

PROJECT REPORT

ON

### “PERSONAL FINANCE TRACKING APPLICATION”

SUBMITTED BY

**SIB23- Jayartha Sengupta**

**SIB26- Shravani Kamble**

**SIB34- Akshay Lakhwal**

**SIB39- Suraj Nakhale**

**SIB40 - Siddharth Ohal**

GUIDED BY

**Dr. Jyoti Surve**

DEPARTMENT OF INFORMATION TECHNOLOGY INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY, PUNE - 411057 SAVITRIBAI PHULE PUNE UNIVERSITY

2023-2024



INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY HINJEWADI, PUNE-57 DEPARTMENT OF INFORMATION TECHNOLOGY

**Certificate**

This is to certify that the Project report entitled

### “URBAN ASSETS”

### PERSONAL FINANCE TRACKING APPLICATION

which is being submitted by

**SIB23- Jayartha Sengupta**

**SIB26- Shravani Kamble**

**SIB34- Akshay Lakhwal**

**SIB39- Suraj Nakhale**

**SIB40 - Siddharth Ohal**

have partially completed the Project entitled “PERSONAL FINANCE TRACKING APPLICATION”, under my guidance in partial fulfilment of the requirement for the Project Based Learning in S.E. Information Technology of International Institute of Information Technology, Hinjewadi, by Savitribai Phule Pune

University for the academic year 2022 – 2023.

Dr. Jyoti Surve Dr. Jyoti Surve

(Guide) (Head of the Department)

Internal Examiner: -

External Examiner:- Date:-

# Acknowledgement

With immense pleasure, we are presenting this project report on “**PERSONAL FINANCE TRACKING APPLICATION**”, as a part of the Project Based Learning in S.E. Information Technology at INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY, HINJEWADI, PUNE.

It gives us the privilege to complete this report under the valuable mentorship of **Dr. Jyoti Surve**.

Her guidance, cooperation, and encouragement have made headway in the project.

We are extremely grateful to **Dr. Jyoti Surve** (H.O.D of Information Technology) and **Dr. Vaishali Patil**, Principal, Information Technology of Information Technology for providing all facilities and help for the smooth progress of the report work. We would also like to thank all the Staff Members of the Information Technology Department, Management, friends, and our family members, who have directly or indirectly guided and helped us with the preparation of this report and gave us unending support right from the stage the idea was conceived.

Students Name & Signature

Siddharth Ohal:

Shravani Kamble:

Suraj Nakhale:

Jayartha Sengupta:

Akshay Lakhwal:

**Contents**

|  |  |  |
| --- | --- | --- |
| **1** | **Introduction to Project Topic** | **Page No.** |
| 1.1 | Project Overview | **7** |
| 1.2 | Brief Description | **7** |
| 1.3 | Problem Definition | **7** |
| 1.4 | Intended Audience and Reading Suggestions | **7** |
| 1.5 | Project Scope | **7** |
| **2** | **Literature Survey** | **9** |
| **3** | **Software Requirements Specification** | **10** |
| 3.1 | Project Scope | 10 |
| 3.2 | Perspective | 10 |
| 3.3 | Product Features | 10 |
| 3.4 | Operating Environment | 10 |
| 3.5 | System Features | 11 |
| 3.6 | Other Non-Functional Requirements | 17 |
| **4** | **System Design** | **18** |
| **5** | **Testing Strategy Plan** | **20** |
| **6** | **Project Plan** | **23** |
| **7** | **Result Discussion** | **25** |
| **8** | **Glossary** | **36** |
| **9** | **Reference** | **37** |

# Graph Contents

|  |  |  |
| --- | --- | --- |
|  | | |
| 4.1 | System Architecture Diagram | 18 |
| 4.2 | Working Diagram | 19 |
| 4.3 | Sequence Diagram | 19 |
| 4.4 | Use Case Diagram | 20 |

# Abstract

The PERSONAL FINANCE TRACKING APPLICATION (PFTA) The use of personal finance smartphone apps results in an improvement in various measures used to assess financial knowledge and skills, attitudes and motivations, and financially capable behaviors for those in low‐income households.

Those provided with smartphone apps demonstrated increased self‐confidence in financial decision‐making and financial literacy and improved their ability to delay self‐gratification and their sense of being able to effect change. Financially capable behavior changes manifested in being better able to keep track of finances and manage unexpected bills. User engagement with finance apps could be improved by targeting users with a specific financial decision‐making problem, personalizing the apps through push notifications to encourage ongoing user engagement, and incorporating game mechanics.

An expense tracker is a tool that enables people to manage their finances by keeping tabs on their spending. A notebook, spreadsheet, or mobile app are just a few tools that can be used to accomplish this. Users can establish budgets for each category in the tracker, which often has predetermined categories like grocery, transportation, and entertainment. Each time a user makes a purchase, they log the amount and the category in the tracker. Users can track their expenditure, assess how it compares to their budget, and make necessary adjustments to help them reach their financial objectives by frequently reviewing the tracker. In general, an expenses tracker is a useful tool for anyone trying to take charge of their finances and make decision.

# Chapter 1

## Introduction to Project Topic

### Project Overview

In today’s era, subscription services are common. Also, you have the usual array of rent and utilities to pay. Plus, money these days is a little tighter than it used to be comparatively speaking.

Therefore, we decided to find an easier way to get rid of this problem. So as the name itself suggests, this project is an attempt to manage our daily expenses in a more efficient and manageable way. Sometimes we can’t remember where our money goes. And we can’t handle our cash flow.

### Brief Description

Our application allows people to eliminate any unnecessary expenses and figure out if they are able to survive themselves or do, they really need to ask help. Typically, people are able to optimize their spending by making some correct decisions. Instead of keeping a diary or a log of the expenses, this application enables the user to not just keep the control on the expenses but also to generate and save reports. So, this application attempts to free the user with as much as possible the burden of manual calculation and to keep track of the expenditure.

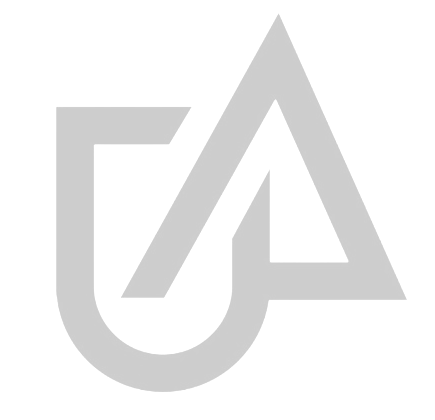
### Problem Definition

As we see in several historical places, individuals take pictures of some ancient sculptures and inscriptions. Most of the traditional pillars or walls have text written within the kind of ancient Brahmi script, however, these days several folks don't perceive this ancient language. We decided to make a project in which we can just take an image and process it to extract the text present in the image. Then we convert this text into our modern languages. It saves tons of your time to read the text from an image and understand it.

### Intended Audience and Reading Suggestions

* For Domestic Purpose
* Retailers

### Project Scope

Our application is for domestic use and for small retailers only. Our application does not meet all the requirements of higher-level businesses. Our application can take advantage of a good market as it is usable by anyone who is willing to manage their expenses and aims to save for future investments and many more.

# Chapter 2

**Literature Survey Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Paper Name** | **Authors** | **Methodology** | **Scope/Future** |
| Personal Finance Application- Grow More | Oshin Raina, Sakshi Mishra, Prof.  Santoshi Pote | Personalized Financial Consultant :Managing finance can be complex. People expect their personal finance apps to do the tricky analytics part for them: the apps should be capable of analysing customers’ data and choosing the best financial products, retirement plans,  investment areas, etc. | Clear Reporting: Reporting features are what customers love most about personal finance solutions. The data in this report, however, should be Presented in clear and simple formats |
| Personal Financial Management Behaviour Using Digital Platforms and Its Domains | Neetu Chhillar,  Swaranjeet Arora | The present quantitative study started with identifying the main dimensions of PFMB from significant studies and empirically examined those using survey data. | This study is the first to provide a thorough and  psychometrically verified instrument covering all the core dimensions of PFMB to measure the DFMB and additionally examining the impact of the core dimensions of PFMB on the DFMB of individual living in metropolitan region of developing country like India. |
| Mobile Bookkeeper: Personal Financial Management Application with Receipt Scanner Using Optical  Character Recognition | Manuel B. Garcia, Julius P Claour | This developmental-usability study focused on designing and developing a budget app with OCR technology. | In summary, a budgeting application with integration of OCR technology, Mobile Bookkeeper, was developed for personal financial management |
| **Paper Name** | **Authors** | **Methodology** | **Scope/Future** |
| Android Based Personal Finance  Management Application: Design and Development | Brilly Andro Makalew | The methodology in specifically chosen to incorporate software development. Unified  Modelling Language (UML) based diagrams were used. Although there are a lot of diagrams to use in the  process of implementing UML, we used Use Case Diagram and Class Diagram to represent the model for  this application. | This application provides novelty features compared to other similar applications |
| Personal Finance Apps and Low‐ Income Households | Elaine Stewart | The authors collaborated with five NI credit unions, and two local technology companies developed four personal finance apps to target, facilitate, and  improve different aspects of an individual's financial capability. | The apps were evaluated concerning the UK Financial Capability Outcome Framework (Bagwell et al., 2014). In this model, financial capability is a combination of ability (knowledge, understanding, and basic skills) and mindset (attitudes and motivations) as well as external factors not within an individual's control (access barriers or social influences). |

# Chapter 3

### Software Requirements Specification

### 3.1 Project Scope:

Our application is for domestic use and for small retailers only. Our application does not meet all the requirements of higher-level businesses. Our application can take advantage of a good market as it is usable by anyone who is willing to manage their expenses and aims to save for future investments and many more.

### 3.2 Perspective:

Our product will play a vital role in budget management. Our product solves the problem of managing the budget on pen and paper and can help users to manage their future savings.

### 3.3 Product Features

* Login
* Registration
* Insert and retrieve income data
* Insert and retrieve expense data
* Update data
* Search income/expense data by date
* Calculation of total income data
* Calculation of total expenses data
* Graphical representation of expense data
* Change password and delete account
* Tax and EMI calculator
* About us and Rate us
* Logout

### 3.4 Operating Environment

|  |  |  |
| --- | --- | --- |
|  | **Development Environment** | **Operational Environment** |
| **HARDWARE REQUIREMENT S** | Processor: intel core i5  RAM 8GB or more  Hard Disk: 80 GB or more | RAM: 1 GB or more  ROM 4 GB or more  Processor: 1 GHZ Dual core or Higher |
| **SOFTWARE REQUIREMENT S** | Android Studio 4.1.2 or higher  Android Application Package (.apk) | Network connectivity  Android 4.0 or higher |

### 3.5 System Features and Requirements

## 3.5.1 Registration Feature

# Description and Priority

This feature will collect and manage information of users like Email and Password. Priority of this feature is High.

# Stimulus/Response Sequences

Stimulus: User requests to enter their email and password.

Response: account veriﬁcation link sent to their registered e-mail. (users need to verify their email account for login to the application.)

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Feature ID | Feature Name | Description |
| FR-UR-1 | Insert User email and set a password | The application shall allow users to insert a new user email and password details. Password should be at least 6 characters long |
| FR-UR-2 | Email veriﬁcation | The application shall allow users to verify their email account for security purposes. |

## 3.5.2 Login Feature

# Description and Priority

This feature will let the users login into the application by checking their credentials. Priority of this feature is High.

# Stimulus/Response Sequences

Stimulus: User requests to enter their registered email and password.

Response: if correct credentials then login successful otherwise login failed.

Stimulus: Users have a choice for the “Remember Me” option.

Response: If the option is checked then the user need not to enter credentials again.

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Feature ID | Feature Name | Description |
| FR-LF-1 | Insert Credential Details of user. | The application shall allow users to insert their credentials for login. |
| FR-LF-2 | Remember me | If the option is checked then the user need not to enter credentials again when entering the app. |

## Forgot Password Feature

# Description and Priority

This feature will provide a facility to set a password again to the user if they forgot their password. An email will be sent to their registered mail account and then they can change their password. Priority of this feature is High.

# Stimulus/Response Sequences

Stimulus: User is requested to enter their registered email for sending the reset password link Response: password successfully changed.

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Feature ID | Feature Name | Description |
| FR-FP-1 | Reset password | The application shall allow user to reset their password through link sent to their registered mail id in case they forgot their password. |

## Dashboard

# Description and Priority

This feature will display the total income, expense, navigation drawer, navigation toolbar, history of income and expenses and ﬂoating button for entering data in income and expense.

Priority of this feature is High.

The application shall allow user to reset their password through link sent to their registered mail id in case they forgot their password.

# Stimulus/Response Sequences

Stimulus: User requests to add a new income information.

Response: System provides a layout to enter amount, Type and note of income to add it in income fragment layout.

Stimulus: User requests to add a new expense information.

Response: System provides a layout to enter amount, Type and note of income to add it in income fragment layout.

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Feature ID | Feature Name | Description |
| FR-DB-1 | “+” ﬂoating button | Allow users to select if they want to enter income or expense details. |
| FR-DB-2 | Add Income details | Application provides a layout to enter amount, Type and note of income to add it in the income fragment layout. |
| FR-DB-3 | Add Expense details | Application provides a layout to enter amount, Type and note of expense to add it in expense fragment layout. |
| FR- DB -4 | Navigation toolbar | The application shall allow user to navigate through di erent fragments like income, expense and dashboard by bottom toolbar |
| FR- DB -5 | Navigation drawer | The application shall allow users to navigate through di erent fragments like income, expense and dashboard by drawer. |
| FR- DB -6 | Income scroll view | The application shall display a scroll view of all previous income details. |
| FR- DB -7 | Expense scroll view | The application shall display a scroll view of all previous expense details. |

## Income Management

# Description and Priority

This feature will collect and manage all information of income and display total income. Users can also update and delete their income data. Priority of this feature is High.

# Stimulus/Response Sequences

Stimulus: User requests to modify income details.

Response: System provides a dialog box for users to modify the details.

Stimulus: User requests to delete income details.

Response: System provides a dialog box for users to delete the details and system will

remove details from .

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Feature ID | Feature Name | Description |
| FR-IM-1 | Update income details | The application shall allow users to make changes to an existing income detail. |
| FR-IM-2 | Delete income details | The application shall allow users to delete an existing income detail. |

## 3.5.6 Expense Management

# Description and Priority

This feature will collect and manage all information of expense and display total expense. Users can also update and delete their expense data.

Priority of this feature is High.

# Stimulus/Response Sequences

Stimulus: User requests to modify expense details.

Response: System provides a dialog box for users to modify the details.

Stimulus: User requests to delete expense details.

Response: System provides a dialog box for users to delete the details and system will remove details from .

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Feature ID | Feature Name | Description |
| FR-EM-1 | Update expense details | The application shall allow users to make changes to an existing expense detail. |
| FR-EM-2 | Delete expense details | The application shall allow users to delete an existing expense detail. |

## Search Income/Expense

# Description and Priority

This feature will search information of income/expense and display all details of

income/expense.

Priority of this feature is High.

# Stimulus/Response Sequences

Stimulus: User requests to search income/expense details by date.

Response: System provides the details of income/expense on the provided date.

Stimulus: User requests to search income/expense details by month (ﬁrst three letters of that month ie. “mar” for march).

Response: System provides the details of income/expense on the provided month

Stimulus: User requests to search income/expense details by year.

Response: System provides the details of income/expense on the provided year.

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Feature ID | Feature Name | Description |
| FR-SIE-1 | Search Income/Expense | The application shall allow users to search existing income/expense details. |

## Tax Calculator

# Description and Priority

This feature will calculate income tax and display gross taxable and deductions

Priority of this feature is medium.

# Stimulus/Response Sequences

Stimulus: User requests to calculate income tax.

Response: System provides the layout to enter basic information to calculate total tax i.e. basic salary, HRA, allowances, deductions etc. and displays total tax.

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Feature ID | Feature Name | Description |
| FR-TC-1 | Tax Calculator | The application shall allow users to calculate the total tax of their basic income. |

## EMI Calculator

# Description and Priority

This feature will calculate EMI and display total interest.

Priority of this feature is medium.

# Stimulus/Response Sequences

Stimulus: User requests to calculate total interest.

Response: System provides the layout to enter principle amount, interest rate and year and displays EMI and total interest.

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Feature ID | Feature Name | Description |
| FR-EC-1 | EMI Calculator | The application shall allow users to insert principle amount, rate and year and should display total interest. |

## Alert of Low balance

# Description and Priority

This feature will alert the user with a message if balance is less than 10% or 5% of total income.

Priority of this feature is High.

# Stimulus/Response Sequences

Stimulus: User has exceeded the limit of expense (balance<10% or 5% of income).

Response: System alerts the user with a dialog box that he/she has exceeded the limit of expense and displays a message that the user needs to control their expenses.

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Feature ID | Feature Name | Description |
| FR-ALB-1 | Alert of Low balance | The application shall alert the user if he/she exceeded the limit for expenses. |

# 

### 3.6 Other Nonfunctional Requirements

## 3.6.1 Software Quality Attributes

The quality attributes of the software are as follows:

Our application is operational, functional and usable for completing or fulﬁlling a user's

**Maintainability**: Our application is understandable, repairable, or enhanced.

**Reliability**: Our Application is completely defect free, does not create downtime, and performs correctly in every scenario.

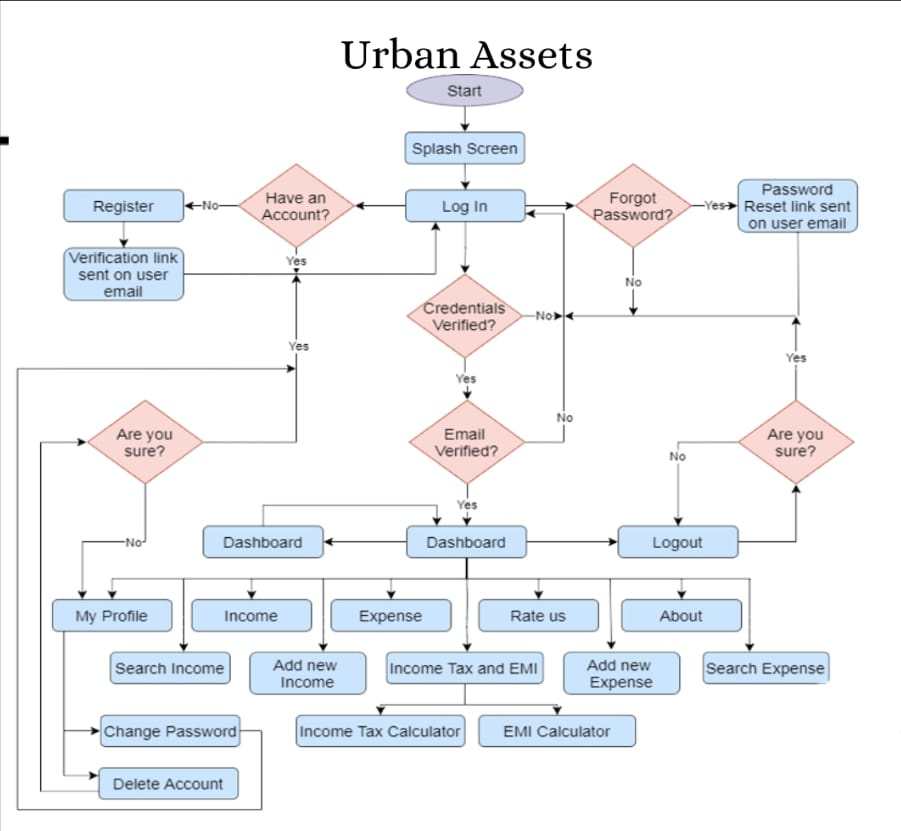
**Usability**: Our application is effective, efficient and satisfactory for all the users.

**Portability**: Our application is portably installed, deployed, accessed and managed - regardless of delivery model, when used on multiple platforms or instantly accessed.

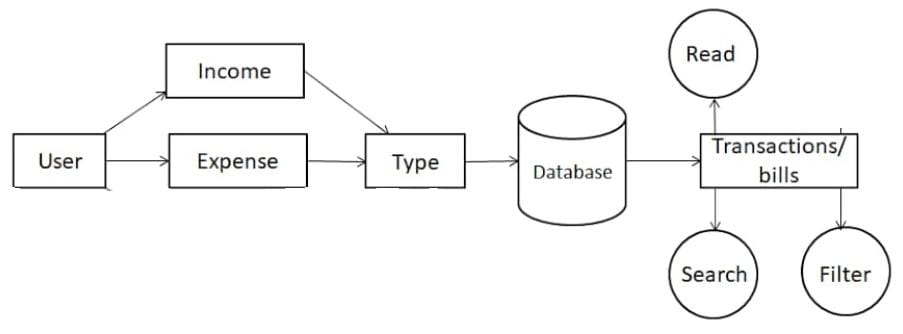
**Correctness:** When our application is used correctly, users are able to interact with the system

# Chapter 4

**System Design**



**Fig 4.1 System Architecture**

****

**Fig 4.2 Working Diagram**

A diagram of a project

Description automatically generated

**Fig 4.3 Sequence Diagram**

A diagram of a person's work flow

Description automatically generated

**Fig 4.4 Use Case Diagram**

# Chapter 5

**Testing Strategy Plan**

**Objective**: Ensure the expense tracker app is reliable, secure, and user-friendly, meeting the needs of users for managing their expenses effectively.

* **Functional testing** of core features such as expense recording, categorization, budget management, and reporting.
* **Usability testing** to assess the app's intuitiveness, ease of navigation, and overall user experience.
* **Security testing** to validate the protection of user data and financial information.
* **Compatibility testing** across different devices (iOS, Android) and screen sizes.
* **Performance testing** to evaluate the app's responsiveness and resource usage under various conditions.
* **Localization testing** if the app supports multiple languages.

**Test Environment:**

* Use real devices and emulators/simulators for testing across different platforms (Android).
* Set up test environments with varying network conditions to simulate real-world usage scenarios.
* Utilize cloud-based testing services for scalability and to cover a wide range of device configurations.

**Test Data:**

* Create diverse sets of test data covering different expense types, amounts, categories, and dates.
* Include boundary cases, negative scenarios, and edge cases to ensure robustness.
* Ensure data privacy compliance by using anonymized or synthetic test data.

**Test Techniques:**

* **Functional Testing**:

Validate expense recording, editing, and deletion functionalities.Verify proper categorization and tagging of expenses.Test budget setting and tracking features. Ensure accurate generation of reports and summaries.

* **Usability Testing:**

Conduct user walkthroughs to assess ease of use and intuitiveness.

Collect feedback on UI design, navigation, and overall user experience.

* **Security Testing:**

Perform penetration testing to identify vulnerabilities.Validate encryption mechanisms for protecting user data.Ensure secure authentication and authorization processes.

* **Compatibility Testing**:

Test the app on various devices, screen sizes, and operating system versions. Verify compatibility with different browsers (if web-based).

* **Performance Testing**:

Measure app responsiveness under different loads and network conditions. Assess resource consumption and optimize for efficiency.

* **Reporting:**

Generate comprehensive test reports highlighting test coverage, pass/fail rates, and critical issues. Provide actionable insights for stakeholders to make informed decisions.

* **Feedback and Iteration:**

Gather feedback from users and stakeholders throughout the testing process. Incorporate feedback into iterative development cycles to improve the app's quality and user satisfaction.

* **Documentation:**

Maintain detailed documentation of test plans, test cases, test data, and test results for future reference and knowledge sharing

**Chapter 6**

**Project Plan**

* **Project Initiation:**

Define project objectives, scope, and requirements. Identify stakeholders and their roles. Establish communication channels and project management tools.

* **Market Research and Analysis:**

Conduct market research to understand user needs and competitor offerings.Analyze user feedback and reviews of existing expense tracker apps.Define key features and unique selling points (USPs) for the app.

* **Requirement Gathering and Analysis:**

Gather detailed functional and non-functional requirements for the app. Prioritize requirements based on importance and feasibility. Create user stories or use cases to capture different scenarios.

* **Design Phase:**

Develop wireframes and mockups to visualize the app's interface and navigation flow. Design database schema for storing expense data securely. Finalize the technology stack and architectural design.

* **Development:**

Break down development tasks into sprints (if following Agile methodology).

Develop core features such as expense recording, categorization, budget management, and reporting. Implement authentication and authorization mechanisms for user accounts.Integrate third-party APIs for currency conversion, receipt scanning (if applicable), and data synchronization. Conduct code reviews to ensure code quality and adherence to coding standards.

* **Testing:**

Develop test cases based on functional and non-functional requirements. Conduct unit testing, integration testing, and system testing. Perform manual and automated testing across various devices and platforms. Address and fix defects identified during testing iterations.

* **Deployment:**

Prepare deployment packages for App Store (iOS) and Google Play Store (Android). Submit the app for review and approval. Plan for a phased rollout or beta testing with a limited audience.

* **Launch and Post-Launch Activities:**

Monitor app performance and user feedback after launch. Address any issues or bugs reported by users promptly. Plan for marketing and promotional activities to increase app visibility. Gather analytics data to track user engagement and app usage patterns.

* **Maintenance and Support:**

Establish a system for handling user inquiries, feedback, and support requests. Release updates and new features based on user feedback and market trends. Regularly maintain and update the app to ensure compatibility with new devices and operating system versions.

* **Documentation and Knowledge Transfer:**

Document all project artifacts including requirements, designs, test cases, and deployment procedures. Conduct knowledge transfer sessions with the development team to ensure continuity and future enhancements.

* **Project Closure:**

Conduct a project review to evaluate achievements against objectives and lessons learned. Archive project documentation and assets for future reference. Celebrate project success and acknowledge contributions from team members.

**Chapter 7**

**Result Discussion**

* 1. **User Interfaces**

|  |  |  |
| --- | --- | --- |
| UI Name | UI image | Description |
| Splash screen |  | Introduction screen of our application |
| Login |  | Users may enter the credentials for login and also have the facility of “remember me”. |
| Registration |  | Users may register and set a password. |
| Forgot password |  | Users may reset their password in case they forgot their password. |

|  |  |  |
| --- | --- | --- |
| Dashboard |  | In the dashboard, users can navigate through the dashboard, income and expense fragments. Displays total income and expenses. Users can add details of new income and expenses through clicking on the ﬂoating button. |
| Drawer layout |  | Drawer layout has options to navigate through income, expense and dashboard fragments. |

|  |  |  |
| --- | --- | --- |
| Insert data layout |  | Users can insert details of new income and expenses like amount, type and note. |
| Income fragment |  | Displays data of total income and separate details of each income. |

|  |  |  |
| --- | --- | --- |
| Expense fragment |  | Displays data of total expense and separate details of each expense. |
| Change Password layout |  | Layout of Change Password: This feature provides the user to Change Password. If a user wants to change a password he can do it using this function. |

|  |  |  |
| --- | --- | --- |
| Delete Account Alert |  | This displays the App's current version and the names of the developers of this Application. |
| Proﬁle |  | This feature helps the user to change his login password and also helps in deleting account from Spendee permanently |

|  |  |  |
| --- | --- | --- |
| Search layout |  | This feature helps the user to ﬁnd a particular income/expense data modiﬁed by date(i.e either Day, Month or Year). |
| Rate Us Layout |  | This feature allows the user to give rating to this application based on their liking and interest. |

|  |  |  |
| --- | --- | --- |
| Logout Alert |  | This displays the App's current version and the names of the developers of this Application. |
| Income Tax and Calculator |  | Users may choose if they want to calculate their income tax or |
|  | EMI/total interest. |

|  |  |  |
| --- | --- | --- |
| Tax details form |  | Users may insert details like basic salary and allowances. Then may choose to calculate total tax or continue. |
| Layout of Tax Calculation |  | This layout displays the information that was inserted by the user and displays total tax. |

|  |  |  |
| --- | --- | --- |
| Layout of EMI Calculator |  | Users may insert details like Principal amount, interest rate and duration. Then display total interest and EMI. |
| Low Balance alert of 5% balance |  | This feature alerts users that, user has less than 5% balance |
|  | remaining, which |
|  | means the user has a |
|  | very high expense |
|  | compared with his |
|  | income. the user has to |
|  | control his expenses. |

|  |  |  |
| --- | --- | --- |
| Low Balance alert of 10% balance |  | This feature alerts the user that, user has less than 10% balance remaining that means the user has a high expense compared with his income. the user should control his expenses. |
| Statistical Dashboard |  | This displays the bar chart of per day expenses on Dashboard Fragment that is expensed by the user. Viewing it, the user comes to know how much he is expensing and now how he needs to control his expenses. |

.

**Chapter 8**

**Appendix A: Glossary**

TBD – To Be Done UI – User Interface

FR – Functional Requirements UR – User Registration

LF – Login Feature FP – Forgot Password DB – Dashboard

IM– Income Management

EM– Expense Management

SIE– Search Income/Expense

TC– Tax Calculator

EC– EMI Calculator

ALB– Alert of Low Balance

**Chapter 9**

**References**

Expensify: <https://www.expensify.com/>

Zoho Expense: <https://www.zoho.com/expense/>

QuickBooks Online: <https://quickbooks.intuit.com/online/>

YNAB (You Need a Budget): <https://www.youneedabudget.com/>

Mint: <https://www.mint.com/>

PocketGuard: <https://www.pocketguard.com/>

Wave: <https://www.waveapps.com/>