



# Software Requirements Specification

Version 1.0

**Theme:** Trip Planner

**Project Name:** ExpenseVoyage

**Category:** Web Application Development

## Table of Contents

<b>1.1 Background and Necessity for the Web Application .....</b>	<b>3</b>
<b>1.2 Proposed Solution .....</b>	<b>3</b>
<b>1.3 Purpose of this Document .....</b>	<b>4</b>
<b>1.4 Scope of Project .....</b>	<b>4</b>
<b>Architecture Diagram for ExpenseVoyage .....</b>	<b>4</b>
<b>Use Case Diagram for ExpenseVoyage .....</b>	<b>5</b>
<b>Entity Relationship Diagram for ExpenseVoyage .....</b>	<b>6</b>
<b>1.5 Constraints .....</b>	<b>6</b>
<b>1.6 Functional Requirements .....</b>	<b>7</b>
<b>1.7 Non-Functional Requirements .....</b>	<b>10</b>
<b>1.8 Interface Requirements .....</b>	<b>11</b>
<b>1.9 Project Deliverables .....</b>	<b>13</b>

## 1.1 Background and Necessity for the Web Application

Traveling, whether for business or leisure, often involves managing multiple expenses. These expenses can quickly accumulate, leading to the necessity for meticulous tracking to stay within budget. Many travelers, however, struggle with maintaining an organized record of their expenditures, leading to overspending, loss of receipts, and difficulty in reimbursement claims.



Traditional methods of tracking expenses, such as manual entry in spreadsheets or keeping physical receipts, are cumbersome and prone to errors. With the rise of digital transformation, there is a growing demand for automated solutions that simplify expense management. The proliferation of smartphones and easy Internet access provide an excellent opportunity to develop a Web application that addresses these challenges effectively.

## 1.2 Proposed Solution

To make it easier for users, a Web application titled **ExpenseVoyage** is to be developed. This Web application will enable users to register, and thereafter facilitate efficient and user-friendly tracking of travel expenses and more. The application will cater to individual travelers as well as groups ensuring seamless and collaborative expense tracking.



### 1.3 Purpose of this Document

The purpose of this document is to present a detailed description of **ExpenseVoyage** Web application. This document will explain the purpose and features of the application, the interfaces of the application, what the application will do, and so on. This document is intended for both stakeholders and developers of the system.

### 1.4 Scope of Project

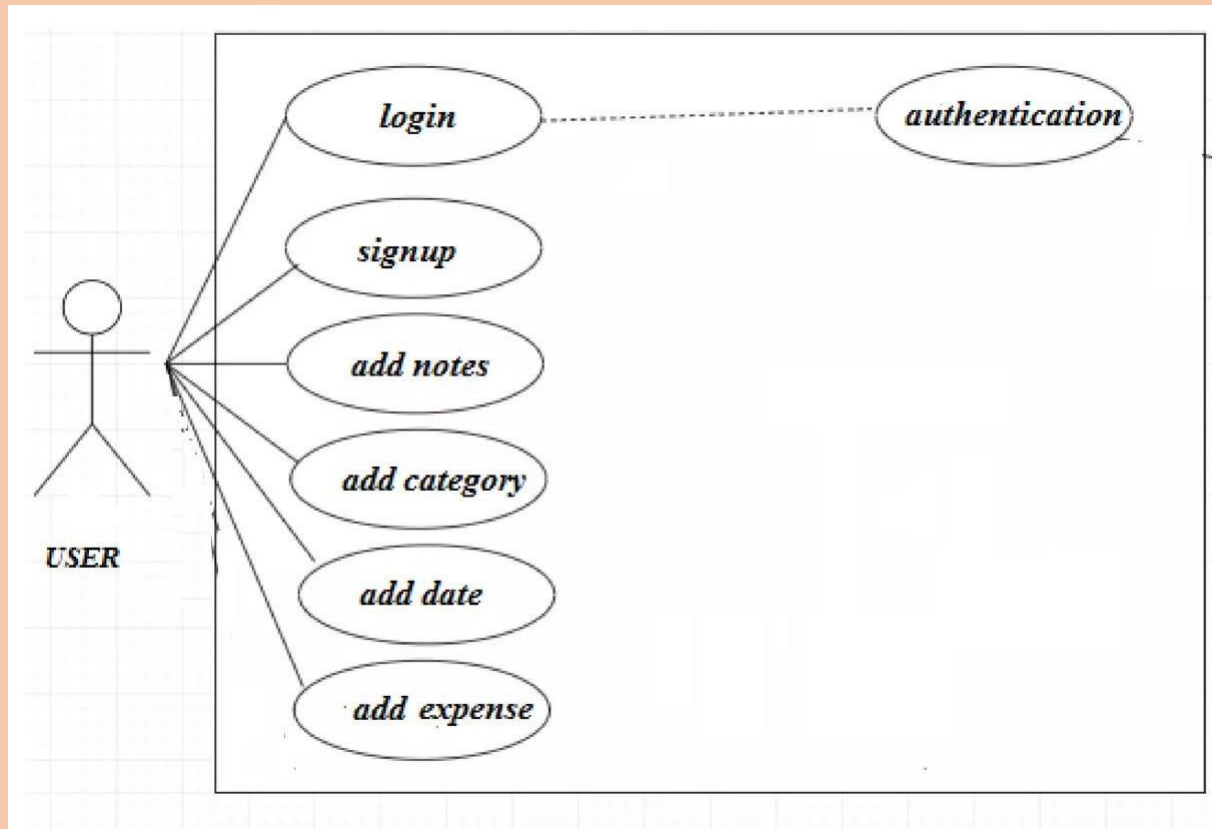
This will be a Web application to be used by individuals. The **ExpenseVoyage** Web application aims to provide a comprehensive and user-friendly solution for travelers to manage and track their travel expenses.



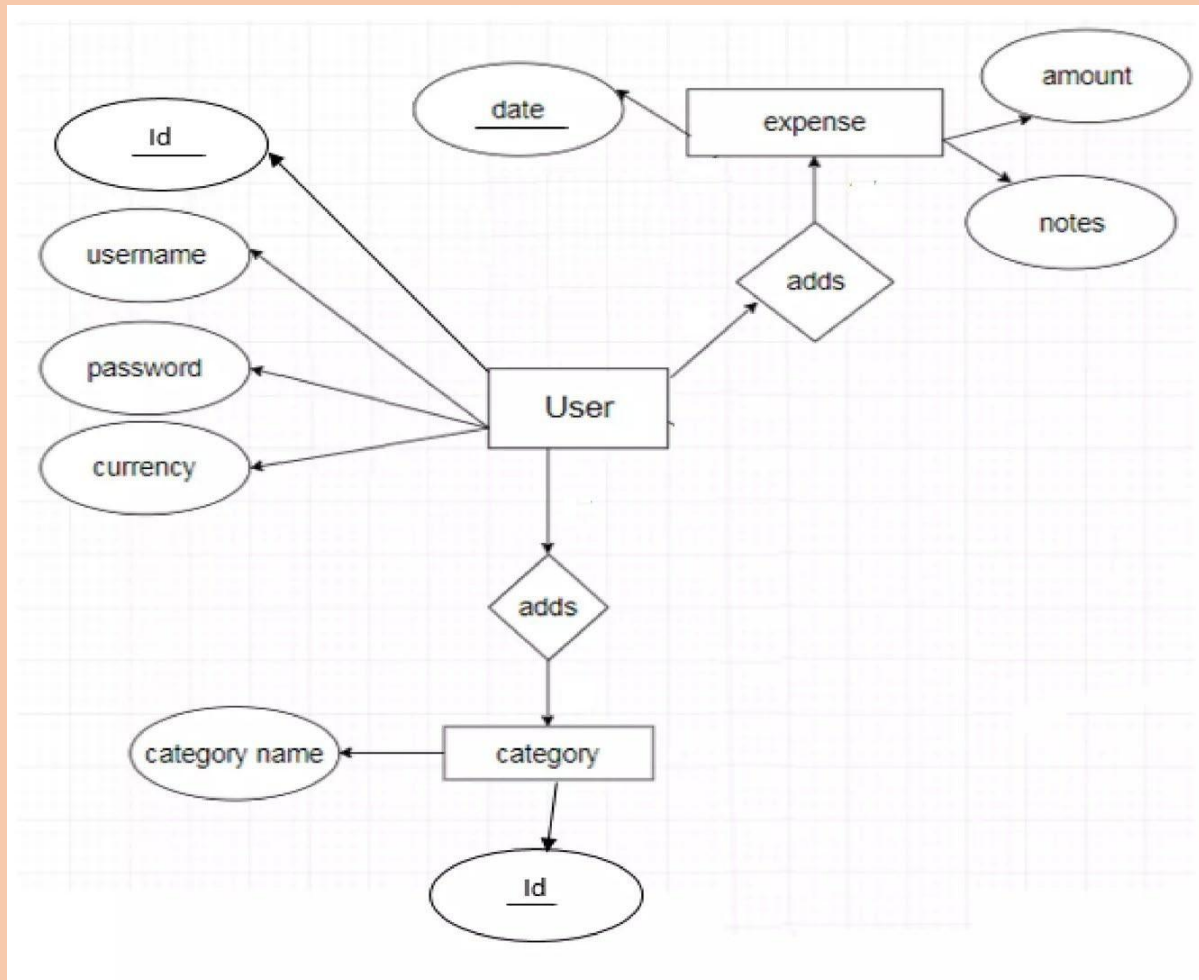
### Architecture Diagram for ExpenseVoyage



## Use Case Diagram for ExpenseVoyage



## Entity Relationship Diagram for ExpenseVoyage



### 1.5 Constraints

The development of the **ExpenseVoyage** Web application must adhere to several constraints to ensure its successful implementation and operation. Technically, the application must be compatible with major Web browsers and responsive across various devices. You may encounter constraints related to data storage, data synchronization, and backup procedures. The usage of various images, and videos may be subject to licensing agreements and copyright restrictions. It is important to understand and comply with these constraints to avoid legal issues.

## 1.6 Functional Requirements



The Web application will be designed with a set of forms/pages with menus representing choice of activities to be performed.

Following are the functional requirements of the application:

**For Users:**

### 1) Home Page and Dashboard

- a) Personalized Welcome Message
  - It will display an attractive visual and menus for various operations including basic functionality such as Registration and Login. After users are logged in, they will be able to access various menus.
  - The home page must display a personalized welcome message with the user's name on logging in.
- b) Current/Upcoming Trip Overview
  - The home page must show an overview of current or upcoming trips including destination, dates, key itinerary items, and remaining budget.
- c) Recent Activities
  - Display a list of recent activities including logged expenses and itinerary updates.





## 2) User Management

- a) User Registration and Login
  - Users must be able to register using their email address and password.
  - Provide password recovery and reset functionalities.
- b) User Profile Management
  - Users must be able to create and edit their profile including personal information, preferred currency, and travel preferences.
  - Users must be able to upload a profile picture.

## 3) Trip Planning

- a) Create and Manage Trips
  - Users must be able to create a new trip by specifying the destination, start date, and end date.
  - Users must be able to edit and delete existing trips.
  - Users must be able to view a list of all their trips.
- b) Itinerary Management
  - Users must be able to add, edit, and delete itinerary items including flights, accommodations, and activities.
  - Itinerary items must be categorizable (for example, transport, lodging, and activities).

## 4) Expense Tracking

- a) Real-Time Expense Logging
  - Users must be able to log expenses manually, specifying the amount, category, date, and notes.
  - Users must be able to edit and delete logged expenses.
- b) Budget Management
  - Users must be able to set an overall budget for each trip.
  - Users must be able to set category-specific budgets (for example, accommodation, food, and transport).
  - The application must provide real-time tracking of expenses against the set budget.
  - Users must receive alerts when they are approaching or exceeding their budget.
- c) Currency Conversion
  - The application must support multiple currencies [at least four, including USD \$, Euro €, GBP £, and your Local Currency] and provide exchange rates.
  - Expenses must be convertible to the user's preferred currency.



## 5) Reports

### a) Expense Reports

- Generate detailed expense reports summarizing expenses by category and trip.

## 6) Notifications and Reminders

### a) Reminders

- Send daily notifications to log expenses.
- Provide reminders for upcoming itinerary items.

### b) Alerts

- Send budget alerts to notify users when they are approaching or exceeding their spending limits.

## For Admin

- **Login:** It will allow Admin to login to the Web Application.
- **Add/Modify/Delete:** It will allow Admin to perform these operations
  - Add and Manage destinations.
  - Add and Manage tours/trips with their itinerary.
  - Add Currency and exchange rates.

## Common to both Admin and Users

- **Search/Filter:** These options enable search or filter for a particular trip/tour or destination and so on based on specific criteria.
- **Sort:** This option enables sorting of data based on specific criteria.
- **About Us:** Description of services offered by the Web application.
- **Image Gallery:** Gallery showcasing various locations.
- **Contact Us:** A form to accept user queries with fields for name, email, contact number, and message.

Implement GPS functionality as appropriate.

**Note:** Boilerplate or readymade HTML template can be used, provided it is only for design aspect and not for implementing application functionality.

## 1.7 Non-Functional Requirements

There are several non-functional requirements that should be fulfilled by the application. These include:

- **Safe to use:** The application should not result in any malicious downloads or unnecessary file downloads.
- **Accessibility:** The application should have clear and legible fonts, user-interface elements, and navigation elements.
- **User-friendliness:** The application should be easy to navigate with clear menus and other elements and easy to understand.
- **Operability:** The application should be reliable and efficient.
- **Performance:** The application should demonstrate high value of performance through speed and throughput. In simple terms, the application should have minimal load time and smooth page redirection.
- **Scalability:** The application architecture and infrastructure should be designed to handle increasing user traffic, data storage, and feature expansions.
- **Security:** The application should implement adequate security measures such as authentication. For example, only registered users can access certain features.
- **Availability:** The application should be available 24/7 with minimum downtime.
- **Compatibility:** The application should be compatible with latest browsers and various devices.



These are the bare minimum expectations from the project. It is a must to implement the FUNCTIONAL and NON-FUNCTIONAL requirements given in this SRS. Once they are complete, you can use your own creativity and imagination to add more features if required.

## 1.8 Interface Requirements

### Hardware

- Intel Core i5/i7 Processor or higher
- 8 GB RAM or higher
- Color SVGA
- 500 GB Hard Disk space
- Mouse
- Keyboard

### Software

#### Technologies to be used:

- **IDE:** Appropriate IDE as per the platform
- **Frontend:**

HTML5, CSS3, Bootstrap (optional), JavaScript, Figma Toolkit, jQuery, AngularJS/Angular 9 or higher/ReactJS 18 or higher, and XML

- **Backend:**

Java 16 or higher, Java EE 7 or higher/Jakarta EE 9 or higher, with Apache NetBeans IDE/Eclipse latest version, Apache Tomcat 10.0 or higher, Glassfish 6.0 or higher, and related libraries

OR

C# 7.2 with ASP.NET MVC and ASP.NET MVC Core (optional), Visual Studio 2019 or higher

OR

PHP 7.0 or higher with Apache Web Server and Laravel Framework Homestead (optional)

OR

Python 3.0 or higher with PyCharm IDE, Django/Flask framework

#### For local hosting (optional):

XAMPP latest version

- **Database:**

MySQL 8.0 or higher/SQL Server 2019 or higher

## Database Design

Users	Trips	Expenses	Categories
user_id (primary key)	trip_id (primary key)	expense_id (primary key)	category_id (primary key)
email	user_id (foreign key)	user_id (foreign key)	user_id (foreign key)
password_hash	trip_name	trip_id (foreign key)	category_name
first_name	start_date	amount	
last_name	end_date	category	
	destination	expense_date	
	budget	notes	

Similarly, you can define other entities and relationships between entities and methods representing activities on the entities.

**Note:** These are just examples, you do not have to adhere to these structures and can design your own table structure with columns.

## 1.9 Project Deliverables

You will require to design and build the project and submit it along with a complete project report that includes:

- Problem Definition
- Design Specifications
- Diagrams such as Flowcharts for various Activities, Data Flow Diagrams, and so on
- Database Design
- Test Data Used in the Project
- Project Installation Instructions (if any)
- **User Credentials for all Types of Users with Passwords**

Documentation is considered as a very important part of the project. Ensure that documentation is complete and comprehensive.

**Documentation should not contain any source code.**

The consolidated project will be submitted as a zip file with a ReadMe.doc file listing assumptions (if any) made at your end and SQL scripts files (.sql) containing database and table definitions.

**Note:** Preferably, host the working Web application on a Website and share the URL for evaluation.

**Submit a video (.mp4 file) demonstrating the working of the Web application, including all the functionalities of the project. This is MANDATORY.**

Over and above the given specifications, you can apply your creativity and logic to improve the system.

**Sitemap:** To understand the flow of **ExpenseVoyage** Web Application, you will have to create a Sitemap and add it to the home page of your application.

*-- End of Document --*