2.1 Introduction to analysis

Here in the analysis I m going to do different tasks like gathering requirements, NLA, use case diagram and class diagram etc. Analysis is the initial phase of this project. In this phase of the project we determine and study different things. It is the phase where we examine the requirements, examine the required classes, examine the relationship between different classes.

It is necessary to perform analysis because it is a necessity if we are developing a project under waterfall model. We cannot risk a project by being haste. The better the analysis the better the project. So, in order to go to the next phases i.e. design analysis needs to be done. Analysis is compulsory because it is the initial phase of the project and it determines how the final project is going to be.

2.2 Feasibility study

Feasibility study is done with the purpose of checking the ability of completing the project smoothly. It is considered as preplan.

1. Economic feasibility

Here economic feasibility means if this project is feasible accordingly with the economic condition. This project is a single user-based project. So, it does not require a huge economic budget to complete. So, it is economically feasible.

1. Technical feasibility

It is a simple project which allows user to keep track of their expenses. It does not have any advanced technology. It is done using C# programming language with the help of Visual Studio. So, it is technically feasible.

1. Schedule feasibility

Here schedule feasibility refers to the completion of the project in scheduled time period. As we know that this project is a simple project so it can be completed in the scheduled time period.

1. Operational feasibility

This project is a desktop-based application. So, it can only be operational on the desktop. It has operational feasibility only on desktop.

1. Legal feasibility

This project doesn’t violate any human rights. It is clearly legal and can be operated for the benefits of the people.

2.3 Requirement analysis

1. Functional

This application will be able to keep track of the expenses that the user has done so far. It will help to manage the expenses that we have done. It helps to see how much savings we have done.

The system should efficiently add the expenses on different things.

Examples: add expenses, delete expenses, search expenses etc.

The system should have a simple login page with a signup button. This login page should lead the user to the dashboard where different menu strips will be located with their own functionality.

Authenticity will be checked with the database. First username and password are required to enter inside the dashboard form. After the username and password are inserted. It is checked with the database.

2. Non-functional

This system can operate offline. It has a local database that stores the expenses that we have done so far. It has low chance of system crash because it does not have any high-level advanced functionality. It is secured because of its offline and single based database.

3. Moscow prioritization

In order to understand what is the most important we have to use Moscow prioritization. It includes different adjectives:

Must have:

The system must have a database because without database the data cannot be recorded, retrieve, update or delete. So, database comes under must have.

Should have:

The system should have an authentication during login. It comes under security that makes it important. Username and password check is important for the security procedure.

Could have:

It could have an automated notification of a detail info of expenses after the end of every month.

Won’t have:

It won’t have a bad design for sure.

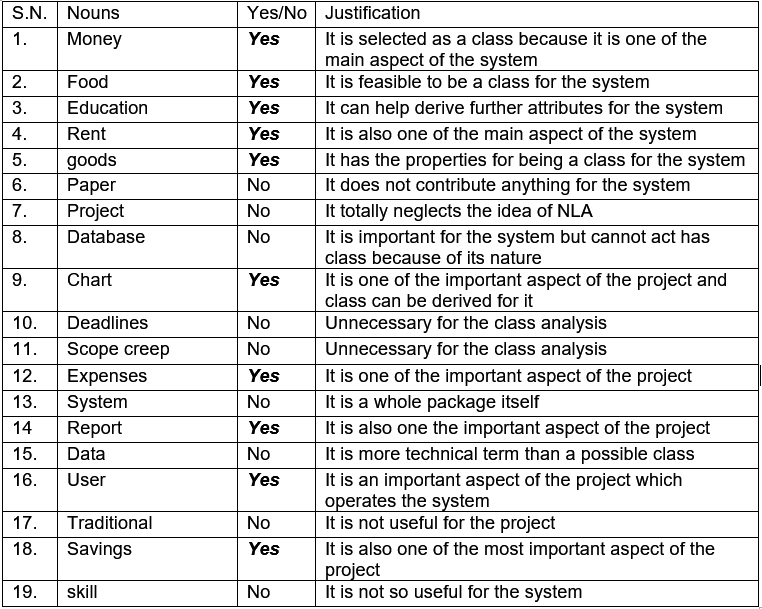
4. SRS

Natural Language Analysis

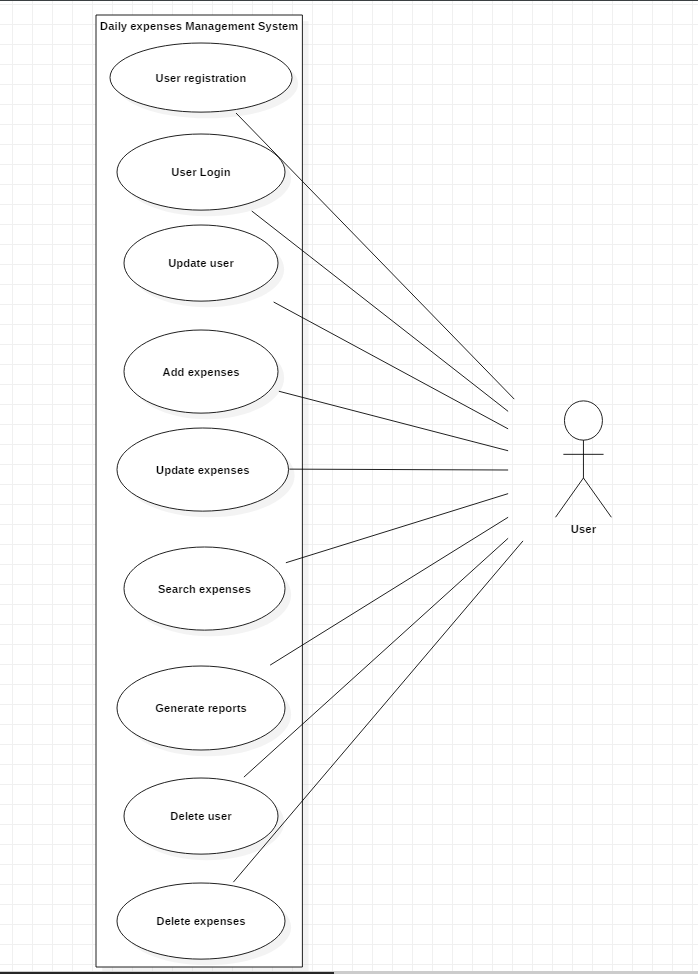
Natural Language Analysis is the initial analysis process which helps to find out the candidate class which we work on later. It is a part of analysis and it is done at first. In this process following steps are carried out:

1. First of all the selected nouns from the scenario are listed in a table.
2. Reduce the redundant nouns from the table.
3. Get rid of nouns that are improper for the system.

Here is the list of nouns for possible candidate class of the Daily expenses Management system:



Use case Diagram

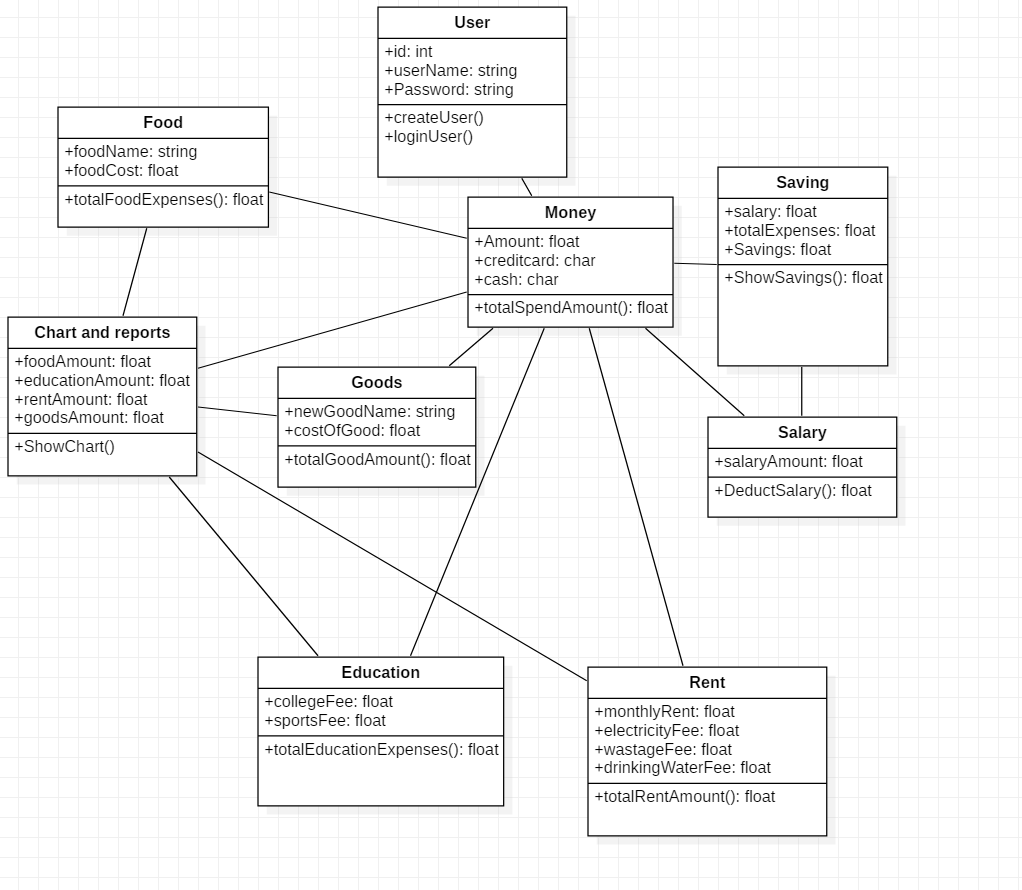


*Fig: Use case diagram of Daily Expenses management system*

The above use case diagram is the graphical representation of tasks that are done by the user. Use case diagram helps to show the tasks that are done by a user. It also shows the functionality of how the system works. Here above in the diagram we can see user registration as a use case. It is a functionality of the system that lets the user to register for their login. There is user login task that can also be done by the user. There are tasks such as update delete and other reports generating tasks that can also be done by the user. Daily expenses management system is for the single user. This project is solely handled by only one user so the use case diagram for this project has only one actor i.e. user.

Class Diagram

A class diagram is a graphical of representation of the classes that are interrelated with each other. Class diagram helps to show how each class are related with each other. The class diagram for this project is presented below:



*Fig: Class diagram of daily expenses Management System*