SplitBill - Expense Management system like splitwise

**PROJECT MANAGEMENT DOCUMENT**

*Version :* 3.0

*Date :* 29/04/2022

*Devesh Modi (201901173)*

*Khalsa Ravinder Singh (201901179)*

*Hastin Dave (201901225)*

*Srushti Menpara (201901307)*

*Devyanish Koul (201901444)*

Dhaval Vaidya(201901462) (Leader)

*Reporting TA :* Pinak Gajera

# **Activity List**

# Formulation Of the problem

* Estimated time : 15 hours
* Probable dates : 15 March - 05 April 2022

# Reading Relevant background information

* Estimated time : 05 hours
* Probable dates : 15 - 25 March 2022

# Understanding and documenting information

* Estimated time : 05 hours
* Probable dates : 26 March - 01 April 2022

# Discussions

* Estimates time : 05 hours
* Probable dates : 02 - 05 April 2022

# Designing a solution, documentation

* Estimated time : 12 hours
* Probable dates : 06 - 15 April 2022

# Relevant learning

* Estimated time : 30 hours
* Probable dates : 11 - 25 April 2022
* <https://www.youtube.com/playlist?list=PLu0W_9lII9ah7DDtYtflgwMwpT3xmjXY9>
* <https://www.geeksforgeeks.org/django-tutorial/>
* <https://www.geeksforgeeks.org/reactjs-tutorials/>

# Coding and unit testing

* Estimated time : 10 hours
* Probable dates : 21 April - 10 May2022

# Documentation

* Estimated time : 03 hours
* Probable dates : 11 - 12 May 2022

# Testing

* Estimated time : 08 hours
* Probable dates : 13 - 14 May 2022

# Reviews

* Estimated time : 05 hours
* Probable dates : 13 May 2022

# Re-work and debugging

* Estimated time : 04 hours
* Probable dates : 14 - 15 May 2022

# **Project plan**

| **Activity** | **Start date** | **End date** | **Person responsible** |
| --- | --- | --- | --- |
| Formulation of problem | 15 March | 5 April | All members |
| Designing a solution, documentation | 6 April | 15 April | Devesh Modi(201901173), Hastin Dave(201901225), Srushti Menpara(201901307) |
| Relevant learning | 11 April | 25 April | All members |
| Coding and unit testing | 21 April | 10 May | All members |
| Documentation | 11 May | 12 May | Devyanish Koul(201901444), Srushti Menpara(201901307), Dhaval Vaidya(201901462) |
| Testing | 13 May | 14 May | Devesh Modi(201901173), Dhaval Vaidya(201901462), Srushti Menpara(201901307),Hastin Dave(201901225) |
| Reviews | 13 May | 13 May | All members |
| Re-work and debugging | 14 May | 15 May | All members |

# **Testing Strategy**[**\***](#_yr9xsulwv3x5)

* Strategy is to test each module in isolation to make concurrent development and testing possible.
* Firstly, we will test the test cases through the design that we create, i.e. White Board Testing, and then comes the part of Software Testing.

# For each requirement, test transaction, expense results[**\***](#_yr9xsulwv3x5)

* Front end testing

(a) Schedule : 13 May 2022

(b) Responsible person : Srushti Menpara(201901307), Dhaval Vaidya(201901462)

(c) Placement of test cases, test data, test transaction, expected results : Test data required is dummy pages for every page of the software. Easy to use front end and guide to use the software.

* Test user authentication

(a) Schedule : 13 May 2022

(b) Responsible person : Devesh Modi(201901173), Dhaval Vaidya(201901225)

(c) Placement of test cases, test data, test transaction, expected results : Large set of dummy user credentials .Users have to enter his unique identification and safe password and ask for access to his profile. Only new users can sign up. Existing users can login in using unique email and/or user id and correct password.

* Testing business logic

(a) Schedule : 14 May 2022

(b) Responsible person : Devesh Modi(201901173), Srushti Menpara(201901307)

(c) Placement of test cases, test data, test transaction, expected results : Custom real-life scenarios. It should be transparent to the user. The splitting of bills is fair. The algorithm for simplifying debt is correct, that is it gives the payment method with a minimum number of transactions. There is consistency among the group related to the amount of debt.

* Testing payment gateway/ methods

(a) Schedule : 14 May 2022

(b) Responsible person : Devesh Modi(201901173), Hastin Dave(201901225), Srushti Menpara(201901307), Dhaval Vaidya(201901462)

(c) Placement of test cases, test data, test transaction, expected results : Traffic of transaction using different payment methods .User selects the amount to pay or selects the recommended amount to pay by the software and mode of payment. Based on the payment method selected, the user has to enter the credential. The requests are redirected correctly and timely to the banks.

* Test connection between business logic and database

(a) Schedule : 14 May 2022

(b) Responsible person : Srushti Menpara(201901307), Hastin Dave(201901225)

(c) Placement of test cases, test data, test transaction, expected results : Dummy requests to the databases. This should be transparent to the user. The information of the lending and borrowing of a particular user or a group stays correct after every transaction.

* Non-Functional(eg. performance, scalability etc.) testing

(a) Schedule : 14 May 2022

(b) Responsible person : Devesh Modi(201901173), Hastin Dave(201901225), Srushti Menpara(201901307), Dhaval Vaidya(201901462)

(c) Placement of test cases, test data, test transaction, expected results : Current standard performance metrics. Users should be able to access the software practically anytime and anywhere. The backend logic used is optimum. The servers on which the software is hosted should follow SLR(Service Level Agreement).

# Test cases for the design[**\***](#_yr9xsulwv3x5)

* An user enters correct login credentials then he should be able to login.
* If a user is new then he should be redirected to sign up.
* An user can register only using unique identification and should satisfy the requirements of the passwords.
* An user trying to make a new group.
* An user adds an individual expense
* An user adds a group expense.
* Users choose different policies for bill split.
* An user settles debts.
* An user wants to simplify debts.
* An user dissolves a group.
* An user changes the name and description of a group.
* An user who is the group owner locks/unlocks a group.
* An user selects a method of choice for paying bills.
* An user approves/disapproves the expense added.
* An user changes the settings of a group.
* An user adds a new member(s) to an existing group.

# Test cases and expected results for integration[**\***](#_yr9xsulwv3x5)

1. Module integration**\***

Integration testing is defined as a type of testing where software modules are integrated logically and tested as a group.

1. Schedule : 13 May
2. Responsible person : All members
3. Placement of test cases, test data, test transaction, expected results : All the paths from the use case diagram would be tested. Test data is the input to the system for different scenarios. Expected result is that each part is expected to work as if it is isolated as mentioned above.
4. User Interface Integration**\***
5. Schedule : 13 May
6. Responsible person : Srushti Menpara(201901307), Hastin Dave(201901225)
7. Placement of test cases, test data, test transaction, expected results : NA

# Test data, expected results for unit testing[**\***](#_yr9xsulwv3x5)

* Unit testing is a type of software testing where individual units or components of a software are tested. The purpose is to validate that each unit of the software code performs as expected.

(a) Schedule : 13 - 14 May 2022

(b) Responsible person : All members

(c) Placement of test cases, test data, test transaction, expected results :

# **Any other special activities**

* Any terms in this document which are new and **not yet introduced** are filled with the information available to us so far and marker **[\*]** which means **“submitted / updated in the upcoming version”** as mentioned in the **“SE-Project-Delivery-Guidelines-2022.pdf”** .