SettleUp : Debt Simplification & Analyzer

A

Project Report

submitted in partial fulfillment of the

Minor Project - II
In
Third year – Sixth Semester of
Bachelor of Technology
In
Computer Science & Engineering
Specialization in
Business Analytics and Optimization.

Avik, Sourabh, Vaishnavi, Vivek

Digitally signed by Avik, Sourabh, Vaishnavi, Vivek
Date: 2021.05.01 19:26:39 +05'30'

under the guidance of

Mr. Rahul Kumar Singh, Department of Informatics

Submitted By:

	SAP ID	ROLL NO.	BRANCH
Avik Singla	**500067990**	**R103218190**	**CSE-BAO**
Sourabh Bhandari	**500071057**	**R103218207**	**CSE-BAO**
Vaishnavi Jaiswal	**500071059**	**R103218199**	**CSE-BAO**
Vivek Raj	**500069212**	**R103218189**	**CSE-BAO**

UPES
UNIVERSITY WITH A PURPOSE

Department of Informatics

School of Computer Science

University of Petroleum & Energy Studies

Bidholi, Via Prem Nagar, Dehradun, UK

May – 2021

DECLARATION

I/We hereby certify that the project work entitled **"SettleUp: Debt Simplification & Analyzer"** in partial fulfilment of the requirements for the award of the Degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING specialization Business Analytics and Optimization and submitted to the Department of Informatics, School of Computer Science, University of Petroleum & Energy Studies, Dehradun, is an authentic record of my/ our work carried out during a period from **Jan.2021** to **May. 2021** under the supervision of **Rahul Kumar Singh**.

The matter presented in this project has not been submitted by me/ us for the award of any other degree of this or any other University.

Avik Singla	**Sourabh Bhandari**	**Vaishnavi Jaiswal**	**Vivek Raj**
R103218190	**R103218207**	**R103218199**	**R103218189**

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date: 1 May 2021

Name of Mentor
Rahul Kumar Singh

ACKNOWLEDGEMENT

We wish to express our deep gratitude to our mentor(s) **Mr. Rahul Kumar Singh**, for all advice, encouragement and constant support he has given us throughout our project work. This work would not have been possible without his support and valuable suggestions.

We sincerely thank to our respected Head of Department, Informatics, **Dr. Thipendra Pal Singh,** for his guidance and support as and when required.

We are also grateful to **Dr. Priyadarsan Patra, Dean SCS**, for providing the necessary facilities to carry out our project work successfully.

We would like to thank all our friends for their help and constructive criticism during our project work. Finally, we have no words to express our sincere gratitude to our **parents** who have shown us this world and for everything they have given to us.

Avik Singla	**Sourabh Bhandari**	**Vaishnavi Jaiswal**	**Vivek Raj**
R103218190	**R103218207**	**R103218199**	**R103218189**

ABSTRACT

Just splitting bills/fares/rents can become complicated with people's feelings. People who got a less nice room sometimes feel quietly upset or resentful, but they don't want to negotiate. This atmosphere makes it harder on everyone.

Paying everyone a fair share without negotiation can be simple using SettleUp.

Simplifying debts is a feature of our application SettleUp 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 payments.

Now a days, a user must track multiple expenses in a group and manage expenses across various categories which can become a tedious task if done manually.
This work tries to simplify this entire process in a user-friendly manner.

This work tries to simplify this entire process in a user-friendly manner and provide an overview of his/her expenses.

TABLE OF CONTENTS

INTRODUCTION

Now a days, we transact a lot on daily basis for different requirements and with different people.

Well, tracking them all might become a tedious task when you are transacting through different mediums and
to different places.

Let us consider an example:
4 friends go for a movie  . But only 2 of them pay for all the expenses. The expense must be split between the 4. Maybe 1 pays for movie tickets  and the other pays for food  .

So, how exactly the split of bills should be among the friends such that it is fair, and nobody ends up arguing?

This work covers to solve the task of Fair expenses and debt sharing with different parties, which covers how different intermediaries in a payment network can be simplified to make settlements easier.

The work begins by studying the concepts behind the fair division strategies and by performing tasks like sharing and splitting of rents in an optimal manner, reasonable distribution across the payment nodes and presenting user his/her spending habits.

Problem Statement

- Building a budget splitter application using fair division techniques.
- By applying the concepts used in maximum flow algorithm on stored transaction dataset, resulting in simplified transactions with fair division method.
- Completing the overall settlement in an efficient manner.
- Storing and maintaining user's data and transactions details.
- Visualizing the expenses of a user to help him/her understand his/her expenditures.

Objectives

This project will help the user to simplify their debts among a group and visualize their monthly expenses: -

1) **Settle all dues:** 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.

2) **Fair Transactions:** 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.

3) **Visualizing and Analyzing Spending Habits:** We will visualize the past transactions to help the user understand about his/her spending habits in a user friendly manner.

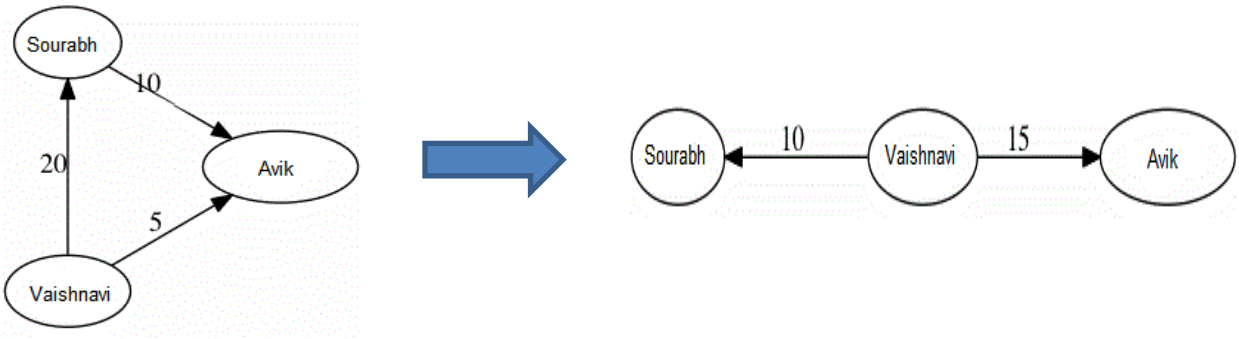
Design

➢ Methodology:

- Simplifying debts 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**.**

- In our project each person 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.



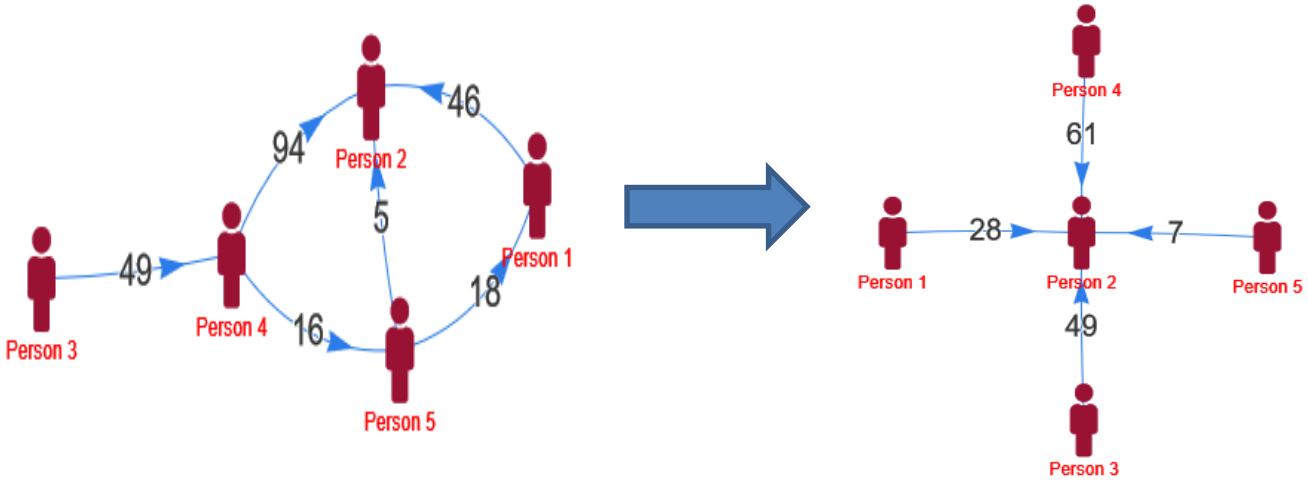
The goal, then, in the general case is to take a debt graph and simplify it such that:

- ✓ No node has edges pointing both in and out of it.
- ✓ No one owes more money in total than they did before the simplification.

- Further, a user can visualize his/her expenses across different categories.

- Even for a complex network of people where multiple people are involved in many transactions across the network SettleUp settles the transactions in the most optimal way possible.

- The approach is greedy where at every step, all amounts of one person is settled and the program recurs for remaining (n-1) persons until the entire network is settled by constructing an equivalent graph of settled transactions.



➢ ALGORITHMS:

Ford Fulkerson Algorithm

Input: A graph G with source as u and sink as v

Output: Residual graph and maxflow capacity of the input source & sink

max_flow=0

for each edge(u,v) in G:

flow(u,v)=w(u,v)

while there is a path p from s->t in residual graph:

min_cap = minimum residual capacity in path p.

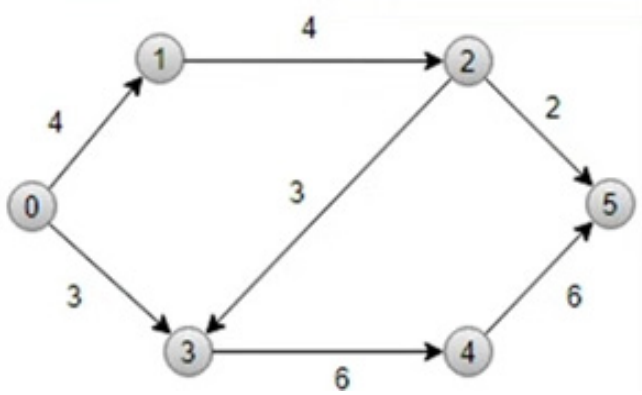
max_flow += min_cap

for each edge(u,v) in p:

flow(u,v) = flow(u,v) - min_cap

flow(v,u) = flow(v,u) + min_cap

return max_flow

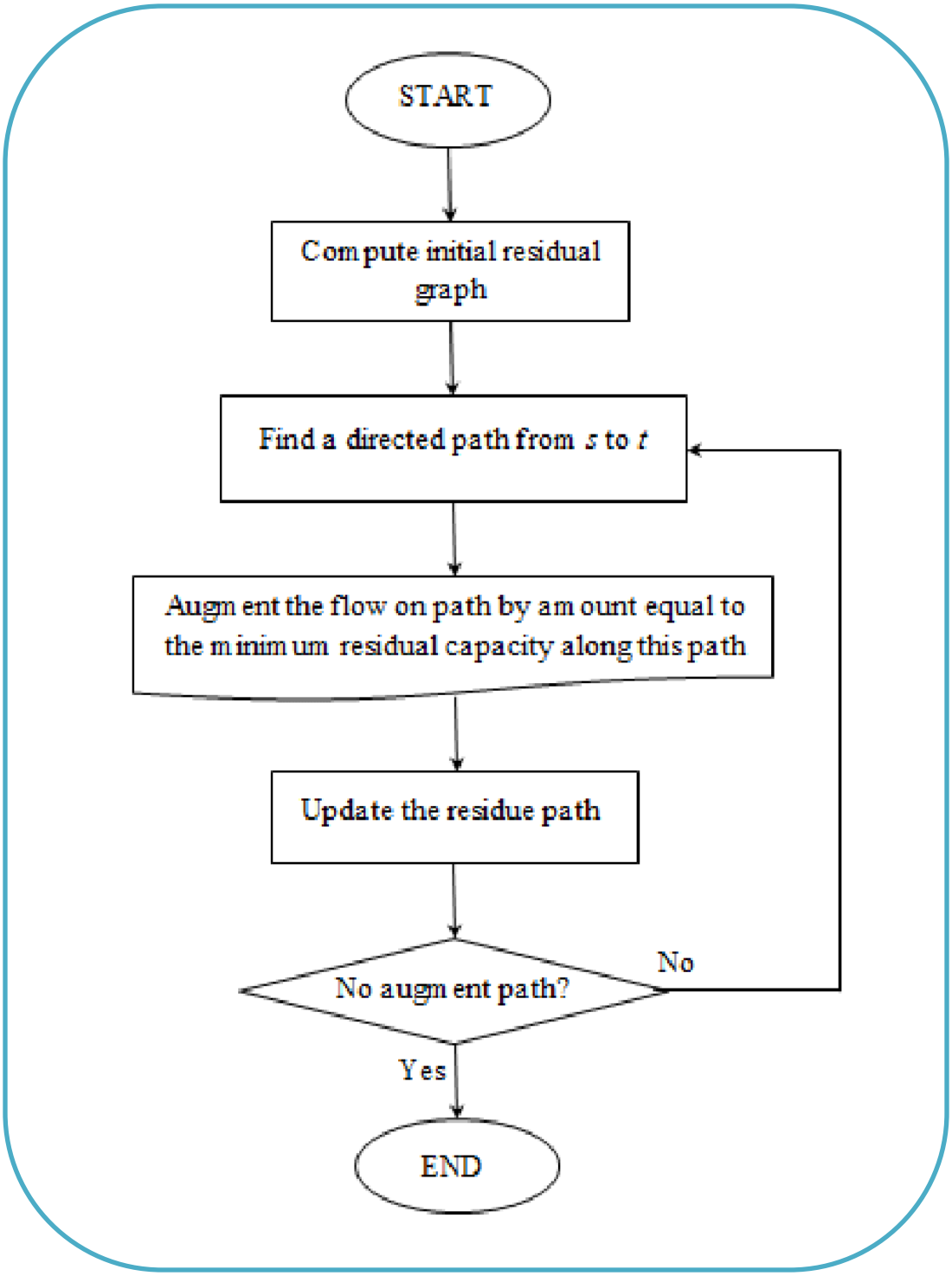


For example, here:

Source = 0 , Sink = 5

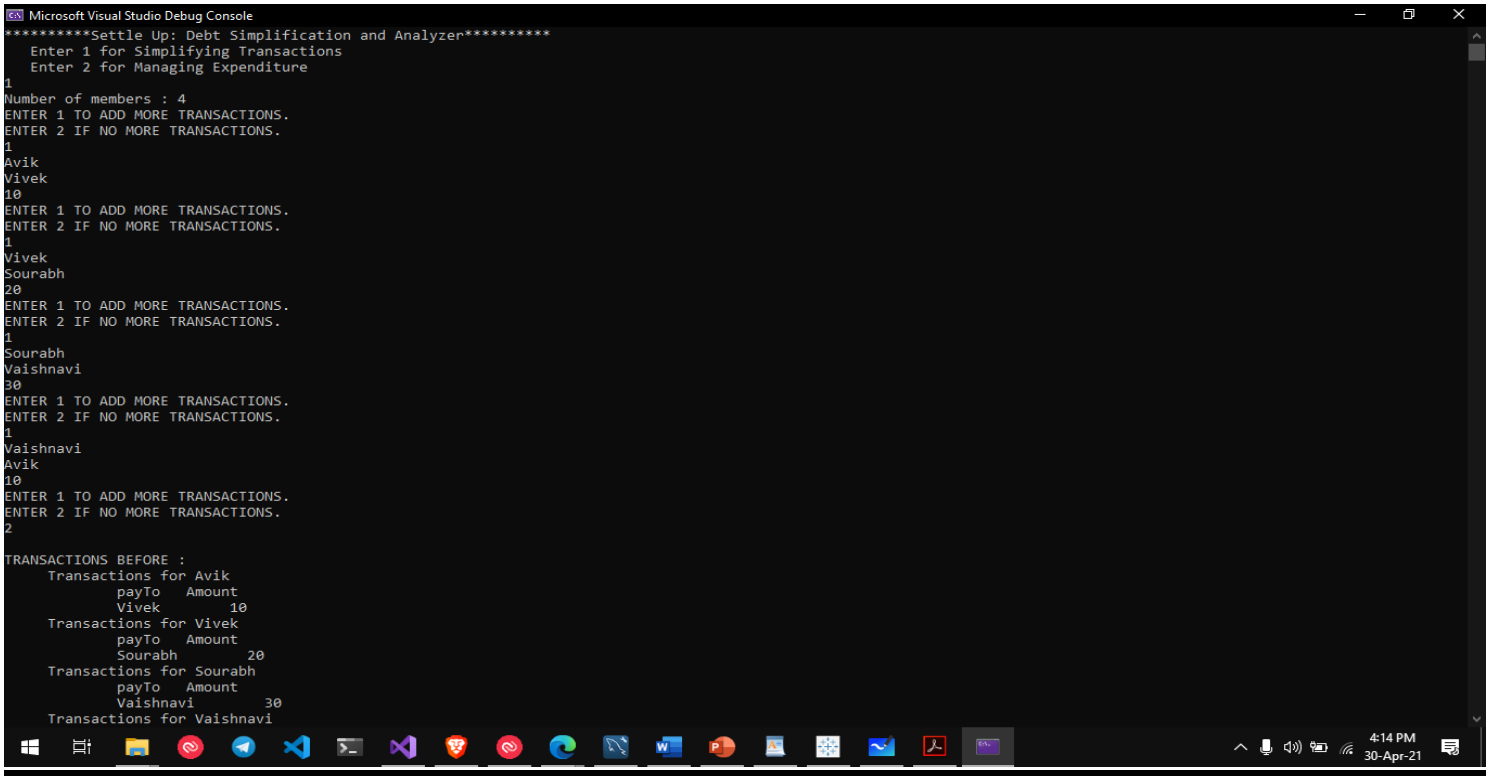
Time Complexity : O(V^2E)

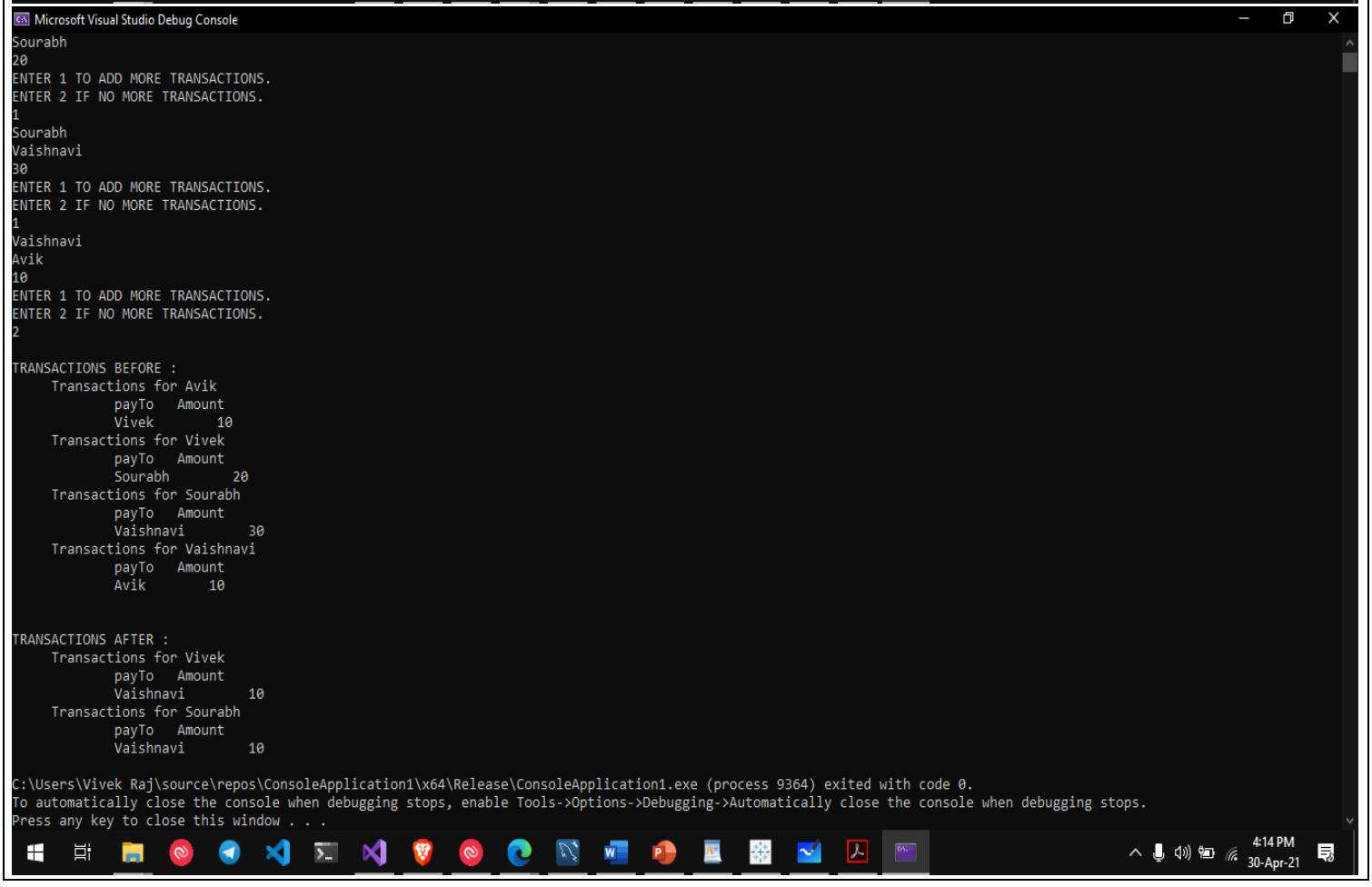
FLOWCHART

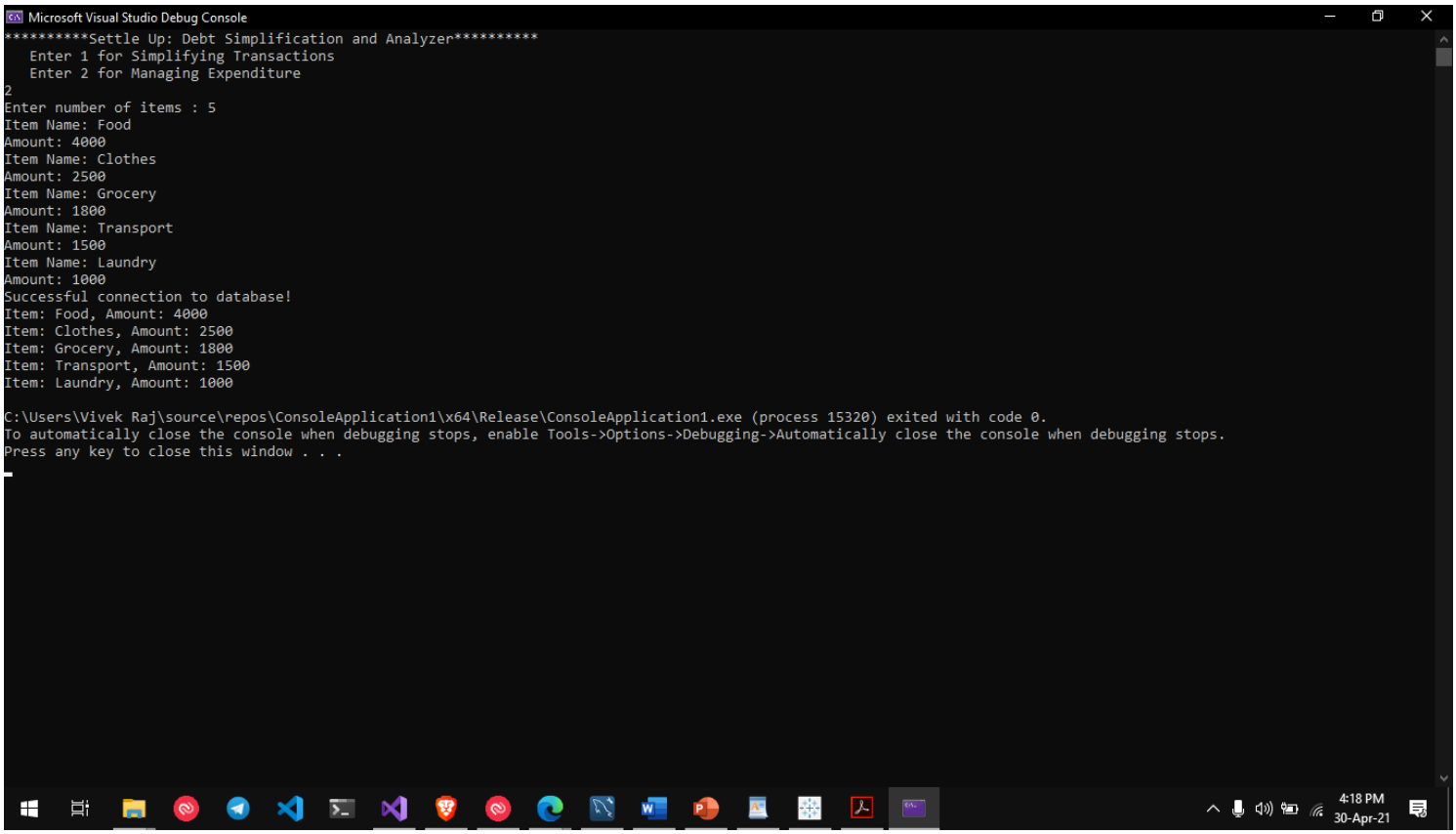

START
Compute initial residual graph
Find a directed path from s to t
Augment the flow on path by amount equal to the minimum residual capacity along this path
Update the residue path
No augment path?
No
Yes
END

Implementation

➢ Output Screens:

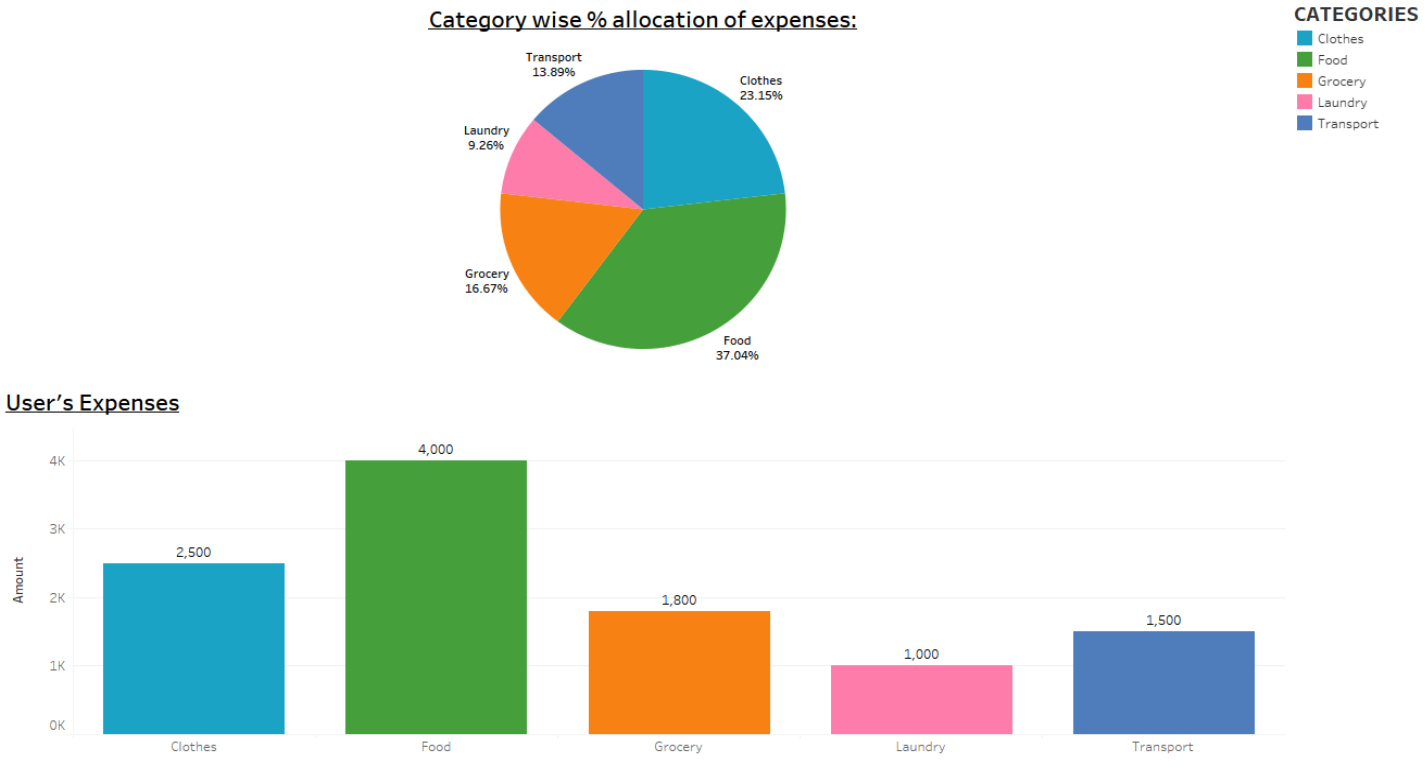






Result Analysis

Here, in the above output we can see that we have simplified the entire process by minimizing the number of transactions to made among the users in a fair manner and visualize the expenses of a user as shown below to help a user understand his/her expenditures.

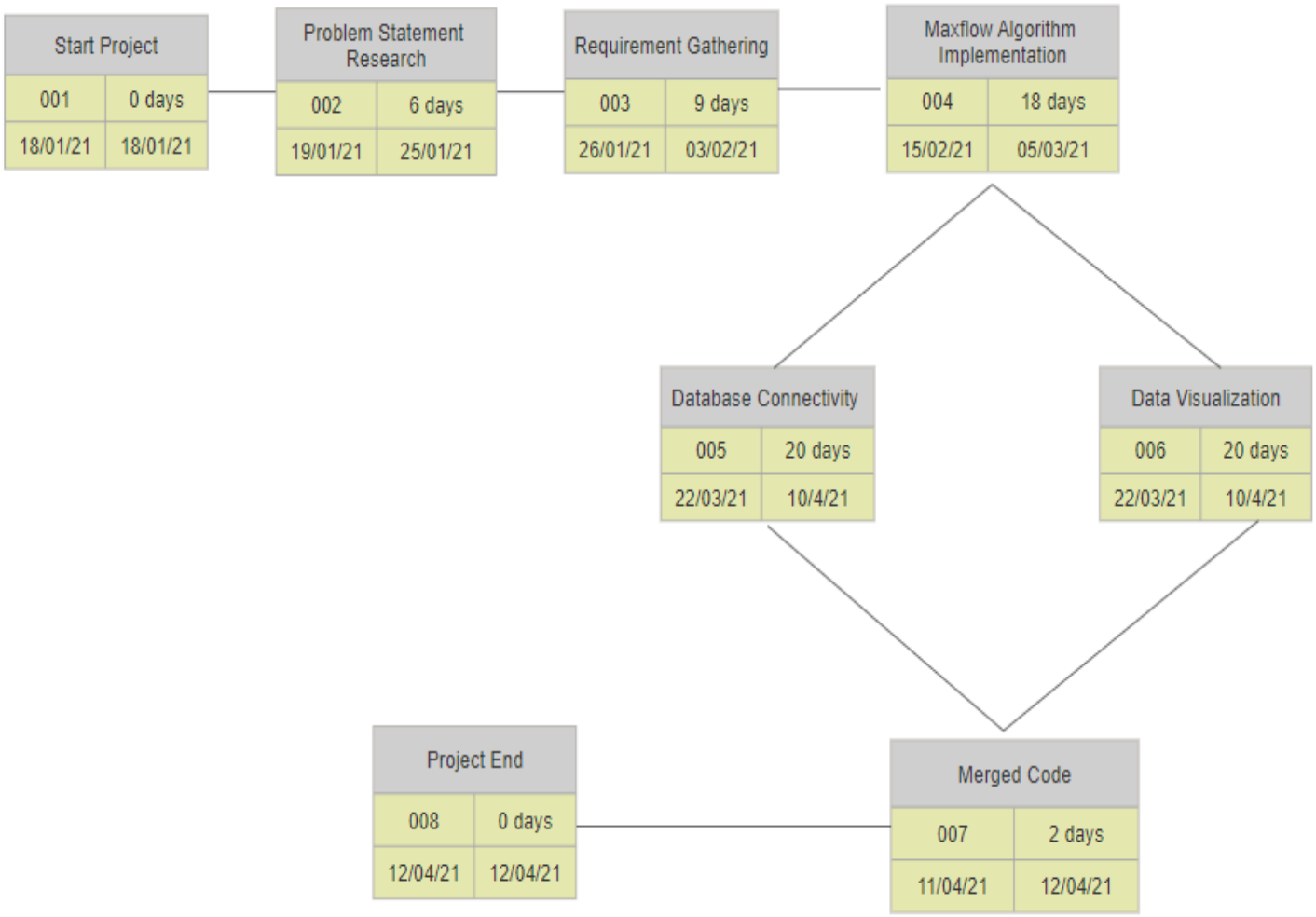


Conclusion and Future Scope

By doing this project SettleUp: Debt Simplification & Analyzer, we learnt about the properties and implementation of graph and maxflow algorithms like Ford Fulkerson and Dinic's maxflow algorithm. We tried to minimize total number of transactions to be made among a group of people to ensure fair distribution of money in the group and understand about the user's expenses by visualizing his/her expenditure.

Further, in future we shall extend our work to build an application to provide break up of the spending summary for a particular group, splitting of taxes on the expenses with due dates reminder which shall help the user track and manage their expenses making life a bit easier.

PROGRESS REPORT (PERT CHART)



APPENDIX I PROJECT CODE

The entire project code and its implementation is managed on **GitHub** by us at: ***vivekrajx/SettleUp-Debt-Simplification-and-Analyzer***

References

[1] Debt Simplification Feature | by Mithun Mohan K |, Debt Simplification Techniques, Mithun Mohan K, 2019

[2] A Fast and Simple Algorithm for the Maximum Flow Problem (researchgate.net) , James B. Orlin, Ravindra K. Ahuja, October 1989

[3] NP-Complete Splitting

[4] An Empirical Evaluation of Fair-Division Algorithms , Dupuis-Roy, Nicolas Gosselin, Frederic, 2009

[5] Ford Fulkerson's algorithm for the maximum possible flow in a network https://youtu.be/_UcOALraATY