

# EXPENSE TRACKING



A Minor Project Report

in partial fulfillment of the degree

**Bachelor of Technology**  
in  
**Computer Science & Artificial Intelligence**

**By**

2103A52135

CHANDANA.G

2103A52110

SAIPRIYA TAKKARSU

2103A52107

SRIHITHA.P

2103A52127

SINDHU RAJ.T

2103A52171

NIHAL.T

**Under the Guidance of**

**Dr.T.SAMPATH KUMAR**

**Submitted to**



**SCHOOL OF COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE**  
**SR UNIVERSITY, ANANTHASAGAR, WARANGAL April, 2024.**



**Department of Computer Science and Artificial Intelligence**

**CERTIFICATE**

This is to certify that project entitled “**EXPENSE TRACKING**” is the bonafied work carried out by **Saipriya.T, Chandana.G, Nihal.T, Sindhu Raj.T ,Srihitha.P** as a Course Project for the partial fulfillment to award the degree **BACHELOR OF TECHNOLOGY** in **ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING** during the academic year 2024-2025 under our guidance and Supervision.

**Dr.T.Sampath Kumar**

Asst. Professor,  
S R University,  
Ananthasagar, Warangal.

**Dr. M.Sheshikala**

Prof. & HOD (CSE),  
S R University,  
Ananthasagar, Warangal.

## **ACKNOWLEDGEMENT**

We express our thanks to Course co-coordinator **Dr.T.Sampath Kumar, Asst. Prof.** for guiding us from the beginning through the end of the Course Project. We express our gratitude to Head of the department CS&AI, **Dr. M.Sheshikala, Associate Professor** for encouragement, support and insightful suggestions. We truly value their consistent feedback on our progress, which was always constructive and encouraging and ultimately drove us to the right direction.

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved Dean, School of Computer Science and Artificial Intelligence, **Dr INDRAJEET GUPTA** for his continuous support and guidance to complete this project in the institute.

Finally, we express our thanks to all the teaching and non-teaching staff of the department for their suggestions and timely support.

## **ABSTRACT**

The goal of this project is to create an application for expense tracking using Flutter for mobile app development and Node.js for the backend database. Users of the application can create reports, track and classify their expenses, and set budgets. The backend is made with Node.js and stores user data, categories, expenses, and budgetary information in a database. The database utilized to effectively store and handle the data is Mongo DB. For the development of mobile apps, Flutter is utilized, offering an iOS and Android cross-platform solution. RESTful APIs are used by the app to connect with the Node.js backend, allowing for smooth data transfer between the database and mobile app. Among the application's primary features are Authorization and authentication of users, Keeping track of and classifying expenses, establishing and monitoring budgets, Producing Expense Reports. Through the use of Flutter for mobile app development and Node.js for the backend, the application guarantees a reliable, scalable, and easy-to-use expense tracking solution that helps people manage their money.

# Table of Contents

<b>Chapter No.</b>	<b>Title</b>	<b>Page No.</b>
1.	Introduction	
	1.1. Overview	1
	1.2. Problem Statement	2
	1.3. Existing Methods	3
	1.4. Present work	4
	1.5. Literature survey	5
2.	Requirement Specifications	
	2.1 System Design	7
	2.2 ER Diagram	7-8
	2.3 Flow chart	8
	2.4 Risk Analysis	11
3.	Project Implementation	
	3.1 Proposed System	13
	3.2 Procedure	13-16
4.	Simulation Setup and Results	
	4.1 Simulation setup	17-19
	4.2 Results	20-21
5.	Conclusion with challenges	
	5.1 Conclusion	24
	5.2 Challenges	24
	5.3 References	25

## LIST OF FIGURES

S.No.	FIGURE NAME	Page
1	System Design	7
2	ER Diagram	8
3	Flowchart	9
4	Code lib	17
5	Screens	18
6	Services	19
7	widgets	19
8	App Logo	20
9	Login / Create Account	21
10	How Application Works	21

# CHAPTER 1

## INTRODUCTION

### 1.1 OVERVIEW

Effective cost management is essential for both individuals and enterprises in the fast-paced world of today. Our solution meets this demand by fusing state-of-the-art technologies to deliver a flawless expense tracking experience. Through the use of Node.js for backend development, Flutter for creating mobile apps, and React.js for frontend web development, we provide a flexible platform that accommodates a wide range of user needs and preferences.

**Node.js Backend:** Node.js, which is well-known for its scalability and effectiveness in managing concurrent connections, powers our backend infrastructure. Real-time data processing and seamless front-end-to-backend database connectivity are made possible by Node.js. Node.js's event-driven architecture maximizes efficiency, which makes it the perfect platform for managing complicated tasks like data management and spending tracking.

**Database Integration:** Our solution uses a strong database system to safely store and handle expense-related data, and it integrates seamlessly with Node.js. By utilizing the capabilities of contemporary databases, we guarantee data scalability, integrity, and effective retrieval, enabling customers to easily access their financial data while upholding strict security protocols to safeguard sensitive data.

**Flutter Mobile App Development:** Using Flutter, we created an elegant and user-friendly expenditure tracking application in response to the widespread use of mobile devices. We can provide a uniform user experience on both iOS and Android devices thanks to Flutter's cross-platform capabilities, which also reduce development time and guarantee widespread accessibility. Real-time spending monitoring, adjustable categories, interactive visualizations, and smooth database synchronization are just a few of the features that our Flutter app has to offer.

**Front-end Web Development Using JavaScript, HTML, and CSS Frameworks:** Using HTML, CSS, and JavaScript frameworks such as React.js, we have created a dynamic frontend to cater to users that prefer web-based interfaces. via any device with a web browser, users may manage their costs via a responsive and feature-rich web interface. Thanks to interactive charts and illuminating data visualizations, users can input, categorize, and evaluate their spending with ease thanks to a modern and accessible design.

**React.js Integration:** Our frontend interface provides a seamless and engaging user experience by utilizing React.js to its full potential. Because of React.js's component-based architecture, we can create reusable user interface elements quickly and with ease. We guarantee that our spending monitoring technology stays extremely responsive by using React.js, which lets users interact with their financial data and take actions with the least amount of latency.

## **1.2PROBLEM STATEMENT**

Keeping up with personal finances has grown more difficult in today's hectic society. People frequently find it difficult to keep track of their expenditures, which can result in overspending, trouble making budgets, and stress related to money. Conventional approaches to tracking expenses, such writing notes by hand or managing spreadsheets, take a lot of effort and are prone to mistakes. The issue is further made worse by the absence of accessibility and real-time insights.

A comprehensive and easy-to-use mobile application for tracking expenses is desperately needed to tackle these issues. This program should include easy-to-use tools for creating budgets, classifying transactions, keeping track of spending, and producing informative reports. To protect users' financial information, security and privacy should also come first.

Important stakeholders—individuals, families, and small businesses—need a solution that fits easily into their daily schedules and gives them the practical knowledge they need to manage their finances. The program should also be flexible enough to accommodate different methods to financial management and tailored to the individual needs of the user.

The overall goal of creating a dependable and effective mobile application for tracking expenses is to provide users with the knowledge and resources they need to manage their money better, set and meet goals, and take charge of their financial future.

Even with the increasing recognition of the significance of financial management, people find it difficult to keep an accurate log of their spending because there are insufficiently effective solutions available. Present-day cost monitoring apps frequently have poor usability, insufficient capabilities, and a dearth of customization options, which irritate users and eventually cause them to give up. A feature-rich and easy-to-use mobile application for tracking expenses is therefore desperately needed in order to meet the various needs of users, offer real-time insights into spending habits, and help them make better financial decisions. People frequently find it difficult to maintain discipline in managing their expenses in today's fast-paced lifestyle, which leads to stress and financial mismanagement. Current cost-tracking software frequently lacks customizable features and user-friendly interfaces, making it difficult to engage users. As a result, consumers terminate these programs too soon, which exacerbates their financial problems. Therefore, there is an urgent need for a cutting-edge mobile application for cost monitoring that not only makes the process of recording expenses easier but also provides tools for budgeting, actionable insights, and features that can be customized to meet the unique spending habits and financial objectives of each user.

Even with the widespread availability of digital tools for tracking expenses, many people still find it difficult to keep precise and current records of their outlays. Users' ability to manage their funds properly is hampered by the complexity, limited support for numerous currencies and payment methods, and lack of device synchronization of existing mobile applications. As a result, there is an immediate need for a flexible and



user-friendly mobile application for tracking expenses that can be integrated with a wide range of banking systems, has an easy-to-use interface, and uses cutting-edge algorithms to analyze spending patterns so that users can save money and make wise financial decisions.

### **1.3 EXSISTING METHODS**

- **Pen & Paper:** Using a real notepad or journal, this is the conventional way of recording expenses. Each expense is manually entered by the user, along with the category, date, and amount. Although straightforward and easily obtainable, this approach is error-prone, does not provide real-time insights, and necessitates manual labor for spending pattern analysis.
- **Spreadsheets:** To keep track of their spending, many people use spreadsheet programs like Google Sheets or Microsoft Excel. With columns for the date, amount, category, and remarks, they design personalized templates. Although spreadsheets are more flexible and organized than pen and paper, they still need human data entry and don't have the automated elements that are included in programs specifically designed to manage expenses.
- **Receipt tracking:** Gathering and sorting receipts is one way that some people keep tabs on their spending. For future use, they scan hard copy receipts into digital formats or save them in folders. Although this approach produces thorough records, it can be laborious and lead to missed or lost receipts.
- **Bank Statements:** Using bank statements as a guide for tracking spending is another popular technique. Users classify expenses and track spending patterns by frequently reviewing their bank transactions. Although bank statements provide precise documentation, they might not enable personalized classification or offer in-depth insights into certain expenditures.
- **expenditure Tracking Apps:** As smartphones become more popular, a growing number of people choose to use specialized expenditure tracking apps. With these apps, users can manually enter their costs or have transactions from their connected credit cards and bank accounts synchronized automatically. They provide tools for budgeting, expenditure statistics, scanning of receipts, and configurable categories. Mint, YNAB (You Need a Budget), and Expensify are a few well-known examples.
- **Digital wallets:** To keep track of costs associated with certain transactions, some users rely on digital wallet programs like PayPal, Venmo, or Apple Pay. Usually, these apps offer transaction histories, which let users check previous purchases and keep an eye on their spending in real time.

Every approach has benefits and drawbacks, and the decision often comes down to personal tastes, lifestyle and degree of comfort using technology.

## **1.4 PRESENT WORK**

In this project we take control of finances with personalized budget settings. Our app provides real-time updates on spending against allocated budgets, empowering users to make informed financial decisions.

## **1.5 LITERATURE SURVEY**

Reviewing scholarly articles, business reports, and other pertinent sources would be part of a literature evaluation on mobile cost monitoring software. This would help one gain an understanding of the state of the field's advancements, difficulties, and trends.

[1]. Yolande E. Chan, Thomas Kvan, and Erik Stettler (2016) "Understanding Mobile Money Management: Insights into the Use of Personal Financial Management Applications" This study looks at user behaviors and preferences while examining the use of mobile financial management applications, which include capabilities for tracking expenses.

[2]. The effect of mobile banking on household welfare and financial literacy" by Rajalaxmi Kamath and Amrit Judge (2017): This study investigates how financial literacy and household welfare are affected by mobile banking in India, which frequently offers features for tracking expenses.

[3]. Leslie A. Carr and Lorraine S. Kisselburgh's "Evaluating Personal Financial Management Tools" (2018): Through user assessments and usability testing, this study investigates the usefulness and usability of several personal financial management tools, such as programs for tracking expenses.

[4]. Peter Tufano's (2018) article "Empowering consumers in financial markets: a behavioral economics perspective" In order to empower customers and enhance their ability to make financial decisions, this article explores the application of behavioral economics in the design of financial services and products, such as apps for tracking spending.

[5]. Stephen K. Karanja, Adelina Gschwandtner, and Paschal B. Mihyo, "The Effectiveness of Financial Technology Applications in Fostering Financial Inclusion: Lessons from M-PESA" (2020): This study looks at how well M-PESA, a mobile money app with capabilities for tracking expenses, works in Kenya to encourage financial inclusion.

[6]. José Viterbo and Tiago Oliveira's article "Understanding users' perception of financial mobile applications" was published in 2021. This study looks into how users view financial mobile applications, taking into account user happiness, adoption and usage characteristics, and perceived usefulness.

[7]. Ankita Gupta, Sanjay Kumar Dubey, and Anand Nayyar's "Design guidelines for mobile expense tracking applications: A review of literature" (2022): This study examines the body of research to determine design standards with an emphasis on usability, user experience, and feature needs for mobile cost monitoring apps.

[8]. Hernandez, E. D. Mynatt, and T. Jaeger's article "Designing a Mobile Expense Tracking Application to Promote Self-Reflection on Personal Finances" In order to encourage introspection on personal finances, this article investigates the conception and assessment of a mobile spending tracking application. It goes into the features, UI design, and findings of a user study that assessed the app's efficacy.

[9]. Rahmati, A. Klemmer, and L. Cranor, "Understanding Users' Needs for Mobile Expense Tracking Applications". The results of a qualitative study examining consumers' requirements and preferences with regard to mobile cost tracking apps are presented in this paper. It talks about typical problems customers have keeping track of their spending and offers advice on how to make apps that work better.

[10]. By M. Hogqvist, J. Persson, and C. Hall, "Mobile Personal Finance Management". This study looks at how mobile personal finance management apps can assist users in creating budgets, keeping track of their spending, and managing their money more skillfully. It explores the attributes and capabilities of current applications and makes suggestions for their future develop

## CHAPTER 2

### HARDWARE / SOFTWARE TOOLS

#### REQUIREMENT SPECIFICATION (S/W & H/W)

##### Hardware Requirements

✓ System	: Pentium 4, Intel Core i3, i5, i7 and 2GHz Minimum
✓ RAM	: 4GB or above
✓ Hard Disk	: 10GB or above
✓ Input	: Keyboard and Mouse
✓ Output	: Monitor or PC

##### Software Requirements

- Tools used :-

Front end : HTML

CSS

Java script Framework – React JS

Back end : Database

Node JS

Mobile App Development : Flutter

## 2.1 SYSTEM DESIGN

System design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. It involves specifying the hardware, software, and network infrastructure needed to implement the system, as well as designing the interactions between the various components of the system.

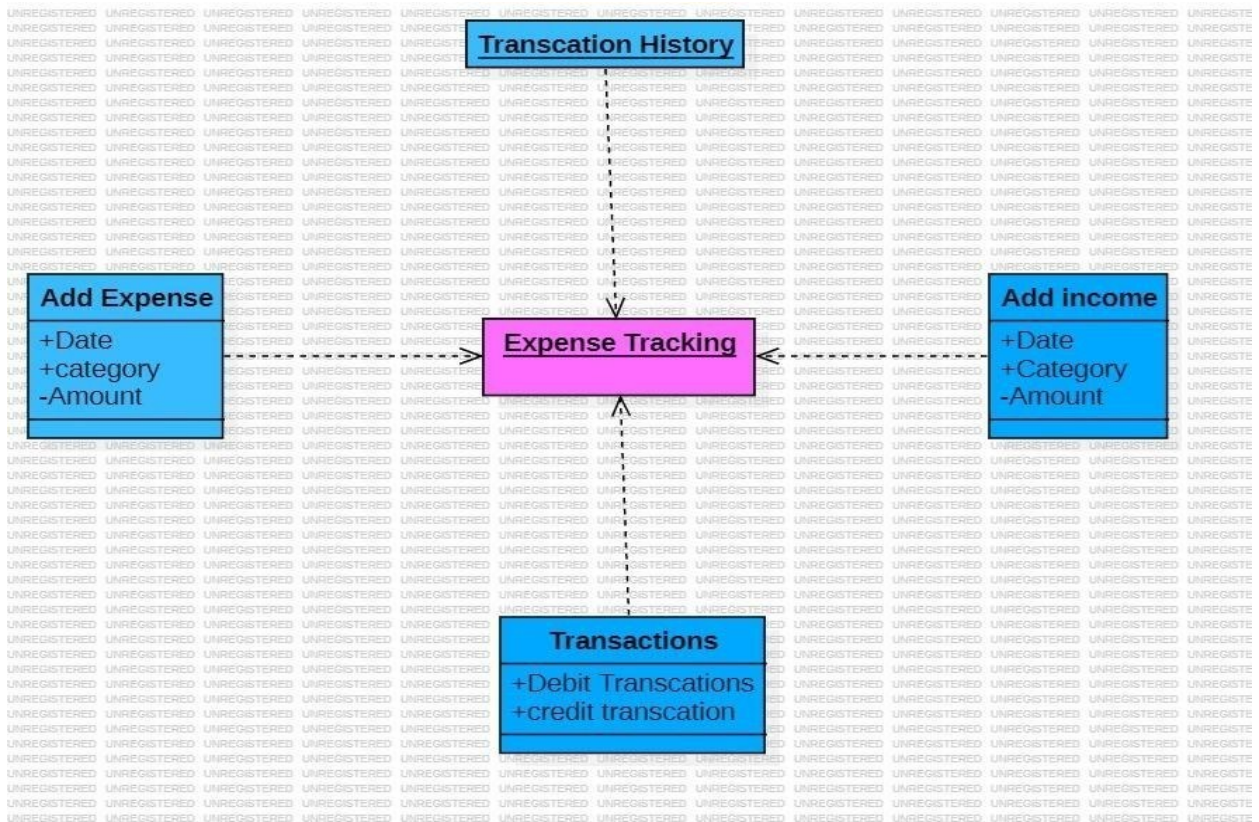
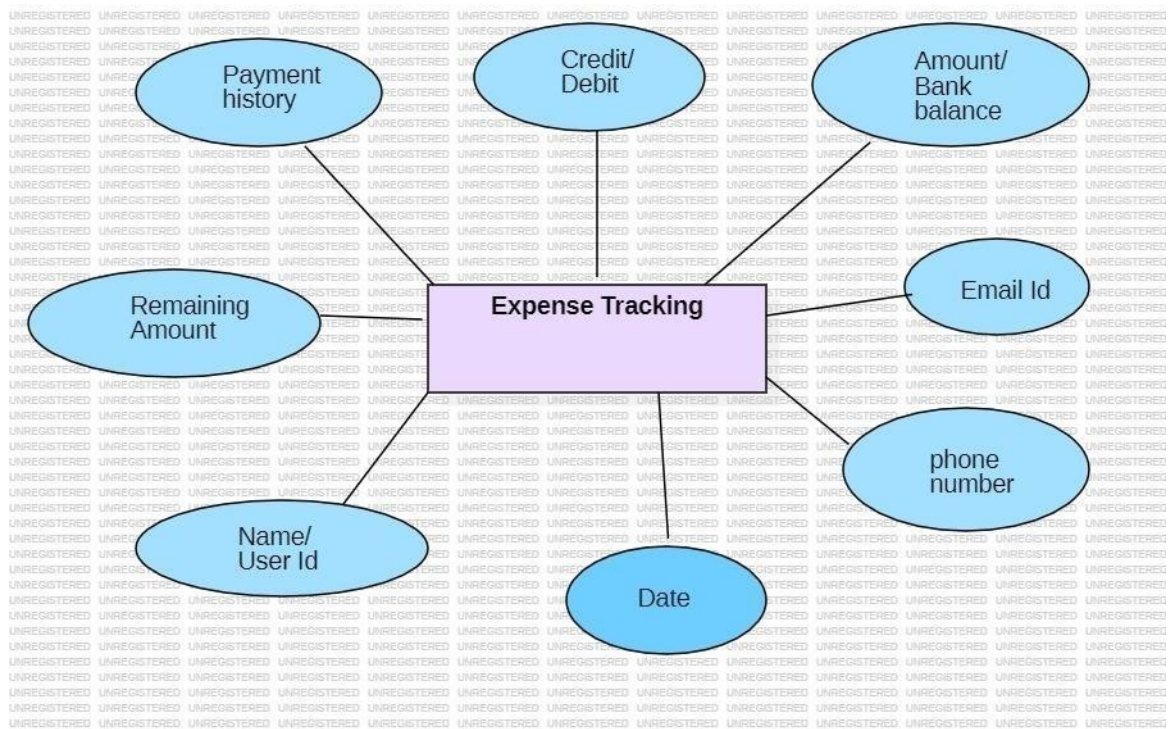


Fig 1. System Design

## 2.2 ER Diagram

The relationships and entities in a database are represented graphically in an entity-relationship (ER) diagram. It is a modeling tool that makes it easier to visually portray a database's structure by showing the relationships between different objects.

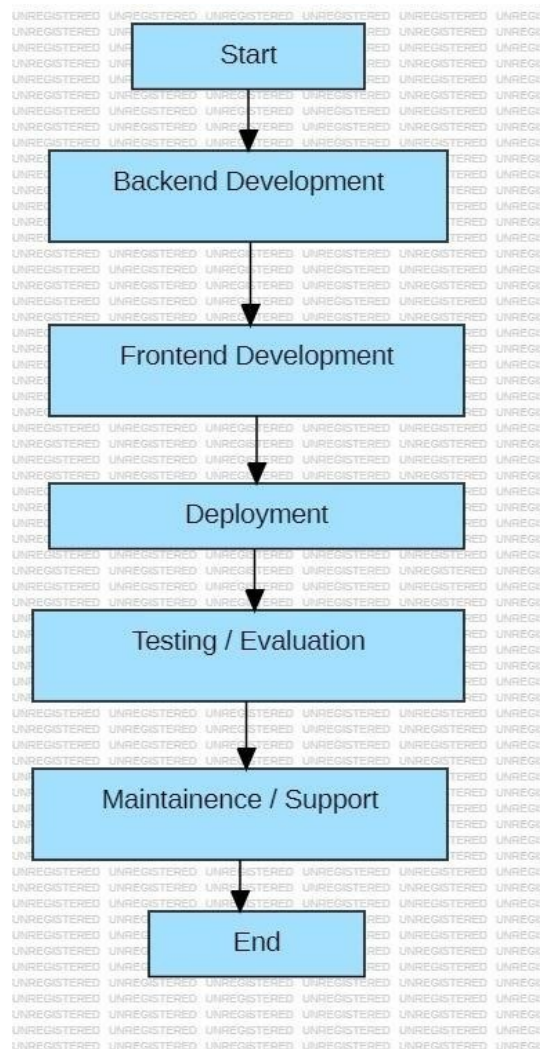
Database developers and other stakeholders can better grasp the relationships and structure of a database system by using ER diagrams. They make it easier to create, share, and use database schemas, guaranteeing that data is successfully and precisely arranged to satisfy system or application requirements.



**Fig 2. ER Diagram**

## 2.3 FLOWCHART

A flowchart is a graphic representation of a computer program, system, or process. They are extensively used in many different sectors to plan, analyze, document, and communicate—in simple, understandable diagrams—often complex processes. Flowcharts, which are sometimes spelled "flow charts," specify the type of step using shapes like diamonds, ovals, rectangles, and perhaps many more. They also employ connecting arrows to define flow and sequence. They can be as basic as hand-drawn charts or as intricate as computer-drawn schematics that show several processes and paths. Taking into account all of the different variations, flowcharts rank among the most widely used diagrams worldwide, being utilized in a wide range of fields by both non-technical and technical individuals. There are occasions when flowcharts are referred to by more technical terms like Process Flowchart, Process Map, and Functional



**Fig 3** Flow chart

### **Backend Development:**

Reverse Engineering Flow Diagram:

Start: The process of developing the backend has begun.

Identify Requirements: Compile the specifications needed for the backend development procedure.

Design Database Schema: Create a database structure that will be used to hold spending information.

Select Technologies: Choose the technologies (Express.js, Node.js, etc.) that will be utilized for backend development.

Establish Project: Make a fresh Node.js project and configure it as needed.

Set up API Endpoints: Provide RESTful API endpoints to manage the Create, Read, Update, and Delete (CRUD) processes related to costs.

Apply Authentication: Apply JWT (JSON Web Tokens) to user authentication.

Put Business Logic Into Practice: Include business logic for filtering, statistics production, and expense classification.

Test Endpoints: Make sure the implemented API endpoints function as intended by testing them.

Record APIs: Keep a record of the APIs for future use.

## **Front-end Programming :**

### **UI/UX Design:**

Use Flutter to create designs and wireframes for the cost tracking app.

Create screens for user authentication, viewing, adding, and other functions as part of the implementation of the user interface.

Connect to Backend: To interact with the Node.js backend in Flutter, use HTTP libraries such as http or dio.

Handle User Authentication: Use JWT tokens to implement user authentication in the Flutter app.

## **Backend Development and Deployment:**

Start: Get the process of developing and deploying the backend started.

Describe the Conditions: Compile the specifications needed for the deployment and development of the backend.

Database Schema Design: Create the database's structure in order to store the expense data.

Select Technologies: Choose the technologies (such as Node.js, Express.js, Heroku, etc.) that will be utilized for backend development and deployment.

Establish Project: Make the required setups and start a new Node.js project.

Set up API Endpoints: Provide RESTful API endpoints to manage the Create, Read, Update, and Delete (CRUD) processes related to costs.

Apply Authentication: Apply JWT (JSON Web Tokens) to user authentication.

Test Endpoints: Make sure the implemented API endpoints function as intended by testing them.

Install Backend: Install the backend on a hosting platform such as Heroku.

Record APIs: Record the APIs for

## **Testing and evaluation:**

Start: Get the testing and assessment process going.

Describe Test Situations: Define test cases and scenarios in accordance with the specifications.

Unit Testing: Test individual components, such as business logic, authentication, and API endpoints.

Integration testing verifies that various components are integrated and function as intended.

Use user acceptance testing (UAT) to make sure the system satisfies their needs by involving users in the



process.

**Bug Fixing:** Address any problems or bugs discovered during testing.

**Retesting:** To make sure bugs have been fixed, retest the system after applying fixes.

**Assessment:** Examine the system in light of user input and test findings.

Testing and evaluation are completed.

### **Maintenance and support:**

**Start:** Initiate the process of maintenance and support.

**Get Feedback:** Find out what problems or requests for new features users and stakeholders have.

**Resolve** any reported defects or issues by analyzing and fixing them.

**Implement New Features:** Based on user requirements and input, develop and implement new features or enhancements.

**Release upgrades:** Release bug fixes and upgrades to the live environment.

**System Monitoring:** Keep an eye out for any problems or poor system performance.

**Offer Assistance:** Assist users with any problems they may run into.

Process of maintenance and support comes to an end.

### **Safety:**

Secure access with user authentication with JWT (JSON Web Tokens).

Validate input to stop injection attacks.

encryption for private information.

## **2.4 RISK ANALYSIS**

Risk analysis is essential for identifying potential factors that could lead to the failure of a project or the inability to complete it by the deadline. Here are some factors that pose risks to the success and timely completion of a expense tracking mobile application project.

**Scope creep :** When it comes to expenditure tracking, scope creep is the term used to describe the ongoing, unforeseen growth of the application's features, functionalities, or requirements beyond what was first specified or agreed upon.

**Resource constraints :** "Resource constraints" in the context of expense tracking relate to limits or restrictions on the resources that the application has access to.

**Technical challenges :** A safe, scalable, and user-friendly expense tracking application must be ensured by meticulous planning, implementation, and continual testing in the face of technical hurdles.

**Poor communication:** A safe, scalable, and user-friendly expense tracking application must be ensured by meticulous planning, implementation, and continual testing in the face of technical hurdles.

**quality Assurance Issues :** You may guarantee the dependability, security, and smooth operation of your spending monitoring program by carrying out extensive quality assurance testing. This promotes user and guarantees the long-term viability of the application.

**Team dynamics and market dynamics :** You may create an expense tracking application that satisfies user needs, stands out in the market, and succeeds over the long run by managing team dynamics well and keeping up with market developments.

## CHAPTER 3

### PROJECT IMPLEMENTATION

#### 3.1 PROPOSED SYSTEM

Our suggested solution guarantees a strong and safe environment for users to efficiently track and control their spending. Our mobile application for tracking expenses is designed to provide users an easy-to-use interface, an abundance of functionality, and cutting-edge technologies, all combined to create a seamless and intuitive experience.

##### **Key Components:**

- Frontend: React.js, HTML, and CSS

The user interface's structure and style are provided via HTML and CSS.

React.js is a JavaScript package for UI development. It makes it possible to create UI components that are reusable.

- Backend (Node.js, Express.js):

Utilizing Node.js as the backend runtime environment for building scalable network applications.

- Implementing Express.js, a web application framework for Node.js, for building web APIs to handle user requests.
- Database (MongoDB): Using the NoSQL database application MongoDB to store user information and spending. The flexibility of MongoDB makes managing documents that resemble JSON easy.
- Mobile App Development (Flutter): Build a cross-platform mobile application with Google's open-source Flutter UI software development kit. This guarantees that a single codebase can execute the program flawlessly on both iOS and Android devices.

#### 3.2 PROCEDURE

##### 1. Frontend: HTML, CSS, and React.js

HTML & CSS: Used for the structure and styling of the user interface.

React.js: A JavaScript library for building user interfaces. It allows for the creation of reusable UI components.

##### 2. Backend: Node.js

Node.js: A JavaScript runtime built on Chrome's V8 JavaScript engine. It's used for building scalable network applications.

Express.js: A web application framework for Node.js. It's used for building web applications and APIs.

### 3. Database: MongoDB

MongoDB: A NoSQL database program. It uses JSON-like documents with optional schemas.

### 4. Mobile App Development: Flutter

Flutter: An open-source UI software development kit created by Google. It's used to develop applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase.

### 5. System Architecture:

Frontend:

HTML/CSS for UI structure and styling.

React.js for building reusable UI components.

Backend:

Node.js with Express.js for building the API.

MongoDB for storing user data and expenses.

Mobile App:

Flutter for developing cross-platform mobile applications.

### 6. Features:

User authentication and authorization.

Adding, editing, and deleting expenses.

Categorizing expenses.

Generating expense reports.

Setting budgets and receiving notifications when close to the budget limit.

Data visualization for expenses.

### 7. Security:

User authentication using JWT (JSON Web Tokens).

Input validation to prevent injection attacks.

Encryption for sensitive data.

### 8. Deployment:

Frontend and Backend hosted separately.

Frontend: Hosted on services like Netlify, Vercel, or AWS S3.

Backend: Hosted on services like Heroku, AWS, or Azure.

Mobile App: Deployed to Google Play Store and Apple App Store.

### 9. Third-party Integrations:

Payment gateways for premium features (if applicable).

Analytics tools for tracking user behavior.

Push notification services for budget alerts.

### 10. Testing:

Unit testing for frontend and backend using Jest, Enzyme, or Mocha.

Integration testing for API endpoints.

UI/UX testing for mobile app using Flutter testing framework.

#### 11. Monitoring and Maintenance:

Logging and monitoring using tools like Sentry or AWS CloudWatch.

Regular updates and patches for security vulnerabilities and feature enhancements.

#### 12. Scalability:

Horizontal scaling for both frontend and backend to handle increased user load.

Database sharding for handling large amounts of data.

#### 13. User Support:

Helpdesk or ticketing system for user queries and issues.

Documentation for user guidance.

This proposed system aims to provide users with a seamless and secure experience for tracking their expenses, whether through the web interface or the mobile application.

### **FRONTEND :**

**HTML :** The common markup language used to create online pages and web apps is called HTML (Hypertext Markup Language). It uses a range of tags and attributes to specify the content structure of a web page.

Text that is shown on a computer or other electronic device that has links (hyperlinks) to other text that the reader can view right away is referred to as hypertext.

A technique for annotating a document that is syntactically distinct from the text is called markup language. This is accomplished in HTML by using tags.

HTML is made up of a number of elements that you can employ to wrap or enclose different parts of the material to give it a certain look or behavior.

### **CSS :**

Cascading Style Sheets, or CSS, is a language for style sheets that describes how an HTML content is presented. It specifies how elements ought to appear on screen, in writing, spoken word, or in other media.

### **JavaScript Frameworks:**

Pre-written JavaScript code libraries known as JavaScript frameworks give developers an organized way to create online applications. These frameworks provide a starting point for creating dynamic and interactive

webpages. Pre-written JavaScript functions, scripts, and utilities are frequently included in them to aid expedite the development process and preserve uniformity throughout the program.

**React.js:** Facebook maintains React.js, an open-source JavaScript library. Its quick and scalable nature makes it ideal for creating user interfaces, especially for single-page apps. With React, developers can build sizable web apps that update data without refreshing the page. Its main goals are to offer simplicity, speed.

## **BACKEND**

**Node.js:** Based on the V8 JavaScript engine in Chrome, Node.js is an open-source server-side JavaScript runtime environment. It enables programmers to create network applications that are quick, scalable, and real-time. Node.js is lightweight and efficient because it operates on an event-driven, non-blocking I/O model. It's frequently used to create APIs and backend services.

**Database:** Generally kept electronically in a computer system, a database is a well-organized collection of structured data. Users can engage in a variety of interactions with the data by using a database management system (DBMS). You'll probably be using a NoSQL database like MongoDB or a relational database like MySQL, PostgreSQL, or SQLite in the context of your application.

## **MOBILE APP DEVELOPMENT**

**Flutter:** Google developed an open-source UI software development kit called Flutter. It is used to create desktop, online, and mobile apps from a single codebase. Developers can write code only once and have it run on both the iOS and Android platforms thanks to Flutter. It makes use of the Google-developed Dart programming language.

## CHAPTER - 4

### 4.1 SIMULATION SETUP

Gain an understanding of the fundamentals of finance by studying about budgeting, keeping track of spending, and creating financial objectives. With this knowledge, you will be able to comprehend the features and requirements of an expense tracker software better. Programming abilities: Programming in pertinent languages.

Depending on which platform (i.e., iOS, Android, web) you choose for development, you may use Objective-C for iOS, Kotlin or Java for Android, HTML/CSS/JavaScript for web development, etc. Make sure you are proficient in both these programming languages and the related development frameworks. Resources for Developing Mobile Apps: Assemble the tools and development environments needed for the platform or platforms of your choice. Integrated development environments (IDEs) like Visual Studio Code for web development, or Xcode for iOS and Android Studio for Android .

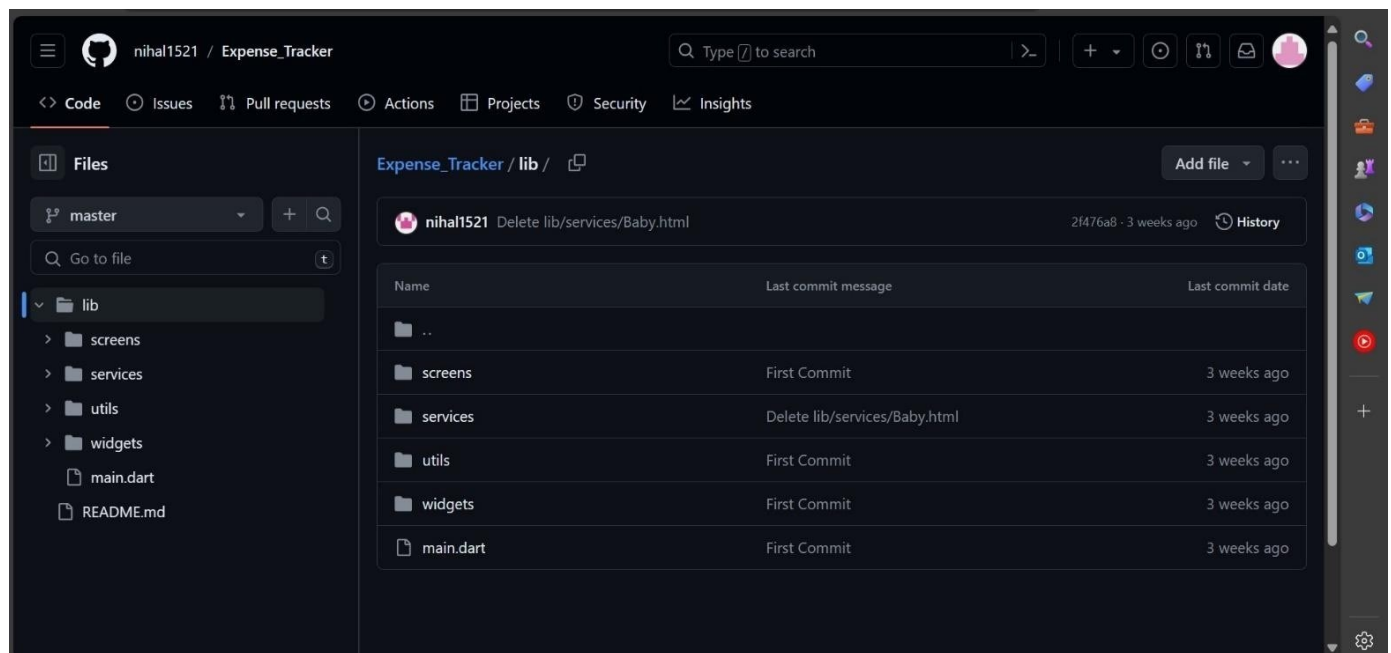


Fig .4 code files lib

**Milestone 1:** Data Collection is the primary source

**Milestone 2:** In this milestone we are going to develop Backend and middle end using visual studio

**Activity 1:** Backend development in this we have 2 type of services

1. auth\_service: This is used for data retrival from the database
2. db: This is the database. The database that we are using 'package:cloud\_firestore/cloud\_firestore.dart

**Activity 2:** Middle end development in this we use node.js

1. iconlist: in this we map all the icons what ever we want
2. validator: in this we are going to send a verification mail to the new user mailid to verify the user is valid or not.

**Activity 3:** Frontend development In this we have 2 Screens and Widges.

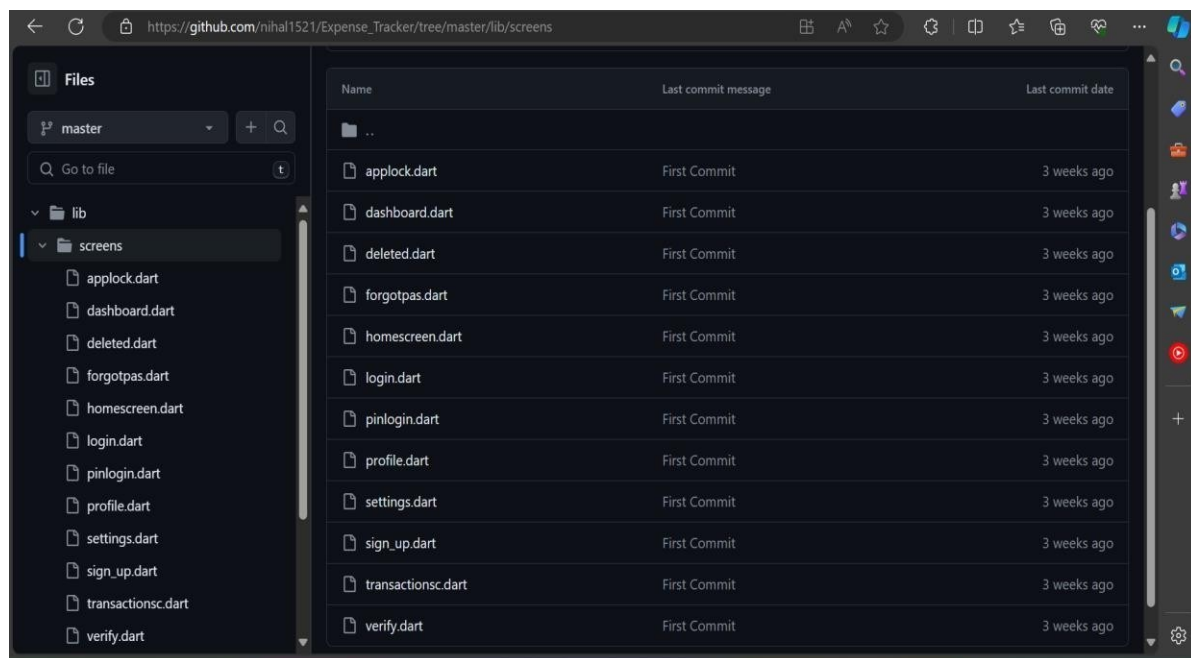


Fig .5 screens



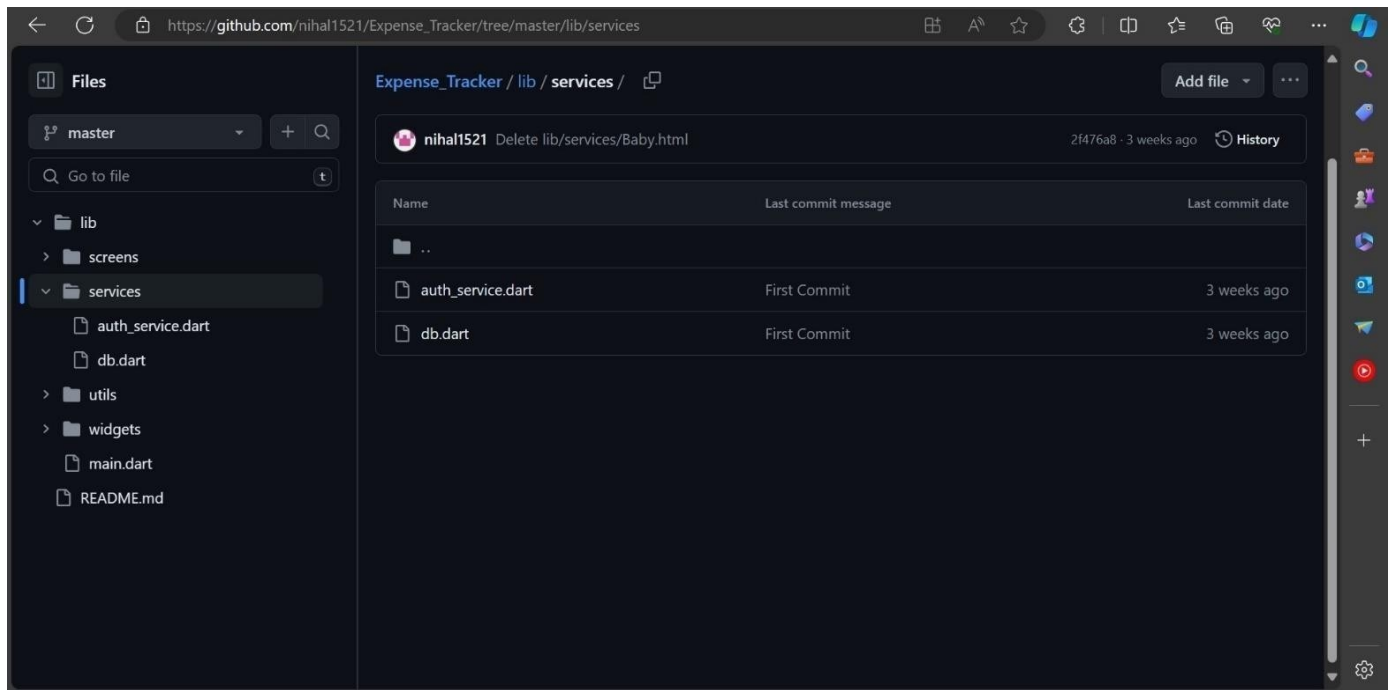


Fig.6 Services

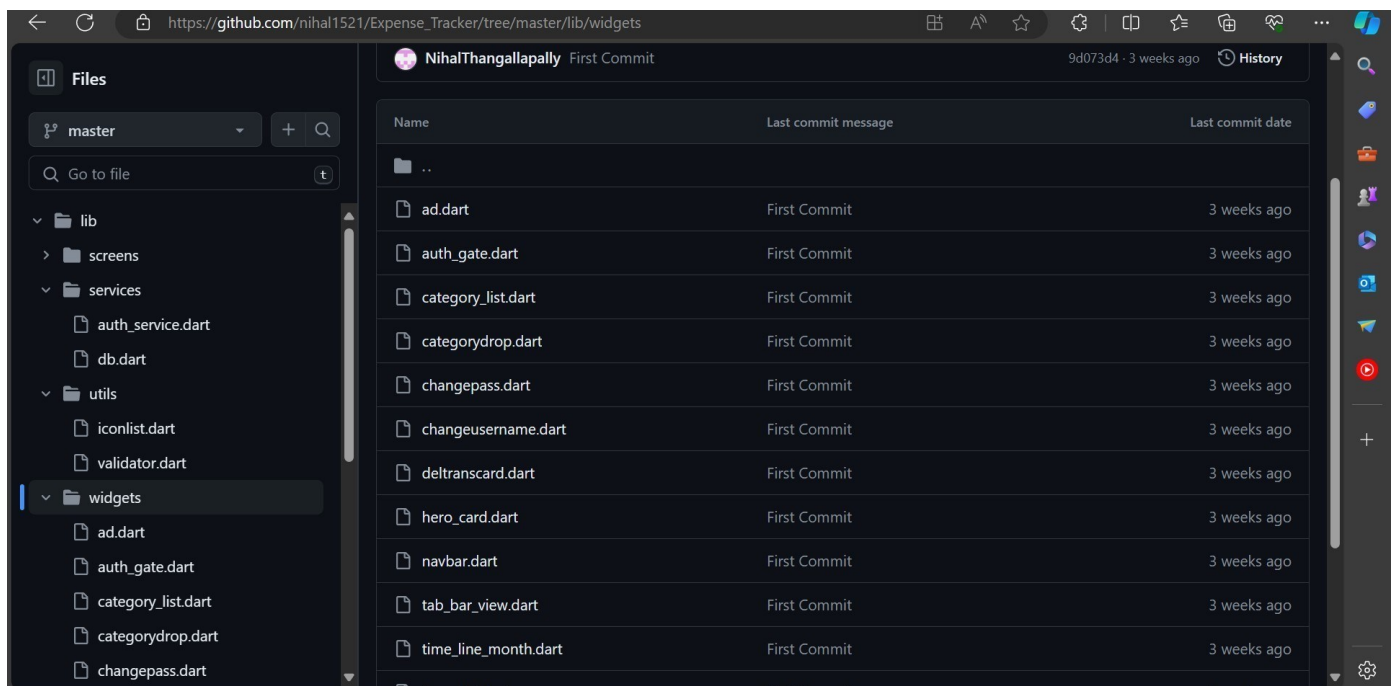


Fig.7 Widgets

## 4.2 RESULT

### PROJECT RESULT:

Expense tracking mobile application result

#### Front-end:

CSS and HTML for the organization and design

The frontend framework is React.js.

#### Backend:

Database: Node.js for the server-side logic

database (such MongoDB, PostgreSQL, MySQL, etc.) that is compatible with Node.js for mobile app development.

Flutter is used to create the mobile application

**Our project Expense tracker mobile application name is Budget Bliss**



Fig 8 App logo

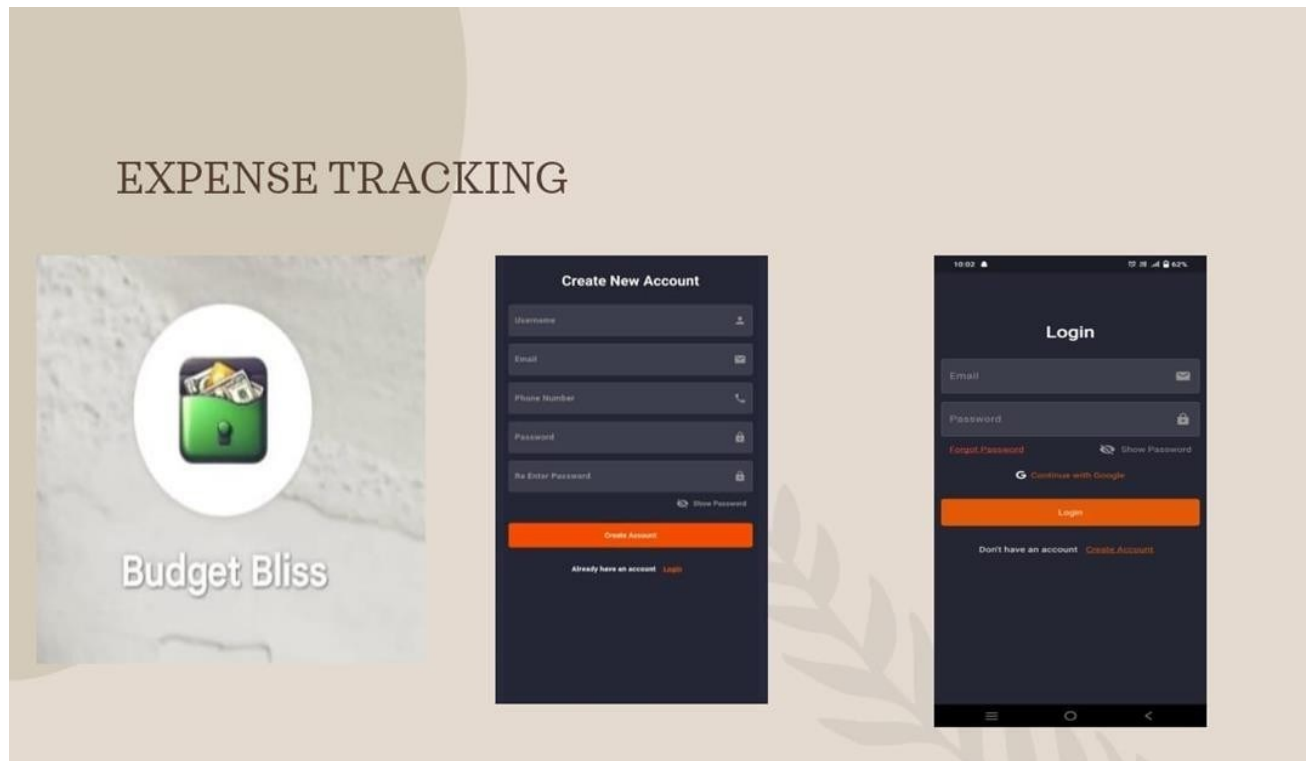


Fig .9 Login /create account

## HOW APPLICATION WORKS

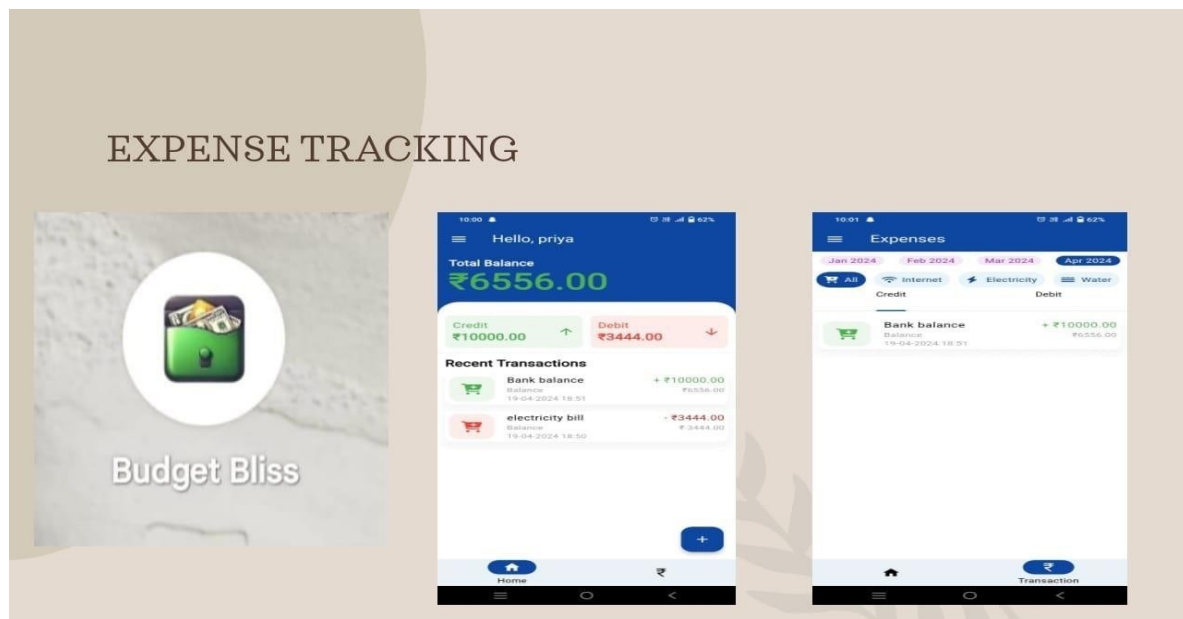


Fig .10 How Application works

## HOW EXPENSE TRACKEING WORKS AND STEPS INCLUDE :

- Create Account

- Login credentials
- Credit / Debit Amount
- Add date to create and enter the transaction details like debit or credit amount
- History shows the remaining bank balance

## 4.3 RESULT COMPARISON AND ANALYSIS

### EXPENSE TRACKING APPLICATION COMPARING WITH EXISTING MINT (NATIVE MOBILE APP)

#### **Mint**

Although Mint's backend technology is not made available to the public, it probably makes use of a strong and expandable technology stack.

Development of Mobile Apps:

Flutter application:

**Advantages:** Cross-platform development is possible with Flutter, allowing you to write code once and publish it to both the iOS and Android operating systems.

The Flutter hot reload functionality expedites the development process.

For creating stunning user interfaces, Flutter offers a wide range of customisable widgets.

**Cons:** Although it's pretty near, it might not deliver the identical native performance.

The native mobile app Mint:

**Positives:** The native Mint app offers a smooth user experience and outstanding performance.

It is optimized for the iOS and Android operating systems.

**Cons:** Compared to using a cross-platform framework like Flutter development time may be greater.

## **ANALYSIS OF COMPARING OUR PROJECT EXPENSE TRACKING WITH (MINT) NATIVE MOBILE APP**

Our expense tracking mobile application, which uses React.js for the frontend, Node.js database for the backend, and Flutter for mobile app development, offers benefits in terms of development speed, code maintainability, and cross-platform compatibility, while Mint provides a native mobile experience with optimized performance.

### **4.4 LEARNING OUTCOME**

A thorough project that covers many facets of contemporary online and mobile development is developing an expense tracking mobile application using HTML, CSS, React.js for the frontend, Node.js for the backend, and Flutter for mobile app development.

**Frontend Development:** HTML/CSS: This course teaches you how to organize and style web pages with CSS.

**React.js:** You may learn component-based UI development, state management, and handling user interactions by building the frontend with React.js.

**Backend Development:** Node.js: Acquire the skills necessary to process HTTP requests, communicate with databases, and create a RESTful API using Node.js.

**Database:** You will learn how to deal with and execute CRUD operations on databases, such as MongoDB or MySQL.

#### **Mobile application development :**

Mobile application development using a single codebase can be learned by developing a mobile application with Flutter. You will gain knowledge of navigation, state management, and Flutter widgets.

**Learning outcome :** Full-stack development: You will acquire expertise in front-end and back-end programming, enabling you to build an entire web application from the ground up.

RESTful API development: Gain knowledge about how to create and implement a RESTful API for front-end and back-end communication.

**Database integration:** Know how to manage data, execute database operations, and link your application to a database.

Develop your skills in mobile app development with Flutter to be able to make cross-platform iOS, and android platforms.

## CHAPTER 5

### 5.1 CONCLUSION WITH CHALLENGES

The expense tracking concludes with how our cutting-edge technology makes it simple and effective for people and companies. Bid farewell to laborious manual data entry and relish instantaneous insights into your spending patterns. Budget Financial Management With customized budget settings, you can take charge of your money. Our software gives users access to real-time information on expenditure relative to budgets, enabling them to make wise financial decisions.

### 5.2 CHALLENGES

**Integration Complexity:** If you're unfamiliar with any of the three technologies, integrating React.js, Node.js, and Flutter can be difficult.

**Learning Curve:** Your team may need some time to become fluent in all three technologies as React.js, Node.js, and Flutter each have their unique learning curves.

**Performance Optimization:** It can be difficult to guarantee optimal performance on all systems, particularly when handling big data sets.

**Security:** It takes close attention to detail to protect the application from common security risks like SQL injection and cross-site scripting (XSS).

**Problems with Compatibility:** It can be difficult to ensure compatibility across various devices, screen sizes, and operating systems, particularly with Flutter, which is still in its early stages of development.

Overall, while using React, Node.js, and Flutter to create an expense tracking mobile application has many advantages.

### 5.3 REFERENCES

- [1]. Manchanda, Angad. Expense Tracker Mobile Application. Diss. San Diego State University, 2012.
- [2]. Vu, Bao. "The design and implementation of web based expense tracking application." (2013).
- [3]. Velmurugan, A., J. Albert Mayan, P. Niranjana, and Richard Francis. "Expense manager application." In *Journal of Physics: Conference Series*, vol. 1712, no. 1, p. 012039. IOP Publishing, 2020.
- [4]. Sabab, S.A., Islam, S.S., Rana, M.J. and Hossain, M., 2018, September. eExpense: a smart approach to track everyday expense. In *2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEICT)* (pp. 136-141). IEEE.
- [5]. Khandelwal A. Developing an Expense Tracking Application using React and Node. js.
- [6]. Khandelwal A. Developing an Expense Tracking Application using React and Node. js.
- [7]. Vanitha, M., K. Alekhya, and A. Sai Gowthami. "Expenses Management System." *International Journal of Advanced Networking and Applications* (2019): 208-211.
- [8]. Johri E, Desai P, Soni P, Jain H, Sanganeria N. Expense Management System. In 2023 4th IEEE Global Conference for Advancement in Technology (GCAT) 2023 Oct 6 (pp. 1-6). IEEE.
- [9]. Bhatele, P., Mahajan, D., Mahajan, B., Mahajan, D., Mahajan, N. and Mahajan, P., 2023, May. TrackEZ Expense Tracker. In *2023 4th International Conference for Emerging Technology (INCET)* (pp. 1-5). IEEE.
- [10]. Hahn, Young-ae, Jiyouon Lee, and Minhyuk Kim. "Envisioning personal finance and expense tracking for a sustainable future." (2022).
- [11]. Vira, D., Savla, S., Chheda, D. and Chavan, V., 2020, June. Cashless Employee Expense and Transacting System. In *Proceedings of the International Conference on Recent Advances in Computational Techniques (IC-RACT)*.
- [12]. Tziala, Aikaterini. "Affective system monitoring personal expenses, helping the user to stay on budget." (2019).
- [13]. Thakare, R., Thakare, N., Sangtani, R., Bondre, S. and Manekar, A., 2023. Expense Tracker Application using Naive Bayes.
- [14]. Gupta A, Dwivedi SK, Ali MS. EXPENSE TRACKER FOR SALES.