# 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.

**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 Wide**

**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 Wide** 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:

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 is describe in another part.

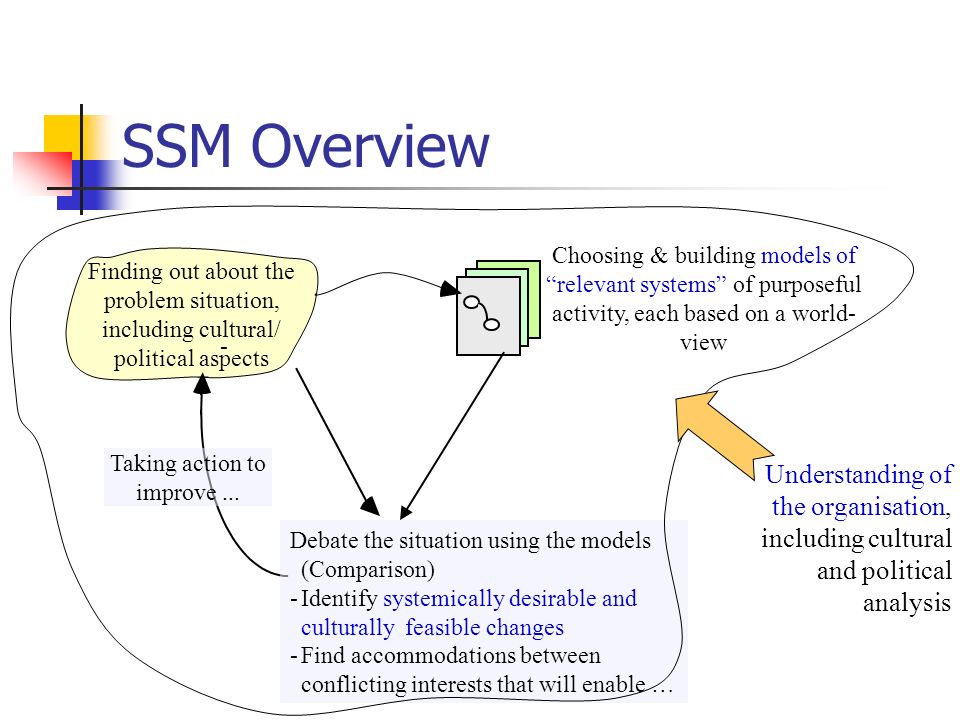
1. **Cultural aspects**

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:**

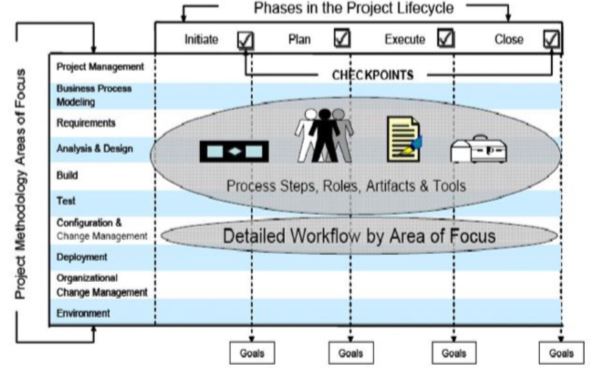


Above screenshot represents the overview of the soft system methodology. The problem arises due to cultural and social aspects is manage with the use of this methodology. Human interaction is an important factor to control the problem. In addition, human interaction managed the misunderstanding between 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.



## 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:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.N** | **Types of Feasibility Study** | **Questions (?)** | **Relation with the project** |
| 1 | Operational Feasibility | How this project helps to solve the problems and catch advantages and opportunities? | This project provides an opportunities and advantages to both the users and developers because in the context of our country there is lack of android application to manage our daily expenses. |
| 2 | Technical Feasibility | Can current technology, resources and skills meet the system project or not? | I have sufficient technological knowledge and skills to develop this project. |
| 3 | Economic Feasibility | Financial resources to complete the project is feasible or not? | This project contain external components like online payment system of light, water and many more. It may be obstacles in first phase of development cycle. |
| 4 | Schedule Feasibility | Is this project is completed within a time? | With the use of gantt chart and work breakdown structure, I have scheduled to complete this project. According to the work breakdown structure and gantt chart there is a planning for specific tasks. |
| 5 | Marketing Feasibility | Is this project useful or applicable for different aged-group users? Who want the system after application has developed? | This project based on all aged-group user for marketing purpose. |
| 6 | Cultural Feasibility | Is this project, well acknowledged by the local people? | This is an android-based application helps to calculate daily expenses and increase effectiveness to manage expenses. |

# 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:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Requirements** | **Description** | **Rational** | **Dependencies** |
| FR1 | Registration | To get login access into the system registration is required. | To create new client or user. | N/A |
| FR2 | Login | To get access into the system login is required. | To use system by the registered user. | FR1 |
| FR3 | Add Expenses | Logged user can add their expenses. | To add expenses of logged user. | FR2 |
| FR4 | Add Income | Logged user can add their income. | To add income of logged user. | FR2 |
| FR5 | Add Expenses Category | Logged user can add their expenses category like food, fuel and many more. | To add expenses category of logged user. | FR2,FR3 |
| FR6 | Add Incomes Category | Logged user can add their income categories. | To add income category of logged user. | FR2,FR4 |
| FR7 | Calculate Saving | Logged user can calculate their saving. | To calculate the saving amount of user. | FR2,FR5,FR6 |
| FR8 | Edit Expenses Category | Logged user can edit their expenses category. | To edit the expenses category of logged user. | FR2,FR5 |
| FR9 | Edit Income Category | Logged user can edit their income category. | To edit income category of logged user according to their needs. | FR2,FR6 |
| FR10 | Delete Expenses Category | Logged user can delete their expenses category. | To delete expenses category of logged user. | FR2,FR5 |
| FR11 | Delete Income Category | Logged user can delete their income category. | To delete income category of logged user. | FR2,FR6 |
| FR12 | Edit Profiles | Logged users can edit their own profile and access to modify them done by admin only. | To edit the profile i.e. data of logged user. | FR2 |
| FR13 | Delete Profiles | Logged user delete their own profiles and access to delete them done by admin also. | To delete the profile of logged user. | FR2 |
| FR14 | View Reports | Logged user can view their individual reports. In addition, admin can view the reports of all users. | To view the reports of a logged user and made plan accordingly. | FR2 |
| FR15 | Change Password | Logged user can change password. | For security purposes. | FR2 |
| FR16 | Rating | Rating describes the quality, merit, or amount of a system. | To show the feedback of user. | FR2 |
| FR17 | Logout | Logged user can out from system according to their needs. | To get out from the system. | FR2 |

## 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:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Requirements** | **Description** | **Rational** | **Dependencies** |
| NFR1 | Authentication | Authorized user only access to system and unauthorized user cannot be access to system. | To maintain network secure by permitting authorized user only. | FR1,FR2 |
| NFR2 | Availability | When the clients need to utilize the system, the system should be accessible and perform the task they need. | To maintain client satisfaction and conveying the requirements in their accommodation. | N/A |
| NFR3 | Documentation | Documentation describes the overall concept of the system i.e. concept, guidance, how to use the system. | Provide learning material to user about how to use the system. | N/A |
| NFR4 | Usability | Design of system should be attractive and easy to use. | To maintain user satisfaction by using this system. | N/A |
| NFR5 | Maintainability | Errors and bugs are easily maintain while using the system. | To maintain sustainable running and to improve the application. | N/A |
| NFR6 | Legal | The application that I have made is legal in the context of law. | To prevent from cybercrime and laws. | N/A |
| NFR7 | User Friendly | The features of the system should be easily understand by the user. | Easy to use and understandable. | N/A |
| NFR8 | Scalability | Scalability includes the RAM, ROM, Database design of system. | To meet user requirements according to the user needs. | N/A |
| NFR9 | Performance | The system that I have made has smooth performance. | To maintain smooth user experience and efficient. | N/A |
| NFR10 | Reliability | Reliability of system includes good performance of hardware, software and firmware. | To deal with misunderstandings and problem of user and to maintain trust. | N/A |
| NRF11 | Confidentiality | Confidentiality defines the privacy of the system. To ensuring confidentiality, data encryption is required. | To control unauthorized access to system. | N/A |
| NRF12 | Integrity | Integrity refers the data that we enter into the system is real and accurate. | To maintain consistency, accuracy and trustworthiness of data. | N/A |
| NRF13 | Security | Here security of user data and security of system occurred. | To maintain security of the user data. | N/A |

# 2.4.3 Hardware/Software Specification

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

|  |  |
| --- | --- |
| Hardware | Software |
| * RAM: 8GB * Processor: Core i5 and above * Hard disk space: 300GB * Good internet connection | * Operating System: Windows 7,8 ,10, Linux * Database: SQL Lite |

## 

## 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 will not be implemented now, but may be considered in future.

|  |  |  |
| --- | --- | --- |
| **ID** | **Functional Requirements** | **MoSCoW** |
| FR1 | Registration | Must have |
| FR2 | Login | Must have |
| FR3 | Add Expenses | Must have |
| FR4 | Add Income | Must have |
| FR5 | Add Expenses Category | Should have |
| FR6 | Add Incomes Category | Should have |
| FR7 | Calculate Saving | Must have |
| FR8 | Edit Expenses Category | Must have |
| FR9 | Edit Income Category | Must have |
| FR10 | Delete Expenses Category | Should have |
| FR11 | Delete Income Category | Should have |
| FR12 | Edit Profiles | Must have |
| FR13 | Delete Profiles | Should have |
| FR14 | View Reports | Should have |
| FR15 | Change Password | Could have |
| FR16 | Rating | Could have |
| FR17 | Logout | Must have |

|  |  |  |
| --- | --- | --- |
| **ID** | **Non-functional Requirements** | **MoSCoW** |
| NFR1 | Authentication | Must have |
| NFR2 | Availability | Must have |
| NFR3 | Documentation | Should have |
| NFR4 | Usability | Must have |
| NFR5 | Maintainability | Should have |
| NFR6 | Legal | Should have |
| NFR7 | User Friendly | Must have |
| NFR8 | Scalability | Should have |
| NFR9 | Performance | Must have |
| NFR10 | Reliability | Must have |
| NRF11 | Confidentiality | Must have |
| NRF12 | Integrity | Must have |
| NRF13 | Security | Must have |

## 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:**