SOFTWARE REQUIREMENTS SPECIFICATION

**For**

**EXPENSE TRACKER**

**Prepared by:-**

*R. Sri Sathya Priya*

*R. Sandhiya*

*R. Sri Siva Sakthi*

# Introduction

## Purpose

The main objective of this document is to illustrate the requirements of the project personal Expense Tracker. The purpose of this project is to track the expenses and give a detailed report. This project describes the hardware and software interface requirements using ER diagrams and UML diagrams. This project provides a user-friendly environment. The main work of business or personal expense tracker is to keep finances organised and create an effective budget that can be used to expand the business later on.

## Document Conventions

* + - Entire document should be justified.
    - Convention for Main title

Font face: Times New Roman Font style: Bold

Font Size: 14

* + - Convention for Sub title

Font face: Times New Roman Font style: Bold

Font Size: 12

* + - Convention for body

Font face: Times New Roman Font Size: 12

## Scope of Development Project

Expense Tracker System is basically updating the expenses manually into an internet-based application so that the users can know the details of their expenses and maximum limit for

Spending the monthly salary.

The project is specifically designed for our personal use. The product will work as a complete user interface for tracking the expense process .It can be used by anyone to manage their expenses and helps the user to understand their statistics. It is especially useful for people who needs to control their expenses and gain awareness for financial decisions.

The project can be easily implemented under various situations. We can add or update values and making expenses accordingly .The language used for developing the project is Java as it is quite advantageous than other languages in terms of performance, tools available, cross platform compatibility, libraries, cost (freely available), and development process.

## Definitions, Acronyms and Abbreviations

JAVA -> platform independence SQL-> Structured query Language ER-> Entity Relationship

UML -> Unified Modeling Language

IDE-> Integrated Development Environment SRS-> Software Requirement Specification

## References

* + - Books

 Monthly Bill Planner and Organizer: Finance Monthly and Weekly Budget Planner Expense Tracker Bill Organizer Journal Notebook Budget Planning Budget Worksheets Person Business

Money workbook by Jada Correia.

* + - Website

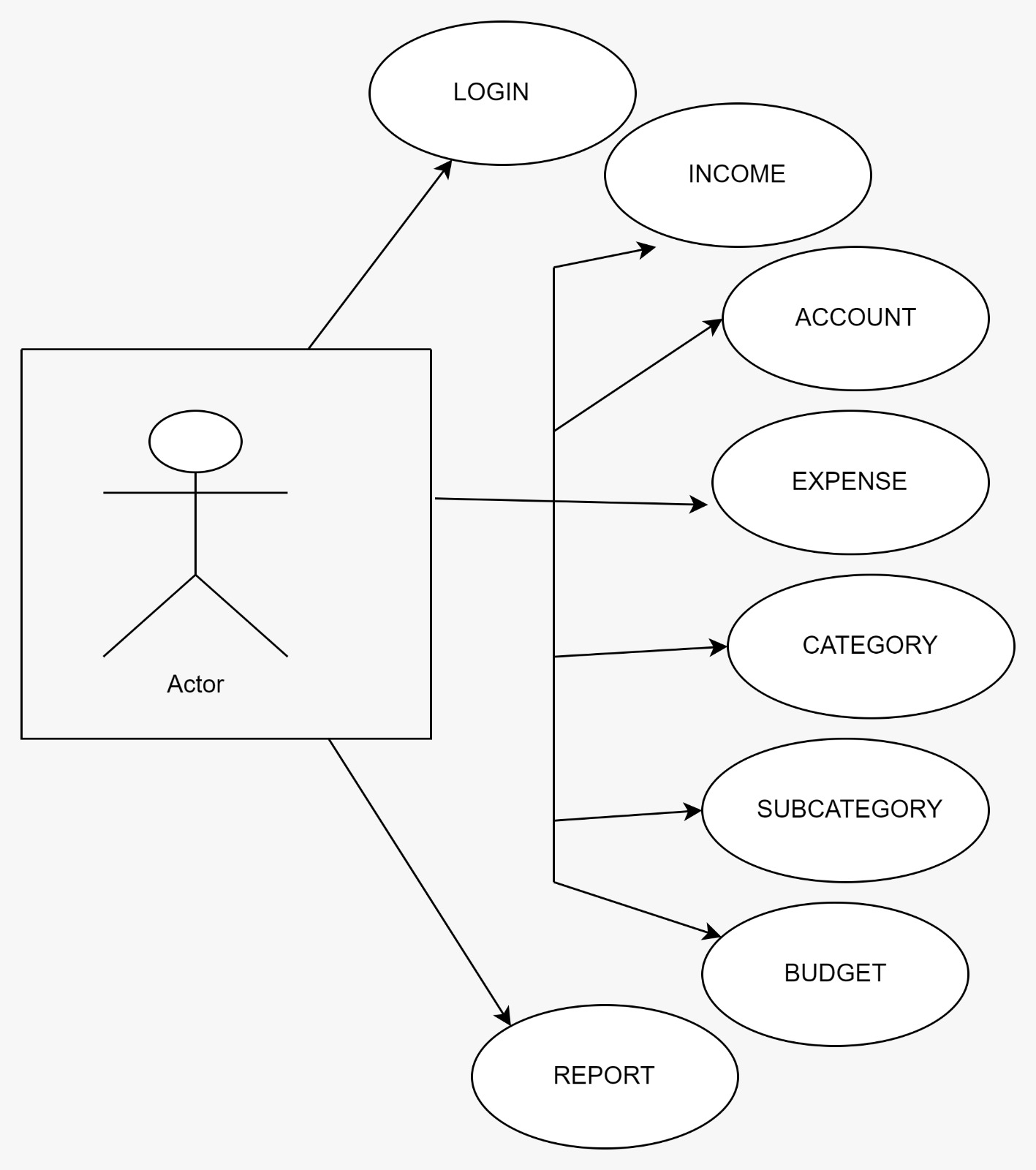
https://www.slideshare.net/RashnaMaharjan2/daily-expense-tracker

**https://www.zoho.com/in/invoice/help/expense/expense-tracking.html**

# Overall Descriptions

## Product Perspective

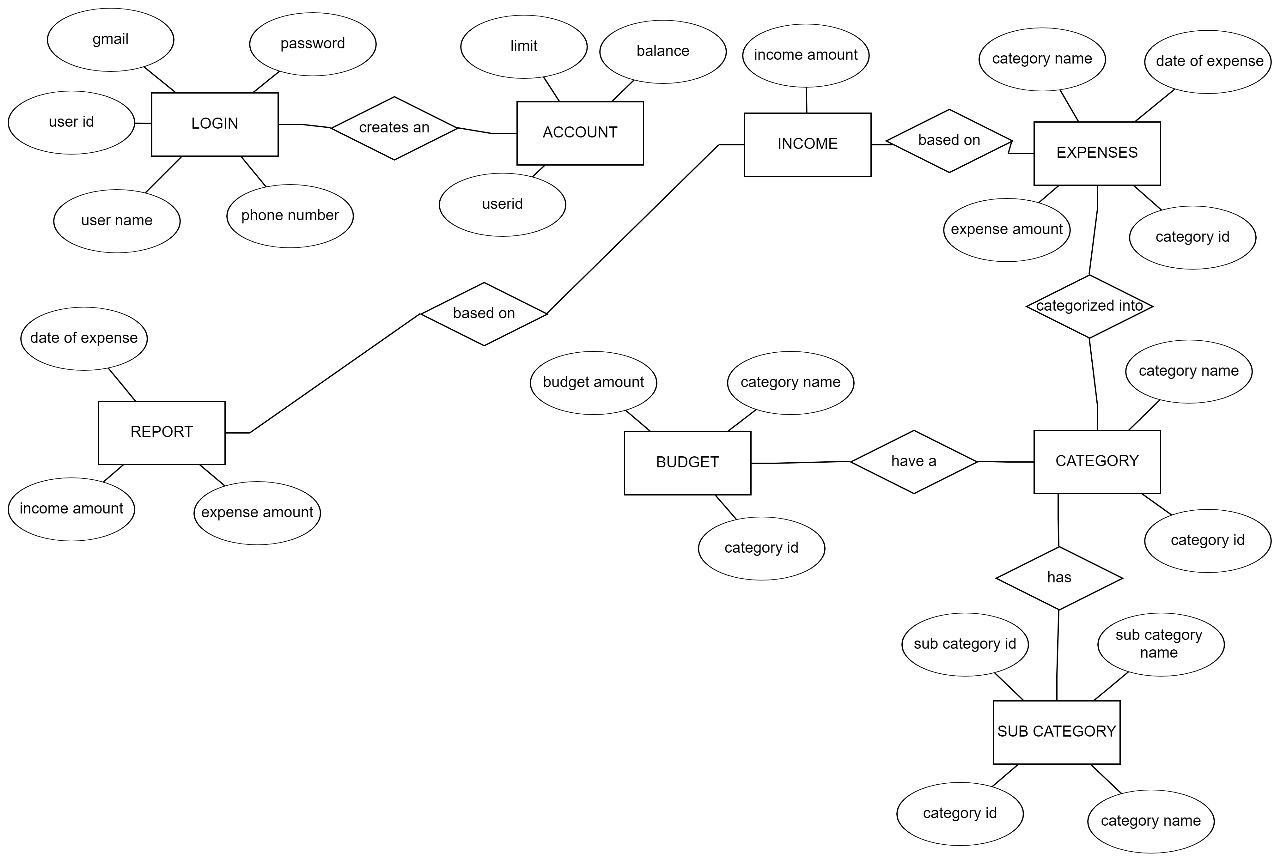
Use Case Diagram of Library Management System



This is a broad level diagram of the project showing a basic overview. The users is the person using it. This System will provide a functionality to manage expense. This will be based on various categories viz. online payment and offline payment. Further the user can add/update the details and the transactions made.It helps the user to understand their expenditure.

## 2.1 Product Function

Entity Relationship Diagram of Expense Tracker



The Expense Tracker provides online real time information about the expense and account balance and the user information. The main purpose of this project is to reduce the track the expenses. This software is capable of managing expenses ,account balance , Generating various Reports for expense tracking according to end user requirements. The user will act as the administrator to control expense and manage salary. The member’s status of expense is maintained in the expense database. The user’s details can be fetched by the user from the database as and when required. The user is only allowed to view their account information.

## User Classes and Characteristics

The system provides different types of services based on the type of users . The user will be acting as the controller and he will have all the privileges of an administrator. The member is the user who will be managing the expenses online.

The features that are available to the user is:-

* + - User can input the income.
    - Can view the different categories of expenses.
    - Can view the subcategories of the expenses.
    - Can set the budget limit.
    - Add/update the income.
    - Edit the information of various expenses.
    - Can check the report of the expenses.
    - Can check the account balance.
    - Can access all the categories and limit the expense accordingly.
    - Can have a transaction limit.
    - Can view the monthly report of expenses.

## Operating Environment

The product will be operating in windows environment. The Library Management System is a website and shall operate in all famous browsers, for a model we are taking Microsoft Internet Explorer,Google , Chrome,and Mozilla Firefox.Also it will be compatible with the IE 6.0. Most of the features will be compatible with the Mozilla Firefox & Opera 7.0 or higher version. The only requirement to use this online product would be the internet connection.

The hardware configuration include Hard Disk: 40 GB, Monitor: 15” Color monitor, Keyboard: 122 keys. The basic input devices required are keyboard, mouse and output devices are monitor, printer etc.

## Assumptions and Dependencies

The assumptions are:-

* + - The coding should be error free
    - The system should be user-friendly so that it is easy to use for the users
    - The information of users must be stored in a database that is accessible by the website
    - The system should have more storage capacity and provide fast access to the database
    - The system should provide monthly report and support quick transactions
    - The expense tracker system is running 24 hours a day
    - Users may access from any computer that has Internet browsing capabilities and internet connection.
    - Users must have their correct usernames and passwords to enter into their online accounts and do actions.

The dependencies are:-

* + - The specific hardware and software due to which the product will be run
    - On the basis of listing requirements and specification the project will be developed and run
    - The end users (admin) should have proper understanding of the product
    - The system should have the general report stored
    - The information of users must be stored in a database that is accessible by the System
    - Any update regarding the income from the user is to be recorded to the database and the data entered should be correct

## Requirement

Software Configuration:-

This software package is developed using java as front end which is supported by sun micro system. Microsoft SQL Server as the back end to store the database.

Operating System: Windows NT, windows 98, Windows XP Language: Java Runtime Environment, Net beans 7.0.1 (front end) Database: MS SQL Server (back end)

Hardware Configuration:- Processor: Pentium(R)Dual-core CPU Hard Disk: 40GB

RAM: 256 MB or more

## Data Requirement

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts. In this project the inputs will be the queries as fired by the users like create an account, selecting category and putting into account. Now the output will be visible when the user requests the server to get details of their account in the form of date and which expense are currently in the account.

# External Interface Requirement

## GUI

The software provides good graphical interface for the user and the administrator can operate on the system, performing the required task such as create, update, viewing the details of the book.

* + - It allows user to view quick reports like Book Issued/Returned in between particular time.
    - It provides stock verification and search facility based on different criteria.
    - The user interface must be customizable by the administrator
    - All the modules provided with the software must fit into this graphical user interface and accomplish to the standard defined
    - The design should be simple and all the different interfaces should follow a standard

template

* + - The user interface should be able to interact with the user management module and a part of the interface must be dedicated to the login/logout module

Login Interface:-

In case the user is not yet registered, he can enter the details and register to create his account. Once his account is created he can ‘Login’ which asks the user to type his username and password. If the user entered either his username or password incorrectly then an error message appears.

Search:-

The user can enter the date he is looking for and then he can search for the expense report of that particular date.

Categories View:-

Categories view shows the categories of expenses available and provides ability to the user to add/edit or delete category from the list.

User’s Control Panel:-

This control panel will allow user to add/remove users; add, edit, or remove a resource. And manage lending options.

# System Features

The users of the system should be provided the surety that their account is secure. This is possible by providing:-

* User authentication and validation of members using their unique member ID
* Proper monitoring by the administrator which includes updating account status, showing a popup if the member attempts to issue number of transaction or budget that exceed the limit provided by the expense policy, assigning error to users who skip the date .
* Proper accountability which includes not allowing a member to see other member’s account. Only administrator will see and manage all his accounts.

# Other Non-functional Requirements

## Performance Requirement

The proposed system that we are going to develop will be used as the performance system within the different users which interacts with the users details. Therefore, it is expected that the database would perform functionally all the requirements that are specified by the user.

* + - The performance of the system should be fast and accurate
    - Expense Management System shall handle expected and non-expected errors in ways that prevent loss in information and long downtime period. Thus it should have inbuilt error testing to identify invalid username/password
    - The system should be able to handle large amount of data. Thus it should accommodate high number of category and transactions without any fault.

## Safety Requirement

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup so that the database is not lost. Proper UPS/inverter facility should be there in case of power supply failure.

## Security Requirement

* + - System will use secured database
    - Normal users cannot read information and they cannot edit or modify anything except their personal and some other information.
    - System will have different types of users and every user has access constraints
    - Proper user authentication should be provided
    - No one should be able to hack users’ password
    - There should be account for admin and such that no member can access the database and only admin has the rights to update the database.

## Requirement attributes

* + - There may be multiple admins creating the project, all of them will have the right to create changes to the system. But the members or other users cannot do changes
    - The project should be open source
    - The Quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database
    - The user be able to easily download and install the system

## Business Rules

A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, make a decision, or infer new data from existing data.This includes the rules and regulations that the System users should abide by. This includes the cost of the project and the discount offers provided. The users should avoid illegal rules and protocols. Neither admin nor member should cross the rules and regulations.

## User Requirement

The users of the system is the person using who act as administrator to maintain the system. The members are assumed to have basic knowledge of the computers and internet browsing. The administrators of the system should have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper user interface, user manual, online help and the guide to install and maintain the system must be sufficient to educate the users on how to use the system without any problems.

The admin provides certain facilities to the users in the form of:-

* + - Backup and Recovery
    - Forgot Password
    - Data migration i.e. whenever user registers for the first time then the data is stored in the server
    - Data replication i.e. if the data is lost in one branch, it is still stored with the server
    - Auto Recovery i.e. frequently auto saving the information
    - Maintaining files i.e. File Organization
    - The server must be maintained regularly and it has to be updated from time to time

# Other Requirements

## Data and Category Requirement

There are no multiple users. Depending upon the category of user the access rights are decided.It means if the user is an administrator then he can be able to modify the data,delete, append etc. All other users only have the rights to retrieve the information about database. Similarly there will be different categories of expenses available. According to the categories of expenses their relevant data should be displayed. The categories and the data related to each category should be coded in the particular format.

## Appendix

A: Admin, Abbreviation, Acronym, Assumptions; B: Books, Business rules; C: Class, Client, Conventions; D: Data requirement, Dependencies; G: GUI; K: Key; L: Library, Librarian; M: Member; N: Non-functional Requirement; O: Operating environment; P: Performance,Perspective,Purpose; R: Requirement, Requirement attributes; S: Safety, Scope, Security, System features; U: User, User class and characteristics, User requirement;

## Glossary

The following are the list of conventions and acronyms used in this document and the project as well:

* + - Administrator: A login id representing a user with user administration privileges to the software
    - User: A general login id assigned to most users
    - Client: Intended users for the software
    - SQL: Structured Query Language; used to retrieve information from a database
    - SQL Server: A server used to store data in an organized format
    - Layer: Represents a section of the project
    - User Interface Layer: The section of the assignment referring to what the user interacts with directly
    - Application Logic Layer: The section of the assignment referring to the Web Server. This is where all computations are completed
    - Data Storage Layer: The section of the assignment referring to where all data is recorded
    - Use Case: A broad level diagram of the project showing a basic overview
    - Class diagram: It is a type of static structure diagram that describes the structure of a system by showing the system’s cases, their attributes, and the relationships between the classes
    - Interface: Something used to communicate across different mediums
    - Unique Key: Used to differentiate entries in a database

## Class Diagram

A class is an abstract, user-defined description of a type of data. It identifies the attributes of the data and the operations that can be performed on instances (i.e. objects) of the data. A class of data has a name, a set of attributes that describes its characteristics, and a set of operations that can be performed on the objects of that class. The classes’ structure and their relationships to each other frozen in time represent the static model. In this project there are certain main classes

which are related to other classes required for their working. There are different kinds of relationships between the classes as shown in the diagram like normal association, aggregation, and generalization. The relationships are depicted using a role name and multiplicities. Here ‘User’, ‘Expense’ and ‘Income’ are the most important classes which are related to other classes.

