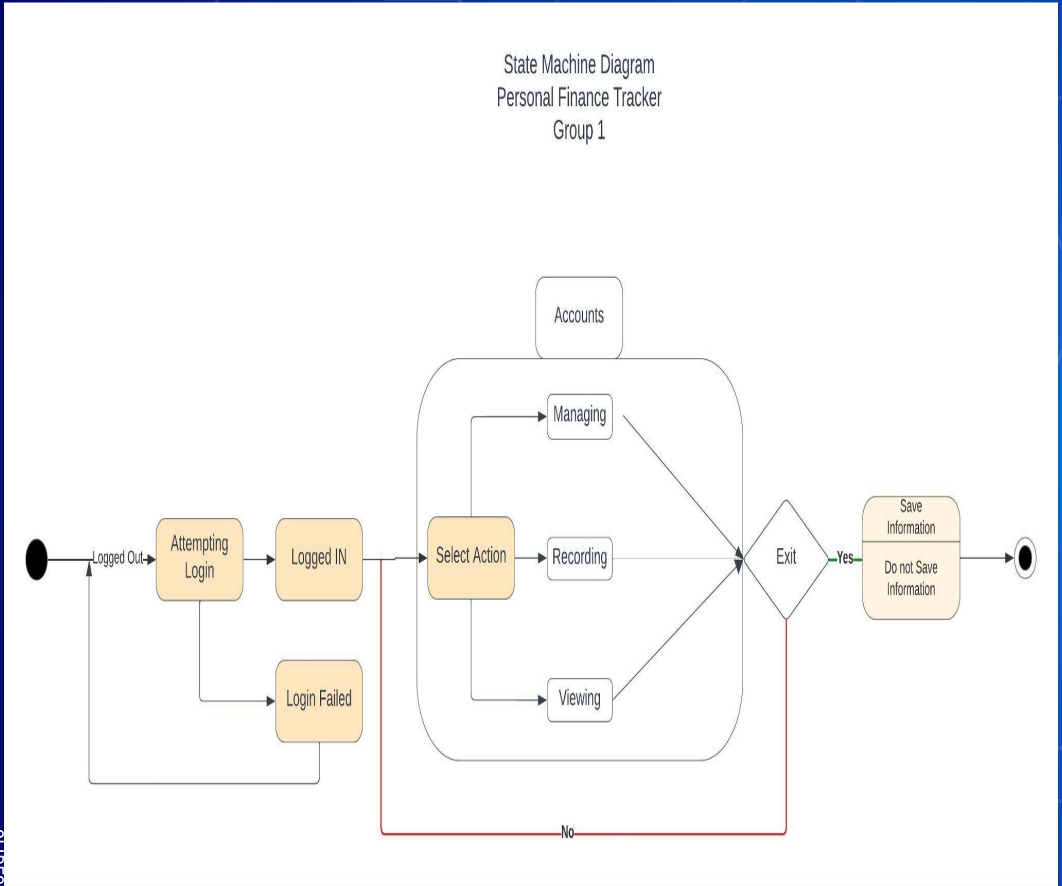
12.Monthly Edit:Options for monthly adjustments.

13.Add Transactions:Guides user in entering new transactions.

14.Spending Habit Pie Chart:Generates pie chart illustrating spending habits.

15.Yearly Line Graph:Generates line graph comparing revenue vs. spending.

Brief Description



Logged Out:
- Initial state when the user is not logged in.

Attempting Login:
- Transition from Logged Out when the user tries to log in.
- If login fails, transition back to Logged Out.
- If login is successful, transition to Logged In.

Logged In:
- State indicating a successful login.
- Leads to the Selection state.

Selecting Options:
- Transition from Logged In
- Leads to either Managing, Viewing or Recording inside user account based on the user's choice.

Managing:
- State where the user manages their personal finance data.

Viewing:
- State where the user views transactions or reports.
- User can choose not to save information and proceed to Exit State

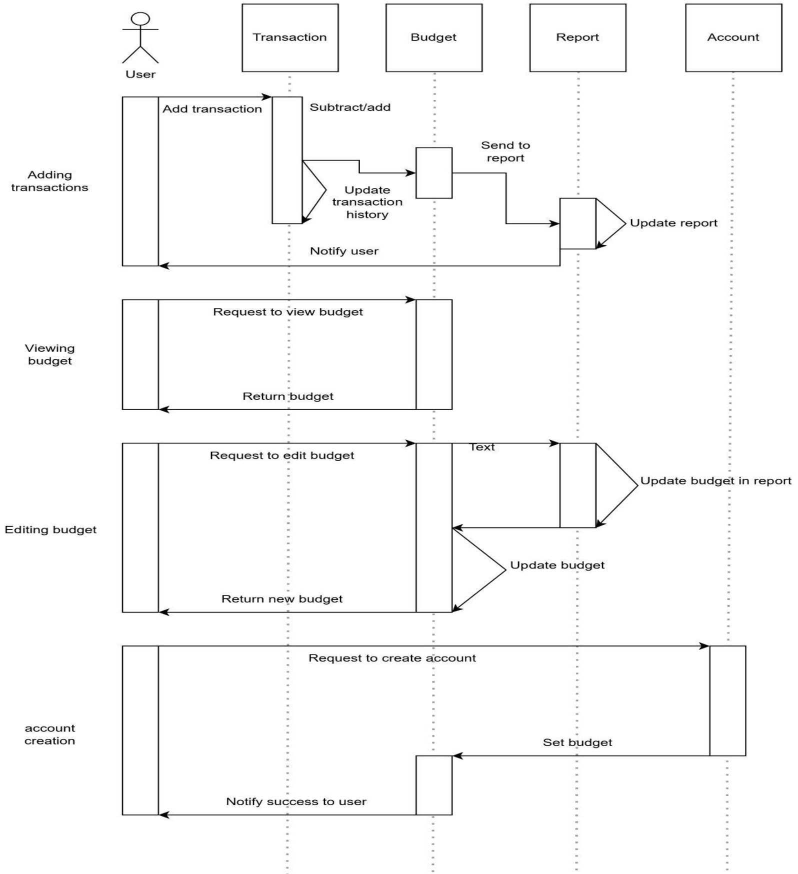
Recording:
- State where the user records a new transaction.
- User can choose to save information and proceed to Exit State

Exit:
- Transition from Selection Action user decides to exit.
- Leads to the either Save Information or Do not Save Information

Save Information :
- Transition from Exit when the user decides to exit and save information.
- Leads to the end state where the user is logged out

Do not Save Information :
- Transition from Exit when the user decides not to save the viewed information.
- Leads to the end state where the user is logged out

Sequence Diagram
Group 1
Personal Finance Tracker



Brief Description

1.User Adds Transaction:
- User adds transaction.
- System updates account.
- Notifies user.

2.User Requests Budget:
- User requests to view/edit budget.
- System processes request, returns budget or updates as needed.

3. User Creates Account:
- User creates a new account.
- System processes request.
- Notifies user of success.

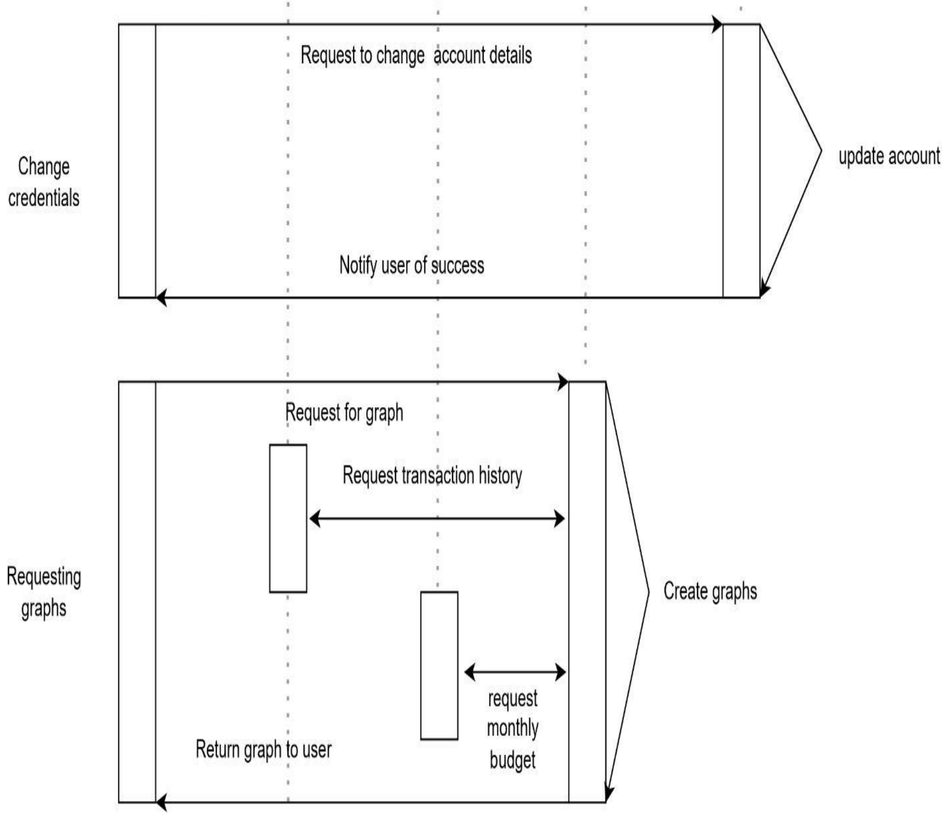
4. User Changes Account Details:
- User requests account detail change.
- System updates account.
- Notifies user.

5. User Requests Graphs:
- User requests graphs.
- System generates and returns graphs.

6. User Changes Credentials:
- User changes login credentials.
- System updates credentials.
- Notifies user.

Continued

Sequence Diagram
Group 1
Personal Finance Tracker



Brief Description

7. User Requests Monthly Budget Update:
- User requests monthly budget update.
- System updates budget in report.
- Returns updated budget.

8. System Updates Transaction History:
- Updates transaction history as transactions are added.

9. System Updates Report:
- Updates financial report based on budget and transaction changes.

10. Notification to User:
- System sends notifications for transaction success, account changes, and updates.

Thank you!

Fonts used: Roboto Mono and Gugi