# Chapter 2: Analysis

## 

## 2.1 Introduction to Analysis:

Analysis is the phase of the software development life cycle (SDLC) which describes the requirements phase of the project and helps to solve the problem arises among clients. Requirements specification is the final product at the end of this stage. By outlining the goals and without considering the implementation of project, requirements gathered with the help of user interaction.

In spite of the fact that requirement gathering is the main aim of this stage, analysis focuses on:

* Information should be gathered and analyzed for feasibility purposes.
* Requirements should be defined and enhance the aims of system
* Requirements should be prioritized and define the alternatives for requirements.
* System Requirement Specification (SRS) should prepared that helps in specification of hardware, software, functional and non-functional requirements of project.
* Conceptual System Design should prepared that helps to give direction for the Management Information System.

## 2.2 Analysis Methodology:

The development team followed a series of structure to develop a system called System Development Life Cycle (SDLC). SDLC consists stages like planning, analysis, design, implementation and maintenance of the project. Analysis and Design are the core part of the SDLC. Like SDLC analysis phase also have own techniques and methodologies that helps us to gather user requirements and functionalities of the system that is being developed.

For this project I have chosen **Soft Approach** as an analysis methodology among different other analysis methodology like **People-Oriented, Process-Oriented, Object-Oriented, Hard Approach, Combined Approach** etc.

**Soft Approach:**

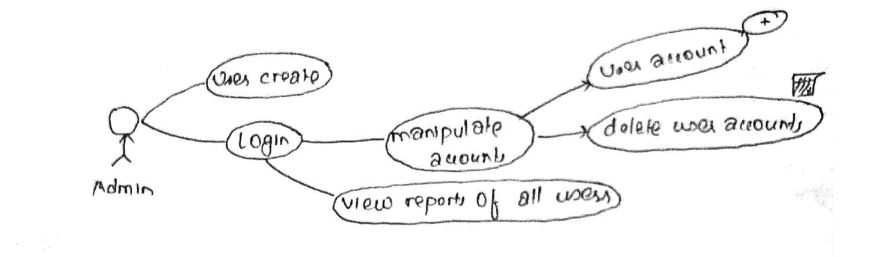
Soft Approach for system analysis consists of both technical and human aspects to develop a complete system. This approach based on the people-focused analysis and user-interaction acts as an important technique. The consideration of human, social activity and technical functional requirements belongs into this approach.

Following are the number of steps undertaken while using this methodology:

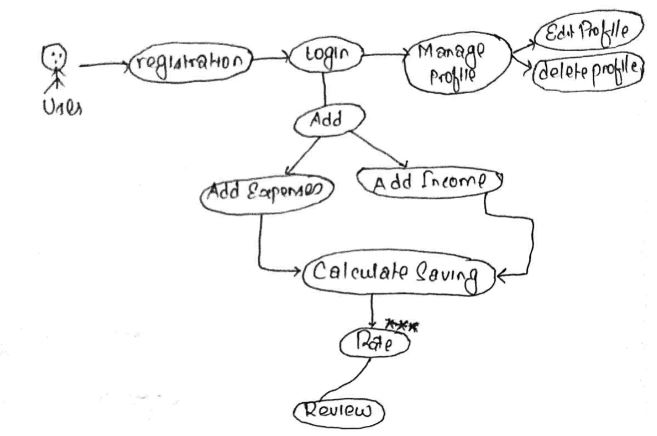
1. **Human activity analysis of soft approach involves rich picture, root definition and conceptual model**

**Rich picture:**

Rich picture is a collection of pictures, symbols, text and many other things that helps to show the relationships, connections and problem arises between these things.



Screenshot 1: Rich picture of Admin



Screenshot 2: Rich picture of User

**Root definition:**

The processes and problem of the system clarify with the help of root definition. There are two types of root-definition i.e.:

Primary task root-definition: focuses on system processes

Issued-based root definition: focuses on problem statement

**WHAT, HOW AND WHY** are the three elements of root definition that defines:

WHAT: defines the aim of the system

HOW: reason for achieving the aim

WHY: longer term aim

**CATWOE** analysis helps to define a root definition.

**CATWOE**

**CATWOE** stands for

**C Customers/Clients**

**A Actors / Agents**

**T Transformations**

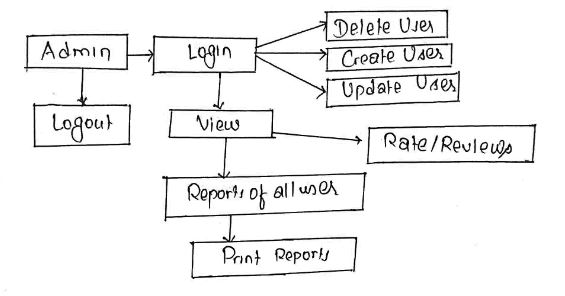
**W World View**

**O Owners**

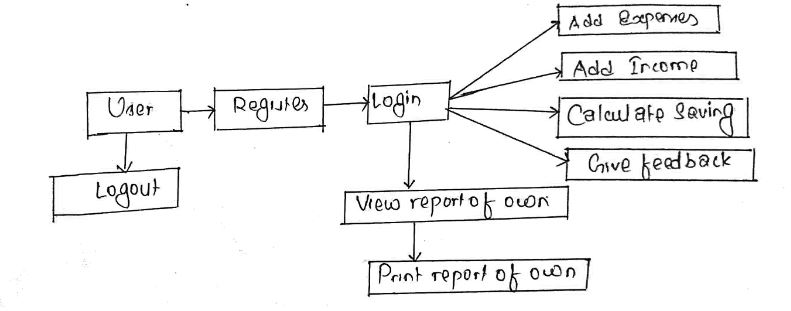
**E Environment**

**Customers** are the important things to run the business. Customers also known as users they accept services through using system**. Actors** must be defined employees, suppliers, agencies and officials. **Transformation** is changes occurred by the system for the easiness of user. **World View** covers the issues arises in the system. Issues may be short-term or long-term. **Owner** is the person who owns the problem they must be the parts of its solution or both. Available resources, budget, laws and regulations comes under **Environment**.

**Conceptual Diagram**

The conceptual diagram of expenses management system represented as follows:

Screenshot 3: Conceptual model of Admin



Screenshot 4: Conceptual diagram of User

1. **Social-technical aspects of analysis and design**

Social and technical aspects of analysis and design of Expenses Management System is cover by **Feasibility Study,** which is describe in another portion**.**

1. **Human-Computer Interface design**

To remove duplication of data and diagram human-computer interface design of Expenses Management System is used.

1. **Design of technical aspects**

Technical aspects of Expenses Management System cover System Requirement Specification (SRS). System Requirement Specification (SRS) portion of the project described in another part.

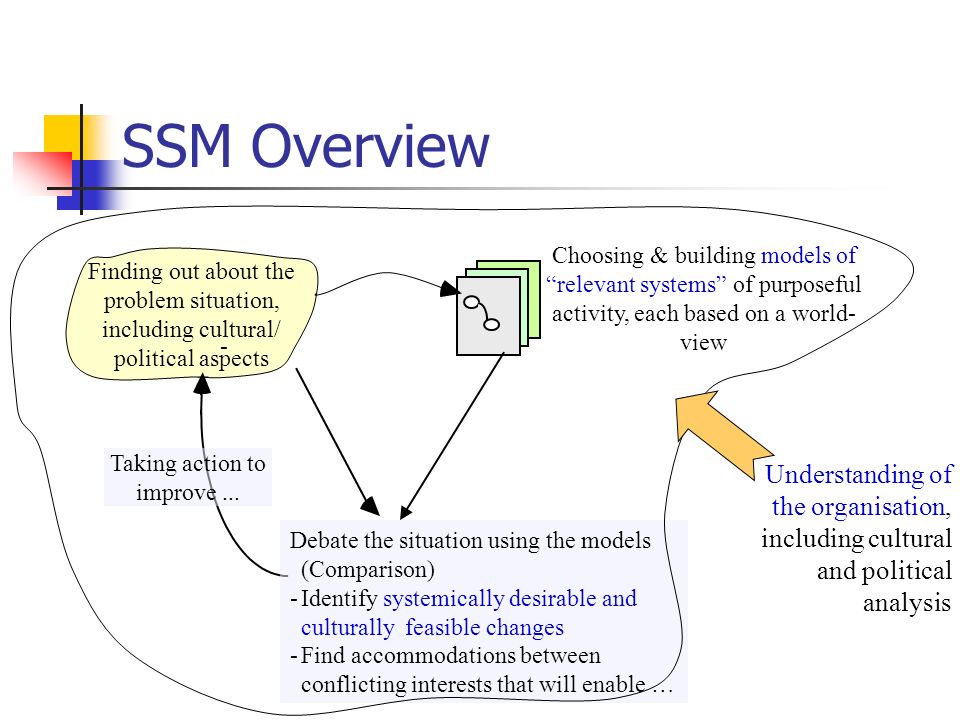
1. **Cultural aspects**

Cultural aspects represents the cultural beliefs and practices of the society. Beliefs of the society do not effect with the use of this application.

1. **Political aspects**

The application that we will develop does not effects the politics policies of the country. Before to develop this application we have to know the political policies of the country.

**Overview of the Soft System Methodology:**



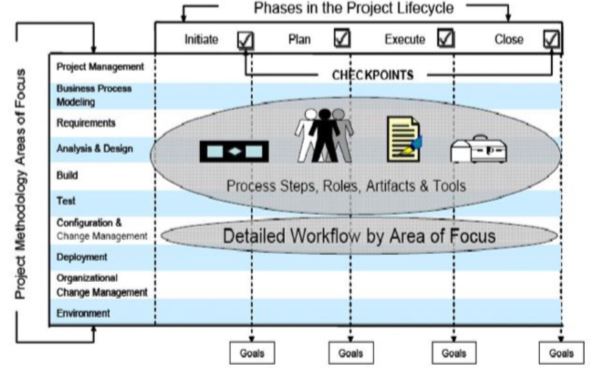
Screenshot 5: Overview of Soft System Methodology

Above screenshot represents the overview of the soft system methodology. The problem arises due to cultural and social aspects is manage with the use of this methodology. Human interaction is an important factor to control the problem. In addition, human interaction managed the misunderstanding between developer and users. For sustainable running this application and to build trust between users and developer, human interaction plays important role.

**Advantages of Soft Approach over other Methodology and its areas of focus:**

I have chosen this methodology because this methodology consider both human and technical factors of the system. Following are the reasons to choose soft approach over other approach are:

* This methodology cover the human, technical, cultural and political part of the system analysis.
* User participation involved during the analysis and design phase of the system.
* To get more user-friendly application user’s knowledge and skills taken as an input parameter for analysis proposes.
* This methodology provides cooperation and stakeholder’s participation to develop complete system.

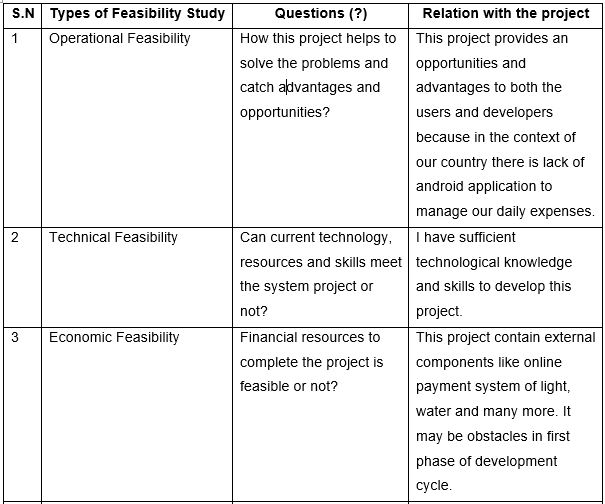
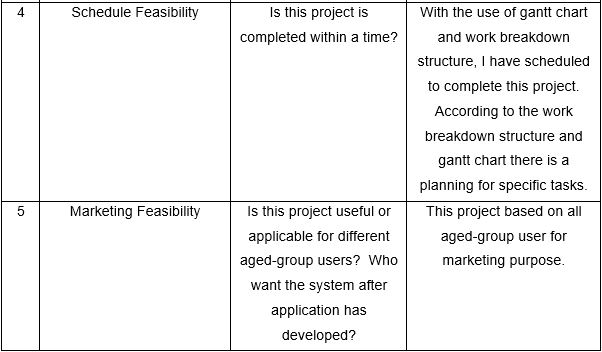


Screenshot 6: Human and technical aspects integration

## 2.3 Feasibility Study

The real world scenario in which a proposed system is financially, technically and operationally practical and workable called feasibility study. Feasibility study comes under social and technical aspects of analysis and design and covers description of product, financial data, legal requirements, tax obligation and many more.

Types of feasibility study and their relation with Expenses Management System are:



# 

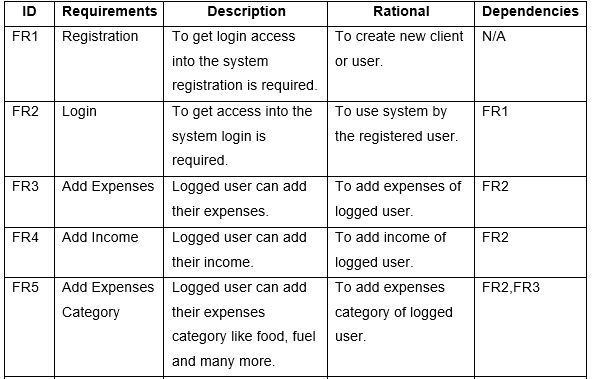
Screenshot 7: Feasibility Study table

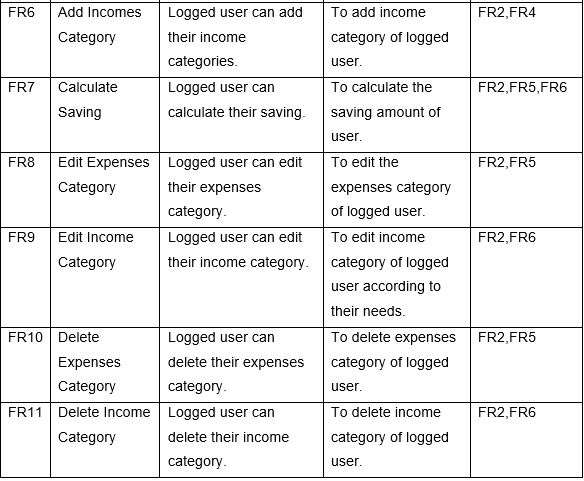
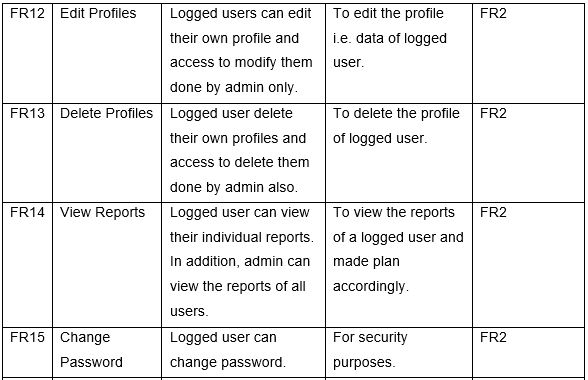
# 2.4 Requirement Analysis

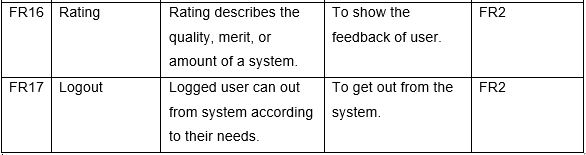
Requirement analysis is the hardest part of building a software system. The goal of requirement analysis is to understand the requirements of user’s and documented them properly. Requirements of users include functional and non-functional requirements that implemented in a system.

# 2.4.1 Functional Requirements

Functional requirements define all the services or functions required by the users they must have provided by system. Functional requirements are easy to test in comparison to non-functional requirements. It describes “what” the software should do. It related to the individual system features. Failure to meet the individual functional requirement may degrade the system.

Functional requirements of Expenses Management System described as follows:



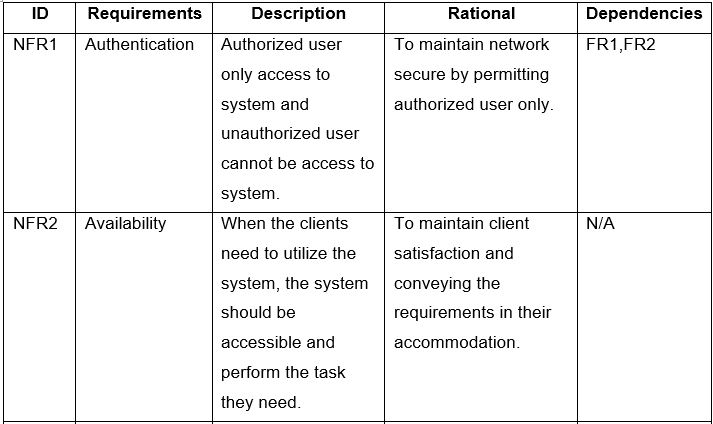


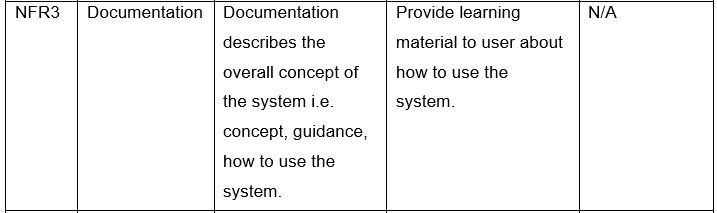
Screenshot 8: Functional requirements table

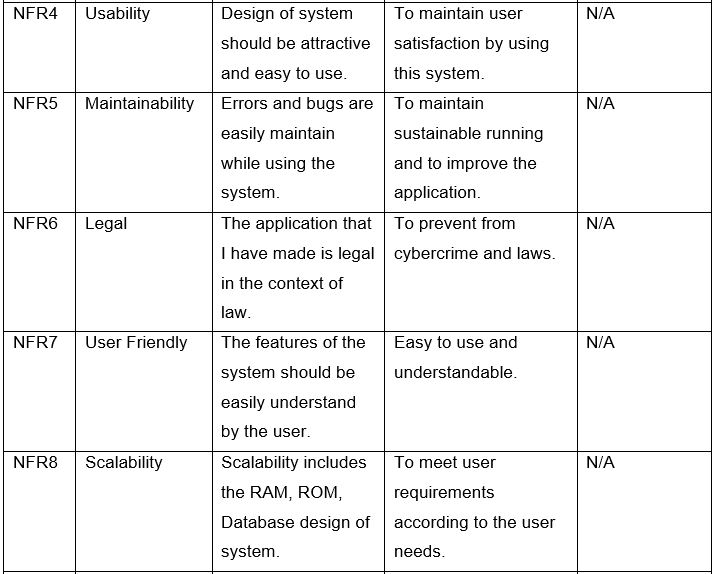
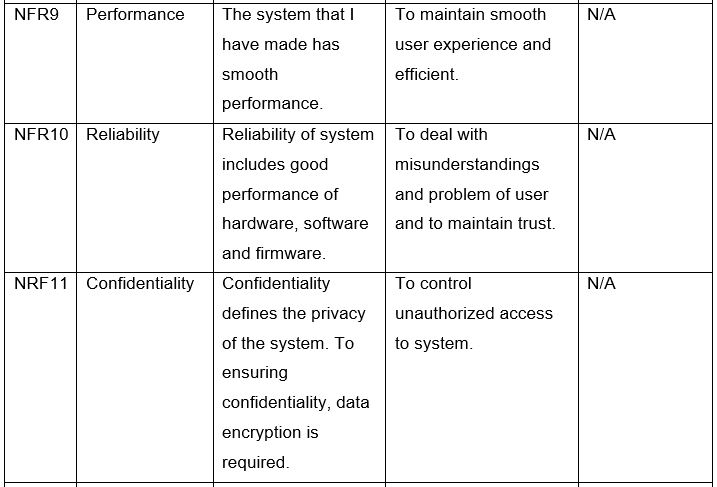
## 2.4.2 Non-functional Requirements

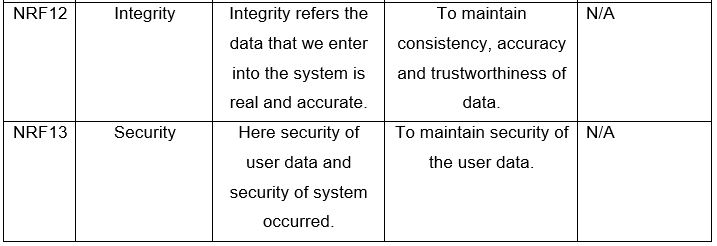
Non-functional requirements define the system properties and constraints. Non-functional requirements are difficult to test in comparison to functional requirements. . It describes “how” the software will do it. It related to system as a whole. Failure to meet non-functional requirements may make the system unusable.

Non-Functional requirements of Expenses Management System described as follows:





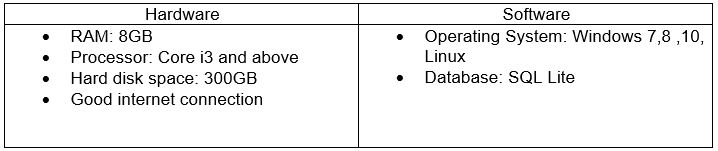




Screenshot 9: Non-functional Requirements Table

# 2.4.3 Hardware/Software Specification

The application that I have been developing requires following hardware/software specification:



## 2.4.4 MoSCoW Prioritization

I have chosen MoSCoW requirement prioritization to prioritize the functional and non-functional requirements of Expenses Management System among different requirement prioritization like grouping assignment, ranking, bubble sort technique, Analytic Hierarchy Process, hundred dollar method.

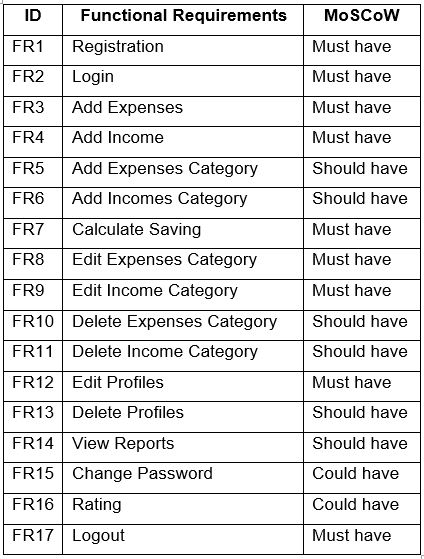
Due to the time and budget stakeholders does not meet their requirements. Prioritization of requirement occurs to satisfy stakeholders. MoSCoW stands for Most have (M), Should have (S), Could have (C) and Would have (W).

**Must have (M)** - Vital requirements for the project i.e. system cannot perform without those requirements.

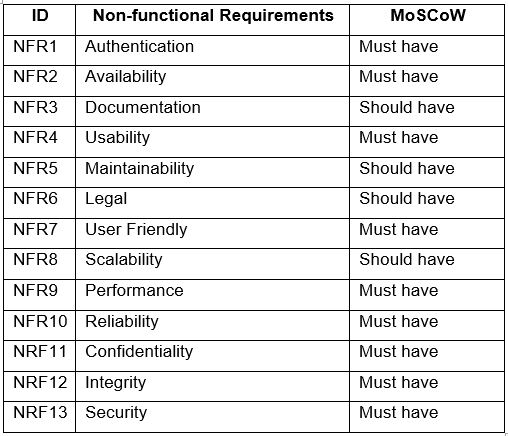
**Should have (S)** - Requirements considered as important but not vital. If we have remaining time it can added in future.

**Could have (C)** – Requirements that considered as desirable but not necessary

**Would have (W)** - A requirement that cannot be implemented in current version due to the lack of time and resources, but may be considered in future.



Screenshot 10: MoSCoW prioritization of functional requirements



Screenshot 11: MoSCoW prioritization of Non- functional requirements

## 2.5 Use-Case Diagram

In UML, a use-case diagram is also known as dynamic or behavioral diagram. A use-case diagram shows the set of use-cases, actors (special kind of class) and their relationship in graphical way. To manage the system requirements i.e. identify, simplify and organize use-case diagram plays important role during analysis.

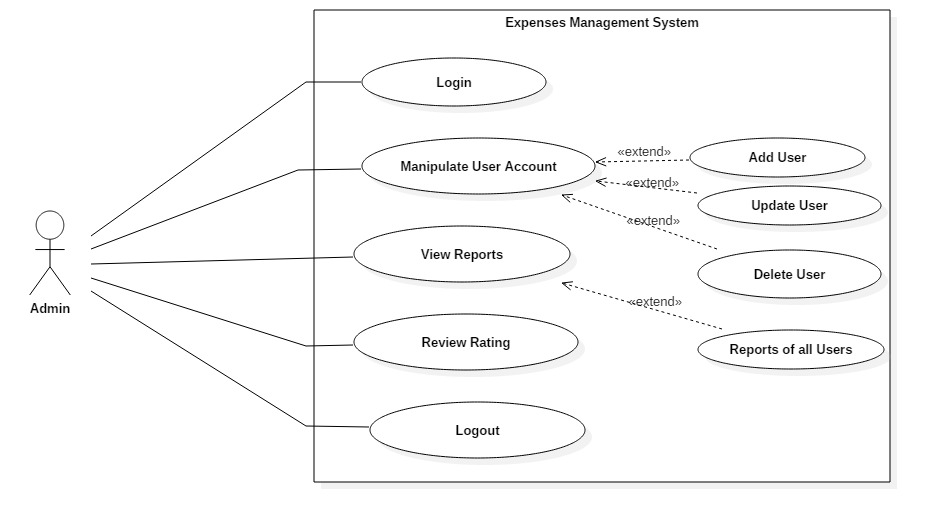
In object modeling and real-world system it has own standard notations. They are boundary, actors, use-cases and relationships.

**Advantages of Use-Case Diagram:**

* Functional requirements of the project gather in use case diagram.
* Use-case diagram gather additional behavior of system and improve in system robustness.
* It describes that how the actor connect with system.
* It can provide the summary of the whole system.
* During requirement gathering this diagram is good choice for system analyst because it is easy to draw and easily understandable.

**Disadvantages of Use-Case Diagram:**

* Use-case diagram cannot gather the non-functional requirements of the product.
* Some products and system have complexity to write and understand the diagram for both developers and users.
* It only summarized the relationship between cases, systems and actors.



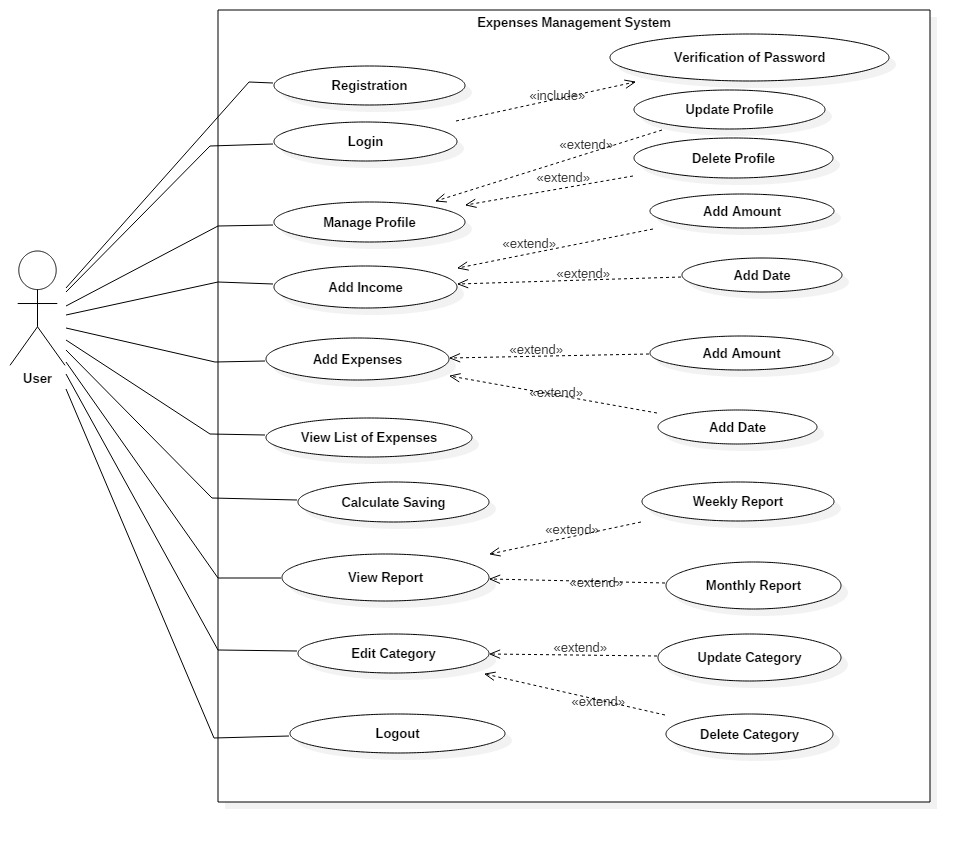
Screenshot 12: Admin Usecase diagram

**Scenario Description**

Actor: Actor represents Admin

Admin can access to overall performance of the system. According to the diagram, Admin can perform following things:

* Admin can access to login into system of all user.
* Admin can access to manipulate user account. Manipulation of user account includes Add User, Update User, and Delete User etc.
* Admin can access to view reports, review and rating of all user.
* Admin can logout from system.



Screenshot 13: User usecase diagram

**Scenario Description**

Actor: Actor represents User.

User can access to following things of system:

* User can register to login into the system.
* User can login to use system.
* User can manipulate their own profile. Manipulation of the profile include Update Profile and Delete Profile.
* User can add their income and expenses.
* User can view their list of expenses.
* User can calculate their saving with the use of expenses and income.
* User can view their own report.
* User can edit their own category like food, clothes, hospital etc.
* User can logout from system.

# 2.6 Natural Language Analysis

Expenses Management System is an android application that helps to manage your expenses and calculate saving of income. Reducing paper work, saving time and money is the main aim of this project. It also helps to reduce human errors and provide satisfaction to the user.

This project consists of two actors i.e. Admin and User. The database of this project stores user’s data. User data includes Id, First name, Last name, Email, Password, expenses of user, category of expenses, income of user, category of income and many more. System also includes review and rating of user towards application. Future plans of application based on review and rating from the users.

The system provide many features to the user with proper access control like Login, Registration, add date, expenses and income. User can able to enter into the system after registration and access to manipulate their own account. Manipulation of account include add, edit, update and delete their own personnel profile and data.

Natural Language Analysis is a process to separate noun, adjective and verb from the scenario.

* Noun represents candidate class
* Adjective represents candidate attributes
* Verb represents the operations or methods

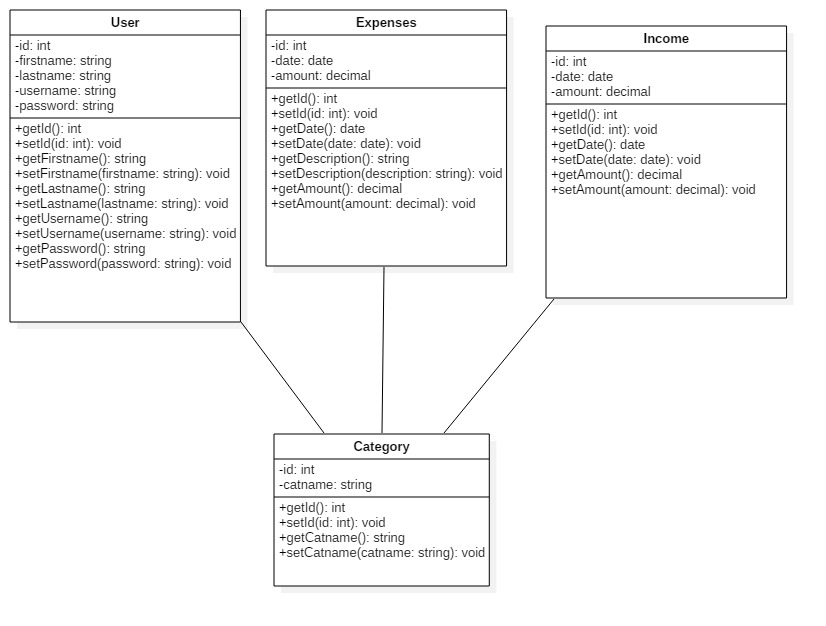
Following are the Candidate class, attributes and methods from the Natural Language Analysis:

|  |  |  |
| --- | --- | --- |
| **Noun (Candidate Class)** | **Adjective (Candidate Attributes)** | **Verbs (Candidate Methods)** |
| User, Category, Expenses , Income | First name, Last name, Email, Password | Add, Edit, Update, Delete, Review, Rate |

# 2.7 Initial Class Diagram

The popular UML diagrams and represented by the blue print of a system called class diagram. It shows the relationship and graphical representation between class, object, attributes and methods of a system. Class diagram is directly mapping with object-oriented languages thus it can used widely due to the phase of construction. It also helps in constructing executable code for application.

To draw class diagrams we have used many rectangular boxes with three sections: 1st section of the rectangular box represents the class name of the system, 2nd section represents the attributes of the class and finally third section represents the methods of the class. Association, Aggression and Composition are the components to show the relationship between different classes.



Screenshot 14: Initial Class Diagram