SOFTWARE REQUIREMENT SPECIFICATIONS (SRS) FOR MONEY MANAGEMENT SOFTWARE

DEVANSHI GUPTA	**180101019**
MANSHI SHARMA	**180101041**
NISHTHA SHARMA	**180101052**
SHIVANGI KUMAR	**180101075**

INDEX

1.Introduction

1.1 Purpose

The purpose of this project is to provide a detailed overview and determine the parameters of our software product, which aims at Money Management and budgeting. This document describes the essential features and requirements keeping in mind the target audience: college and university students for whom money management can feel monstrous and overwhelming.

We shall also predict and sort out how we hope this product will be used in order to gain a better understanding of the project, outline concepts that may be developed later, and document ideas that are being considered, but may be discarded as the product develops. It defines how our client, team and audience see the product and its functionality. It helps any designer and developer to assist in software delivery lifecycle (SDLC) processes.

1.2 Scope

This software system will be a money management and budgeting software. Primarily, the scope pertains to the Money Management related issues students face in their day-to-day lives. It focuses on the college/university going students for whom money management can feel monstrous and overwhelming and enables them to primarily keep a track of their credit and debit transactions while maintaining a budget to stick to.

2.Functional Requirements

2.1: Sign Up

Description: This function is used to register a new user by storing the details as taken in input and storing them. It gives a success message if the user is registered correctly or an error message otherwise.
Input: Name, Email, Password, Phone Number
Output: Success/Error Message

2.2: Login

Description: The function is used to log in an existing user by taking the details, verifying them. It gives a success message and logs the user in if the user exists and gives an error message otherwise.
Input: Email, Password
Output: Success/Error Message

2.3: Create Budget

Description: This function is used to create a budget for a custom time period, the user specifies that. The users can create multiple income and expense categories and specify the amount they earn and intend to spend under each of those categories, respectively. Such a categorised budget will be more manageable for the users. It will be more informative for them since they can easily monitor their various sources of income and they'll know under which category they are spending the most and so on. Accordingly, they can manage their expenses. Also, the user will be able to specify a saving amount. This will specify how much the user expects to save in the time that the budget is created for. This will help him keep a track of his savings as well.

- 2.3.1: Specify time period for budget
    - **Input**: Start date and end date
    - **Output**: Prompt to create category
    - **Description:** The user can specify the start date and end date and thus the time period for which the user intends to create that budget for. The function then asks the user to move to the next step and create various categories for their income and expenses.

- 2.3.2: Create Category

  Description: This function is used to create multiple income and expense categories.

        - 2.3.2.1 Income Category
            - **Input:** Name, Amount
            - **Output:** Display created category
            - **Description:** This function is used to create categories for income (for eg, parental allowance, part-time jobs, etc.) and fix an amount for that category. The user can create multiple categories and specify the amount that they earn in that category. This will help the users keep a track of their various sources of income. The function then displays the income categories that the user just created.

        - 2.3.2.2 Expense Category
            - **Input:** Name, Amount
            - **Output:** Display created category
            - **Description:** This function is used to create categories for income (for eg, parental allowance, part-time jobs, etc.) and fix an amount for that category. The user can create multiple categories and specify the amount that they earn in that category. This will help the users keep track of their various sources of income. The function then displays the income categories that the user just created.

- 2.3.3: Get saving amount
    - **Input**: Amount
    - **Output**: Display budget created
    - **Description**: This function takes the saving amount for the set time period of the budget specifying that the user aims to save this much amount in the stipulated time. The output is the final budget created along with all the details gathered in other functions above it in the hierarchy.

2.4: Register transaction

Description: This function registers a transaction (income or expenditure depending on the category). It updates the amount from the corresponding category in the budget and a warning message is generated if the spent amount exceeds the savings specified or the budget amount for that category. It can also take the receipt of the transaction and store it.
Input: Category, Amount, Receipt
Output: Success/Warning message

2.5: Coupons

Description: This function is used to add a new coupon or use an existing coupon.

- 2.5.1: Add a coupon
    - **Input:** Coupon name, coupon code
    - **Output:** Display coupon
    - **Description:** This function is used to add a new coupon and displays the coupon just added.

- 2.5.2: Use a coupon
    - **Input:** Use coupon option
    - **Output:** Updated coupon list
    - **Description:** This function deletes the coupon from the list of existing coupons and displays the updated list.

2.6: Record Credit or Debit Transactions

Description: This function is used to register any amount credited or debited in personal transactions (for ex. Borrowing from people, lending to people) to help the user keep a record of these.

- 2.6.1: Register credit
    - **Input**: Name, Amount
    - **Output**: Updated credit list
    - **Description**: This function takes the lender's name and the credit amount, updates the list of credit transactions with these attributes and displays it.

- 2.6.2: Register debit
    - **Input**: Name, Amount
    - **Output**: Updated debit list
    - **Description**: This function takes the borrower's name and the debit amount, updates the list of debit transactions with these attributes and displays it.

2.7: Export data

Description: This function exports the data for a particular budget (out of many that the user may have) to a particular file format as selected by the user (csv, excel, etc).
Input: Budget, file format
Output: File

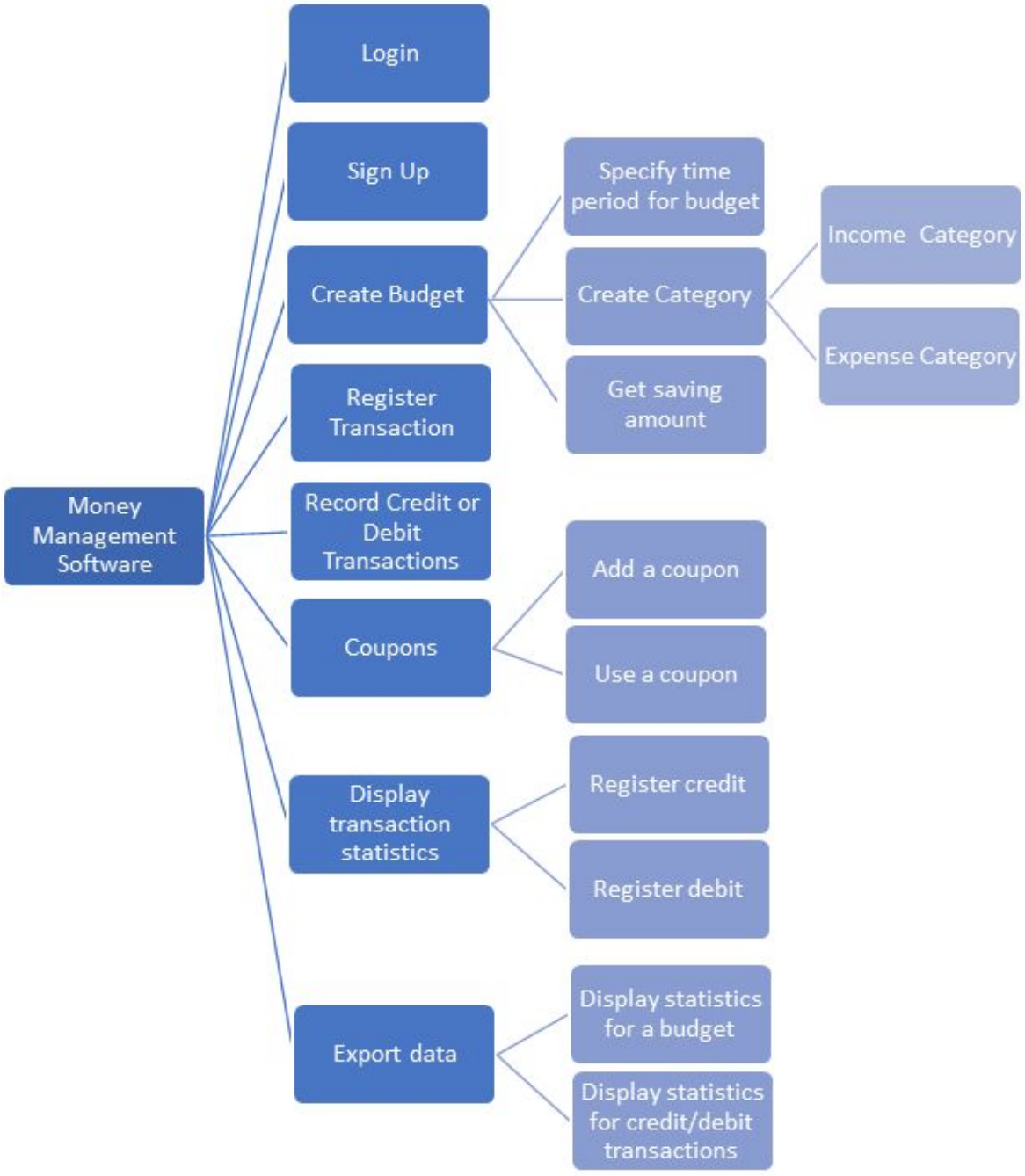
2.8: Display transaction statistics

Description:

- 2.8.1: Display statistics for a budget
  - **Input**: Budget
  - **Output**: Graphs
  - **Description**: This function takes a particular budget (out of many that a user may have) and generates multiple graphs corresponding to the data of that budget and displays them. This gives the users a visual analysis of their income and expenses.

- 2.8.2: Display statistics for credit/debit transactions
  - **Input**: Start date, end date
  - **Output**: Graphs
  - **Description**: This function takes the start date and end date and the credit and debit transactions within that time period are considered. It generates multiple corresponding graphs and displays them to give the users a visual analysis of their transactions.

HIERARCHY DIAGRAM:

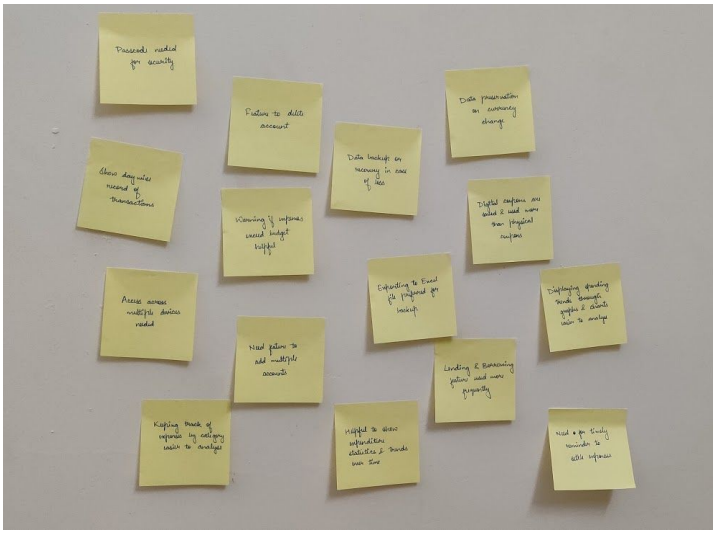


3.Nonfunctional Requirements

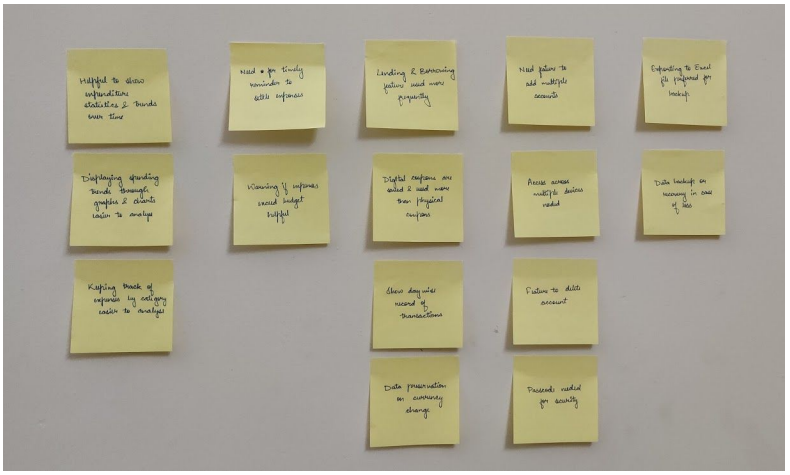
3.1 CONTEXTUAL INQUIRY

3.1.1 Process and Environment

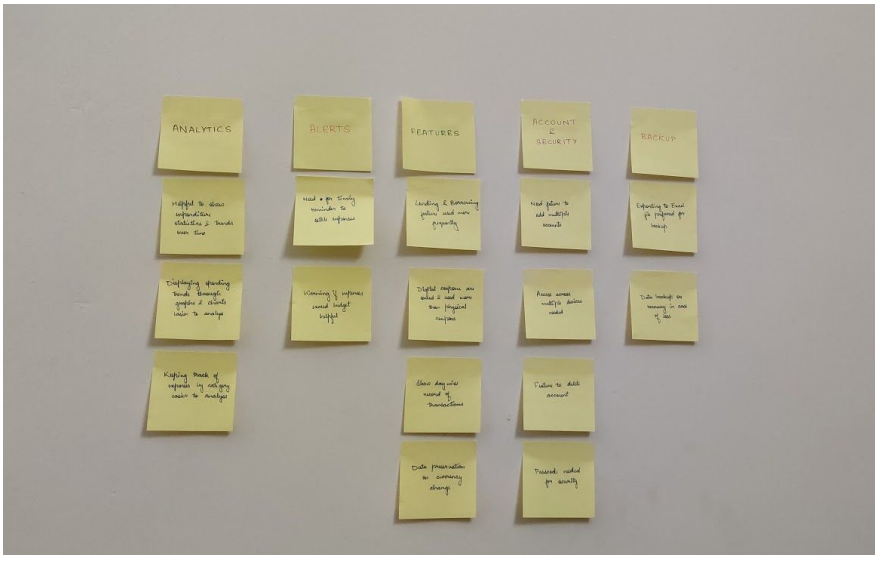
For this contextual inquiry, we observed a number of our friends over video conferencing from their home where they had access to their mobile phones and computer. First, we made them use their current money management software to create a dummy budget for a week. We made them share their screen and observed them in the process. We also stayed in constant contact with them over that week and observed how they entered data while asking them various questions about anything that we did not understand or found particularly interesting. Also, we made them register some credit as well as debit transactions on their software and observed that as well. We kept recording these observations and shared them at the end of the day. We noted them on post-its on a wall and later sorted them and created the header cards accordingly. We typically concluded our sessions by thanking them and asking for suggestions.



Generating random ideas



Sorting and arranging the ideas



Creating header cards

FINISHED DIAGRAM:

Analytics	Alerts	Features	Account and Security	Backup
Helpful to show expenditure statistics and trends over time	Need for a timely reminder to settle expenses	Lending and borrowing feature used more frequently	Need a feature to add multiple accounts	Excel file preferred for data export over other options
Displaying spending trends through graphs and charts easier to analyse	A warning if expenses exceed budget is usually helpful	Digital coupons are saved and used more than physical ones	Access across multiple devices needed	Data backup or recovery if lost
Keeping track of expenses by category makes it easier to analyse		Show day-wise record of transactions	Make it possible to delete account	
		Data preservation on currency change	Passcode needed for security	