VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi - 590 018

A Mini -Project Work on

SPLITTER

A Dissertation work submitted in partial fulfillment of the requirement
for the award of the degree

Bachelor of Engineering
In
Information Science & Engineering

Submitted by
RISHABH DUBEY - 1AY19IS076

Under the guidance of
Prof. SUBHASH KAMBLE
ASSISTANT PROFESSOR

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

ACHARYA INSTITUTE OF TECHNOLOGY

(AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI.APPROVED BY AICTE, NEW DELHI, ACCREDITED BY NAAC, NEW DELHI)

Acharya Dr. Sarvepalli Radhakrishnan Road, Soldevanahalli, Bengaluru-560107

2021-22

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

ACHARYA INSTITUTE OF TECHNOLOGY

(AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI.APPROVED BY AICTE, NEW DELHI, ACCREDITED BY NAAC, NEW DELHI)

Acharya Dr. Sarvepalli Radhakrishnan Road, Soldevanahalli, Bengaluru-560107

ACHARYA

Certificate

This is to Certify that the Mini-Project work entitled SPLITTER is a bonafide work carried out by **RISHABH DUBEY (1AY19IS076)** in partial fulfillment for the award of the degree of **Bachelor of Engineering** in **Information Science and Engineering** of the **Visvesvaraya Technological University**, Belagavi during the year 2021-22. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The Project has been approved as it satisfies the academic requirements in respect of Project work prescribed for the Bachelor of Engineering Degree.

Prof. Subhash Kamble
Guide

Prof. Marigowda C K
HOD

Name of the Examiners **Signature with date**

1. ______________________ ______________________

2. ______________________ ______________________

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of this mini-project would be incomplete without the mention of the people who made it possible through constant guidance and encouragement.

We would take this opportunity to express our heartfelt gratitude to **Sri. B. Premnath Reddy**, Chairman, Acharya Institutes, and **Dr. Prakash M R**, Principal, Acharya Institute of Technology for providing the necessary infrastructure to complete this mini-project.

We wish to express our deepest gratitude and thanks to **Prof. Marigowda C K**, Head of the Department, Information Science, and Engineering.

We wish to express sincere thanks to our guide **Prof. Subhash Kamble** ,Assistant professor, Department of Information Science and Engineering for helping us throughout and guiding us from time to time.

A warm thanks to all the faculty of the Department of Information Science and Engineering, who have helped us with their views and encouraging ideas.

RISHABH DUBEY-1AY19IS076

ABSTRACT

Splitter is a bill splitting web application. It allows the user to maintain, view, edit and track the expenses made, this allows for hassle free maintenance of transactions and easy settlement.

The aim of Splitter is to automate the existing manual system with the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required hardware and software are easily available and easy to work with.Users can create own projects and add their expenditure for the group expenses, the description of the expense, so that while going through the process of settlement of the bills a clear description will also be provided as a reference to other members of the group.

TABLE OF CONTENTS

TABLE OF FIGURES

CHAPTER 1

INTRODUCTION

It is common for a group of friends be to go on a vacation or a trip these days. Most common problem is that it becomes difficult to identify which person owes whom how much. Even they are friends if the amount is large, it can't be ignored. That is where SPLITTER comes into picture and tells which person owes who how much.

1.1 Introduction to DBMS

DBMS stands for **D**ata**b**ase **M**anagement **S**ystem. We can break it like this DBMS = Database + Management System. The database is a collection of data and a Management System is a set of programs to store and retrieve those data. Basically, DBMS is a software tool to organize (create, retrieve, update and manage) data in a database.

The main aim of a DBMS is to supply a way to store up and retrieve database information that is both convenient and efficient. By data, we mean known facts that can be recorded and that have embedded meaning. Normally people use software such as DBASE IV or V, Microsoft ACCESS, or EXCEL to store data in the form of a database. A datum is a unit of data. Meaningful data combined to form information. Hence, information is interpreted data – data provided with semantics. MS. ACCESS is one of the most common examples of database management software.

Database systems are meant to handle large collections of information. Management of data involves both defining structures for the storage of information and providing mechanisms that can do the manipulation of those stored information. Moreover, the database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access.

1.1.1 Why DBMS?

- To develop software applications in less time.
- Data Independence and efficient use of data.
- For uniform data administration.
- For data integrity and security.
- For concurrent access of data, and data recovery from crashes.
- To use user-friendly declarative query language.

1.1.2 Database applications

- **Telecom:** There is a database to keep track of the information regarding calls made, network usage, customer details, etc. Without the database systems, it is hard to maintain that huge amount of data that keeps updating every millisecond.

- **Industry:** Where it is a manufacturing unit, warehouse, or distribution center, each one needs a database to keep the records of ins and outs. For example, distribution centers should keep a track of the product units that are supplied into the center as well as the products that got delivered out from the distribution center on each day; this is where DBMS comes into the picture.

- **Education sector:** Database systems are frequently used in schools and colleges to store and retrieve the data regarding student details, staff details, course details, exam details, payroll data, attendance details, fees detail etc. There is a hell lot of interrelated data that needs to be stored and retrieved efficiently.

- **Online shopping:** You must be aware of the online shopping websites such as Amazon, Flipkart etc. These sites store the product information, your addresses and preferences, credit details and provide you the relevant list of products based on your query. All this involves a Database management system.

- **Banking system:** For storing customer info, tracking day to day credit and debit transactions, generating bank statements etc. All this work has been done with the help of Database management systems.

1.1.3 Advantages of DBMS

A DBMS manages data and has many advantages.

- **Data Independence:** Application programs should be as free or independent as possible from details of data representation and storage. DBMS can supply an abstract view of the data for insulating application code from such facts.

- **Efficient data access:** DBMS utilizes a mixture of sophisticated concepts and techniques for storing and retrieving data competently and this feature becomes important in cases where the data is stored on external storage devices.

- **Data integrity and security:** If data is accessed through the DBMS, the DBMS can enforce integrity constraints on the data.

- **Data administration:** When several users share the data, integrating the administration of data can offer major improvements. Experienced professionals understand the nature of the data being managed and can be responsible for organizing the data representation to reduce redundancy and make the data to retrieve efficiently.

- **Providing backup and recovery:** A DBMS must provide facilities for recovering from hardware or software failures. The backup and recovery subsystem of the DBMS is responsible for recovery.

- **Permitting inferencing and actions using rules:** Some database systems provide capabilities for defining deduction rules for inferencing new information from the stored database facts.

1.1.4 Components of DBMS

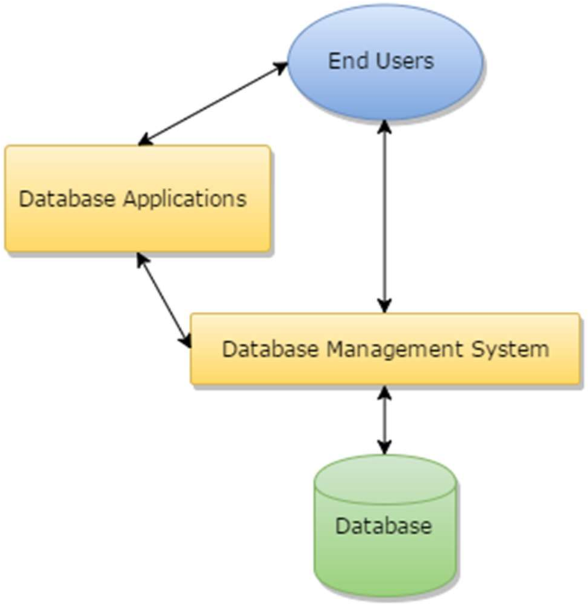


Fig-1.1: Components of a Database Management System

- **Users:** Users may be of any kind such as DB administrator, System developer or database users.

- **Database application:** Database application may be Departmental, Personal, organization's and / or Internal.

- **DBMS:** Software that allows users to create and manipulate database access.

- **Database:** Collection of logical data as a single unit.

- **Database access language:** This is used to access the data to and from the database, to enter new data, update existing data, or retrieve required data from databases. The user writes a set of appropriate commands in a database access language, submits these to the DBMS, which then processes the data and generates and displays a set of results into a user readable form.

1.1.5 Three-Schema architecture

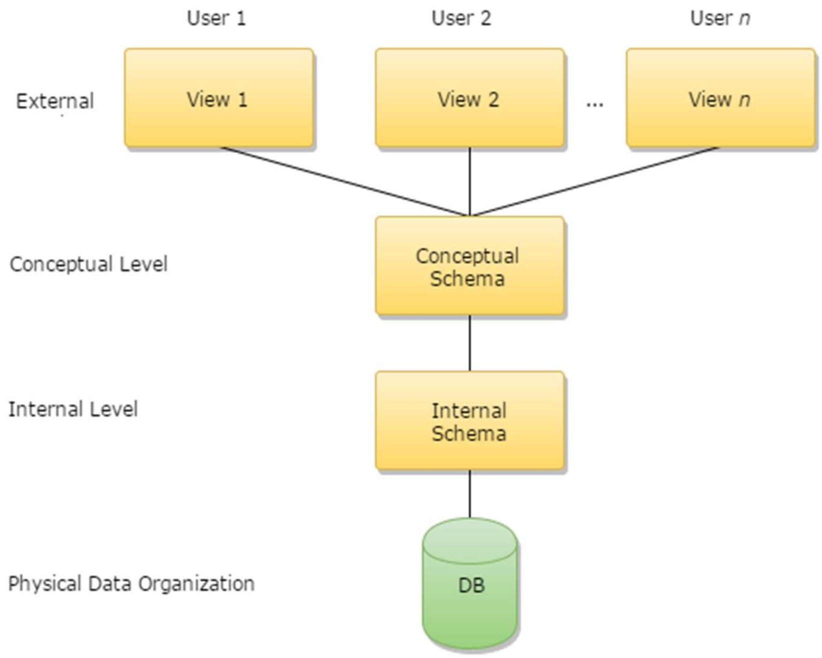


Fig-1.2: Architecture of Database System

The levels form a three-level architecture that includes an external, a conceptual, and an internal level. The way users recognize the data is called the external level. The way the DBMS and the operating system distinguish the data is the internal level, where the data is actually stored using the data structures and file. The conceptual level offers both the mapping and the desired independence between the external and internal levels.

CHAPTER 2

SYSTEM REQUIREMENTS

2.1 Hardware Requirements

- **Processor:** Intel Core2 Quad @ 2.4Ghz on Windows® Vista 64-Bit / Windows® 7 64-Bit / Windows® 8 64-Bit / Windows® 8.1 64-Bit.
- **RAM:** 2GB of RAM
- **Memory:** 256GB Hard drive
- **Keyboard:** MS-compatible keyboard
- **Mouse:** MS-compatible mouse

2.2 Software Requirements

- **Operating system:** Windows® Vista 64-Bit / Windows® 7 64-Bit / Windows® 8 64-Bit / Windows® 8.1 64-Bit.
- **Front end:** HTML, CSS, BOOTSTRAP
- **Back end:** PHP, XAMPP
- **IDE:** VS Code

CHAPTER 3

DESIGN

3.1 ER Diagram

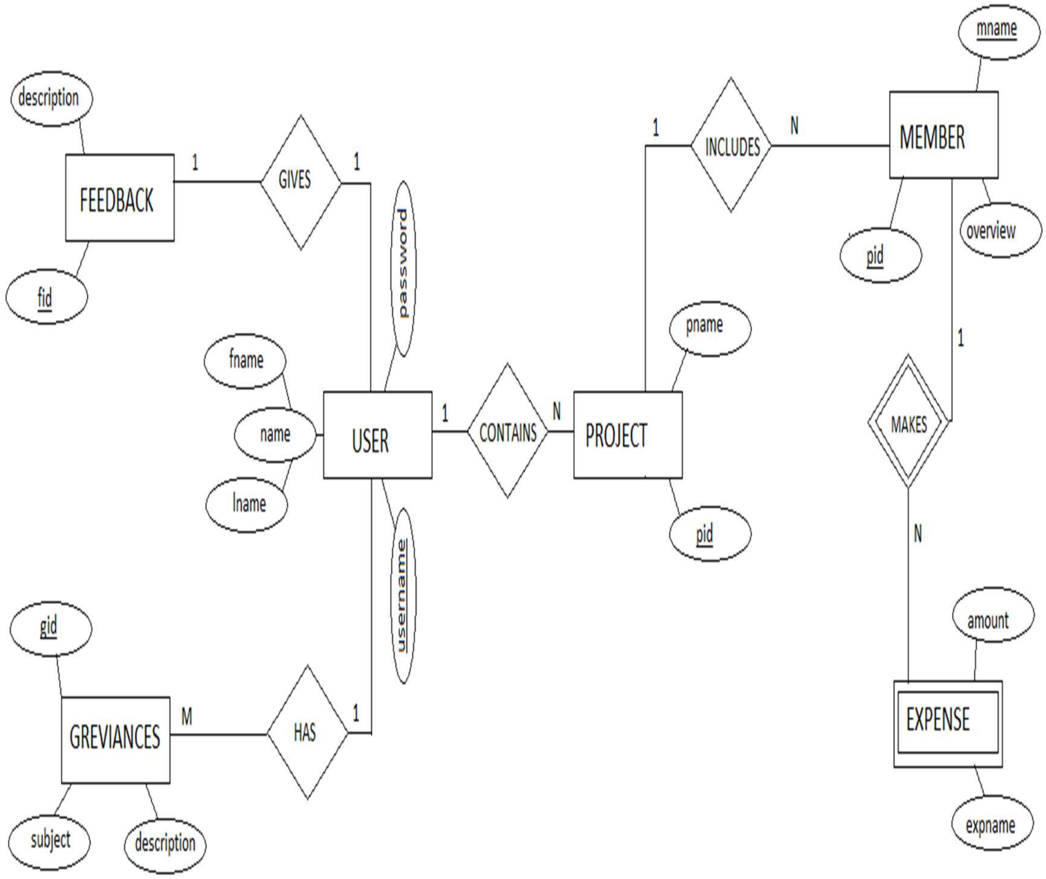


Fig-3.1: Entity Relationship Diagram

1: N

- One user contains many projects.
- One project can have many members.
- One member can have many expenses.
- One user can have many greviances.

1:1

- One user can give only one feedback.

3.2 Schema Diagram

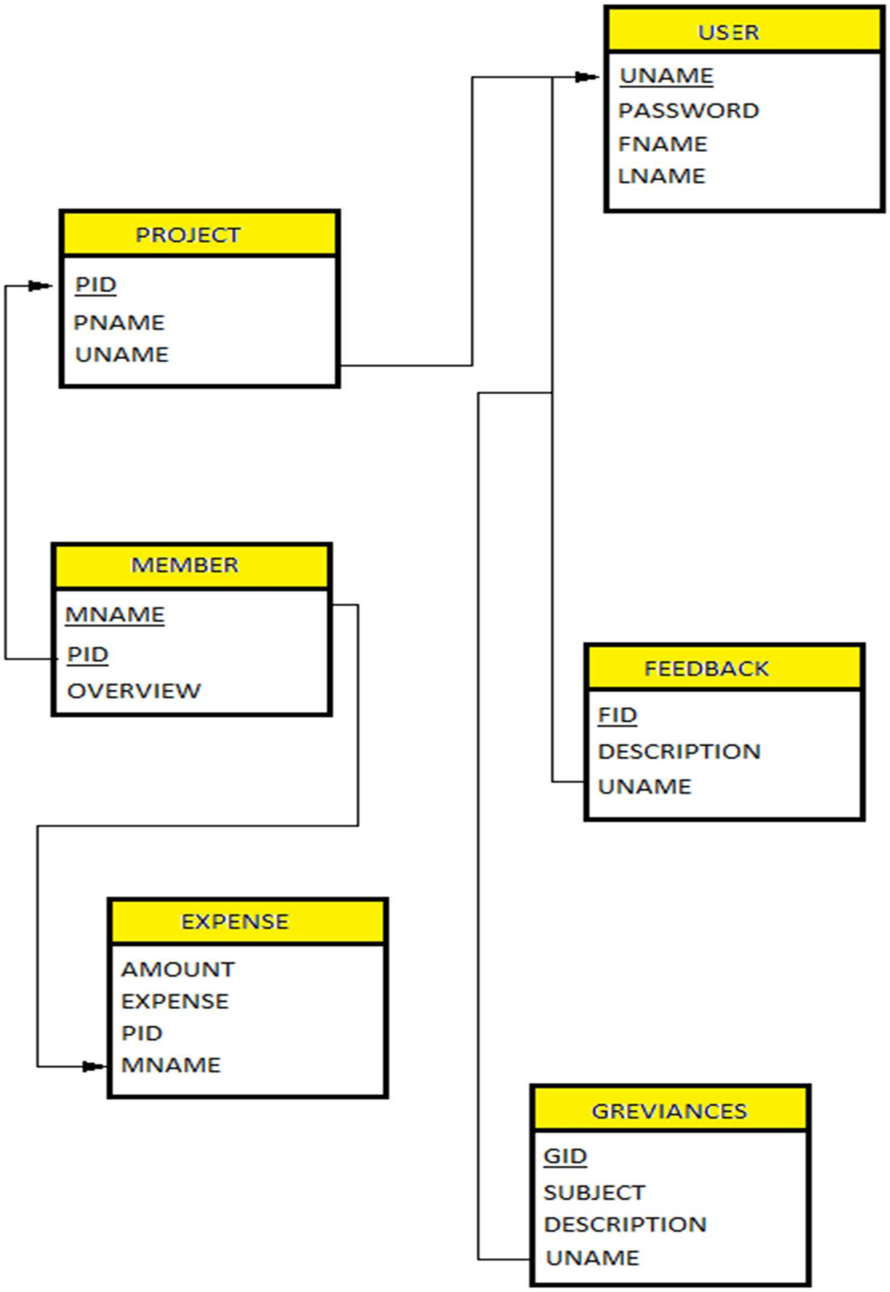


Fig-3.2: Schema Diagram

CHAPTER 4

IMPLEMENTATION

4.1 Tables

4.1.1 USER

Entity name	**ATTRIBUTES**	**CONSTRAINTS**
USER	username	Primary key, not Null
	fname	Not null
	lname	Not null
	password	Not null

4.1.2 PROJECT

Entity name	**ATTRIBUTES**	**CONSTRAINTS**
PROJECT	pid	Primary key, not Null, Auto increment
	uname	FK
	pname	Not null

4.1.3 MEMBER

Entity name	**ATTRIBUTES**	**CONSTRAINTS**
MEMBER	mname	Primary key, not Null
	pid	FK
	overview	Not null

4.1.4 EXPENSE

Entity name	**ATTRIBUTES**	**CONSTRAINTS**
EXPENSE	expname	NOT Null
	mname	NOT Null
	amount	NOT Null
	pid	FK

4.1.5 GREVIANCE

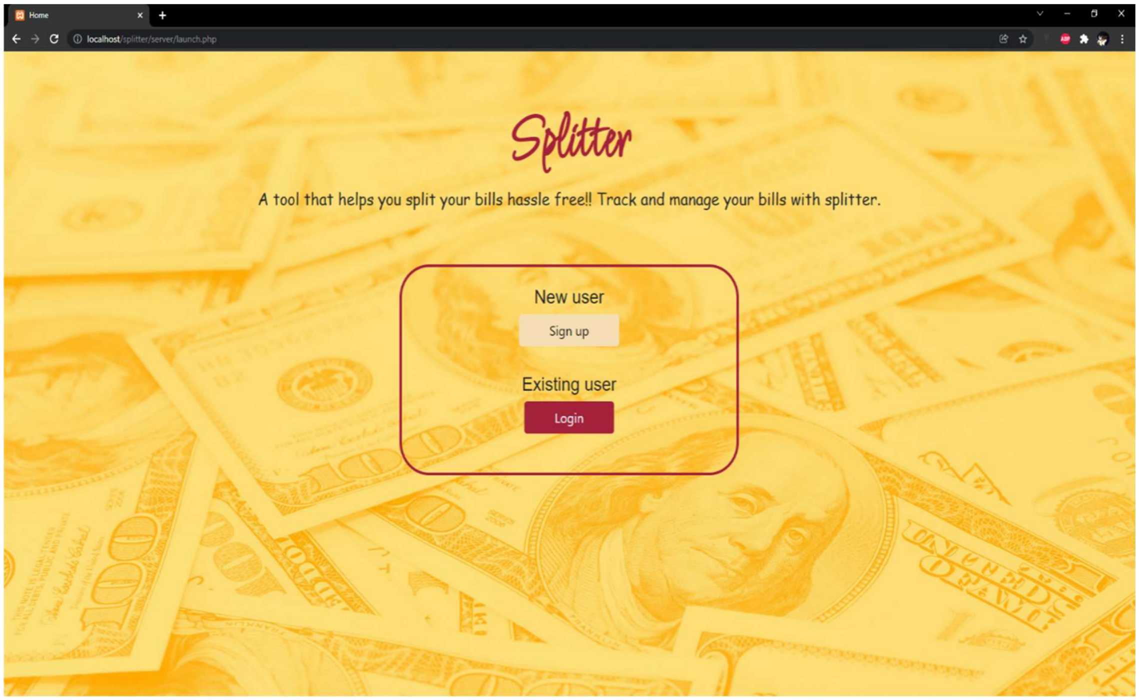
Entity name	**ATTRIBUTES**	**CONSTRAINTS**
GREVIANCE	gid	Primary key, not Null, Auto increment
	username	FK
	subject	Not null
	greviance	Not null

4.1.6 FEEDBACK

Entity name	**ATTRIBUTES**	**CONSTRAINTS**
FEEDBACK	fid	Primary key, not Null
	uname	FK
	feedback	Not null

CHAPTER 5

SNAPSHOTS

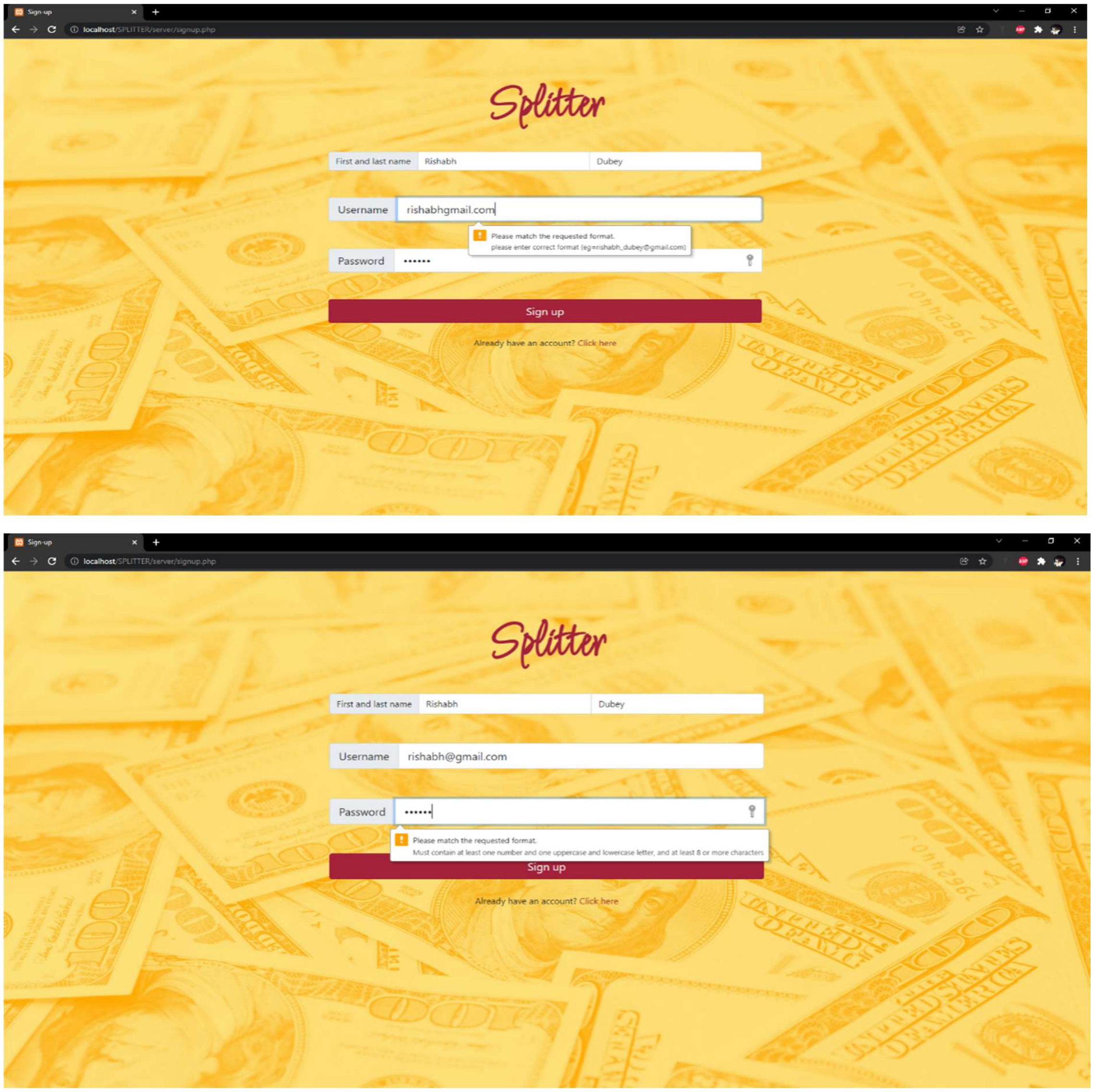


A tool that helps you split your bills hassle free!! Track and manage your bills with splitter.

Fig-5.1: Snapshot of Root Page/Welcome Page

The first page is root page or the launch page which will allow the user to either sign up or login via the two-option button provided. The following snapshot contains the root page or launch page

- The welcome screen contains a header describing the use of the application.
- It contains the link for new user which will redirect them to the signup page.
- It contains the link for existing user which will redirect them to the login page.


Sign-up
localhost/SPLITTER/server/signup.php
Splitter
Sign up
Already have an account? Click here

First and last name: Rishabh | Dubey

Username: rishabhgmail.com

Please match the requested format.
please enter correct format (eg=rishabh_dubey@gmail.com)

Password: ••••••

First and last name: Rishabh | Dubey

Username: rishabh@gmail.com

Password: ••••••

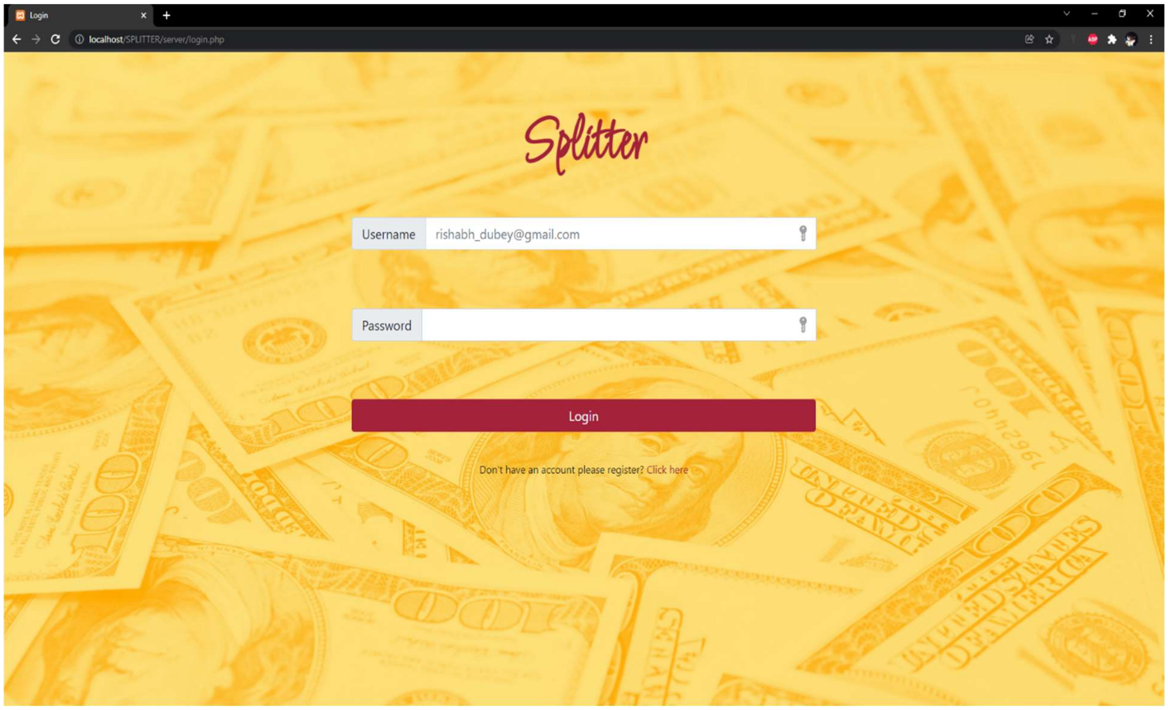
Please match the requested format.
Must contain at least one number and one uppercase and lowercase letter, and at least 8 or more characters

Fig-5.2: Snapshot of Sign-up Page

The new user will be redirected to this page after selecting the option of signup in the launch page. User needs to enter his first name, last name, email and a password to register.

Validators given

- Email validation is given by a validating with a regex expression pattern="[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,}$".
- The warnings are based on the number of uppercase letters, lowercase letters, numbers and special characters.
- If the user already has an account, then there is a link for sign in.



Login
localhost/SPLITTER/server/login.php
Splitter
Login
Don't have an account please register? Click here

Username: rishabh_dubey@gmail.com

Password:

Fig-5.3: Snapshot of Sign-in Page

Once the user has successfully registered, he can login with his register username and password.

Validators given

- A user will be able to sign in only when he has been registered.
- When he completes the registration Signed Up user will be redirected to this page giving a prompt of Successfully registered.
- If either the password or the username mismatches it show a message as INVALID CREDENTIALS.
- If the user has not registered, he can click the link given at the bottom which will be redirected to the signup page.
- After the completion of the Sign in and sign up the user will be redirected to the home page or profile page.

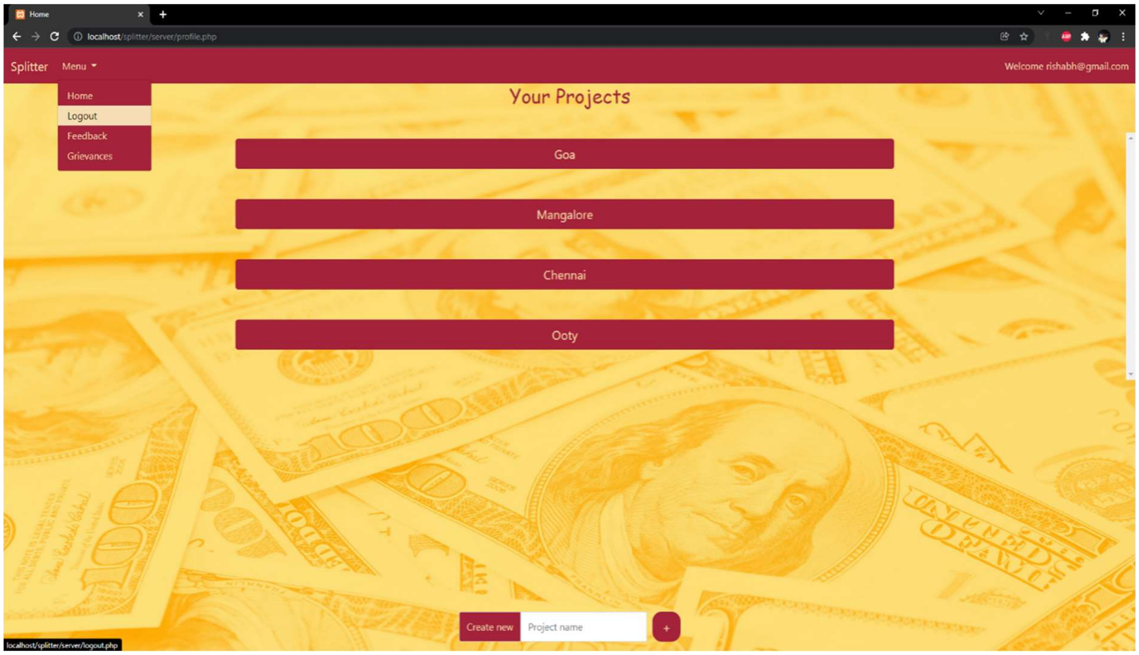


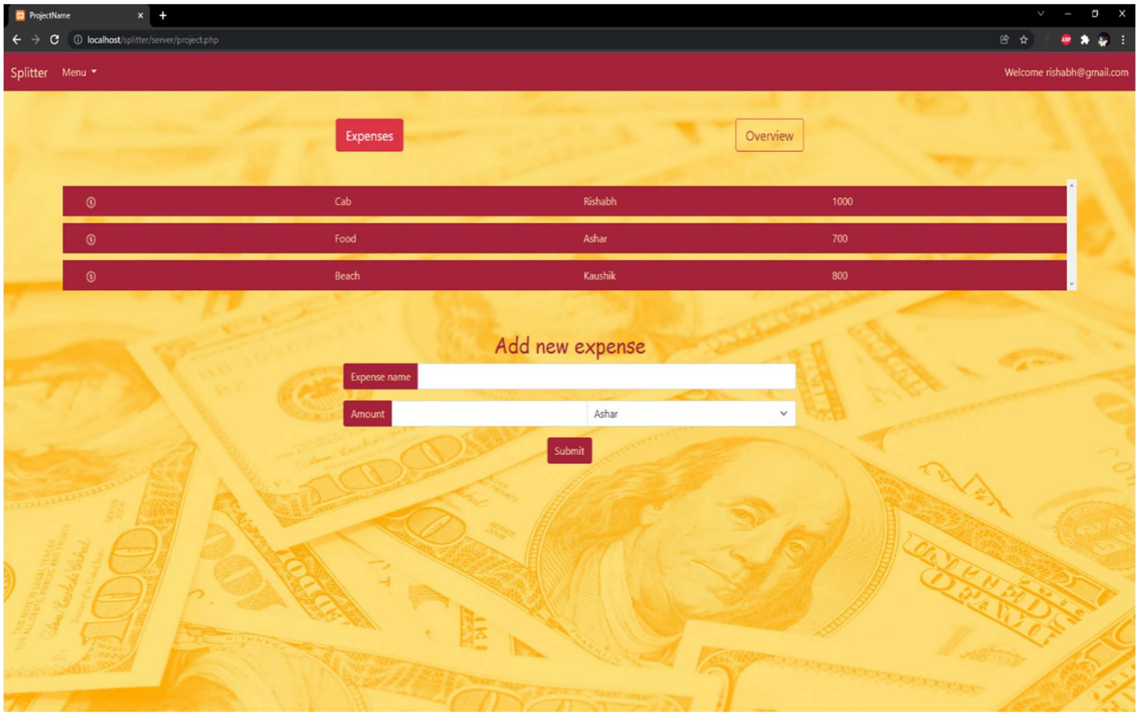
Fig-5.4: Snapshot of Home /Profile Page

The above snapshot is seen as soon as the user registers for SPLITTER. The above page will show all the projects that the particular user has created (Here project means nothing but the different places that user has travelled and want to keep the records of the expenses).

- It contains a nav which from now onwards will be available for the user to access which contains following sections.
    - **Home** which will redirect the user to profile page or this same page.
    - **Logout** which will end the current session and will redirect the user to login page.
    - **Feedback** which will redirect the user to the feedback page.
    - **Grievances** which will redirect the user to the grievance page.
- The top left sections welcome the user by showing the text welcome and logged in username.
- It contains all the projects links which will redirect the user to the particular project session.
- At the bottom contains a field which allows a user to create a new project.

Validators given

- The following page will not be rendered if the user is not successfully logged in.


ProjectName
localhost/splitter/server/project.php
Splitter
Menu
Welcome rishabh@gmail.com
Expenses
Overview
Submit

$	Cab	Rishabh	1000
$	Food	Ashar	700
$	Beach	Kaushik	800

Add new expense

Expense name

Amount Ashar

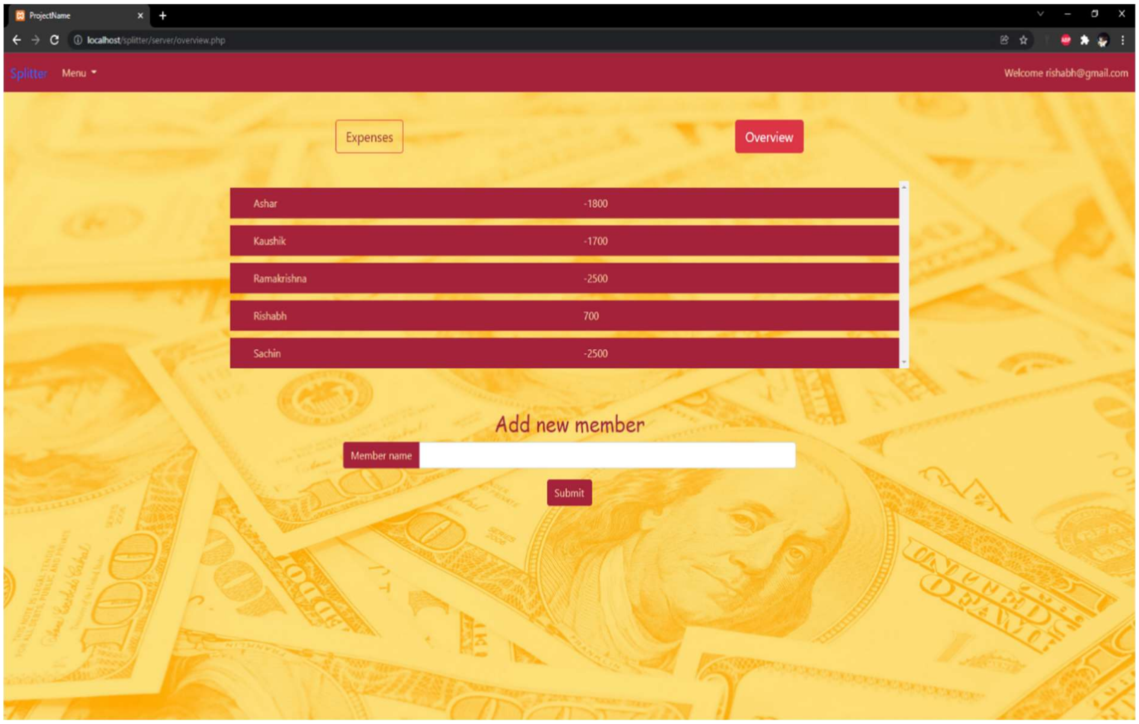
Fig-5.5: Snapshot of Project/Expense Page

The is a snapshot of the project page which tells the events of expenses occurred during travelling of the particular project.

- The top of the page contains two buttons which allows us to switch between two pages which are expenses and overview.
- We are currently at the expense which tell us the expense events that has already occurred. Which contains the following details.
    - Expense name.
    - Name of the member who paid the amount for the event.
    - Amount that has been paid.
- The bottom contains a field which allows the user to add a new event by the above-mentioned details.

Validators given

- The following page will not be rendered if the user has not successfully logged in.
- Only the members of the project are allowed to take part in expenses.



Ashar	-1800
Kaushik	-1700
Ramakrishna	-2500
Rishabh	700
Sachin	-2500

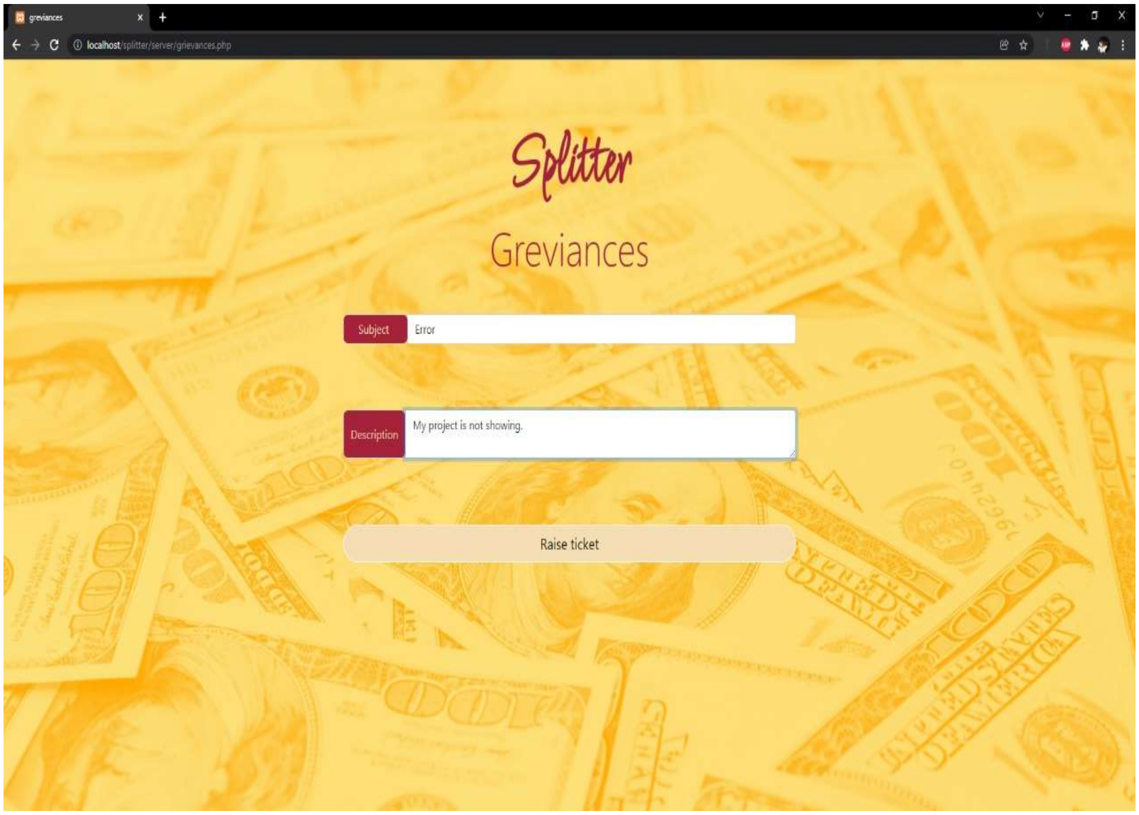
Fig-5.6: Snapshot of Overview Page

The is a snapshot of the overview page which is the main part of the application which splits the bills among the members currently present in the project.

- A table is used in order to show the members present in the project with the following details.
    - Name of the member.
    - Amount he/she has to take or pay. (Eg- 1000 means he will get 1000 Rs from the group in total, -1500 means he will pay 1500 Rs to the group in total).
- The bottom contains a field which is used to add a new member in the project. (The moment the member is joined from that point the expenses will be added for him).

Validators given

- The following page will not be rendered if the user has not successfully registered.
- Only the members of the project are allowed to take part in expenses.



Subject: Error

Description: My project is not showing.

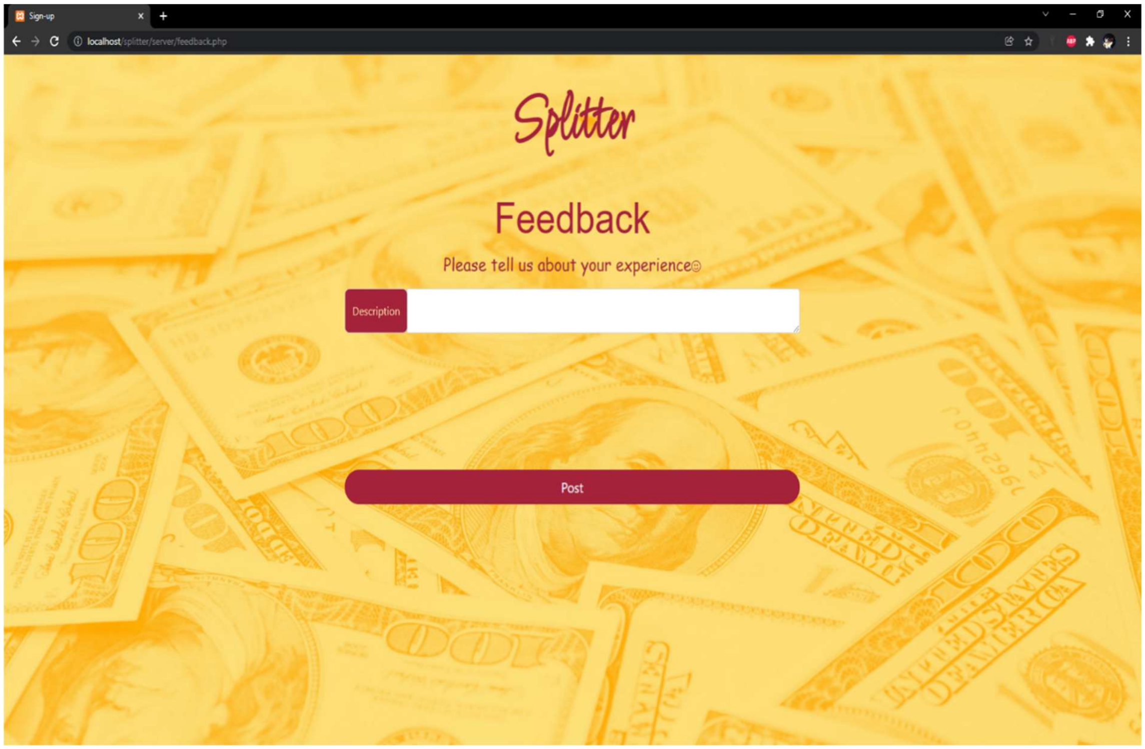
Fig-5.7: Snapshot of Grievances Page

In case of any discrepancies the user is allowed to raise a ticket corresponding to the ambiguity regarding to his/her data.

- When a user selects the grievance button on the navigation bar user will be directed to this page.
- The grievance of the user is stored in the database once the user raises the ticket.
- It contains the following contents:
  - Subject
  - Description
  - A button to raise ticket

Validators Given:

- This page will not be rendered if the user is not successfully logged in.



Splitter

Feedback

Please tell us about your experience☺

Fig-5.8: Snapshot of Feedback

For feedback the user needs to enter the description and post his/her feedback.

- When a user selects the feedback button on navigation bar user will be directed to this page.
- It contains a description column for a user to enter the feedback about the experience of the webapp.

Validators given

- This page will not be rendered if the user has not successfully logged in.
- Only one feedback per user is allowed.

TRIGGERS

A SQL trigger is a database object which fires when an event occurs in a database. We can execute a SQL query that will "do something" in a database when a change occurs on a database table such as a record is inserted or updated or deleted. For example, a trigger can be set on a record insert in a database table.

Trigger for storing the username and first name as a backup

```
CREATE TRIGGER REGISTER_LOG
AFTER INSERT ON USER
FOR EACH ROW
WHEN
INSERT INTO signup_logs (username, name) VALUES (NEW.username, NEW.fname);
```

PROCEDURES

A stored procedure is a set of Structured Query Language (SQL) statements with an assigned name, which are stored in a relational database management system (RDBMS) as a group, so it can be reused and shared by multiple programs.

Procedure for showing all the projects of the particular user.

```
CREATE PROCEDURE getprojects @username VARCHAR(30)
AS
SELECT * FROM PROJECT WHERE UNAME=@username
GO;
```

CONCLUSION & FUTURE ENHANCEMENT

Conclusion

Splitter is a web application that helps people to track, view and edit their expenses which are made for a group of people.

As in a real-world scenario different users would pay for different expenses for the group if going out on a trip or maybe expenditure on some project, splitter helps to automate the manual process of calculation of each person's share and settlement of the expenditure.

Future Enhancement

The future enhancement of the project can be done in the following ways:

- Improvement of the user interface.
- Introduction of dynamic user database.
- Creating a view for the total expenses and each user's share as a pie chart.
- Addition of multiple currency and conversion support.
- Making a pdf for the settlement amounts which can be shared via social media.

The above-mentioned points are the future enhancement scope for the project.

BIBLIOGRAPHY

Web References

- https://www.w3schools.in/dbms/intro/
- https://getbootstrap.com/
- https://developer.mozilla.org/en-US/
- https://www.google.com/
- https://www.youtube.com/c/CodeWithHarry

Course References

- YouTube
- Web development bootcamp course

Book References

- Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, Pearson, 2017
- Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, McGraw Hill,2014
- Coronel, Morris, and Rob, Database Principles Fundamentals of Design, Implmentation and Management, Cengage Learning 2012