

**MES3023**

**SOFTWARE REQUIREMENT AND SPECIFICATION**

**PROJECT: MONEY PLANNER**

|  |  |  |
| --- | --- | --- |
| **NO.** | **NAME** | **MATRIC NUMBER** |
| 1. | ANAS SAFWAN BIN ABDUL HALIM | D20171076716 |
| 2. | MUHAMMAD AKMAL HANIF BIN HAMZAH | D20171076746 |
| 3. | SHIMOL KHAN | D20171079547 |
| 4. | NUR AZILA BINTI AZMI | D20171076785 |
| 5. | SHOLESTICA ELMIE ANAK DANSY | D20171076804 |
| 6. | SHAHIRAH BINTI MALUHI | D20171076762 |

**GROUP : C**

**LECTURER’S NAME : DR. AZNIAH BINTI ISMAIL**

**MONEY PLANNER**

**SOFTWARE REQUIREMENT AND SPECIFICATION**

**VERSION 1.0**

**12 DECEMBER 2018**

ANAS SAFWAN BIN ABDUL HALIM

**LEAD SOFTWARE ENGINEER**

**TEAM MEMBERS:**

MUHAMMAD AKMAL HANIF BIN HAMZAH

SHIMOL KHAN

NUR AZILA BINTI AZMI

SHOLESTICA ELMIE ANAK DANSY

SHAHIRAH BINTI MALUHI

**PREPARED FOR**

**MES3023**

**SOFTWARE REQUIREMENT AND SPECIFICATION**

**LECTURER: DR. AZNIAH BINTI ISMAIL**

**SEMESTER 1 2018/2019**

**REVISION HISTORY:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| 18/12/2018 | Version 1.0 | Anas Safwan Bin Abdul Halim | - |

**DOCUMENT APPROVAL:**

The following Software Requirements and Specification has been accepted and approved by the following:

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  | Anas Safwan Bin Abdul Halim | Lead Software Engineer | 15 December 2018 |
|  | Dr. Azniah Binti Ismail | Lecturer, MES3023 | 15 December 2018 |

**TABLE OF CONTENTS**

**REVISION HISTORY**…………………………………………………………………….i

**DOCUMENT APPROVAL**……………………………………………………………….i

1. **INTRODUCTION**……………………………………………………………………..1-2
   1. PURPOSE……………………………………………………………………….1
   2. SCOPE…………………………………………………………………………..1
   3. GLOSSARY…………………………………………………………………….1
   4. REFERENCES……………………………………………………………….....2
   5. OVERVIEW…………………………………………………………………….2
2. **GENERAL DESCRIPTION**………………………………………………………….3-5
   1. PRODUCT PERSPECTIVE……………………………………………………3
   2. PRODUCT FUNCTIONS……………………………………………………...3
   3. USER CHARACTERISTICS…………………………………………………..4
   4. GENERAL CONSTRAINTS…………………………………………………..4
   5. ASSUMPTIONS AND DEPENDENCIES…………………………………….5
3. **SPECIFIC REQUIREMENTS**……………………………………………………….
   1. EXTERNAL INTERFACE REQUIREMENTS……………………………….
      1. USER INTERFACES……………………………………………….6-16
      2. HARDWARE INTERFACES………………………………………17-18
      3. SOFTWARE INTERFACES……………………………………….18
      4. COMMUNICATIONS INTERFACES…………………………….18
   2. FUNCTIONAL REQUIREMENTS……………………………………………
   3. USE CASE………………………………………………………………………23
   4. QUALITY REQUIREMENTS…………………………………………………24
   5. LOGICAL DATABASE REQUIREMENT……………………………………25-28
4. **ACTIVITY DIAGRAM**……………………………………………………………….29-35
5. **INTRODUCTION**
   1. **Purpose**

The purpose of this document is to present a detailed description of the requirement for the “Money Planner”. As a student, this system is needed because it will help student to be more systematic in manage their financial problem. This document also will explain and illustrate the purpose and features of the system, the interface of the system, what the system will do, what the system cannot do, the constraints under which it must operate and how the system will react to stimuli. This document is primarily intended to be proposed to a student and a reference for developing the first version of the system for the development team.

* 1. **Scope of Project**

This software, MP will be created for student UPSI who want to manage their financial problem. This system is designed to make things more easier to track their expenses, before this system is created, they frequently facing the difficulty in manage their money, which it always occurs at student that have to facing lack of money in the end of month or the end of semester. More specifically, the system will make a tracker for every transaction that have been done by the student. So that student can find out where they've spent their money in each category.

* 1. **Glossary**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| UPSI | Universiti Pendidikan Sulltan Idris |
| MP | Money Planner |

**Table 1: Glossary for term and definition**

**1.4 References**

[ 1 ] Nur Azila, Muhammad Akmal Hanif, Shimol Khan, Anas Safwan, Sholestica Elmie, Shahirah (2018). Field Report, MES3023. Semester 1 2018/2019, Not Published.

**1.5 Overview of Document**

The next chapter, the General Description section of this document tells about functionality of the whole system.With Money Planner, UPSI student will be more sensitive about managing their financial. , Furthermore, next chapter also will describe about the stakeholders and the basic function that we use in this system. In the third chapter, Requirements Specification section of this document is dedicated for the developers and describes the technical terms and the details of the Money Planner functionality. Both sections of the document describe the software products specifically.

1. **GENERAL DESCRIPTION**

This section will describe the general overview about the system that we have made. It is including the product perspective which what our system is about. We will also explain its functions, type of stakeholder that will use this system, and lastly for this section is we will present the constraints and assumptions for this system.

* 1. **Product Perspective**

This system will only used by students and staff of UPSI. Money Planner system will help them manage their financial. Firstly, they need to create the account of Money Planner which mean they need to sign up before they can use it. After got their account, they need to log in to use this system. Any details of user which is related to the financial management will be asked so that this system may help them manage their financial as well.

While this system is made up of a database, it will save all of user information that they had been inserted. They may edit their information as they want. Any update regarding the system and their financial will be notified.

* 1. **Product Functions**

This system needs the user to create their Money Planner account to enable them use this system. The user need to sign up for the first time either as a lecturer or students. Using the system after they had been sign up just need them to log in whenever they want to use it. As this is a financial management system, the user will be asked to insert their financial details. The dashboard will show them their expenses and their current debit and credit. This system also can let the user to edit their profile to make sure their details is updated besides editing their expenses which they want to add or to remove it. This system will help them manage or plan their financial in every aspects including bills, transportation, communication, clothes, entertainment, food and many more. If there is any financial history that they want to recheck or to keep in their own device, they might do that in the financial report part which they also can download it. Our system provides the tracker for user to check their financial history. Any setting that they want to do regarding their currency, date format, financial report format and etc can be done on the setting part.

* 1. **User Characteristics**

Type of users that will used this system is as mentioned before which is students of staff from the UPSI. They can do all things that had mention in the product functions except tracking the financial history as that will be done by the tracker in the system itself. They only can ask for that financial report. This system is only created for those people in UPSI to help them in managing their financial so another than students or staff from UPSI cannot use this system. For sure those who are not from UPSI cannot create any account in this system.

* 1. **General Constraints**

Since the system is working with the internet, it will be the constraints for this system. The accessibility of Money Planner system will work depend on the internet because the system fetches data from the database over the internet. Besides, we are considering about the safety and security for this system and also the user details. It is because the system will store the user details regarding their financial plan.

Money Planner system can be access by both PC and also smartphone, it will be constrained by the capacity of the database. Since the database is shared between both application it may be forced to queue incoming requests and therefor increase the time it takes to fetch data.

* 1. **Assumptions and Dependencies**

There are assumptions and dependencies in Money Planner system.

1. **The user needs to go to the website before creating the account.**

By searching the website of <https://www.moneyplanner.com.my>, they will find the login page and they may create their Money Planner account.

1. **The user needs to create an account before login and use the system.**

Every user must have an account to use the system. Means that they need to create their username and put the password. That username and password will continuously use whenever they want to login to their Money Planner account.

1. **The user can edit or update their details in the system.**

User can edit their profile and update which part that need to be do so. It will easy for the system to keep them on update regarding their financial plan.

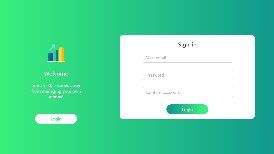
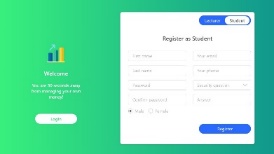
1. **The system will notify if their financial is exceed the plan or if there is any update for the user.**

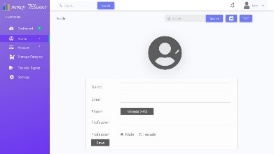
Any update regarding the system or the user financial plan will be given the notification to make sure they know what happen for their financial plan. So that they can take their action for any changes in the system or regarding their own expenses.

**3.0 Specific Requirement**

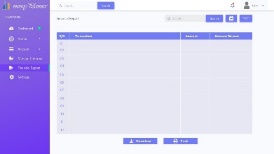
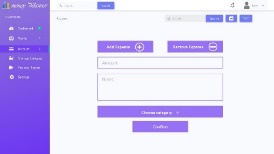
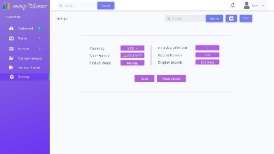
**3.1 External Interface Requirement**

**3.1.1 User Interface**





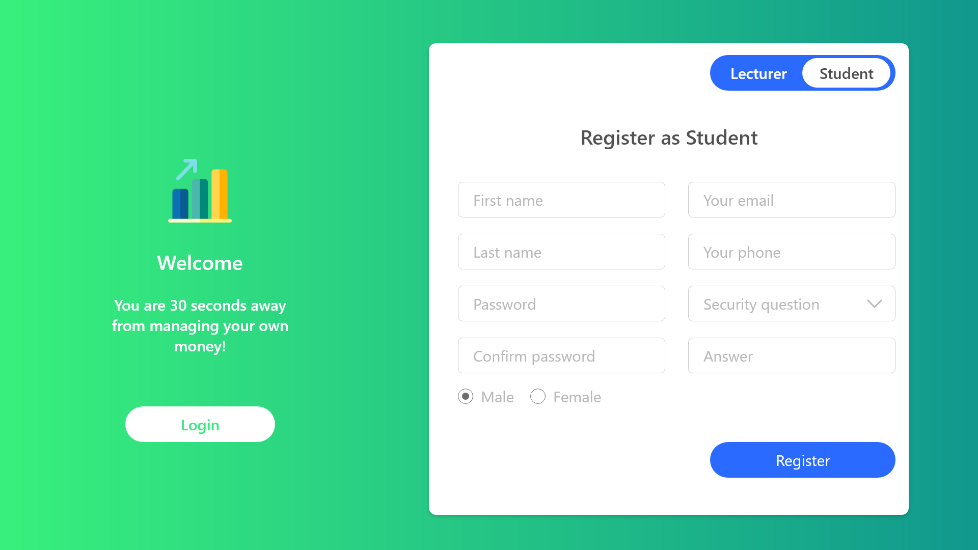




**3.1.1.1 Registration Interface**

Font type: Segoe UI

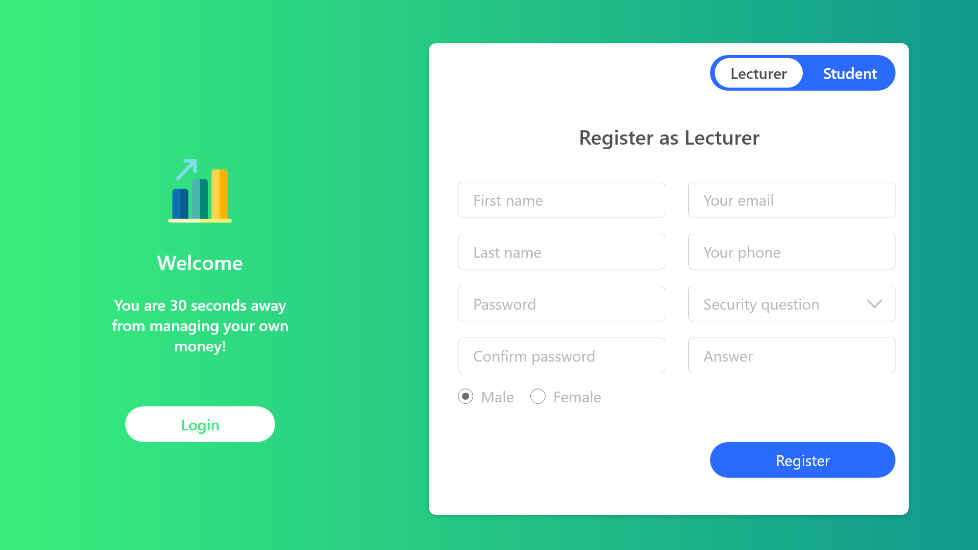
Size : 20, 22, 26 px



**Register**

The first use of Money Planner requires the planner to register into the apps by following their identity (students or lecturer).

* Sign up page for the Money Planner
* For student sign up options, specify for the students detail.



* For lecturers sign up options, specify for the lecturers detail.

**Login**

To go straight to login page

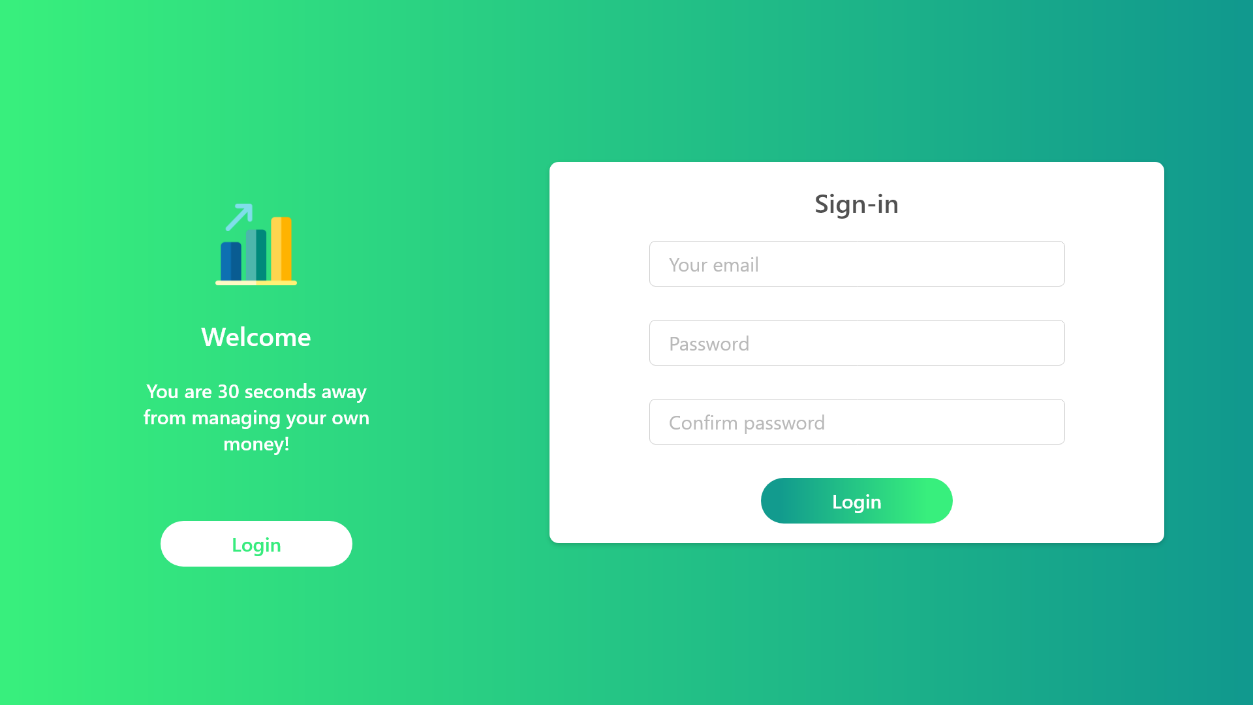
**Figure 3.1.1.1** Registration Interface

**3.1.1.2 Log-In Interface**

Font type: Segoe UI

Size : 20, 22, 26 px

* Require the detail of the planner for sign in options



**Login**

To access the main menu of Money Planner

* The sign-in page is directly after the sign-up page.
* Thus means it only can be access if the planner have pass through the sign up page or already have the account of the apps

**Figure 3.1.1.2** Log-In Interface

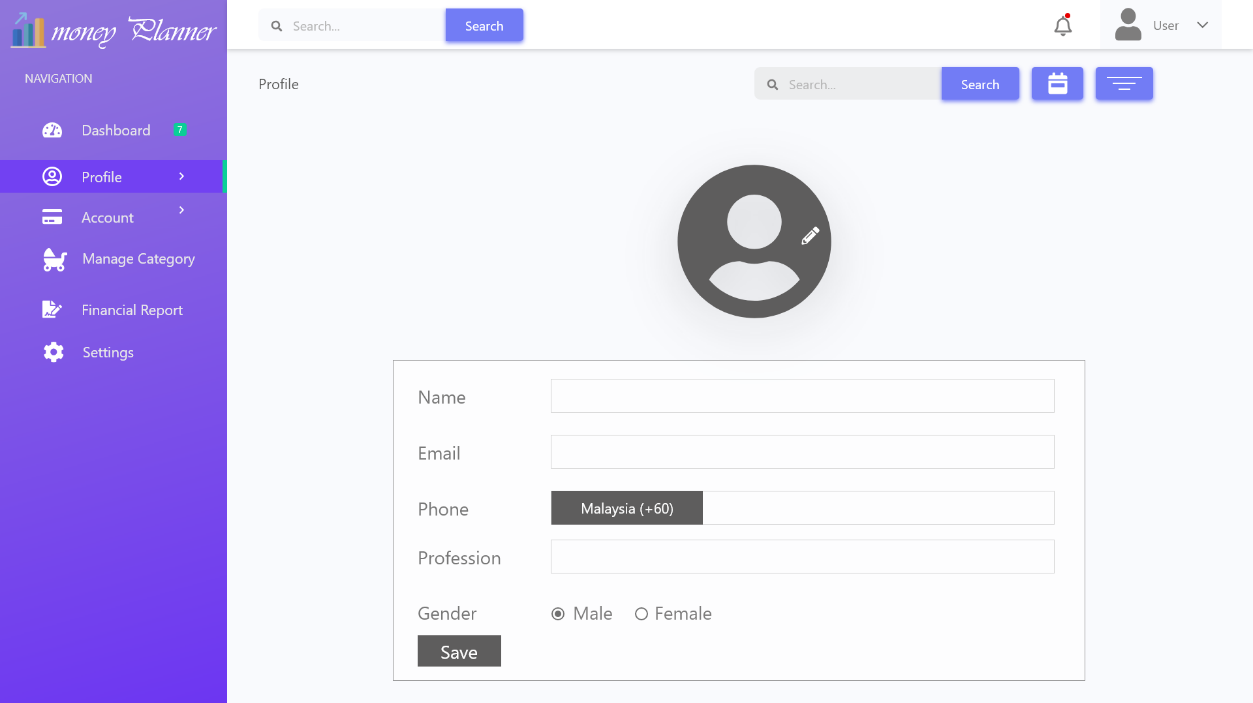
**3.1.1.3 Profile Interface**

Notifications bar and calendar for the use of the planner

Font type: Segoe UI

Size : 50, 20, 22, 26 px

* The profile viewer.
* Giving the planner options to edit their basic informtaion



* The easy access.
* For navigation to another page in the apps itself

Contains the basic information box of the planner.

**Figure 3.1.1.3** Profile Interface

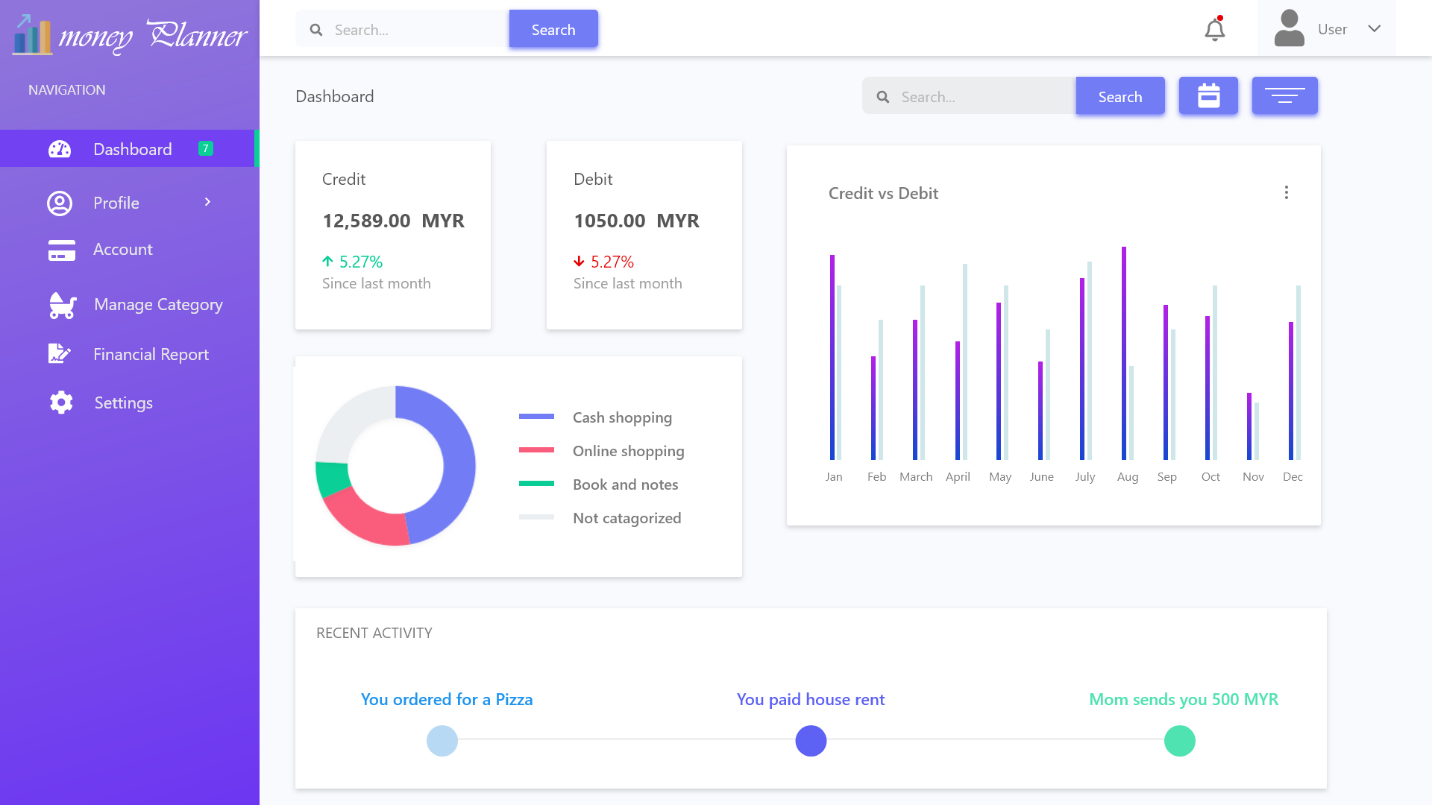
**3.1.1.4 Main Page Interface**

Font type: Segoe UI

Size : 50, 20, 22, 26 px

The dashboard of the apps.

Giving the planner overview of every activity that have been recorded



The value of the expenses regarding the debit and credit.

The diagram that show most activity that consume expenses.

The comparison for the usage of the credit and debit (only available after confirming the use of credit card)

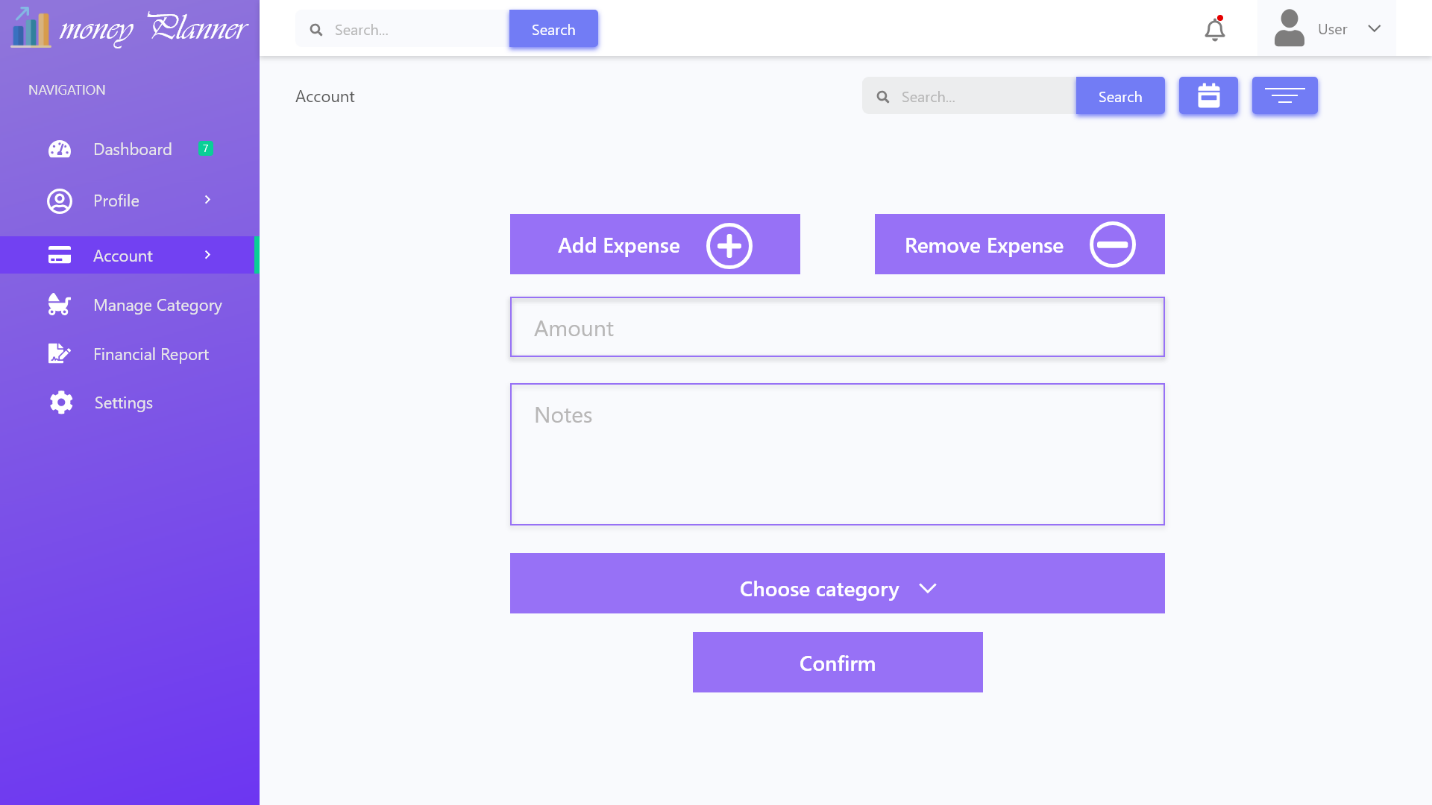
The recent activity that being recorded as a history (income or expenses )

**Figure 3.1.1.4** Main Page Interface

**3.1.1.5 Account Interface**

* To enter the value and notes for the expenses.

Lets the planner to add expenses or remove expenses.



By clicking this button, it will directed the user to a selection of category.

* For choosing the category of the expenses.

Font type: Segoe UI

Size : 50, 20, 22, 26 px

Saving the information and redirect the planner to the dashboard page

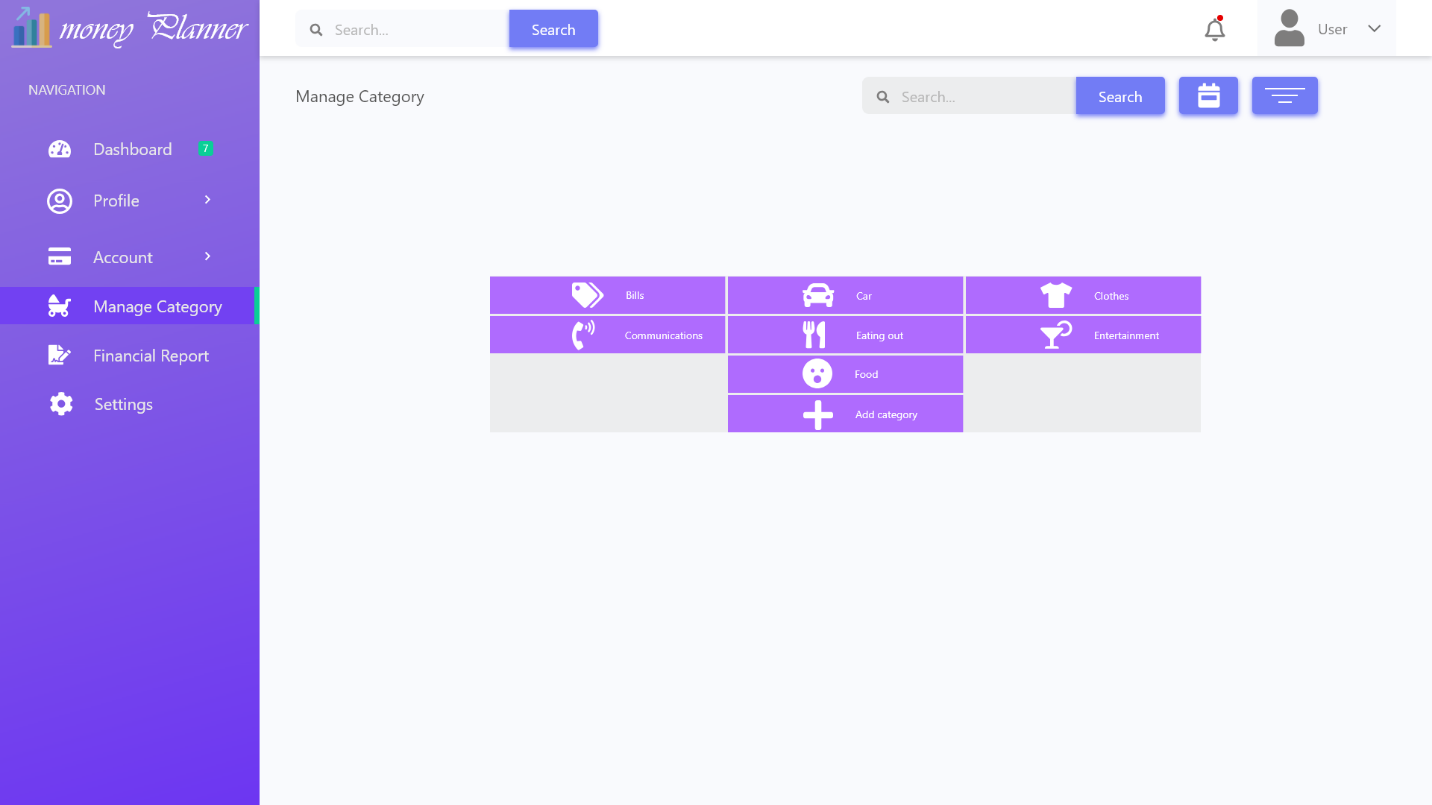
**Figure 3.1.1.5** Account Interface

**3.1.1.6 Expenses Category**

Font type: Segoe UI

Size : 50, 20, 22, 26 px

The category of the expenses



Any selection of the category will redirected back to the earlier page ( the account page)

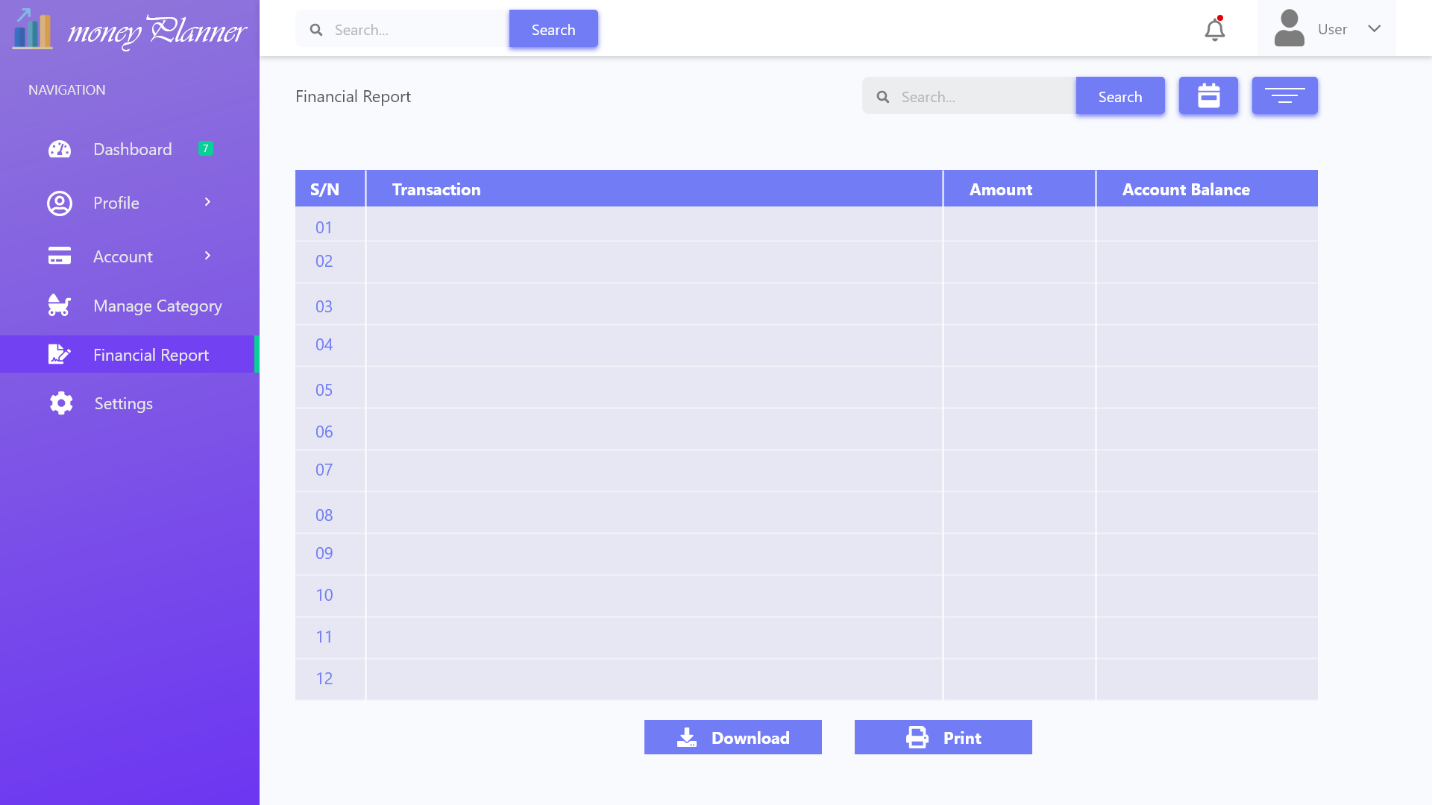
**Figure 3.1.1.6** Expenses Category

**3.1.1.7 Financial Report Interface**

The financial report of the apps, generated based on the recorded data regarding the expenses and income that occur.

Font type: Segoe UI

Size : 50, 20, 22, 26 px



Enabling the report to be downloaded and viewed online.

Giving the options of printing the report. As an option for those who want to keep track of their expenses in physical documents .

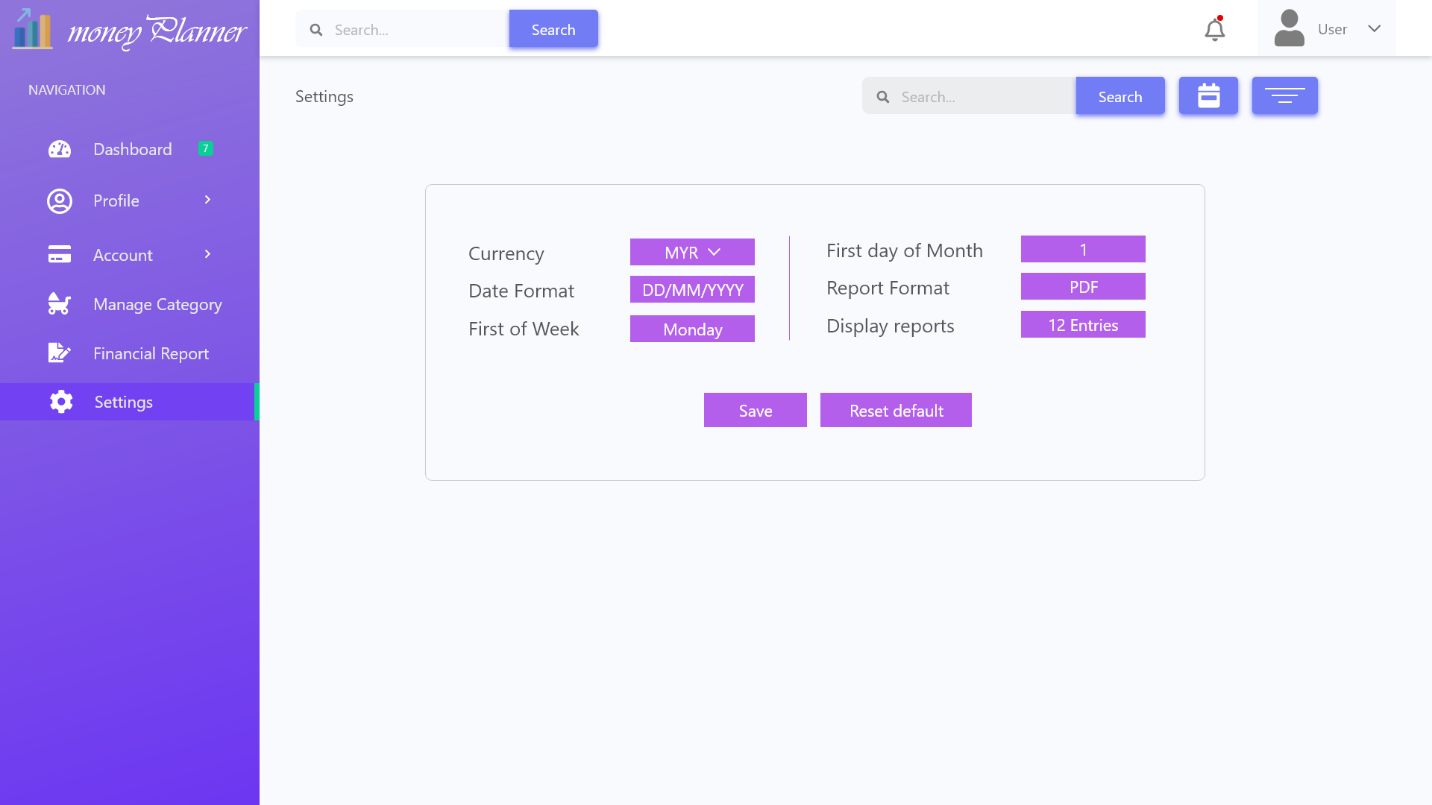
**Figure 3.1.1.7** Financial Report Interface

**3.1.1.8 Configuration Interface**

Font type: Segoe UI

Size : 50, 20, 22, 26 px

Giving the settings consist of currency, date format, report and etc.



Clicking this option will reset every detail in default. The planner will stay on this page and not redirected to dashboard page automatically as to giving the planner options to do any changes in their second thought.

Clicking this option will save all the changes but the planner will stay on this page and not redirected to dashboard page automatically as to giving the planner options to do any changes in their second thought.

**Figure 3.1.1.8** Configuration Interface

**3.1.2 Hardware Interfaces**

These are the recommended requirement in terms of viewing the Money planner apps. It is greatly suggest for the viewing to be in the computer (PC).

|  |  |  |
| --- | --- | --- |
| Hardware Components | Minimum Requirements | Recommended Requirements |
| Processor | 1.0 GHz | 1.4 GHz |
| Random Access Memory (RAM) | 1 GB | 4 GB |
| Hard Drive Storage Capacity | 10 GB | 15 GB |
| Screen Size | 14” | 15.4” and above |
| Resolution | 1024 x 768` | 1280 x 1024 |

**Table 3.1.2.1** Requirement for PC

For using the mobile, this is the recommended requirement:

|  |  |
| --- | --- |
| Hardware Component | Minimum Requirement |
| Screen size (inches) | 4.5” |
| Resolution | 720 x 1280 |
| RAM | 2 GB |
| Processor | 1.4 GHz |

**Table 3.1.2.2** Requirement for Mobile Smartphone

**3.1.3 Software Interfaces**

In order to use the Money Planner apps in a perfect operating environment it is necessarily to use it in this specified browser:

* Google Chrome 28.0 and above
* Mozilla Firefox 52.00 and above
* iOS Safari 9.1.3 (Yosemite OS X)

**3.1.4 Communications Interfaces**

The apps itself required the use of Internet, thus any connections to Internet via Wi-Fi, LAN or mobile data are highly required. The apps will use HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.

**3.2 Functional Requirement**

**3.2.1 Register**

* Planner needs to register before they can use the apps.
* Any information detail such as their full name, gender, age, e-mail and phone number are highly required.
* They need to input their username and password to be used as a login detail
* After successful registration, they need to confirm the email that being used as a security confirmation.

**3.2.2 Login**

* By searching the website of <https://www.moneyplanner.com.my> , they can find the login page for the apps.
* Planner need to use the previously set username and password to get into their account.

**3.2.3 Create Profile**

* Planner needs to create their profile such as name, picture and other important information.
* In this section, they can put all the data they want, as required in the profile.

**3.2.4 Edit Profile**

* It is only for the purpose to edit any part of the profile that already exist.
* Any data that already exist, can be change by the planner.

**3.2.5 Update Profile**

* It only lets the planner to input any data that does not in the profile earlier. As example if the data address was not created, then the user (planner) need to use the update features to bring a new type of information into the apps

**3.2.6 Set Initial Balance**

* Planner needs to input their current amount of money as a deposit (a terms that refer to the beginning of their account).
* It data then will be used and will be subtracted automatically with any input of expenses.

**3.2.7 Add Expense Category**

* Planner needs to add their expenses based on category provided. There are variety of category from the bills, clothes entertainment and others.
* If the expenses that they try to key-in does not have its own category, then they can add a new category.

**3.2.8 Edit Expense Category**

* An option if the planner wants to edit the category that already exist or the category that was created by them.

**3.2.9 Delete Expense Category**

* An option to delete any expense category that already exist or the category that was created by them.

**3.2.10 Add Expense**

* To add the amount of the expenses of the earlier expense category.
* The amount then will be subtracted from the balance automatically.
* The data will be saved inside the history of expenses.

**3.2.11 Edit Expenses**

* To edit the value of the expenses, if there was any mistake in inputting the value earlier.

**3.2.12 Delete Expense**

* Giving the option to delete the expense, probably whether because they are not needed or a redundant data.

**3.2.13 Add Expense Note**

* A feature that for an easy information regarding the expenses.
* Letting the user to write a note about the expenses as a future reference for them.

**3.2.14 Edit Expense Note**

* To edit the note of the expenses, if there was any mistake in making the notes earlier.

**3.2.15 Delete Expense Note**

* Giving the option to delete the expense note, probably whether because they are not needed or a redundant data.

**3.2.16 Print Financial Report**

* Giving the options for the user to print the financial report
* As an ease of access for the user who wants to keep track of their records in physical format.
* All the data regarding the balance, expenditure, income and all process will be included in the report.

**3.2.17 Track All Records**

* Track all records involving the financial of the user.
* For the purpose of recording the history of transactions from category, the adding of income, dates, and amount of expenditure.

**3.2.18 Calculate All Transactions**

* Sums all the transactions between the category (eg: from “Bills” to “Food”) automatically to ease the user.
* Keeping track of the transactions in a big amount.

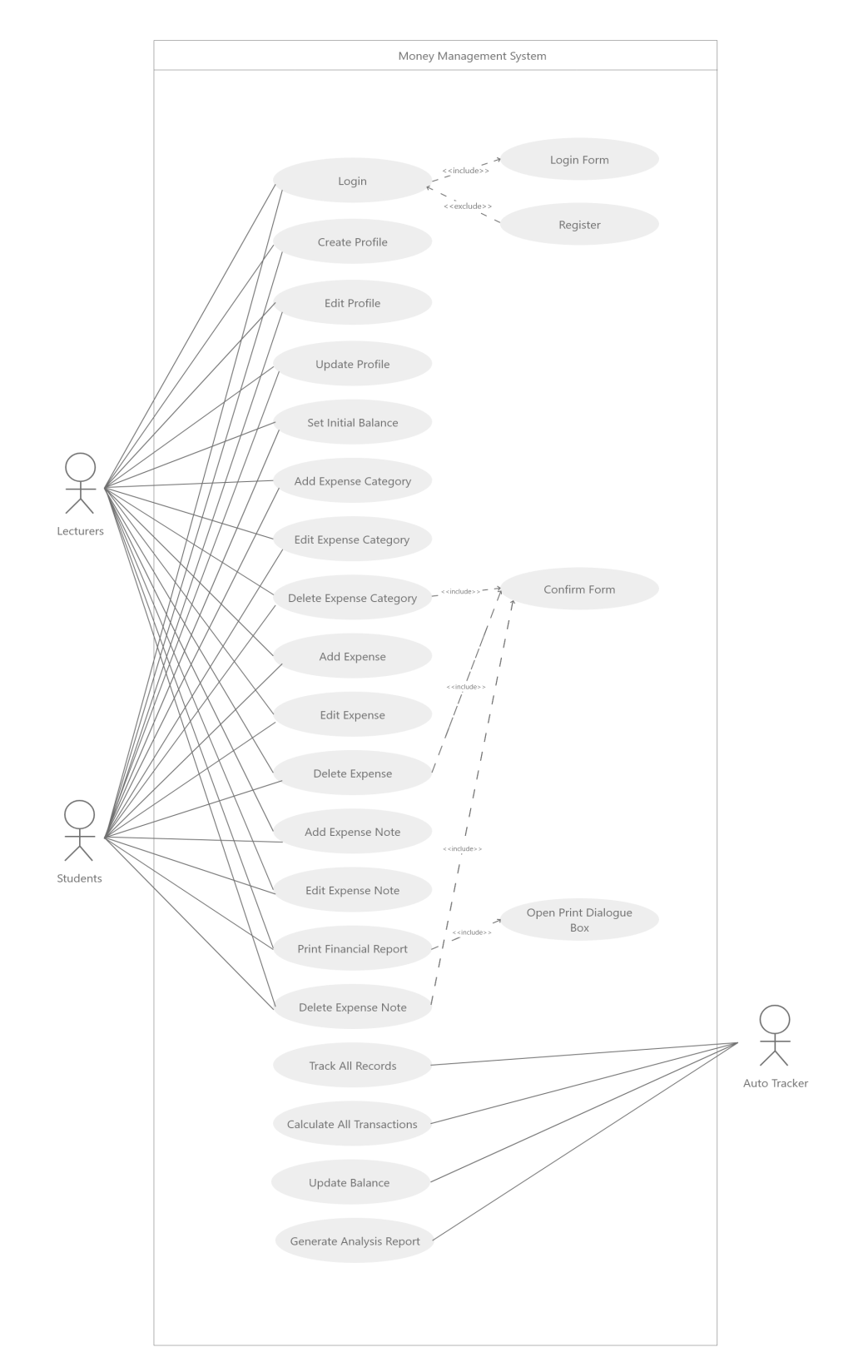
**3.2.19 Update balance**

* Giving the planner an option of updating their balance as probably there will be some income in their financial (allowances, salary).

**3.2.20 Generate Analysis Report**

* Generate an analysis about the use of their money and how they spend it in every month.
* Shows the efficient way of managing their money and set the goals for the user to avoid their limits of expenditures.

## 3.3 Use Case



Use Case of Money Planner Usage

## 3.4 Quality Requirements

In our application, there is reliability measures the probability that a money planner are able to run 24/7 with maximum performance. Users can system or part of a system will work. The system needs to be able to do what is expected or has been promised. Besides, to upload their management documents and access the system at any time system. This application also provide less time when uploading documents and this software also used for users who wants to state their money management or to help developers in expenses. Other users who seek for money management may also use this system.

Other than that, performance. It affect an application’s response behaviour and use of resources. Performance efficiency affects customer satisfaction, workforce productivity, application scalability, response-time degradation, and inefficient use of processing or storage resources. The Performance efficiency of an application lies in each individual component‘s performance, as well as in the effect of each component on the behaviour of the chain of components comprising a transaction in which it participates.

Next, the most impotant part of software quality is security. Security is concerned with how to develop and maintain systems that can resist malicious attacks intended to damage a computer-based system or its data.This system protects user’s data from plagiarism by only allowing the application with proper documentation such as software documentation and patents.

More than that, portibility. Application portability is the ability of an application to be portably installed, deployed, accessed and managed. The term defines an application's flexibility when used on multiple platforms or instantly accessed from the Internet, a desktop or network.For our project, we have selected web based application. Web application is in the top while thinking about portability. It needs less dependency comparing to desktop or mobile application. What we need just web browser. Clients can access our application from a web browser. They do not need to think about the requirements of our application. One thing can be consider, that is the web browser. It is better if client uses latest web browser. Because latest web browser has highest support for we contents to be displayed.

Lastly, quality in maintainability. Maintainability help to encountered, adapted if its enviroment. It might ease with which the program can be corrected if there any errors, changes or enhanced if the user desires a change in requirement.

**3.5 Logical Database Requirements**

|  |  |
| --- | --- |
| **Field** | **Description** |
| Planner\_Name | String data type, the length size must be at least 8 characters and above. |
| Planner\_Email | String data type, the length size must be at least 8 characters and above. |
| Planner\_PhoneNo | Integer data type, the length size must be at least 10 characters and above. |
| Planner\_Username | String data type, the length size must be at least 1 characters. |
| Planner\_Password | String data type, the length size must be a least 8 characters and above. |
| Planner\_Balance | String data type, the length size must be at least 2 characters and above. |
| Planner\_Income | String data type, the length size must be at least 2 characters and above. |
| Planner\_Expenses | String data type, the length size must be at least 2 characters and above. |

Table 3.5.1: Field Description

|  |
| --- |
| **Registration** |
| Planner\_Name  Planner\_Gender  Planner\_Email  Planner\_PhoneNo  Planner\_SecurityQuestion  Planner\_Username  Planner\_Password |

|  |
| --- |
| **Registered Planner Profile** |
| Planner\_Username  Planner\_Password |

Manage Account

|  |
| --- |
| **Accounts** |
| Planner\_Balance  Planner\_Income  Planner\_Expenses |

View

|  |
| --- |
| **Status of Financial** |
| Planner\_FinancialReports  Planner\_AnalysisReports |

Figure 3.5.2: Logical Database of “Money Planner”

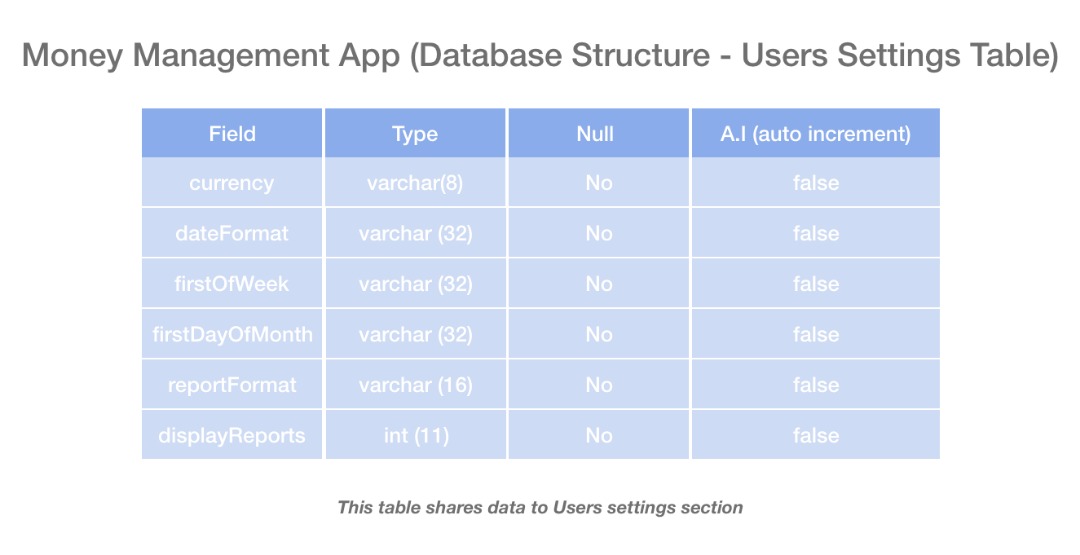


Figure 3.6.2: Database Structure for Users Setting

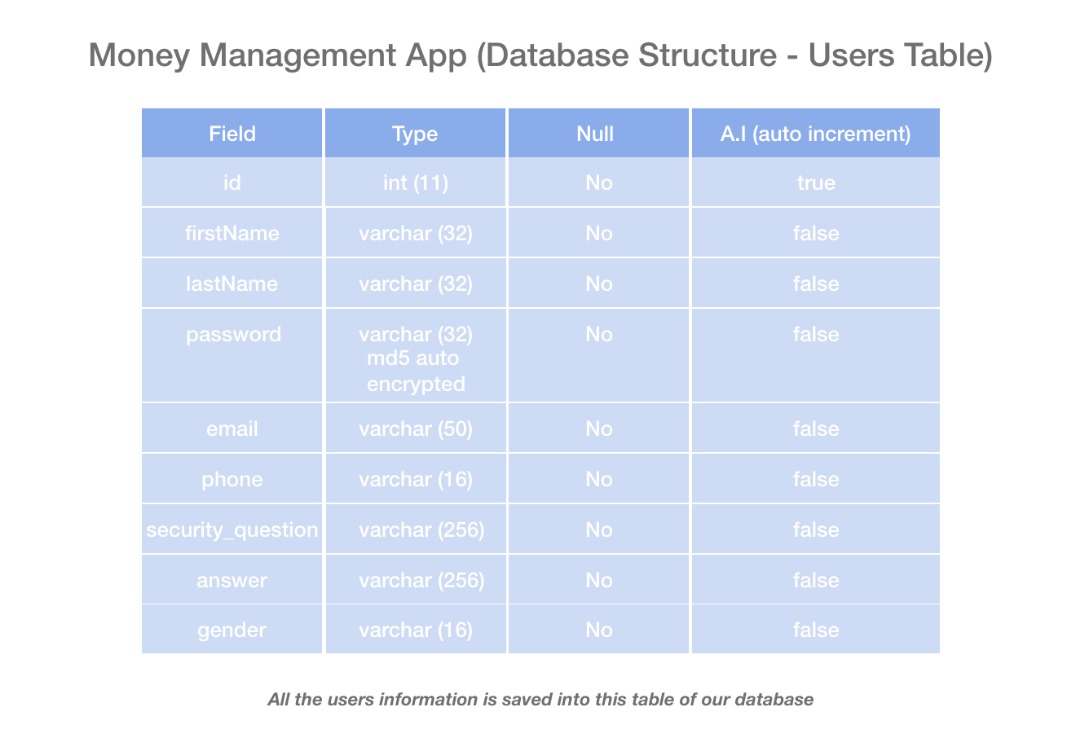


Figure 3.6.3: Database Structure for Users

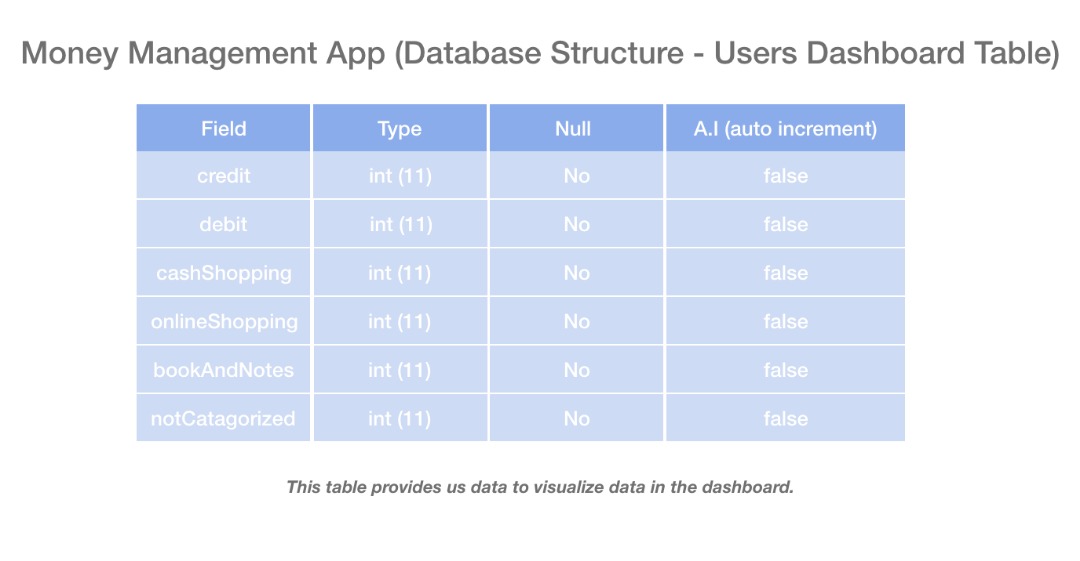


Figure 3.6.4:: Database Structure for Dashboard Users

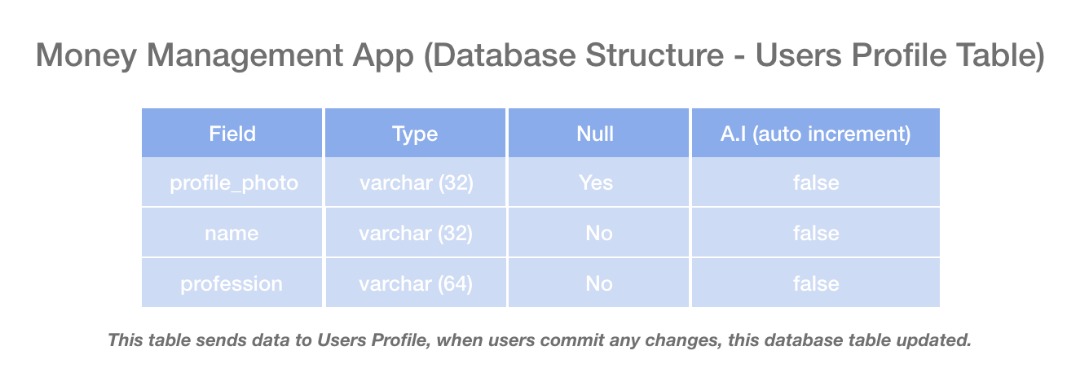
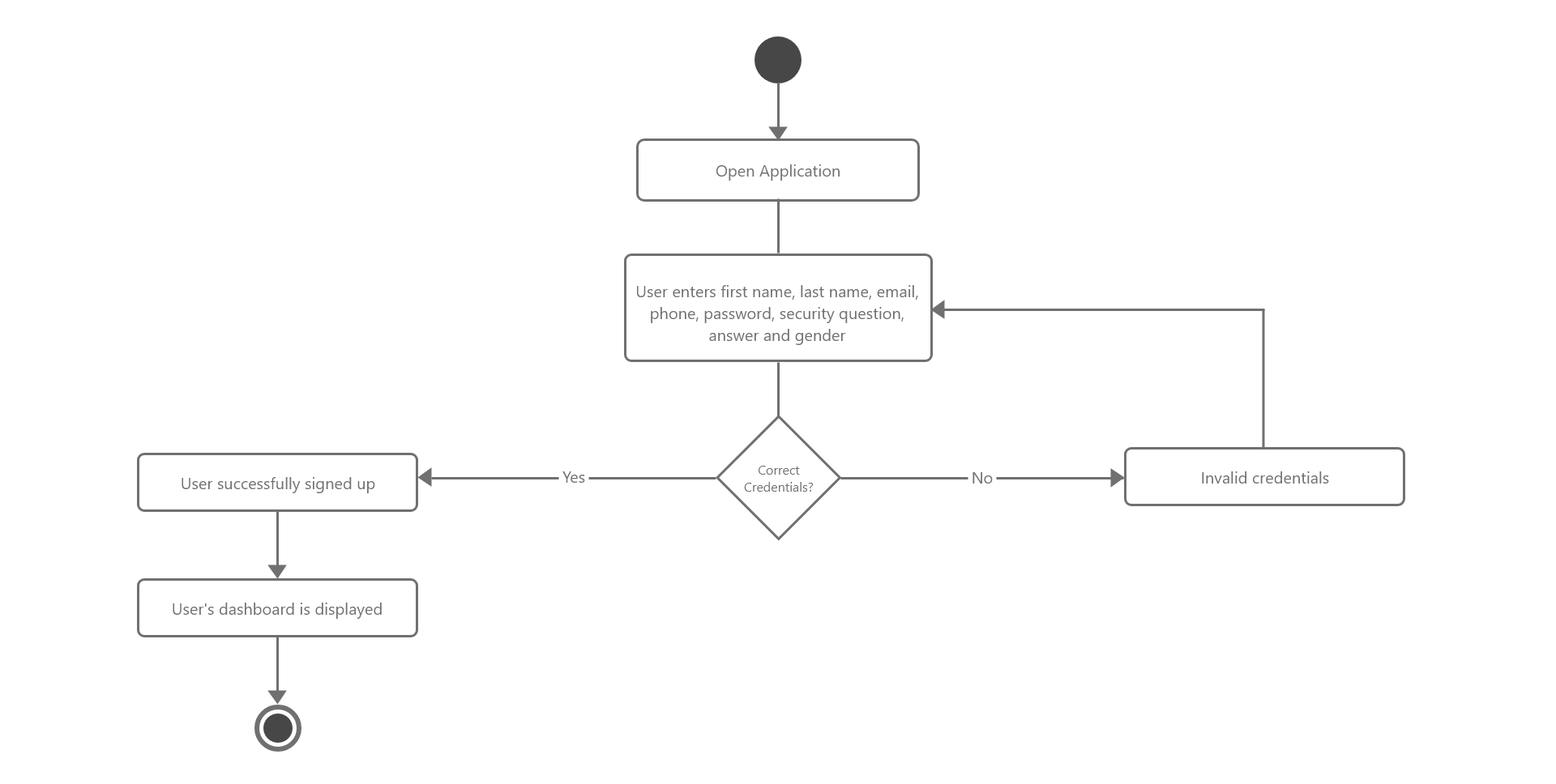


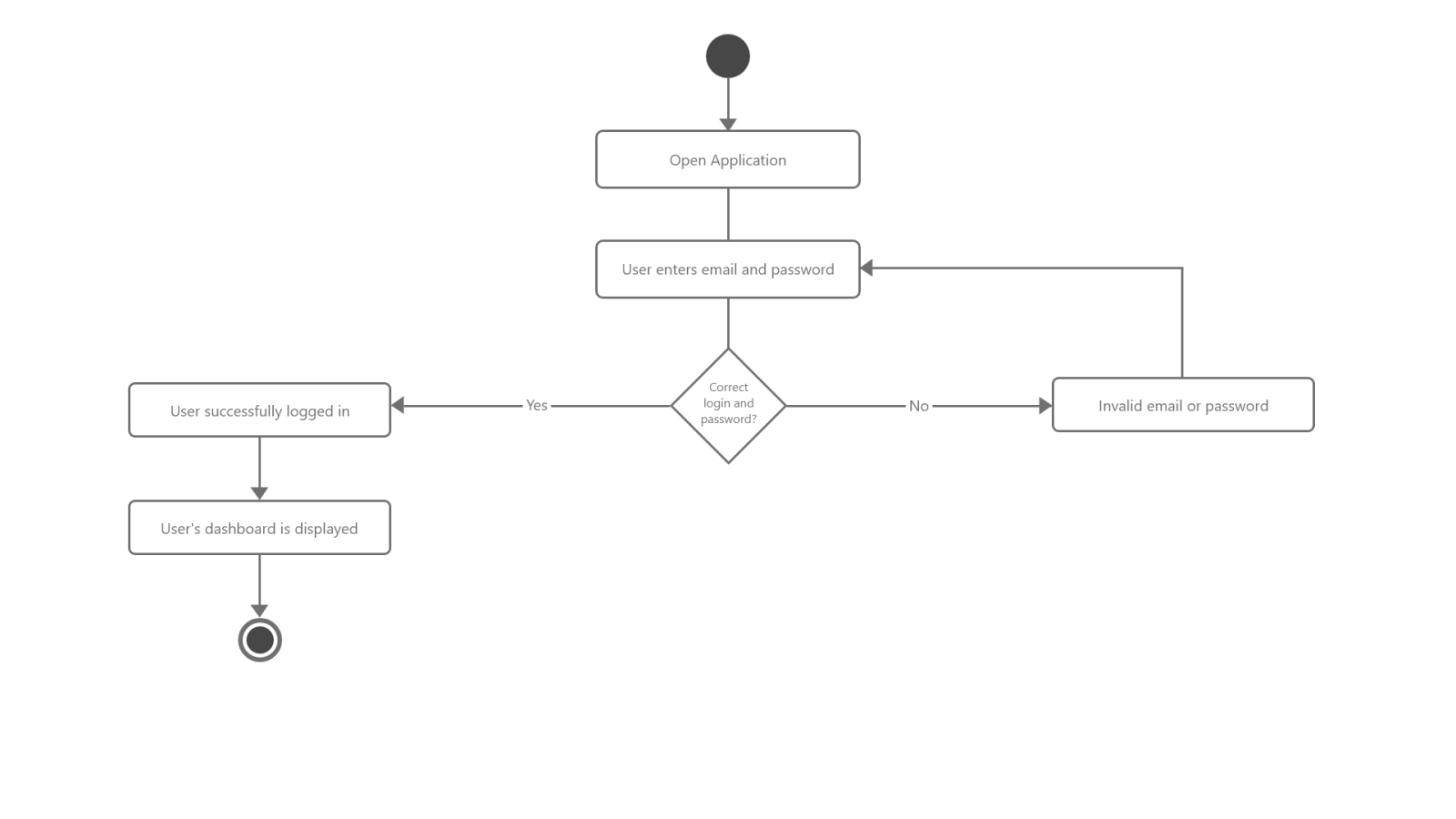
Figure 3.6.5: Database Structure for Users Profile

**4.0 Activity Diagram**

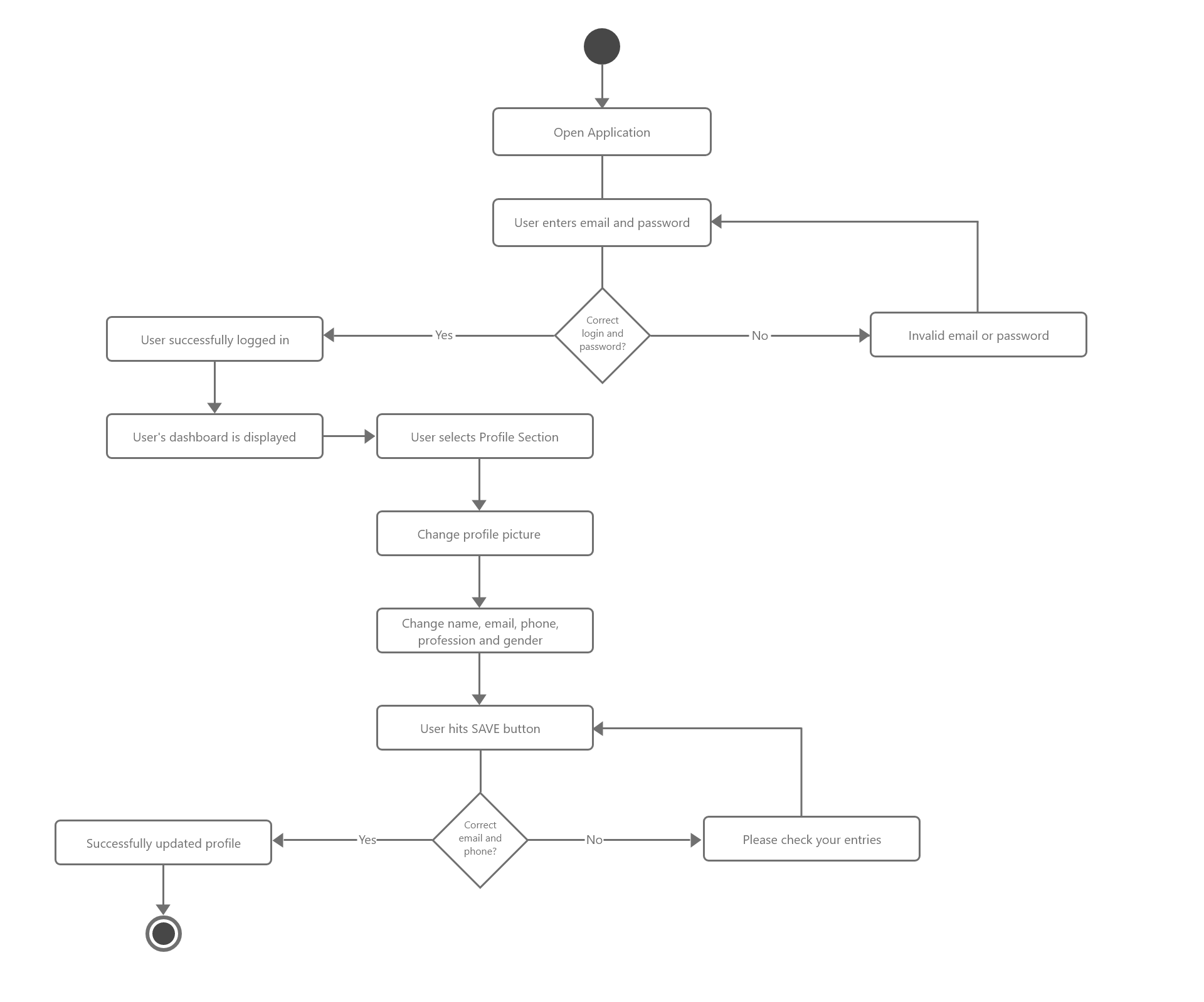
1 (User Registration)



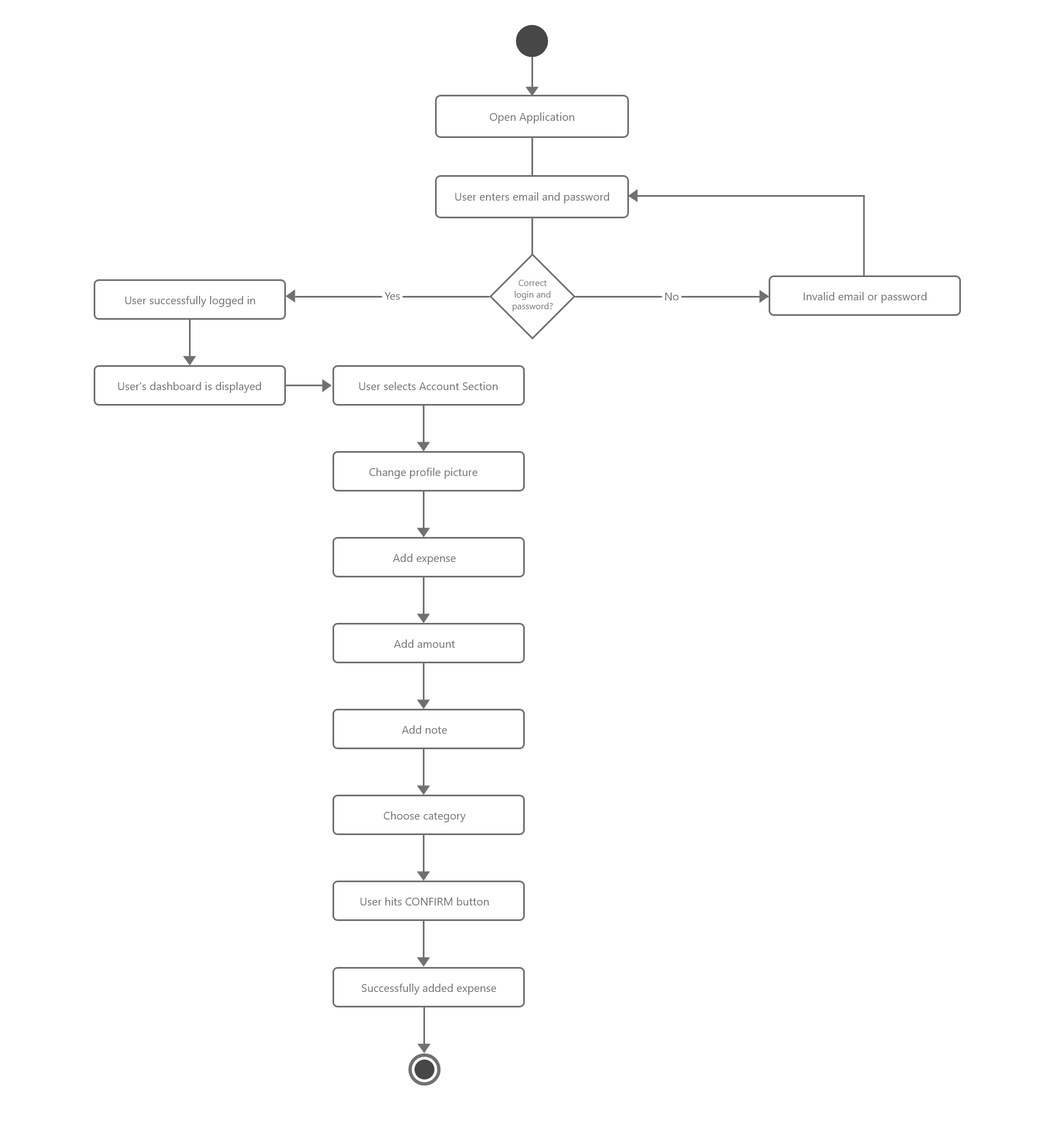
# 2 (User Sign In)



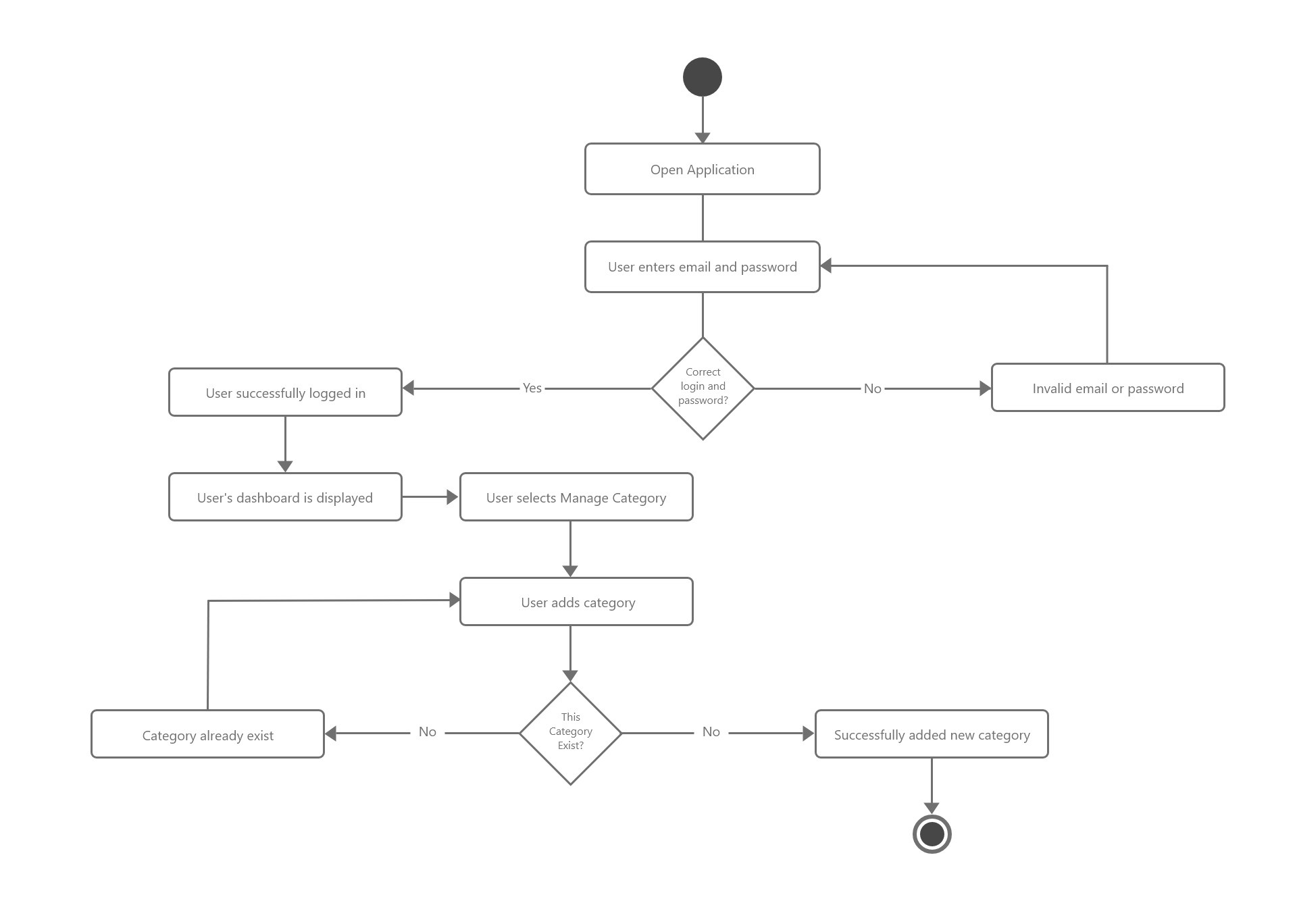
#3 (Profile Activity)



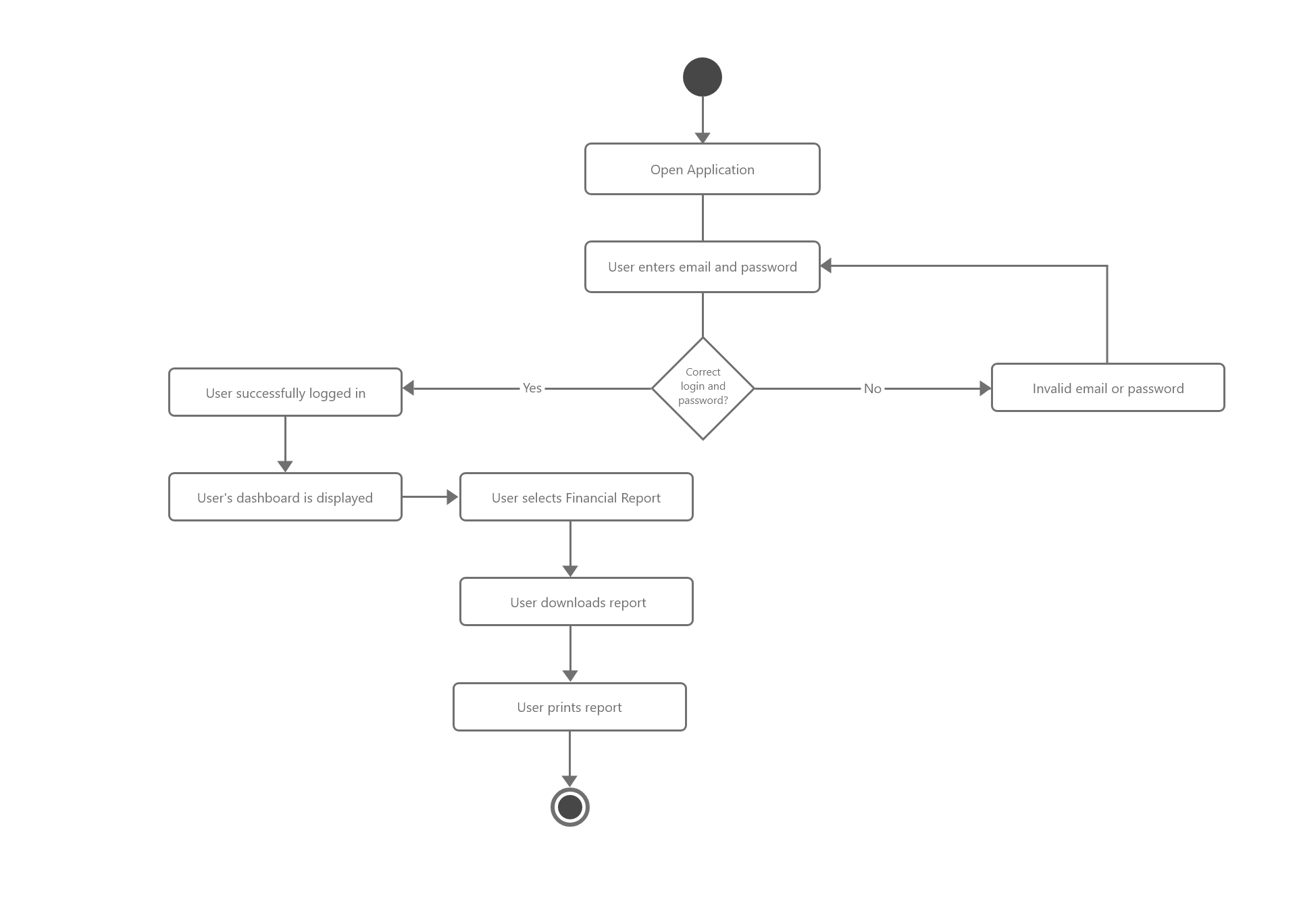
#4 (Account Activity)



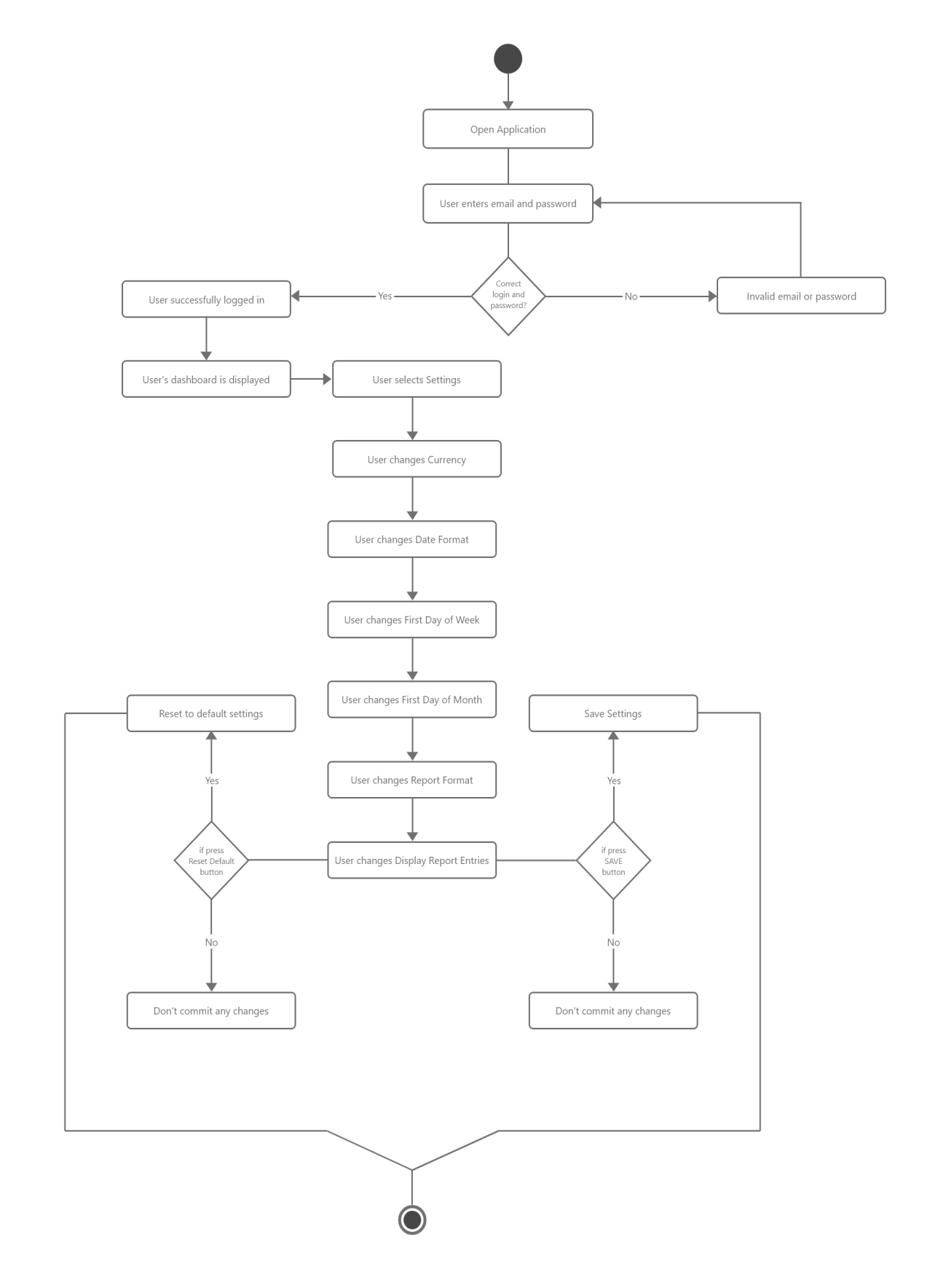
#4 (Manage Category Activity)



#4 (Financial Report Activity)



#5 (Settings Activity)



Backup of Our Project

We have maintained version control while dealing with our project. Since we can not believe our computer’s hard disk drive, anytime it can crush and our project will be lost. We have used GitHub.



Link of our GitHub repository: https://github.com/Shimol-Khan/SRSMoneyPlanner