CHAPTER 1

INTRODUCTION

Mark-up languages are the languages in which the web is written. The most common mark-up language used is HTML, which uses tags to annotate text so that a computer can then manipulate the text.

HTML is the conventional mark-up language used to create and edit web pages and web applications. HTML is used for creating the basic structure of a website. HTML consists of different elements preceded by an opening tag, <tag>, and a closing tag, </tag>. The content between the tags, <html> and </html>, is the content of the webpage. The content between the tags, <head> and </head>, is the title of the webpage. This text is displayed between the <title> and </title> tags. The content between the tags, <body> and </body>, is the main content of the webpage. The content can include links, paragraphs, headings, and various other elements.

1.1 What is the need of DBMS in Web Technology?

Database systems are basically developed for large amount of data. When dealing with huge amount of data, there are two things that require optimization: Storage of data and retrieval of data.

Storage: According to the principles of database systems, the data is stored in such a way that it acquires lot less space as the redundant data (duplicate data) has been removed before storage.

Fast Retrieval of data: Along with storing the data in an optimized and systematic manner, it is also important that we retrieve the data quickly when needed. Database systems ensure that the data is retrieved as quickly as possible.

The choice of a database product is often influenced by factors such as:

- the computing platform (i.e., hardware, operating system)
- the volume of data to be managed
- the number of transactions required per second
- existing applications or interfaces that an organization may have

1.2 Design and Modeling:

The first task of a database designer is to produce a conceptual datamodel that reflects the structure of the information to be held in the database. A common approach to this is to develop an entity-relationship model, often with the aid of drawing tools. Another popular approach is the Unified Modeling Language. A successful data model will accurately reflect the possible state of the external world being modeled: for example, if people can have more than one phone number, it will allow this information to be captured.

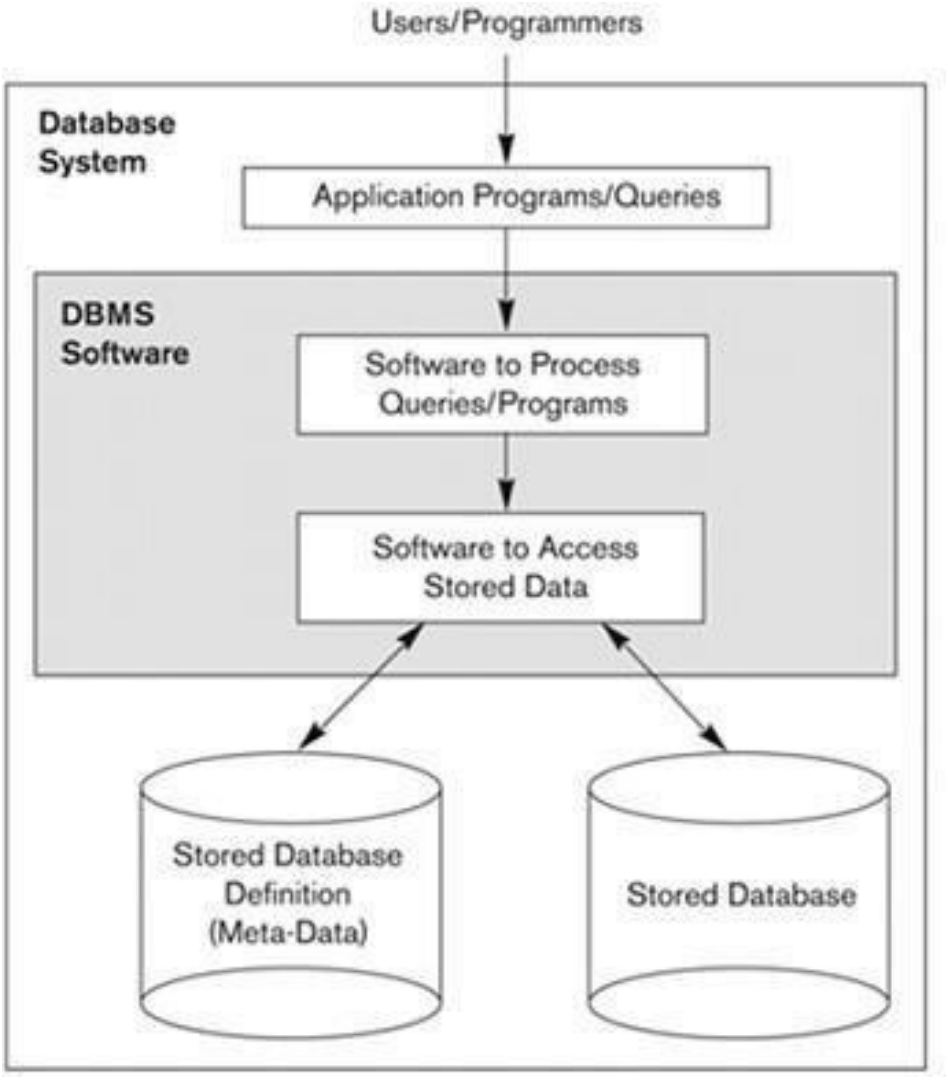


Figure 3.1: Simplified database system environment

1.3 PROBLEM STATEMENT

In the world of software development there lots of improvement in the area of Architectural design and principles. The philosophies and implementation details are changing as the people guiding the development of the application.

Web services are one such area where architects must lean on their creative side and hope that their solutions are still successful. In this report we will explain an exciting voyage down the road of Web services application.

1.4 OBJECTIVE

The main objective of the project on “Know Your Expense” is to manage the expense of the user. User can add their daily expenses which they spend on food, electricity bill, water bill, hotel bills etc. User can also delete their expenses. Know Your Expense aims to helps everyone to track their expenses and user can view their expenses on day wise basis, weekly basis, month wise basis and year wise basis according to periods of time. User can also view their overall expenses. User can view expenses report on day wise basis, month wise basis and year wise basis according to periods of time.

.

CHAPTER 2

SYSTEM SPECIFICATION

HARDWARE & SOFTWARE REQUIREMENT

2.1 SOFTWARE REQUIREMENTS

Programming language	:	HTML, MYSQL, PHP
Operating system	:	ANY OS (Recommended**:** Windows 7 or above)
Web Server	:	XAMPP
Program editor	:	Sublime

2.2 HARDWARE REQUIREMENTS

CPU	:	Pentium IV 2.4 GHz or above
Memory (Primary)	:	512 MB, 1 GB or above
Hard Disk	:	40 GB, 80GB, 160GB or above
Monitor	:	15 VGA color

2.3 Sublime Text 3

Sublime Text 3 (ST3) is a lightweight, cross-platform code editor known for its speed, ease of use, and strong community support. It's an incredible editor right out of the box, but the real power comes from the ability to enhance its functionality using Package Control and creating custom settings.

2.4 Xampp

XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database and interpreters for scripts written in the PHP and Perl programming languages. XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (MariaDB), and scripting language (PHP) – is included in an extractable file.

2.5 Browser

Google Chrome is preferably used to run the web server locally. "Localhost" refers to the local computer that a program is running on. Localhost is used in Web scripting languages like PHP, for defining Apache server the code should run from or where the online book store database is located.

CHAPTER 3

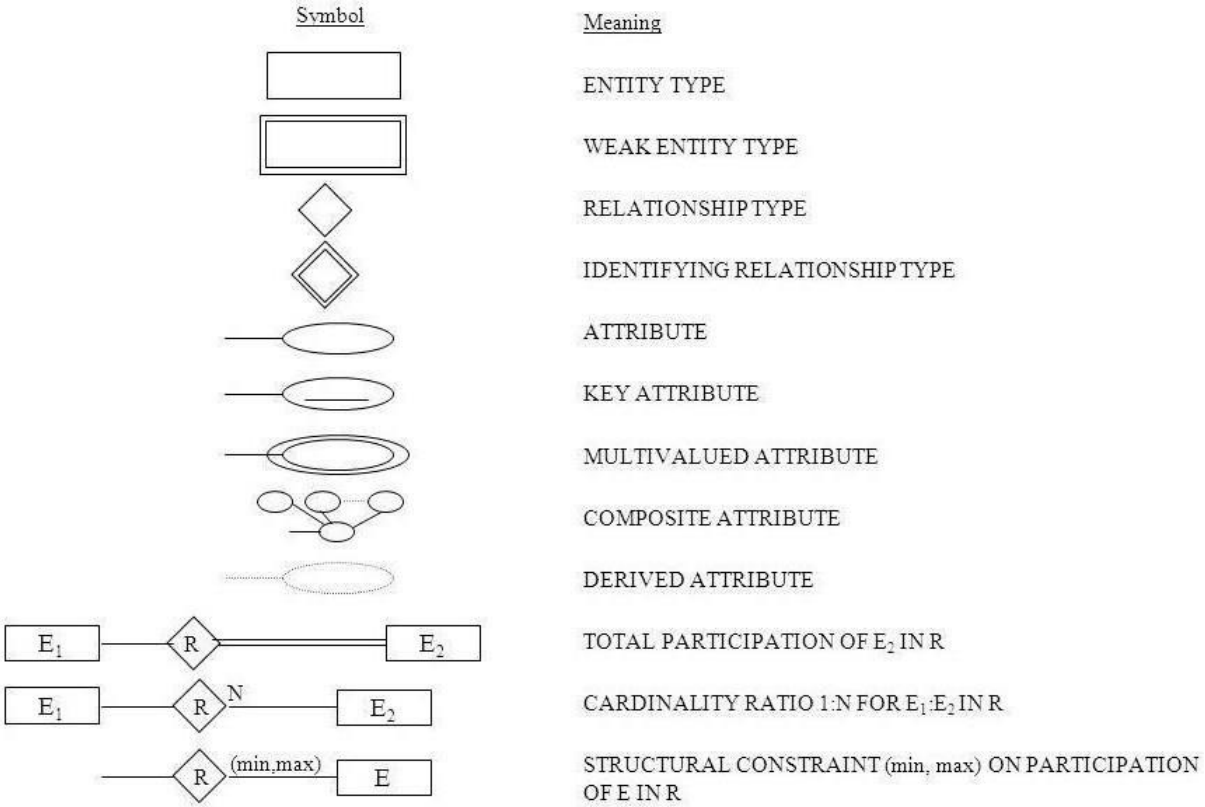
SYSTEM DESIGN

3.1 ER Diagram

An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems. An entity is a piece of data-an object or concept about which data is stored.

The cardinality or fundamental principle of one data aspect with respect to another is a critical feature. The relationship of one to the other must be precise and exact between each other in order to explain how each aspect links together. In simple words Cardinality is a way to define the relationship between two entities.

The following are the notations of the ER diagram:



Following interactions between the entities and their relationships has been depicted in the above ER Diagram.

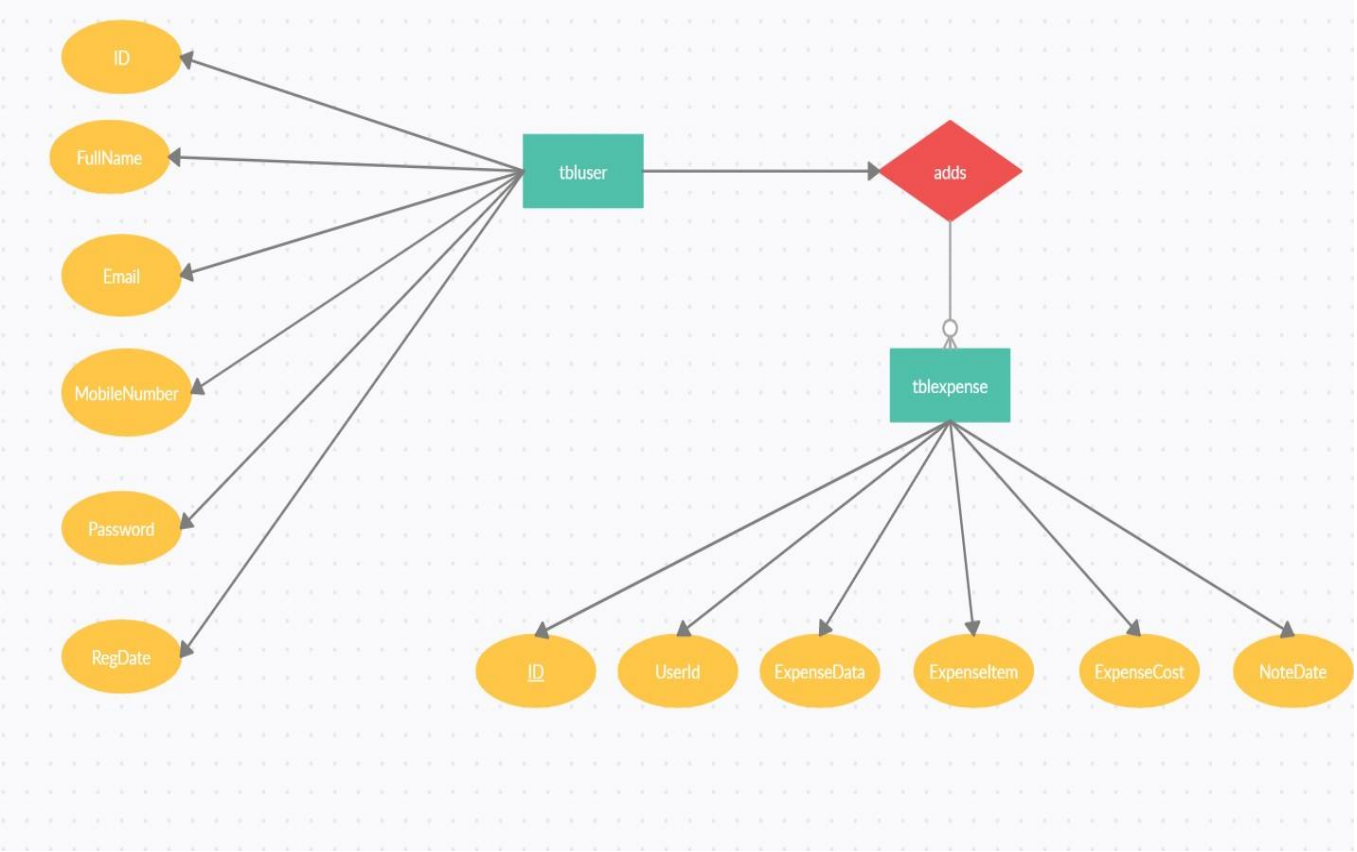


Figure 3.2: ER Diagram

3.2 Schema Diagram

In any data model it is important to distinguish between the description of the database and the database itself. The description of a database is called the database schema, which is specified during database design and is not expected to change frequently.

A displayed schema is called a schema diagram. A schema diagram displays only some aspects of a schema, such as the names of record types and data items, and some types of constraints.

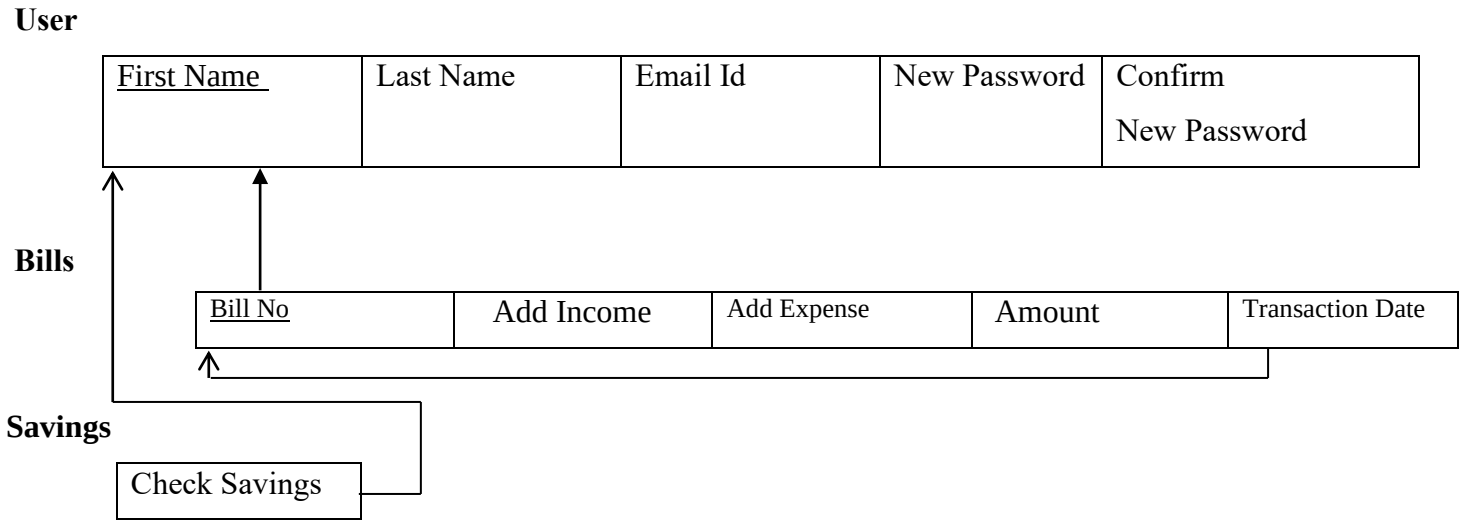


Fig 3.3: Schema Diagram

CHAPTER 4

IMPLEMENTATION

4.1 MYSQL DATABASE

In this project, MySQL is used as the backend database. MySQL is an open source database management system.

The **features** of MySQL are given below:

- **Easy to use:** MySQL is easy to use. You have to get only the basic knowledge of SQL. You can build and interact with MySQL with only a few simple SQL statements.
- **It is secure:** MySQL consist of a solid data security layer that protects sensitive data from intruders. Passwords are encrypted in MySQL.
- **Client/ Server Architecture:** MySQL follows a client /server architecture. There is a database server (MySQL) and arbitrarily many clients (application programs), which communicate with the server; that is, they query data, save changes, etc.
- **Free to download:** MySQL is free to use and you can download it from MySQL official website.
- **It is scalable:** MySQL can handle almost any amount of data, up to as much as 50 million rows or more. The default file size limit is about 4 GB. However, you can increase this number to a theoretical limit of 8 TB of data.
- **Compatible– on many operating systems:** MySQL is compatible to run on many operating systems, like Novell NetWare, Windows* Linux*, many varieties of UNIX*, OS/2, FreeBSD*, and others.
- **Allows roll-back:** MySQL allows transactions to be rolled back, commit and crash recovery.
- **High Performance:** MySQL is faster, more reliable and cheaper because of itsunique storage engine architecture.
- **High Flexibility:** MySQL supports a large number of embedded applications which makes MySQL very flexible.
- **High Productivity:** MySQL uses Triggers, Stored procedures and views which allows the developer to give a higher productivity.

The **drawbacks** of MySQL are given below:

- MySQL version less than 5.0 doesn't support ROLE, COMMIT and stored procedure.
- MySQL does not support a very large database size as efficiently.
- MySQL doesn't handle transactions very efficiently and it is prone to data corruption.
- MySQL is accused that it doesn't have a good developing and debugging tool compared to paid databases.
- MySQL doesn't support SQL check constraints.

4.2 Integrating the Website and Database

Customers ordering from an e-commerce website need to be able to get information purchase, and submit payment information. Vendors need to be able to track customer inquiries and preferences and process their orders. So a well organized database is essential for the development and maintenance of an e-commerce site.

In a static Web page, content is determined at the time when the page is created. As users access a static page, the page always displays the same information. Example of a static Web page is the page displaying company information. In a dynamic Web page, content varies based on user input and data received from external sources. We use the term “data-based Web pages” to refer to dynamic Web pages deriving some or all of their content from data files or databases.

A data-based Web page is requested when a user clicks a hyperlink or the submit button on a Web page form. If the request comes from clicking a hyperlink, the link specifies either a Web server program or a Web page that calls a Web server program.

In some cases, the program performs a static query, such as “Display all items from the Inventory”. Although this query requires no user input, the results vary depending on when the query is made. If the request is generated when the user clicks a form’s submit button, instead of a hyperlink, the Web server program typically uses the form inputs to create a query. For example, the user might select five books to be purchased and then submit the input to the Web server program.

The Web server program then services the order, generating a dynamic Web page response to confirm the transaction. In either case, the Web server is responsible for formatting the query results by adding HTML tags.

4.3 Know Your Expense Application

The objective of this application is to provide the user an online website where they can add their expenses. User can add their daily expenses which they spend on food, electricity bill, water bill, hotel bills etc. User can also delete their expenses. Know Your Expense aims to helps everyone to track their expenses and user can view their expenses on day wise basis, weekly basis, month wise basis and year wise basis according to periods of time. User can also view their overall expenses.

Website consists of the following web pages:

1. add-expense.php
2. change-password.php
3. dashboard.php
4. expense-datewise-reports.php
5. expense-datewise-reports-detailed.php
6. expense-monthwise-reports.php
7. expense-monthwise-reports-detailed.php
8. expense-reports.php
9. expense-reports-detailed.php
10. expense-yearwise-reports.php
11. expense-yearwise-reports-detailed.php
12. forgot-password.php
13. index.php
14. logout.php
15. manage-expense.php
16. register.php
17. reset-password.php
18. user-profile.php

CHAPTER 5

TESTING AND RESULTS

MODULE TESTED	EXPECTED RESULT	FINAL RESULT
User Login with incorrect credentials	Message showing invalid details	Success
User Login with correct credentials	Redirect to users dashboard	Success
Adding zero amount in expense	Total amount is unchanged	Success
Deleting an item	Alert message indicating record successfully deleted	Success
Updating users details	Message showing user profile has been updated	Success
Changing login password	Message showing your password successfully changed	Success

Table 5.1: Test Case

CHAPTER 6

CONCLUSION

Know Your Expense application is successfully designed. After making this application we assure that this application will help its users to manage the cost of their daily expenditure. It will guide them and aware them about their daily expenses. It will prove to be helpful for the people who are frustrated with their daily budget management, irritated because of amount of expenses and wishes to manage money and to preserve the record of their daily cost which may be useful to change their way of spending money. In short, this application will help its users to overcome the wastage of money. Know Your Expense is usable by anyone who are willing to manage their expenses and aiming to save for the future investments. This application has no range criteria or any kind of profession or gender are focused so it will used hugely by any other person.

- Login through their email id
- Add your expenses
- Management of user expense

SCOPE OF ENHANCEMENT

- Know Your Expense System currently designed as a Desktop Application. which can be further developed as mobile app Application. Further days, there will be mails and pay mode embedded with the application. Also, backup details will be recorded on cloud

REFERENCES

Books

[1] Matthew MacDonald, "Creating a Website - The Missing Manual", 3rd ed, 2011, O'Reilly.

[2] Matthew MacDonald, "HTML 5 - The Missing Manual", 2nd ed, 2014, O'Reilly.

[3] David Sawyer McFarland, "CSS 3 - The Missing Manual", 3rd ed, 2013, O'Reilly.

[4] David Sawyer McFarland, "JavaScript and jQuery - The missing manual", 3rd ed, 2014, O'Reilly.

[5] Jonathan Chaffer and Karl Swede bery, "Learning jQuery", 4th ed, 2013, Packt Publishing.

[6] Codd E. F., "A Relational Model of Data for Large Shared Data Banks", Communications of the ACM, vol. 13, issue 6, pp. 377–387, June 1970.

Website

[1] Modern JavaScript from the beginning on www.udemy.com

[2] HTML 5, CSS 3, BootStrap 4, SQL and PHP on www.w3schools.com

[3] Project Repositories on www.github.com and www.stackoverflow.com

[4] developer.mozilla.org

[5] www.coursera.org

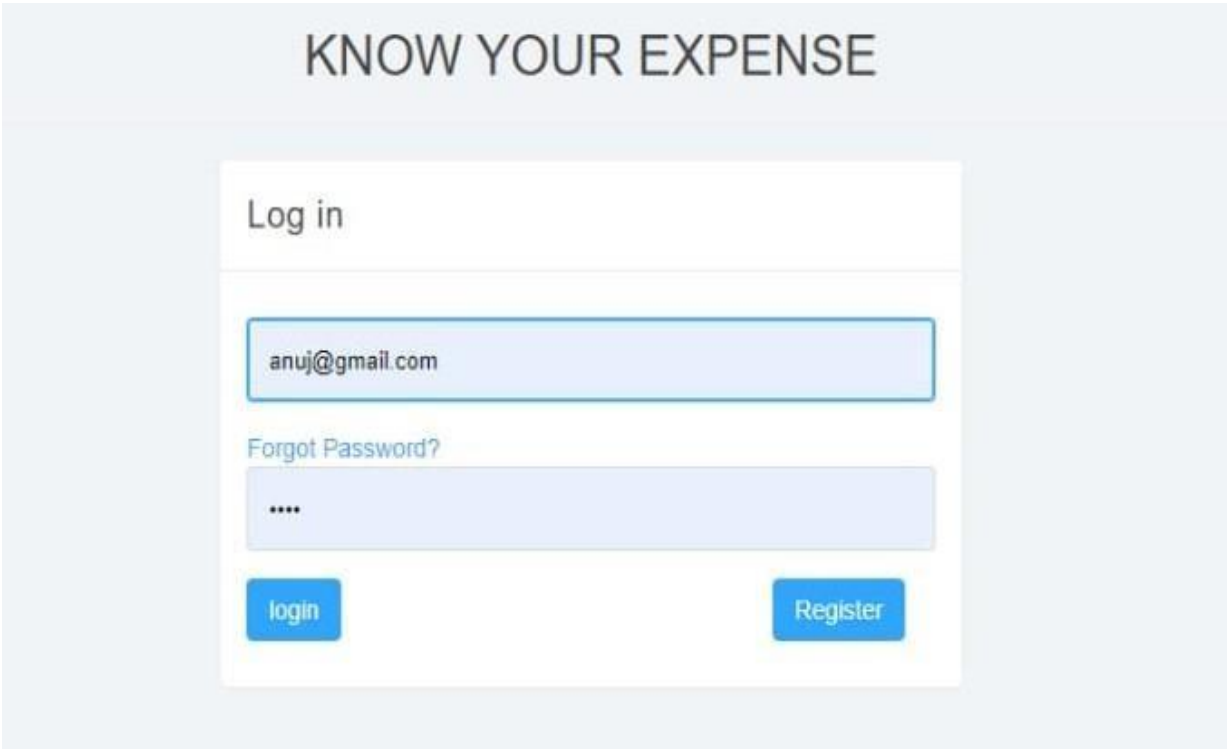
[6] www.youtube.com

[7] www.codingcyber.org

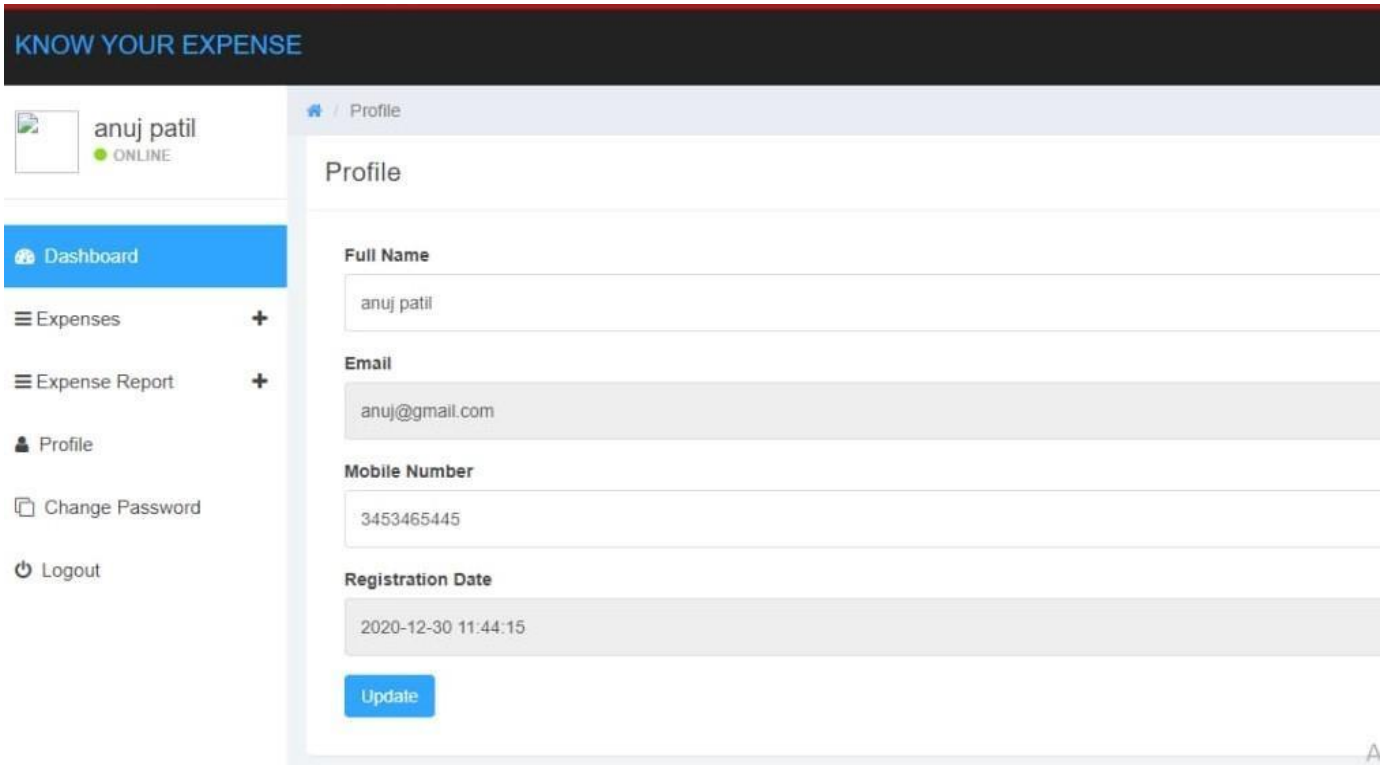
[8] www.wikipedia.org

Appendix A

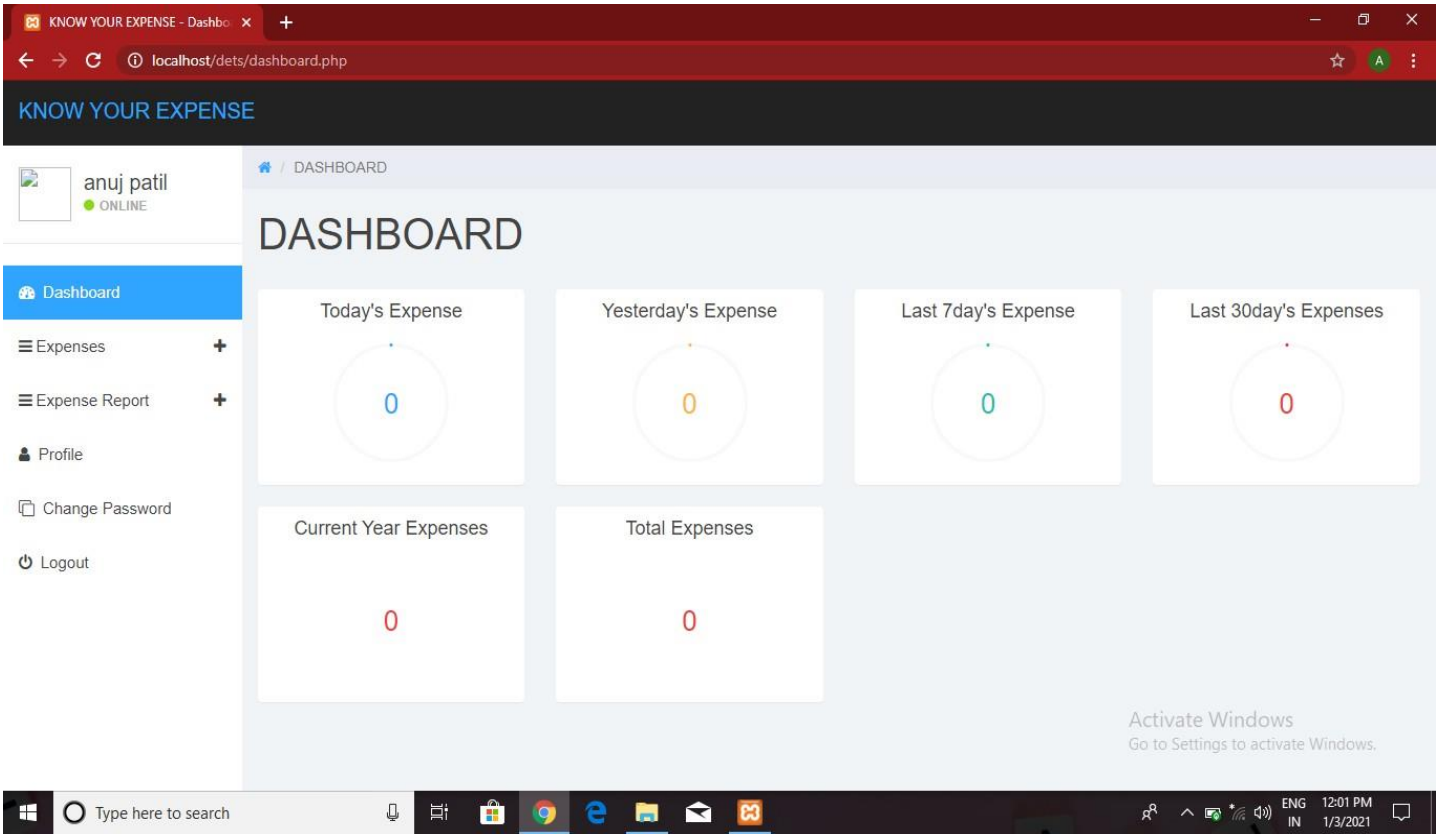
SNAPSHOTS



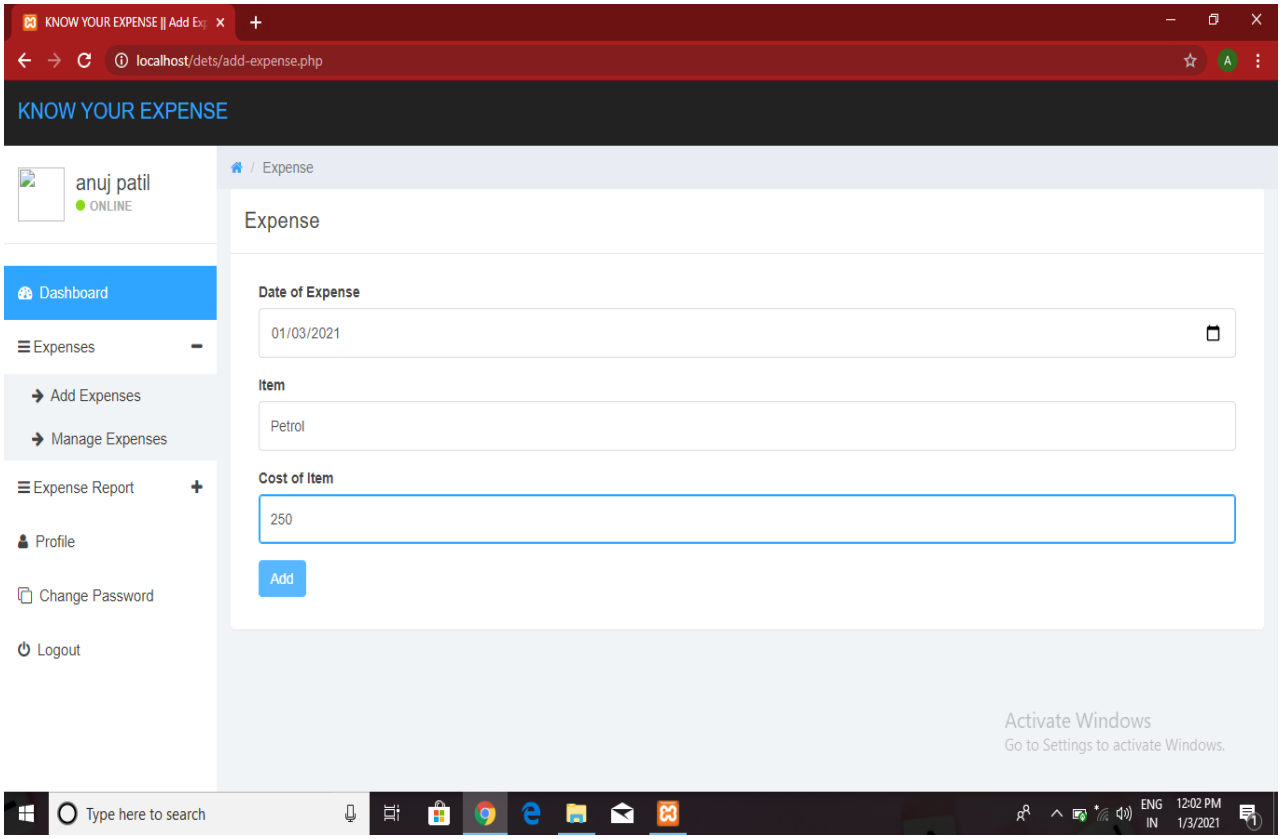
Snapshot 1: Start page of the project



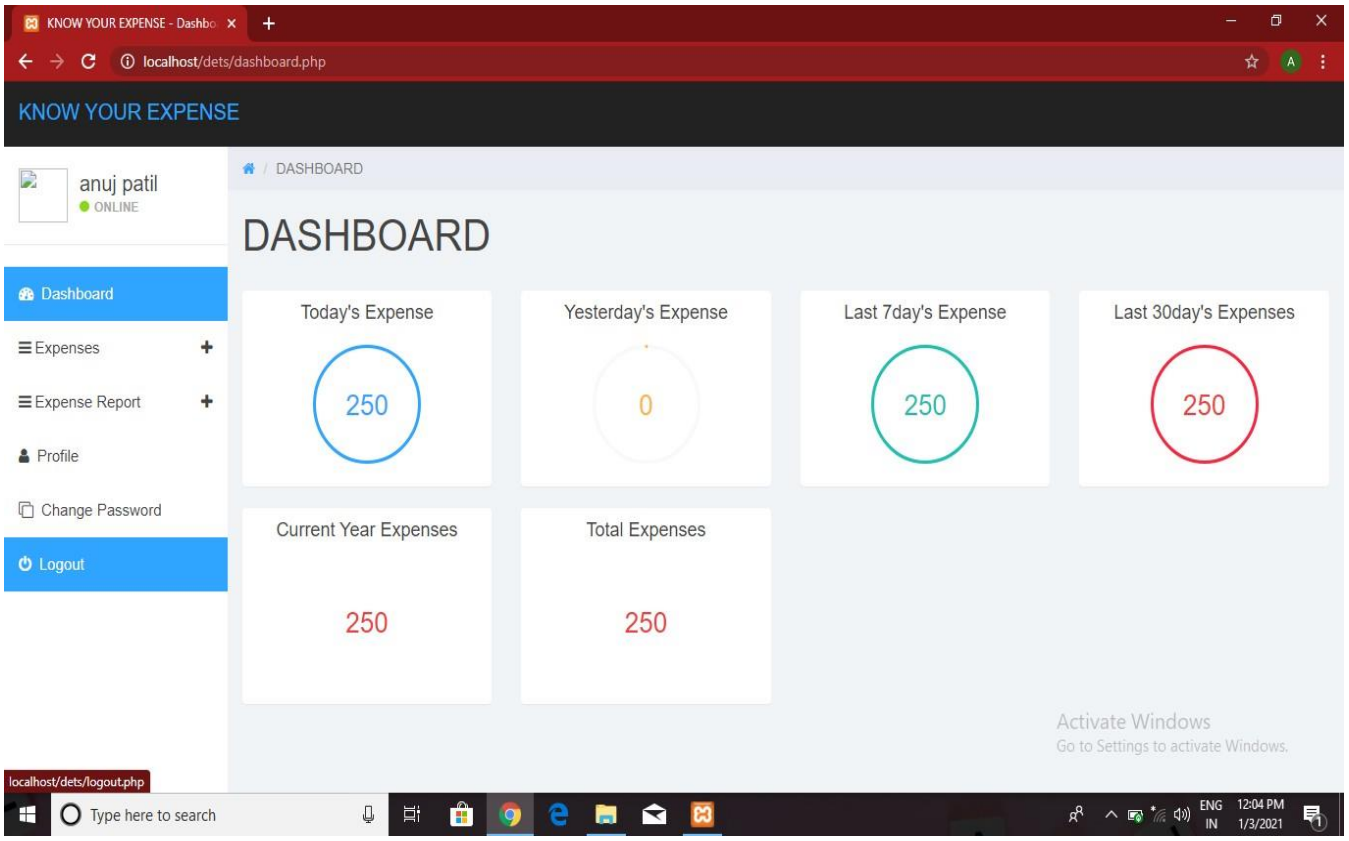
Snapshot 2: create account page of the project



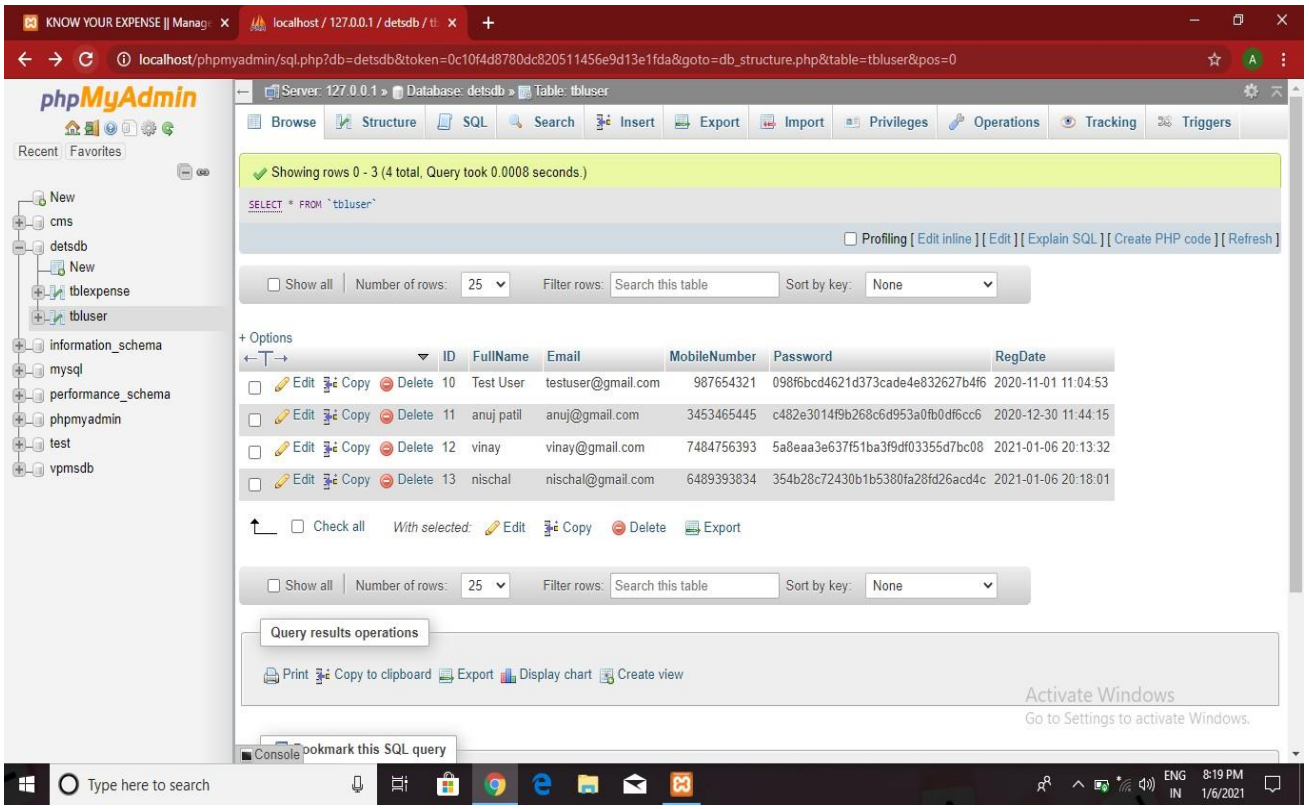
Snapshot 3: Dashboard of the project



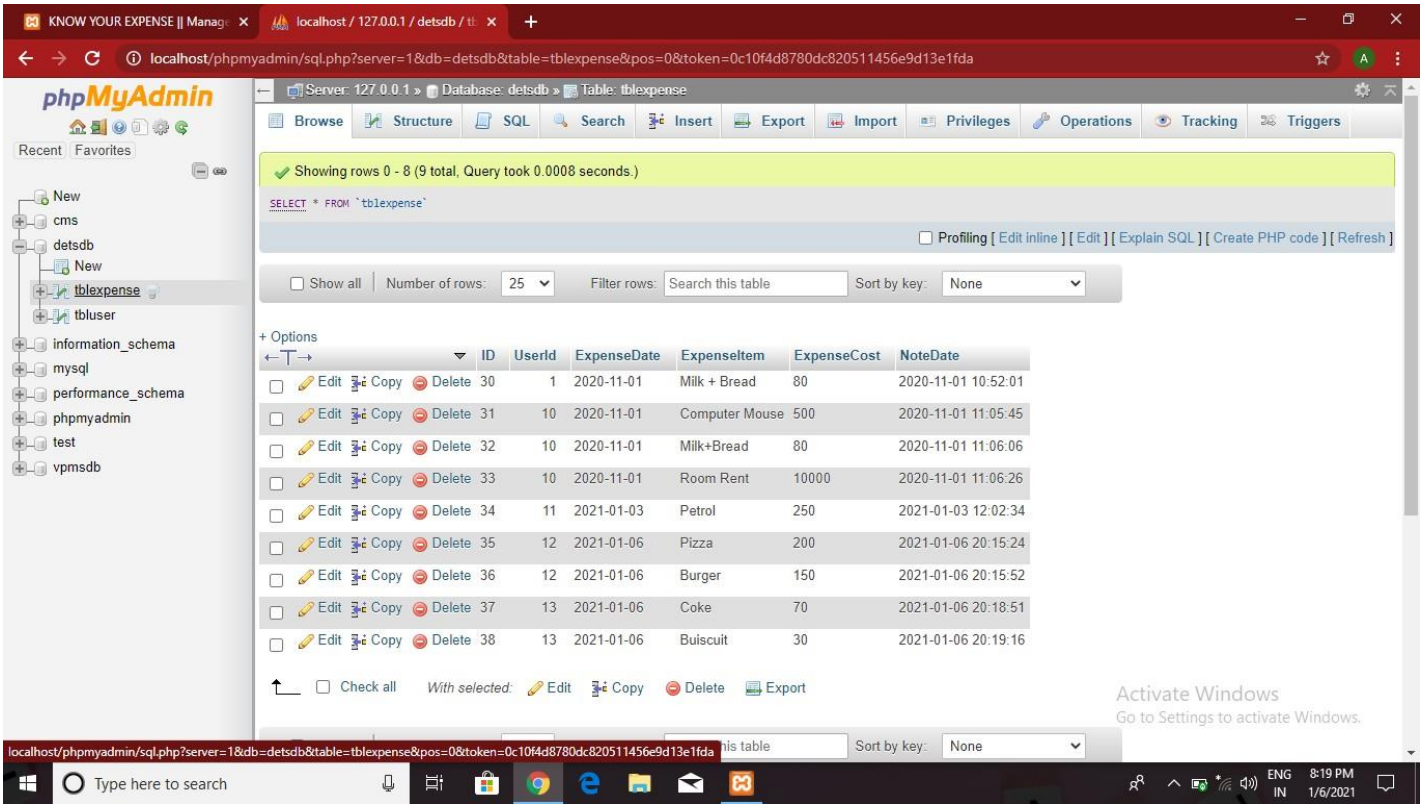
Snapshot 4: User Expense page of the project



Snapshot 5: User's expense details in dashboard



Snapshot 6: user details in database



Snapshot 7: User's expense details in database