SettleUp: Debt Simplification and Analyzer

A Synopsis Submitted
in Partial Fulfillment of the Requirements
for the Course of
Minor Project - II
In
Third year – 6th Semester of
Bachelor of Technology Computer Science
specialization
In
Business Analytics and Optimization

Under the guidance of

Mr. Rahul Kumar Singh

Submitted By:

NAME	SAP-ID	ROLL NO.	Branch/Batch
Vivek Raj	500069212	R103218189	CSE-BAO B4
Avik Singla	500067990	R103218190	CSE-BAO B4
Vaishnavi Jaiswal	500071059	R103218199	CSE-BAO B2
Sourabh Bhandari	500071057	R103218207	CSE-BAO B4

DEPARTMENT OF INFORMATICS
SCHOOL OF COMPUTER SCIENCE
UNIVERSITY OF PETROLEUM AND ENERGY STUDIES, BIDHOLI, DEHRADUN, UTTRAKHAND, INDIA

January, 2021

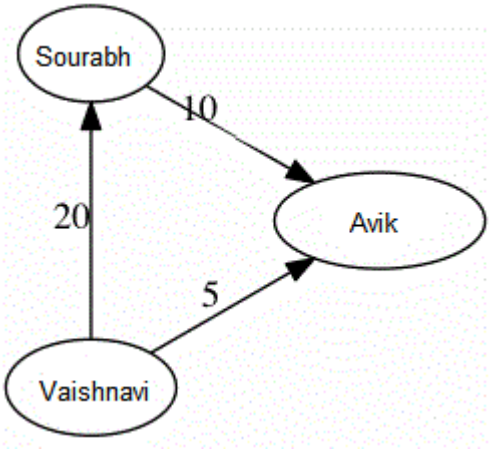
INTRODUCTION

Simplifying debts ("debt simplification") is a feature of our application Settle Up: Debt Simplification and Analyzer that restructures debt within groups of people. It does not change the total amount that anyone owes, but it makes it easier to pay people back by minimizing the total number of transactions.

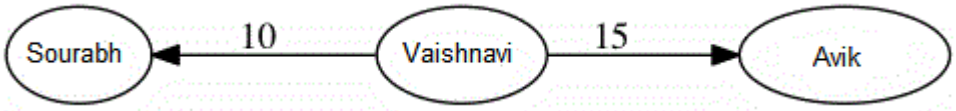
We organize all your shared expenses and debts in one place, so that everyone can see who they owe. Whether you are sharing a ski vacation, splitting rent with roommates, or paying someone back for lunch, our project SettleUp makes life easier.
It organizes all its users' shared expenses and debts in one place so that everyone can see who they owe. It sends its users monthly reminders about outstanding debts.

In our project each debt will be represented in the form of a node of a directed graph and each edge of this graph will represent the amount for the transactions. We will simplify this graph to get least number of transactions.

For example, let us assume 3 friends Sourabh, Avik and Vaishnavi went out for walk in a park.
So, after the walk, we see that:
Vaishnavi owes Sourabh Rs.20 and Avik Rs. 5, Sourabh owes Avik Rs. 10, etc.



This graph should be simplified to the following graph:



There is no sense in Rs.10 making its way from Vaishnavi to Sourabh and then from Sourabh to Avik if Vaishnavi could just give it to Avik directly.

The goal, then, in the general case is to take a debt graph and simplify it (i.e., produce a new graph with the same nodes but different edges) such that:

- No node has edges pointing both in and out of it.
- All nodes have the same "flow" through them as they did in the original graph (it is identical in terms of where the money ends up).

OBJECTIVE

- Our objective is to settle all the dues or debts amongst the group of people. We will try to minimize total number of transactions to be made.

- We must ensure that all the transactions are fair and completed and see that nobody ends up paying anyone less or more than their share.

- We will analyze the past transactions to help the user learn about his/her spending habits to make smart decisions in the future.

MOTIVATION

We see that whenever we go out some payments get mixed up in groups and someone is lacking behind the payments. Bills and food shops/holidays where everyone is pooling money. The dues keep on piling up. We often forget in our busy lifestyle that a particular friend owes certain amount be it small or large. So, this program will help us solve this problem and will be useful in real life and help you manage each and every single transaction during any course of time.

The use case can be as simple as sharing expenses with your hostel roommate to a long trip with a group of friends to Goa.

METHODOLOGY

Analysis: In our day to day lives, when we travel or go to a restaurant with our friends, we face the problem as to what each individual's share of overall bill would be. Whenever a person is lacking behind in payments the dues keep on piling up. The use case can be as simple as sharing expenses with your hostel room mate or splitting a fare from a shared cab. So, to simplify this settlement process we have come up with this project idea.

Project Requirements:

Software Requirements: Visual Studio Code, GCC/G++ Compiler, Microsoft Excel

Hardware Requirements: A system with Windows/Linux OS.

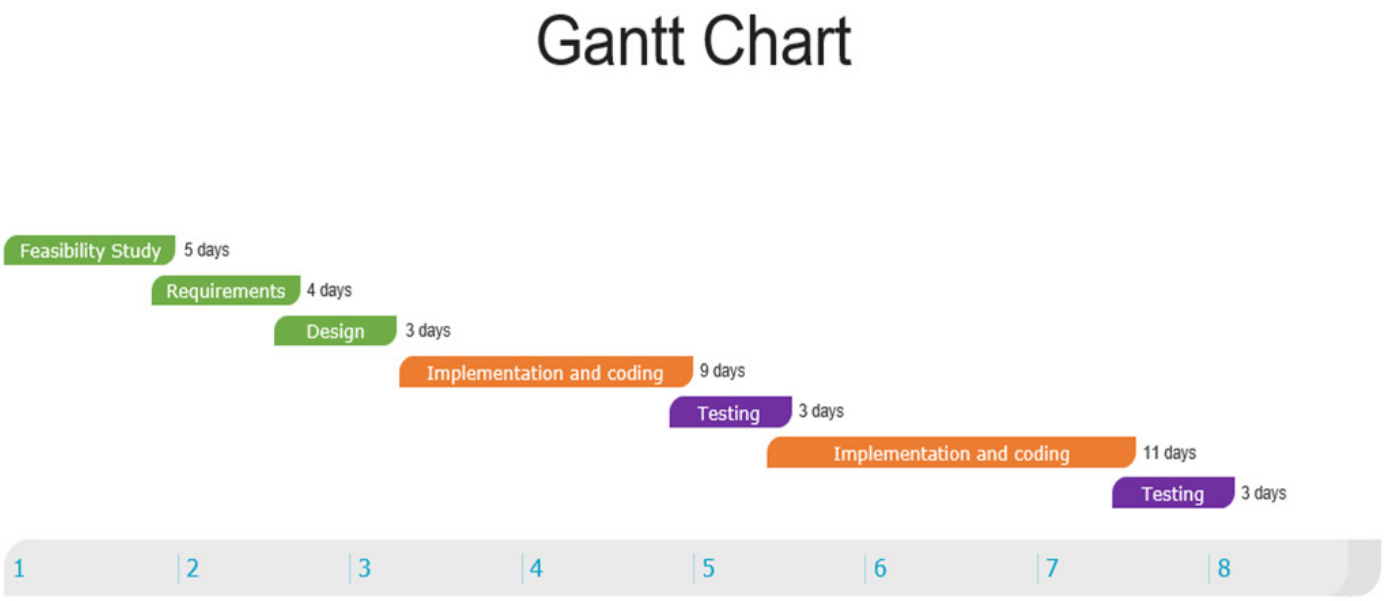
Design: The designing of the project workflow involves thorough analysis of use cases and flow of overall user interaction with the program from input to generating the desired result.

Implementation:

1. Read the dataset for transactions record from csv file.
2. Parsing the input to be processed by the program.
3. Based on the inputs provided, analyze, and calculate the desired result.
4. Present the findings to the end user in an intuitive manner.
5. Help user make informed decisions through the results.

Testing: We will test our dataset through various test cases and scenarios to ensure that the program runs efficiently and produces desired results in different use cases.

PLAN OF WORK


Gantt Chart
Feasibility Study
5 days
Requirements
4 days
Design
3 days
Implementation and coding
9 days
Testing
3 days
Implementation and coding
11 days
Testing
3 days
1
2
3
4
5
6
7
8

REFERENCES

[1] Debt Simplification Techniques :
Debt Simplification Feature | by Mithun Mohan K |

[2] A Fast and Simple Algorithm for the Maximum Flow Problem (researchgate.net)

[3] NP-Complete Splitting