2nd Year Project Report

on

Expense Tracker:
CSV-Based Bank Statement Visualization Using Python

CS213: Introduction To Python Programming

Under The Guidance of

Dr. Dubacharla Gyaneshwar
Computer Science & Engineering Department

Submitter By

Naumisharanya Tirth (CS23B1050) & Sourav Kumar (CS23B1070)

Department of Computer Science And Engineering

ಭಾರತೀಯ ಮಾಹಿತಿ ತಂತ್ರಜ್ಞಾನ ಸಂಸ್ಥೆ ರಾಯಚೂರು
भारतीय सूचना प्रौद्योगिकी संस्थान रायचूर
Indian Institute of Information Technology Raichur

October 6, 2024

Contents

1. Introduction

In today's fast-paced financial world, the ability to analyze and interpret personal or business transaction data is crucial for informed decision-making. With an overwhelming amount of financial data often stored in CSV files, manually processing and interpreting such data can be time-consuming and prone to errors. This project aims to bridge the gap between raw financial data and actionable insights by providing an automated tool for seamless data analysis and visualization.

Leveraging the power of Python, this application allows users to upload transaction data in CSV format and instantly visualize key metrics such as cumulative balances, credits and debits over time, and transaction types. The use of Pandas for data manipulation ensures efficient data processing, while Plotly creates dynamic and interactive visualizations, enabling users to gain deeper insights into their financial activities with just a few clicks.

By automating this analysis process and offering a user-friendly interface built with Tkinter, this project eliminates the complexity of manual data review, empowering users to make better financial decisions, track spending habits, and monitor income patterns with ease.

2. Aim and Objectives

Aim:

To develop a Python-based tool for analyzing and visualizing financial transactions from CSV files, making it easy for users to gain insights into their financial data.

Objectives:

- **Intuitive GUI for Effortless CSV File Selection**
  Unlock a seamless user experience with an elegant graphical user interface (GUI) crafted using Tkinter. Users can effortlessly drag and drop their CSV files or select them through a simple browsing option, making data input a breeze.
- **Smart Data Parsing and Cleaning**
  Dive into the world of financial insights by automatically parsing and cleaning the transaction data. The program intelligently handles inconsistencies and missing values, transforming raw data into a polished dataset ready for analysis and visualization.
- **Interactive and Stunning Visualizations with Plotly**
  Bring data to life with dynamic and visually captivating graphs generated using Plotly. These interactive visualizations allow users to explore their financial transactions in a more engaging manner, providing the ability to hover, zoom, and filter data for deeper insights.
- **Insightful Analysis of Transaction History**
  Empower users with a comprehensive understanding of their financial habits. The program presents key insights, such as cumulative balances and transaction types, allowing users to quickly grasp their spending patterns and make informed decisions.

3. Contributions:

- **Person 1:** Focused on creating the visualizations using Plotly, which allowed for greater interactivity compared to Matplotlib. The move to Plotly was necessary to achieve the desired level of interactivity in the graphs.
- **Person 2:** Designed the Tkinter window, handled CSV file processing, and worked on displaying the visualizations. Both Person 1 and Person 2 collaborated on using webview to maintain interactivity. However, we faced challenges in viewing the graphs within the same window, which led to the decision to use webview for individual displays.

4. Tools and Data Used

Tools:

- **Programming Language:** Python is a versatile, high-level programming language known for its readability and simplicity. It is widely used in various fields, including data science, web development, and automation, making it an ideal choice for this project.
- **Libraries:**
    - **Pandas** is a powerful library for data analysis and manipulation. It provides data structures like DataFrames, which make it easy to handle structured data, perform operations like filtering, grouping, and aggregating, and manage missing values.Plotly for creating interactive graphs.
    - **Plotly** is a library for creating interactive visualizations. It enables the development of graphs and dashboards that allow users to explore data dynamically. This is particularly useful for visualizing financial transactions in an engaging way.Webview for displaying interactive visualizations.
    - **Tkinter** is the standard GUI toolkit for Python. It allows developers to create desktop applications with user-friendly interfaces. In this project, Tkinter is used to build the file selection interface, enabling users to easily upload their CSV files

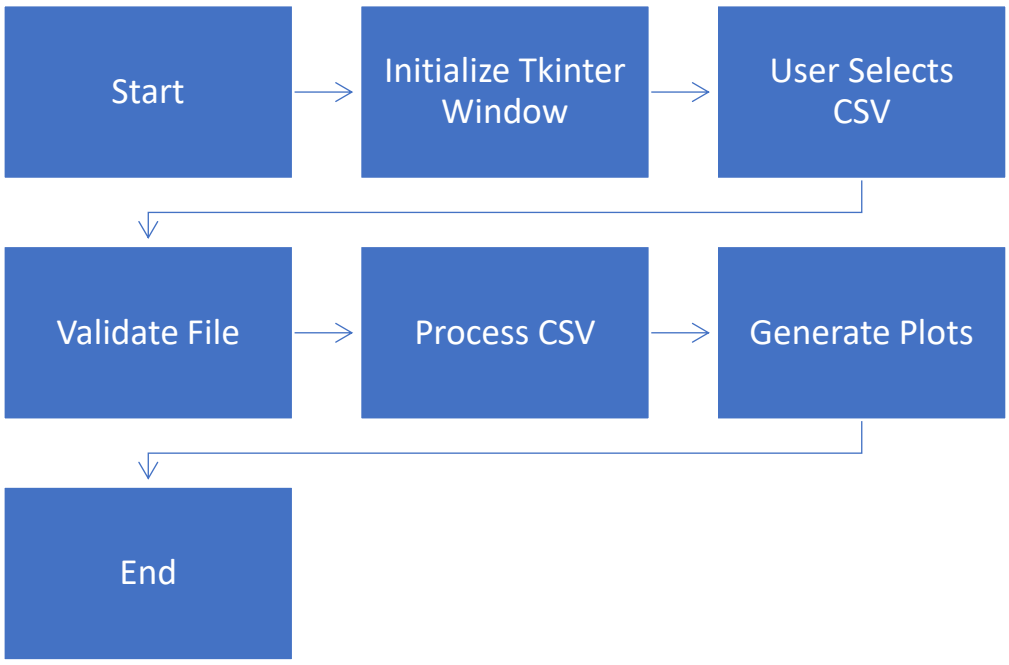
- **Webview** is a lightweight library for creating web-based interfaces in desktop applications. It allows the integration of HTML and JavaScript into a Tkinter application, enabling the display of interactive visualizations created with Plotly.
- **NumPy** is a fundamental library for numerical computations in Python. It provides support for arrays, matrices, and a variety of mathematical functions. In this project, NumPy is utilized for handling missing data and performing numerical calculations.

. **Development Environment:** IDE - VSCode used for development.

Data:

- The data used comes from CSV files containing financial transactions, typically including columns like Date, Credit, Debit etc. This data can be sourced from bank statements or similar transactional data.

5. Flow Diagram



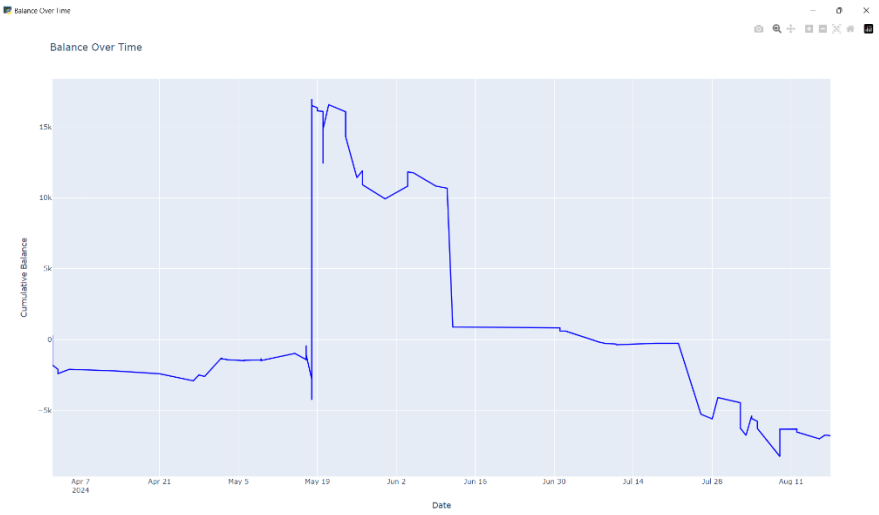
6. Results

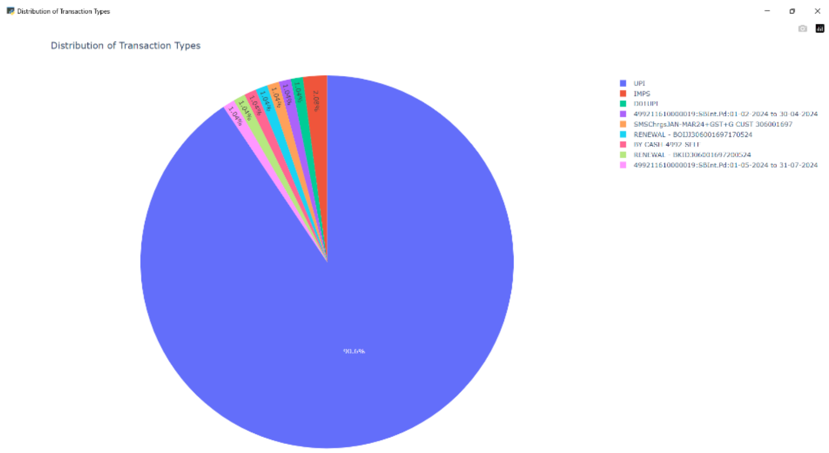
The application successfully processes CSV files, cleans the data, and generates dynamic visualizations that allow users to:

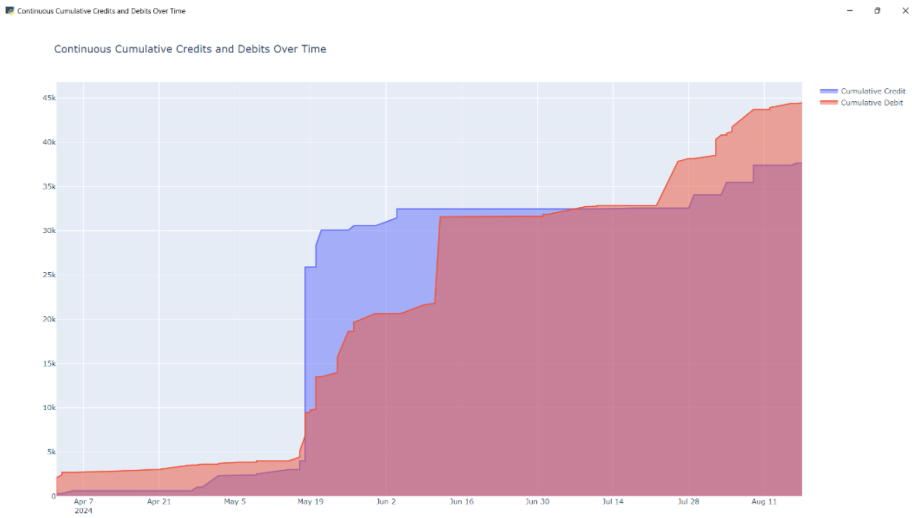
- Track balance changes over time.
- Compare spending and income patterns.
- Visualize the distribution of different transaction categories.
- Analyze cumulative financial trends.

These visualizations enable users to gain insights into their financial data, identify trends, and make informed decisions.

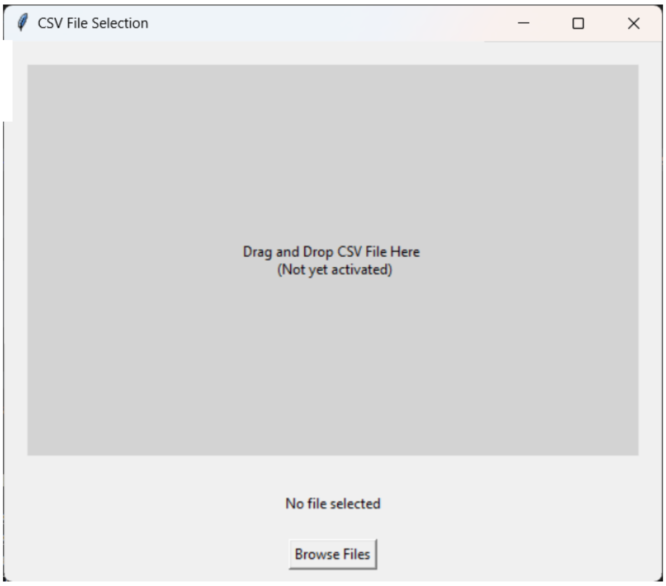
Graphs




Distribution of Transaction Types
Distribution of Transaction Types
UPI
IMPS
4992116100000019:SBInt.Pd:01-02-2024 to 30-04-2024
SMSChrgsJAN-MAR24+GST+G CUST 306001697
RENEWAL - BOIJJ306001697170524
4992116100000019:SBInt.Pd:01-05-2024 to 31-07-2024


Continuous Cumulative Credits and Debits Over Time
Continuous Cumulative Credits and Debits Over Time
Cumulative Credit
Cumulative Debit
45k
40k
35k
30k
25k
20k
15k
10k
5k
0
Apr 7
2024
Apr 21
May 5
May 19
Jun 2
Jun 16
Jun 30
Jul 14
Jul 28
Aug 11

File Dialog


CSV File Selection
Drag and Drop CSV File Here
(Not yet activated)
No file selected
Browse Files

7. Conclusions

The project successfully achieved its aim of providing an efficient tool for processing and visualizing transaction data from CSV files. By leveraging Tkinter for the user interface, Pandas for data manipulation, and Plotly for interactive graphs, the application offers a streamlined experience for users to analyze their financial data with ease.

While the project successfully maintained graph interactivity, challenges were encountered when attempting to view all graphs in a single window. Future improvements will focus on merging all plots into one cohesive display and enhancing the transition from the file input window to the graph visualization, ensuring a more intuitive and fluid user experience.

Furthermore, the project plans to implement a wider range of graphs to provide deeper insights into various spending categories, including shopping expenses, mobile recharges, investment tracking, and many more. These enhancements will enrich the application's functionality, offering users comprehensive tools for managing and visualizing their finances.