

1. Project Description

The Bill Splitter App is a desktop application developed to assist users in efficiently dividing expenses among participants. Built with PyQt5 for a seamless graphical interface and SQLite3 for reliable data storage, the app provides an intuitive experience for managing group expenses, tracking individual contributions, and ensuring accurate bill splitting.

2. Objectives

- Simplify the process of splitting bills among multiple participants.
- Enable efficient tracking and categorization of expenses.
- Ensure accurate calculation of each participant's share based on total expenses.
- Provide an easy-to-navigate and user-friendly interface.

3. Features

- **Participant Management:** Allows adding, editing, and tracking of participants.
- **Expense Tracking:** Supports inputting and categorizing expenses, specifying the amount, payer, and all involved participants.
- **Automatic Bill Splitting:** Calculates each participant's share based on their involvement in the expenses.
- **Database Storage:** Utilizes SQLite3 for secure and persistent data storage.
- **User Interface:** Built with PyQt5 for an intuitive and responsive UI.

4. Overview

Launching the App

Run the app by executing this .bat file:

```
@echo on
REM Go to your drive (if you are not running the bat file in the
same drive) ex: E:
E:
REM Move to project directory
cd "E:\path\to\project\bill-splitter"

REM Activate the virtual environment
call venv\Scripts\activate

REM Run the Python script
python src/main.py

REM Deactivate the virtual environment
deactivate
```

Main View

By default, the app opens with a simple, user-friendly layout:

The screenshot shows the 'Bill Splitter' application window. It has a light gray background and standard window controls (minimize, maximize, close) in the top right corner. The interface is divided into three main sections:

- Left Section:** A 'Name' input field with the placeholder text 'New Participant ...' and an 'Add' button. Below this is a table with two columns: 'Name' and 'Action'.
- Middle Section:** A 'Debt:' label above a large table with four columns: 'Owe', 'Owed', 'Amount', and 'Action'.
- Right Section:** A 'Description:' label above a 'New Description ...' input field and an 'Add' button. Below this is a text area containing the text 'pay debt'. At the bottom of the right section is a 'Transactions' label above a scrollable list area, with a 'New Transaction' button at the very bottom.

Adding Participants

For example, we add three participants:

The screenshot shows the 'Bill Splitter' application window after adding three participants. The 'Name' input field now contains 'chandra' and the 'Add' button is visible. The table in the left section now has three rows of data:

	Name	Action
1	andi	Delete
2	budi	Delete
3	chandra	Delete

The 'Debt:' table in the middle section now has three rows of data:

	Owe	Owed	Amount	Action
1	andi	budi	0.0	Pay
2	andi	chandra	0.0	Pay
3	budi	chandra	0.0	Pay

The right section remains the same, with the 'Description:' input field containing 'pay debt' and the 'Transactions' list area empty.

Adding Spending Descriptions

Add a description for the type of expense:

Name :

	Name	Action
1	andi	<input type="button" value="Delete"/>
2	budi	<input type="button" value="Delete"/>
3	chandra	<input type="button" value="Delete"/>

Debt:

	Owe	Owed	Amount	Action
1	andi	budi	0.0	<input type="button" value="Pay"/>
2	andi	chandra	0.0	<input type="button" value="Pay"/>
3	budi	chandra	0.0	<input type="button" value="Pay"/>

Description:

pay debt
spending

Transactions

Entering a New Transaction

Add a new transaction. This opens a window to enter details. Suppose Andi and Budi pay 9000 and 3000, respectively, and want to split the bill evenly.

Transaction Window

Description:

	Name	Proportion	Amount
1	andi	1	9000
2	budi	1	3000
3	chandra	1	0

Updated View After Transaction

The main view updates to show the new transaction, displaying each participant's balance and debt automatically:

Bill Splitter

Name :

	Name	Action
1	andi	<input type="button" value="Delete"/>
2	budi	<input type="button" value="Delete"/>
3	chandra	<input type="button" value="Delete"/>

Debt:

	Owe	Owed	Amount	Action
1	andi	budi	-1000.0	<input type="button" value="Pay"/>
2	andi	chandra	-4000.0	<input type="button" value="Pay"/>
3	budi	chandra	0.0	<input type="button" value="Pay"/>

Description:

pay debt
spending

Transactions

andi - 2024/11/12 - spending - 9000
budi - 2024/11/12 - spending - 3000

Making a Payment

If Chandra wants to pay Andi 2000, they can click the "Pay" button, which opens this window:

Pay Window

Input:

After clicking OK, the balances update accordingly:

Bill Splitter

Name :

Add

	Name	Action
1	andi	Delete
2	budi	Delete
3	chandra	Delete

Debt:

	Owe	Owed	Amount	Action
1	andi	budi	-1000.0	Pay
2	andi	chandra	-2000.0	Pay
3	budi	chandra	0.0	Pay

Description:

New Description ...

Add

pay debt

spending

Transactions

andi - 2024/11/12 - spending - 9000

budi - 2024/11/12 - spending - 3000

chandra - 2024/11/12 - pay debt - 2000

<

>

New Transaction

Removing a Participant

To delete a participant, click the delete button. The app retains only active participants' information:

Bill Splitter

Name :

Add

	Name	Action
1	andi	Delete
2	budi	Delete

Debt:

	Owe	Owed	Amount	Action
1	andi	budi	-1000.0	Pay

Description:

New Description ...

Add

pay debt

spending

Transactions

andi - 2024/11/12 - spending - 9000

budi - 2024/11/12 - spending - 3000

chandra - 2024/11/12 - pay debt - 2000

<

>

New Transaction