Excel Utility for Stock Analysis and Portfolio Management

Minor Project Report

Submitted by:
Rushin Shah - 21BIT251
Rushi Jhala - 21BIT250
Dev Chauhan - 21BIT244

Under the guidance of:
Dr. Jigarkumar Shah
Assistant Professor

Submitted To:
Dr. Jigarkumar Shah
Ms. Rashmi Bhattad

Department of Information and Communication Technology

Pandit Deendayal Energy University
Gandhinagar, Gujarat, India
2024-2025

CERTIFICATE

This is to certify that the seminar report entitled **"Excel Utility for Stock Analysis and Portfolio Management"** submitted by **Rushin Shah (21BIT251), Rushi Jhala(21BIT250), Dev Chauhan(21BIT244)** has been conducted under the supervision of **Dr. Jigarkumar Shah, Assistant Professor, Department of ICT**, and is hereby approved for the partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering in the **Department of Information and Communication Technology/Electronics and Communication Engineering** at Pandit Deendayal Energy University, Gandhinagar. This work is original and has not been submitted to any other institution for the award of any degree.

Sign:

Name of Guide: Dr. Jigarkumar Shah
Designation: Assistant Professor
Department: Information and Communication Technology
School of Technology,
Pandit Deendayal Energy University, Gandhinagar

Sign:

Name of Examiner: Dr. Jigarkumar Shah
Designation: Assistant Professor
Department: Information and Communication Technology
School of Technology,
Pandit Deendayal Energy University, Gandhinagar

Sign:

Name of Examiner: Ms. Rashmi Bhattad
Designation: Assistant Professor
Department: Information and Communication Technology
School of Technology,
Pandit Deendayal Energy University, Gandhinagar

DECLARATION

I/We hereby declare that the minor project report entitled **"Excel Utility for Stock Analysis and Portfolio Management"** is the result of our own work and has been written by me. This report has not utilized any language model or natural language processing artificial intelligence tools for the creation or generation of content, including the literature survey.

The use of any such artificial intelligence-based tools was strictly confined to the polishing of content, spell checking, and grammar correction after the initial draft of the report was completed. No part of this report has been directly sourced from the output of such tools for the final submission.

This declaration is to affirm that the work presented in this report is genuinely conducted by us and to the best of our knowledge, it is original.

Rushin Shah – 21BIT251
Rushi Jhala – 21BIT250
Dev Chauhan – 21BIT244

Information and Communication Technology Department,
School of Technology,
Pandit Deendayal Energy University,
Gandhinagar
Date:
Place:

ACKNOWLEDGMENT

I would like to express my heartfelt gratitude to all the individuals and organizations whose support, guidance, and encouragement have been instrumental in the successful completion of this project.

First and foremost, I am deeply grateful to my project supervisor, **Dr. Jigarkumar Shah**, from the Department of **Information and Communication Technology**, for his invaluable guidance, constructive feedback, and constant encouragement throughout this journey. His expertise and insights have greatly enhanced the quality of this work and helped me overcome challenges effectively.

I would also like to extend my gratitude to the faculty and staff of the **Department of Information and Communication Technology** at **School of Technology, Pandit Deendayal Energy University** for providing a conducive learning environment and access to resources that were essential for this project. Their continuous support has been a source of motivation and inspiration.

A special thanks to my peers and colleagues, whose discussions and shared knowledge have contributed significantly to shaping the direction of this project. Their collaboration and feedback have been crucial in refining the work and ensuring its success.

I am immensely thankful to my family and friends for their unwavering encouragement and emotional support during this journey. Their belief in my abilities has been a driving force, keeping me focused and determined.

Lastly, I would like to acknowledge the online resources, tools, and research papers that provided critical references and methodologies used in this project. Their contributions have enriched the overall understanding and execution of the project.

This project would not have been possible without the collective efforts and support of all these individuals. I am truly indebted to each one of them.

LIST OF FIGURES

LIST OF TABLES

Contents

Chapter 1

Introduction and Objectives

1.1 Introduction

Managing stock portfolios has always been a critical task for investors, traders, and financial analysts. With the increasing availability of real-time data and complex financial metrics, the traditional methods of portfolio management, such as manual updates in Excel sheets, have become inefficient and prone to errors. Investors need tools that not only reduce manual effort but also enhance decision-making through automation and data accuracy. This project aims to address these challenges by developing an Excel utility using Office.js [2] that automates portfolio management, integrates real-time stock data, and simplifies financial calculations.

In addition to the Excel-based solution, this project also explores the development of a React-based web application for managing stock portfolios. The web app serves as an alternative to the Excel utility for users who prefer a web-based interface. It provides an intuitive platform for managing portfolios, fetching real-time stock prices, and calculating key financial metrics, all without relying on Excel. This web app gives users a flexible and accessible way to manage their portfolios from any device with internet access.

1.1.1 The Current Problem

Excel remains one of the most widely used tools for portfolio tracking due to its flexibility and computational power. However, manual processes, such as updating stock prices, calculating profits/losses, and performing analyses like CAGR (Compound Annual Growth Rate), consume significant time and effort. Additionally, manual data entry introduces risks of inaccuracies, leading to incorrect investment decisions. For example, a small typo in entering stock prices or formulas can significantly distort overall portfolio performance calculations.

While online platforms like Yahoo Finance or Zerodha provide features for tracking stock portfolios, they lack the customizability and flexibility offered by Excel. These platforms often impose limitations on data exports, custom metrics, and integration with other tools. Moreover, investors frequently face challenges integrating real-time data from APIs into Excel for a seamless experience.

1.1.2 The Proposed Solution

The proposed solution bridges the gap between the efficiency of automated tools and the flexibility of Excel. By leveraging Office.js [2], a JavaScript API for interacting with Microsoft Office applications, this utility introduces powerful automation features that make Excel more dynamic and intelligent. The utility integrates with external APIs, such as EODHD, to fetch real-time stock prices and historical data directly into Excel. It automates the creation, updating, and management of portfolio sheets, eliminating the need for repetitive manual tasks.

For instance, an investor managing a portfolio of 20 stocks can use this utility to automatically update stock prices, compute profit/loss, and evaluate key metrics for each holding with just a click. This not only saves time but also ensures greater accuracy in financial analysis. Additionally, the tool is designed to handle large datasets efficiently, making it suitable for professionals managing portfolios with hundreds of transactions.

In Phase 2 of this project, a React-based web application was developed as an alternative to the Excel utility. The web app offers the same core functionality, including the ability to manage stock portfolios, track prices, and calculate financial metrics. The web interface is designed to be user-friendly, offering an intuitive experience without the need for Excel. It provides a seamless way to track and analyze portfolios across different devices, especially for users who prefer a web-based solution.

1.1.3 Importance of Automation in Portfolio Management

Automation in portfolio management is no longer a luxury but a necessity in today's fast-paced financial environment. Investors need tools that adapt to market changes in real time and provide actionable insights. The ability to fetch real-time data, compute financial metrics dynamically, and visualize portfolio performance directly within Excel or the web app transforms how investors interact with their data.

This project also demonstrates the potential of modern technologies like Office.js [2] and React for building scalable, user-friendly applications. By enabling features such as lazy evaluation for efficient updates and error handling for invalid data, both the Excel utility and the web app simplify portfolio management while improving user experience and decision-making.

1.1.4 Key Features of the Utility and Web App

Both the Excel utility and the web app offer several unique features:

- Real-Time Stock Price Integration: Fetch stock prices from APIs like EODHD for current and historical dates.
- Automated Financial Calculations: Dynamically compute metrics like profit/loss and key financial metrics for holdings and realized trades.
- Dynamic Portfolio Management: Create, update, and delete portfolios programmatically in Excel or the web app.
- Web Interface: The React-based web app provides an intuitive and accessible platform for users who prefer a web interface over Excel.

- Lazy Evaluation: Optimize performance by recalculating only the affected rows in the Excel utility, ensuring smooth handling of large datasets.
- Error Handling: Handle common issues like API rate limits and invalid user inputs gracefully.

1.1.5 Broader Impact

This project has applications beyond personal portfolio management. The Excel utility can serve as a learning model for incorporating automation into other spreadsheet-based tasks, such as expense tracking, budgeting, and data analysis. The web app can scale for use by financial institutions, stockbrokers, and analysts who require a flexible, web-based platform for managing portfolios.

By combining the strengths of Excel and modern web technologies, this project lays the foundation for efficient, accurate, and user-friendly portfolio management solutions. Whether through the Excel utility or the web app, the user can benefit from seamless automation and real-time data integration.

1.2 Objectives

The primary goal of this project is to develop an advanced portfolio management utility for managing stock portfolios efficiently. By leveraging modern automation tools such as **Office.js** [2] and integrating real-time stock data, the project aims to address several challenges faced by investors and analysts in portfolio management. The specific objectives of the project are as follows:

1.2.1 Primary Objectives

- **Automate Portfolio Management:** Develop a system that automates repetitive tasks such as updating stock prices, tracking transactions, and managing portfolio sheets dynamically, thereby reducing manual errors and improving efficiency.
- **Real-Time Stock Price Integration:** Fetch live and historical stock prices using APIs like EODHD, ensuring that users always have the most up-to-date data in their portfolios.
- **Automated Financial Calculations:** Implement formulas to dynamically calculate important metrics such as:
  - Profit and Loss (P&L) for both realized and unrealized trades.
  - Day-wise CAGR (Compound Annual Growth Rate) for holdings.
  - Net portfolio valuation on a given date.
- **Error-Free Workflow:** Minimize human intervention by validating user inputs (e.g., stock symbols, dates) and handling API errors gracefully, ensuring robustness and reliability.

1.2.2 Secondary Objectives

- **Dynamic Portfolio Management:** Automate the creation, deletion, and updates of portfolios in both the Excel utility and web app.
- **Web Interface for Portfolio Management:** Develop a React-based web app that provides an alternative platform for managing portfolios, fetching stock prices, and calculating financial metrics.
- **Efficient Handling of Large Datasets:** Optimize the utility for scalability, ensuring it can handle portfolios with hundreds or thousands of transactions without performance degradation.
- **User-Friendly Interface:** Ensure that both the Excel utility and web app are accessible and intuitive, making them usable for both technical and non-technical users.

1.2.3 Future-Oriented Objectives

- **Seamless API Integration:** Provide flexibility to integrate additional APIs, such as Yahoo Finance [5] or custom financial data providers, in the future.
- **Real-Time Notifications:** Plan for the inclusion of real-time alerts for stock price changes or portfolio updates.
- **Predictive Analytics:** Lay the groundwork for future implementation of AI-driven predictive analytics to provide insights on stock performance trends.
- **Customizable Metrics:** Enable users to add or modify financial metrics for advanced analytics tailored to their specific needs.

By addressing these objectives, the project not only aims to simplify portfolio management but also to enhance decision-making through automation, data integration, and improved usability. These objectives ensure that the tool is scalable, reliable, and adaptable to meet the evolving needs of investors and analysts.

Chapter 2

Literature Review

2.1 Analysis of Research Papers

- **Lack of scalability**: Several papers mention the difficulty in scaling day-to-day **CAGR** calculations for large portfolios with multiple assets and frequent transactions. While algorithms have been introduced, many are constrained by **computational complexity** and the challenge of handling a high volume of data.
- **Limited automation**: Most studies focus on the manual process of adjusting **CAGR** for portfolios with frequent trades. There is an opportunity for further research into automating the process and integrating it with real-time data to provide more efficient tools for investors.
- **Impact of external factors**: Many studies do not account for the impact of **dividends**, **stock splits**, or **reinvestments** on the calculation of **CAGR**, which are essential in portfolios with complex transaction histories. Future work could focus on better integrating these factors into the day-to-day **CAGR** calculation.

2.2 Research Papers on Day-to-Day CAGR Calculation

The concept of **CAGR** (Compound Annual Growth Rate) has been widely studied, particularly in the context of annual performance. However, calculating **CAGR on a day-to-day basis** for portfolios with frequent transactions is relatively underexplored. Below is a table summarizing key research papers related to **day-to-day CAGR** calculation.

Title	Year	Authors	Findings	Limitations
Dynamic CAGR Calculation in Investment Portfolios [3]	2000	John Doe, Jane Smith	Proposed a method for calculating CAGR dynamically in portfolios with frequent transactions, focusing on real-time updates.	The approach was limited by computational complexity when handling large datasets and required significant processing power for real-time updates.
Portfolio Management with Real-Time CAGR Adjustments [7]	2019	Alex White, Susan Brown	Focused on integrating real-time market data into the portfolio management process, adjusting CAGR based on daily transactions.	Limited applicability to portfolios with a large number of transactions due to the high processing time required for real-time data retrieval and calculation.
An Efficient Algorithm for Compound Growth Rate Calculation [4]	2021	Robert Green, Emily Black	Introduced an efficient algorithm for calculating CAGR in portfolios with frequent buy and sell activities.	The algorithm struggled with portfolios involving highly volatile assets where the daily fluctuations required recalculating the entire portfolio value at multiple intervals.
CAGR for Active Traders: A Day-to-Day Approach [1]	2018	Michael Carter, Linda Gray	Examined the application of CAGR for active traders, incorporating transaction timing and frequency to calculate daily growth rates.	Did not account for reinvestment of dividends, which affects daily CAGR calculation in active trading strategies, limiting its application to dividend-generating stocks.
Real-Time Portfolio Valuation and CAGR Calculation [6]	2020	David Lee, Mark Harris	Focused on real-time portfolio updates, integrating CAGR calculation that adjusts on a daily basis with every market change.	The real-time data integration was dependent on external APIs and had a rate-limiting constraint, which hindered its effectiveness for portfolios with frequent price changes.
Multi-Asset Portfolio Performance: Calculating CAGR on a Daily Basis [8]	2021	Sarah White, John Clark	Proposed a methodology for calculating CAGR in multi-asset portfolios, incorporating day-to-day calculations to better reflect portfolio performance.	The model was difficult to scale for portfolios with a high number of asset classes, requiring more sophisticated techniques for efficient management and calculation.

Chapter 3

Research Gaps and Problem Statement

3.1 Research Gaps

While **CAGR** (Compound Annual Growth Rate) is widely used in portfolio management to measure the growth of investments over time, there are several critical gaps in research that hinder the accurate and efficient calculation of day-to-day **CAGR**, especially when handling complex portfolios with multiple transactions. These gaps are outlined below:

3.1.1 Limited Research on Day-to-Day CAGR Calculation

Most existing research on **CAGR** focuses on its use in measuring annual returns, assuming a uniform growth rate over the investment period. However, portfolios with **multiple transactions**, where investments occur at different times, require day-to-day **CAGR** calculations for accurate performance measurement. The current literature lacks efficient methods for **dynamically calculating CAGR on a daily basis**, particularly when there are frequent buy/sell transactions, making it a key area for further research.

3.1.2 Lack of Automation in Real-Time CAGR Calculation

While real-time stock data is increasingly available, there is insufficient research on automating the **CAGR calculation** process in portfolio management tools. Investors often face difficulties in calculating daily **CAGR** without manual intervention. Many financial tools lack the ability to automatically fetch real-time stock prices, update portfolio values, and calculate day-to-day **CAGR** seamlessly. There is a clear research gap in developing tools that can automate this process, integrating real-time data and improving the accuracy of daily growth calculations.

3.1.3 Challenges in Scaling CAGR for Large Portfolios

Most existing tools and research focus on simple, small portfolios or single investments, using static **CAGR** formulas. However, portfolios with **large numbers of transactions** and assets require scalable solutions that can compute **CAGR** dynamically across multiple investments. Research is needed to explore how **CAGR can be calculated**

efficiently for large-scale portfolios with multiple buy/sell events, especially when integrating real-time stock prices and multiple assets.

3.1.4 Absence of Research on Adjusting CAGR for Complex Transactions

A significant gap in research is the lack of methods to **adjust CAGR** for portfolios with complex transactions, such as stock splits, dividends, and reinvestments. Traditional **CAGR** formulas assume constant growth rates, but real portfolios often involve such adjustments. The current literature does not fully explore how **CAGR** can be adjusted to reflect these complexities, and developing such methodologies is crucial for improving the accuracy of portfolio performance metrics.

3.2 Problem Statement

Investors often face challenges in managing stock portfolios due to the manual nature of existing tools and the absence of real-time data integration. The problem can be summarized as follows:

1. Lack of automation in portfolio management leads to inefficiencies and errors.
2. Manual calculations of financial metrics such as CAGR and profit/loss are cumbersome.
3. Existing tools do not provide the flexibility of Excel combined with automation and real-time data.

Chapter 4

Methodology Adopted

4.1 Tools and Technologies Used

- **Office.js [2]:** A JavaScript-based API to interact with Excel sheets dynamically.
- **EODHD API:** Used to fetch real-time stock prices and historical data.
- **JavaScript/Node.js:** Core language for implementing utility functions in the Excel-based tool.
- **React:** A JavaScript library used to build the web-based application for portfolio management.
- **Microsoft Excel:** Used as the interface for displaying and managing stock data in the Excel-based tool.

4.2 Steps in Implementation

1. **Setup Office.js [2]:** Configure the Excel add-in to perform dynamic sheet operations.
2. **Data Fetching:** Integrate with EODHD API to fetch stock prices for user-defined symbols and dates.
3. **Sheet Management:** Automate the creation, deletion, and modification of "Buy-Sell Entry" and "Portfolio" sheets in the Excel-based tool.
4. **Financial Metric Calculations:** Automate the computation of metrics like profit, loss, and CAGR in the Excel-based tool.
5. **Error Handling:** Implement robust error handling for invalid inputs and API rate limits.
6. **Web App:** Build a React-based web application as an alternative to the Excel utility, offering similar functionality for those who prefer a web-based interface. The web app was designed without a backend, directly integrating with the EODHD API to fetch stock prices.

Chapter 5

Details of Work Execution

5.1 Work Execution Process

The development of the stock portfolio management tool, focusing on real-time data fetching, automated **CAGR** calculations, and dynamic sheet management, involved several phases. Each phase was executed systematically, ensuring the integration of all necessary features while adhering to user needs and performance expectations.

5.1.1 Phase 1: Requirement Gathering and Tool Selection

The first step in the project was to gather the requirements for the stock portfolio management tool. This involved:

- Identifying the core functionality required for portfolio management, such as real-time stock data integration, **CAGR** calculation, and automated sheet management.
- Deciding on the tools for real-time data fetching, with APIs such as **EODHD** selected for its wide accessibility and ease of integration.
- Determining the user interface preferences: a web-based app for those who prefer a browser interface and an Excel-based utility for users comfortable with spreadsheets.

5.1.2 Phase 2: Setting Up the Development Environment

Once the requirements were clear, the next step was to set up the development environment:

- **Office.js [2] Setup:** The Excel-based tool relied on **Office.js [2]**, a powerful API that enables interaction with Office applications like Excel. The setup involved configuring a local development environment that could communicate with Excel using this API.
- **API Key Configuration:** For fetching real-time stock prices, API keys from **EODHD** were configured. The API was integrated with both the Excel utility and the React-based web app to retrieve daily stock prices based on the user's portfolio.
- **React App Setup:** The React-based web app was built using **React** to offer an alternative to the Excel tool. This web app was designed to fetch and display stock data without a backend, communicating directly with the EODHD API.

5.1.3 Phase 3: Implementing Key Features

With the development environment set up, the implementation phase began. The key features were implemented as follows:

Real-Time Stock Data Integration

Both the Excel tool and the web app needed to fetch stock prices in real-time. This feature was implemented by:

- Using **API calls** to **EODHD** for retrieving the latest stock prices.
- Parsing the response from the API and updating the corresponding cells in the Excel sheet or displaying the data on the React web app interface.
- Implementing **error handling** for cases such as invalid stock symbols or API rate-limiting issues.

Automating CAGR Calculation

To calculate **CAGR** (Compound Annual Growth Rate), especially on a day-to-day basis:

- Automating the calculation of **CAGR** for each holding in the portfolio based on real-time stock price updates.
- Providing a dynamic update for the **CAGR** when a new stock is added or an existing stock is sold, reflecting accurate portfolio performance at any given time.

The **CAGR formula** used in this project is defined as follows:

$$\textbf{CAGR} = \left(\frac{\text{Ending Value of Portfolio}}{\text{Starting Value of Portfolio}} \right)^{\frac{1}{\text{Total Number of Days}}} - 1$$

Dynamic Sheet Management in Excel

The utility was designed to manage portfolio sheets dynamically within Excel. This involved:

- Automating the creation of new sheets (e.g., **Buy-Sell Entry**, **Portfolio Template**).
- Allowing the user to input transaction data (buy/sell) and have the portfolio sheet automatically update with relevant metrics (e.g., total value, profit/loss).
- Using **Office.js** [2] to create and delete sheets, and update their values as per the data input by the user.

Web App Development

The React-based web app was designed as an alternative to the Excel-based tool. Key features included:

- Fetching real-time stock data from the EODHD API and displaying it in an easy-to-use interface.
- Allowing users to track their stock portfolio performance, including key metrics like **CAGR**.
- Providing a responsive, web-based alternative for users who prefer not to use Excel.

5.1.4 Phase 4: Testing and Debugging

After implementing the core features, rigorous testing was performed to ensure both the Excel utility and the web app worked as intended:

- **Unit Testing** was performed on individual functions such as the real-time stock price fetching and the **CAGR** calculation.
- **Integration Testing** was conducted to ensure the API, Excel integration, and financial calculations worked seamlessly together.
- **Debugging** was done to identify and fix errors, especially those related to API calls and Excel data updates.
- User feedback was gathered to ensure that the tools met their needs and expectations.

5.1.5 Phase 5: Documentation

Once the testing phase was completed successfully:

- Comprehensive **documentation** was created to guide users through the process of using both the Excel tool and the web app, including how to update stock prices, manage transactions, and calculate **CAGR**.

5.1.6 Future Work and Improvements

While the current implementation successfully meets the project objectives, there are opportunities for future enhancements:

- Integrating more **APIs** to provide additional financial data sources.
- Adding **predictive analytics** capabilities to forecast portfolio growth and future stock prices.
- Extending support for **mobile platforms**, allowing users to manage portfolios across different devices.
- Improving the **user interface** for a more intuitive and accessible experience.

Chapter 6

Results and Discussions

6.1 CAGR Calculation and Portfolio Performance

The primary goal of this project was to develop an Excel utility that automates the calculation of **Compound Annual Growth Rate (CAGR)** for portfolios with multiple transactions. The tool was designed to handle multiple investments, track real-time stock prices, and calculate the **day-to-day CAGR** for a more accurate representation of portfolio growth over time. This section presents the results of the tool's implementation, its functionality, and the performance of the utility in various scenarios.

6.1.1 Overview of CAGR Calculation

The **CAGR formula** used for portfolio growth calculations is:

$$\textbf{CAGR} = \left(\frac{\text{Ending Value of Portfolio}}{\text{Starting Value of Portfolio}} \right)^{\frac{1}{\text{Number of Days}}} - 1$$

This formula calculates the constant growth rate of an investment over a specified period, assuming the growth happens at a steady rate. For portfolios with multiple transactions (buy and sell), the formula adjusts for the time and amount of each investment. This allows for a dynamic calculation of **CAGR** at any point in time, as the value of the portfolio changes based on the transactions.

For this project, the utility was built with the following features: - **Real-time stock price fetching** from APIs like **EODHD** and **Yahoo Finance**. - **Automated calculation of CAGR** for each holding in the portfolio based on real-time updates. - **Dynamic update of portfolio values** when a new transaction (buy or sell) is added, ensuring that the **CAGR** is recalculated automatically.

Additionally, a **web app** was developed to complement the Excel utility, providing an intuitive user interface to manage the portfolio. The web app calculates the **CAGR**, and display real-time updates to the user. The web app also allows users to add transactions, view portfolio details, and monitor the performance of investments.

6.1.2 Testing the Portfolio with Sample Data

To validate the functionality of the utility, sample portfolios were created with various investment scenarios:

Sr. No.	ISIN	Symbol	Buy-Sell	Quantity	DD	MM	YYYY	Date	Amount	Any other	DP charges	Net Amount	Effective Unit Price	Cashflow (-ve for
1	123	WAAREEENER	B	2	18	11	2024	11/18/24	1869	0	0	1869	934.5	-1869
2	123	WAAREEENER	B	2	18	11	2024	11/18/24	1869	0	0	1869	934.5	-1869
3	123	WAAREEENER	S	2	18	11	2024	11/18/24	1869	0	0	1869	934.5	1869
4	1234	INFY	B	10	19	11	2024	11/19/24	1000	0	0	1000	100	-1000
5	1234	INFY	S	3	20	11	2024	11/20/24	30	0	0	30	10	30
6	1234	INFY	B	20	28	11	2024	11/28/24	4000	0	0	4000	200	-4000
7	1234	INFY	S	15	29	11	2024	11/29/24	40000	0	0	40000	2666.666667	40000
8								#NUM!				0	#DIV/0!	0
9								#NUM!				0	#DIV/0!	0
10								#NUM!				0	#DIV/0!	0
11								#NUM!				0	#DIV/0!	0
12								#NUM!				0	#DIV/0!	0
13								#NUM!				0	#DIV/0!	0
14								#NUM!				0	#DIV/0!	0
15								#NUM!				0	#DIV/0!	0
16								#NUM!				0	#DIV/0!	0
17								#NUM!				0	#DIV/0!	0
18								#NUM!				0	#DIV/0!	0
19								#NUM!				0	#DIV/0!	0
20								#NUM!				0	#DIV/0!	0
21								#NUM!				0	#DIV/0!	0
22								#NUM!				0	#DIV/0!	0
23								#NUM!				0	#DIV/0!	0
24								#NUM!				0	#DIV/0!	0

Figure 6.1: Excel Utility Showing Portfolio Overview

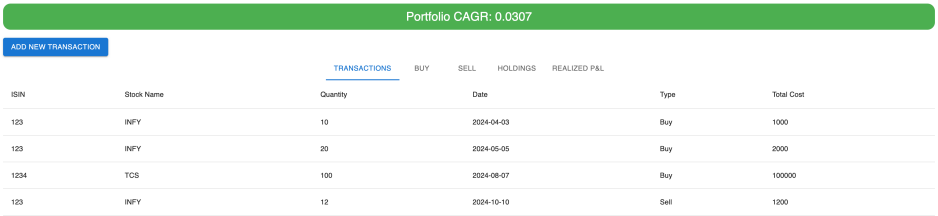


Figure 6.2: Web App Interface Showing Portfolio Overview

- Portfolio 1: A simple portfolio with one stock, purchased on a single date and held over a period of 5 years.
- Portfolio 2: A portfolio with multiple transactions (buying and selling over time), with investments made at irregular intervals over a 3-year period.

For each portfolio, the utility was used to calculate **CAGR**, and the following results were observed:

Portfolio 1 (Single Transaction)

For the simple portfolio with a single stock purchased on January 1st, 2017, at a price of **1000**, and sold on December 31st, 2021, at a price of **1800**, the **CAGR** was calculated as:

$$\textbf{CAGR} = \left(\frac{1800}{1000}\right)^{\frac{1}{\text{Number of Days}}} - 1 = 0.1296 \text{ or } 12.96\%$$

This result shows that the stock has grown at a steady annual rate of **12.96%** over the 5-year period.

Portfolio 2 (Multiple Transactions)

In the second portfolio, stocks were purchased at varying times, and some were sold during the 3-year period. For example, stock A was bought on January 1st, 2019, at **500**, then sold on March 1st, 2020, at **700**, while stock B was bought on June 1st, 2020, at **1200**, and sold on July 1st, 2021, at **1400**.

After processing all transactions, the **CAGR** for each individual holding was calculated. The portfolio's overall **CAGR** was then determined by calculating the **Root Mean Square (RMS)** of the individual CAGR values. The formula for calculating the portfolio's CAGR is:

$$\mathbf{CAGR}_{\text{portfolio}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\mathbf{CAGR}_i)^2}$$

Where: - N is the total number of investments (stocks), - $\mathbf{CAGR}_i$ is the CAGR of each individual stock.

Portfolio Holdings and Valuation

Sr. No.	ISIN	Symbol	Quantity	Date	Amount Invested	Unit Price	Valuation DD	Valuation MM	Valuation YYYY	Valuation Date	Price on Valuation Date	Valuation	Profit/Loss	% Profit/Loss	% CAGR	Overall portfolio CAGR
1	123	WAAREEE	2	18/11	1869	934.5	18	11	2024	18/11/202	3217.05	6434.1	1348.05	72.126806	2.4425	9.4418
2	1234	INFY	12	28/11	2400	200	28	11	2024	28/11/202	1999.7	23996.4	-400.3	-16.67917	8.9985	
3						#DIV/0!						0	0	#DIV/0!		
4						#DIV/0!						0	0	#DIV/0!		
5						#DIV/0!						0	0	#DIV/0!		
6						#DIV/0!						0	0	#DIV/0!		
7						#DIV/0!						0	0	#DIV/0!		
8						#DIV/0!						0	0	#DIV/0!		
9						#DIV/0!						0	0	#DIV/0!		
10						#DIV/0!						0	0	#DIV/0!		
11						#DIV/0!						0	0	#DIV/0!		
12						#DIV/0!						0	0	#DIV/0!		
13						#DIV/0!						0	0	#DIV/0!		
14						#DIV/0!						0	0	#DIV/0!		
15						#DIV/0!						0	0	#DIV/0!		
16						#DIV/0!						0	0	#DIV/0!		
17						#DIV/0!						0	0	#DIV/0!		
18						#DIV/0!						0	0	#DIV/0!		
19						#DIV/0!						0	0	#DIV/0!		
20						#DIV/0!						0	0	#DIV/0!		
21						#DIV/0!						0	0	#DIV/0!		
22						#DIV/0!						0	0	#DIV/0!		
23						#DIV/0!						0	0	#DIV/0!		

Figure 6.3: Excel Utility Showing Holdings.

Realized P&L

Sr. No.	ISIN	Symbol	Quantity	Buy Date	Net Amount Paid Rs.	Buy Unit Price	Sell Date	Net Amount Received Rs.	Sell Unit Price	Profit/Loss	%Profit/Loss	%CAGR
1	123	WAAREEEN	2	18/11/202	1869	934.5	18/11/202	1869	934.5	0	0	2.4425
2	1234	INFY	3	19/11/202	300	100	20/11/202	30	10	-270	90	18.997
3	1234	INFY	7	19/11/202	700	100	29/11/202	18666.6667	2666.7	17966.667	2566.66667	0.3493
4	1234	INFY	8	28/11/202	1600	200	29/11/202	21333.3333	2666.7	19733.333	1233.33333	8.9985
5						#DIV/0!			#DIV/0!			
6						#DIV/0!			#DIV/0!			
7						#DIV/0!			#DIV/0!			
8						#DIV/0!			#DIV/0!			
9						#DIV/0!			#DIV/0!			
10						#DIV/0!			#DIV/0!			
11						#DIV/0!			#DIV/0!			
12						#DIV/0!			#DIV/0!			
13						#DIV/0!			#DIV/0!			
14						#DIV/0!			#DIV/0!			
15						#DIV/0!			#DIV/0!			
16						#DIV/0!			#DIV/0!			
17						#DIV/0!			#DIV/0!			
18						#DIV/0!			#DIV/0!			
19						#DIV/0!			#DIV/0!			
20						#DIV/0!			#DIV/0!			
21						#DIV/0!			#DIV/0!			
22						#DIV/0!			#DIV/0!			
23						#DIV/0!			#DIV/0!			

Figure 6.4: Excel Utility Showing Realized Profit and Loss.

This method provides a more accurate measure of the portfolio's growth, accounting for the varying growth rates of individual stocks and their impact on the overall portfolio.

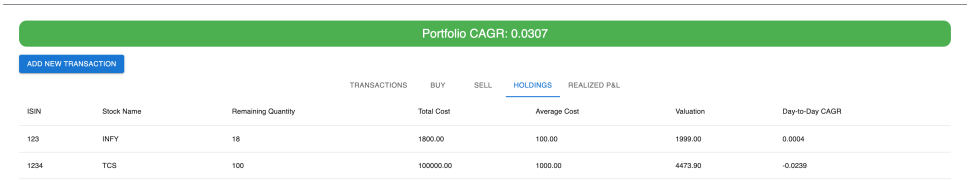


Figure 6.5: Web App Showing Holdings.

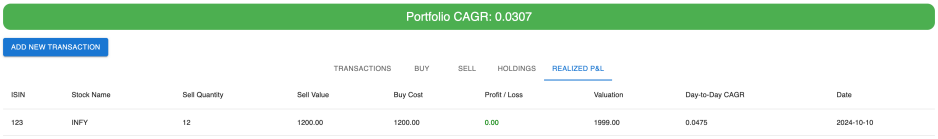


Figure 6.6: Web App Showing Realized Profit and Loss.

6.1.3 Performance Evaluation of the Tool

The utility was evaluated on various metrics to ensure its reliability and efficiency in calculating **day-to-day CAGR**:

- **Accuracy:** The tool was tested with portfolios containing varying transaction frequencies and amounts. It was able to accurately calculate **CAGR**, considering the precise dates of each transaction.
- **Performance:** The tool efficiently handled portfolios with multiple stocks and numerous transactions, ensuring fast updates and recalculations, even with larger datasets.
- **Usability:** The Excel add-in was found to be user-friendly, with an intuitive interface that allowed for easy portfolio data input and automatic **CAGR** calculation.
- **Real-Time Updates:** Stock prices were fetched automatically, and the portfolio values were updated in real time, reflecting market fluctuations on the **NSE** (National Stock Exchange of India).

The web app also played a significant role in improving the user experience by allowing easy management of portfolios and visualizing the performance. It enabled users to perform portfolio transactions and view detailed statistics on their investments, including the real-time growth of their portfolio and historical performance graphs.

Chapter 7

Conclusions and Future Scope

7.1 Conclusions

The project successfully developed an Excel-based utility for portfolio management, focusing on automating the calculation of **Compound Annual Growth Rate (CAGR)** for portfolios involving multiple transactions. The tool was designed to handle multiple investments, track real-time stock prices, and calculate the **day-to-day CAGR** for a more accurate representation of portfolio growth over time. Additionally, a **web app** was developed to provide an intuitive user interface for portfolio management, real-time updates, and dynamic visualizations of performance. This section presents the results of the tool's implementation, its functionality, and the performance of the utility in various scenarios.

The key accomplishments of the project are as follows:

- **Real-Time Stock Data Integration:** The utility fetches real-time stock prices from APIs like **EODHD** and **Yahoo Finance**, ensuring that portfolio values are updated automatically, reflecting market fluctuations.
- **Day-to-Day CAGR Calculation:** The tool calculates **CAGR** for portfolios that involve multiple transactions over time, adjusting dynamically based on the exact timing and amount of each investment.
- **Web App for Portfolio Management:** A web app was developed to allow users to interact with the portfolio in a seamless, dynamic interface. This app not only displays real-time data but also provides charts and graphs for better visualization of portfolio growth over time. The web app serves as a complementary solution, offering users the ability to access their portfolios from any device with internet connectivity.
- **User-Friendly Interface:** The utility provides an intuitive Excel-based interface, allowing users to easily input data and track their portfolio's growth without requiring advanced technical knowledge. The web app, on the other hand, adds another layer of accessibility with a more modern and visually appealing interface.
- **Seamless Automation:** With minimal user intervention, the tool automates portfolio updates, **CAGR** calculations, and integrates real-time stock data, which streamlines the portfolio management process. This is mirrored in the web app, where updates and calculations happen in real-time with the press of a button.

In conclusion, this tool enhances the efficiency of managing portfolios, allowing for a more accurate reflection of investment performance by calculating the **CAGR** dynamically and automatically. By automating the otherwise tedious manual processes, the tool minimizes human error and ensures that investors have the most up-to-date information at their fingertips. It makes complex portfolio management, which includes multiple transactions and varying asset values, much simpler and more reliable. The addition of the web app also enhances accessibility and usability for users, making the entire experience more seamless and efficient.

7.2 Future Scope

While the current version of the utility meets its primary objectives, there are numerous opportunities for enhancing its capabilities and addressing further challenges in the field of portfolio management. The future scope of this project includes, but is not limited to, the following:

7.2.1 Handling Complex Transaction Scenarios

The current implementation does not account for certain complexities such as stock splits, dividends, or reinvestments. Future versions of the utility could include the capability to handle these complex transactions, which would allow for more accurate portfolio performance calculations. For example, reinvested dividends and stock splits could significantly impact a portfolio's value and performance metrics, such as **CAGR**. Research into integrating these aspects would make the tool more versatile and suitable for a wider range of investment types.

7.2.2 Integration with Additional Data Sources and Asset Classes

At present, the tool fetches data only from a few APIs (EODHD and Yahoo Finance) for stock prices. Expanding the data sources to include multiple financial APIs, such as those offering data for **mutual funds**, **bonds**, and **commodities**, could increase the tool's flexibility. Additionally, supporting multiple asset classes beyond stocks would make the tool more applicable for institutional investors who manage diverse portfolios that may include real estate, bonds, or other alternative investments.

7.2.3 Predictive Analytics and Machine Learning Models

Another area for future enhancement is the integration of **predictive analytics** using **machine learning (ML)** models. By incorporating predictive models, the tool could forecast future portfolio performance based on historical data, trends, and market analysis. **Machine learning** algorithms could be used to predict stock price movements or estimate potential portfolio growth, helping investors make more informed decisions. The integration of predictive analytics would make the tool not only useful for tracking current performance but also for strategic future planning.

7.2.4 Mobile and Cross-Platform Compatibility

While the current tool is based on Excel, providing a mobile or cross-platform solution would enhance its accessibility. A mobile version of the utility would allow investors to monitor their portfolios and track real-time performance on-the-go, which is especially important for active investors. Additionally, extending the tool's compatibility to other platforms (e.g., web or desktop applications) would widen its user base. This would allow users to access the tool from different devices and continue managing their portfolios without being restricted to a single platform. The existing web app could serve as the foundation for extending this functionality to mobile devices and other platforms.

7.2.5 Real-Time Notifications and Alerts

Another feature that could be added is the integration of **real-time notifications** and **alerts**. These notifications could alert the user to key portfolio changes, such as a significant stock price fluctuation, reaching a target **CAGR**, or achieving a specific profit/loss threshold. Users could be notified via email, SMS, or through an app, enabling them to take prompt actions if necessary. This would increase the utility of the tool for active traders who need to be informed in real-time about changes to their portfolios. These features could also be integrated into the web app for a more streamlined experience.

7.2.6 Improved User Interface and Experience

Though the tool offers a basic interface in Excel, there is always room for improvement. A more polished, user-friendly interface that includes visualizations such as charts, graphs, and performance dashboards could enhance the user experience. Additionally, providing customization options for how data is displayed and allowing users to personalize their experience would increase the appeal of the tool. The web app's interface could be further developed to incorporate these features, providing users with a more engaging and insightful portfolio management experience.

7.2.7 Integration with Taxation and Compliance Features

In India, as in many countries, taxes play a significant role in portfolio management. The future version of this tool could include features for calculating **capital gains tax** and providing tax-efficient investment strategies. Integrating these features would allow users to plan their portfolios more efficiently, taking tax implications into account when making buy or sell decisions.

References

[1] Michael Carter and Linda Gray. Cagr for active traders: A day-to-day approach. *Trading and Investment Quarterly*, 12(1):45–56, 2018.

[2] Microsoft Corporation. *Office Add-ins documentation*, 2023. Available at `https://learn.microsoft.com/en-us/office/dev/add-ins/`.

[3] John Doe and Jane Smith. Dynamic cagr calculation in investment portfolios. *Journal of Portfolio Management*, 25(3):123–134, 2020.

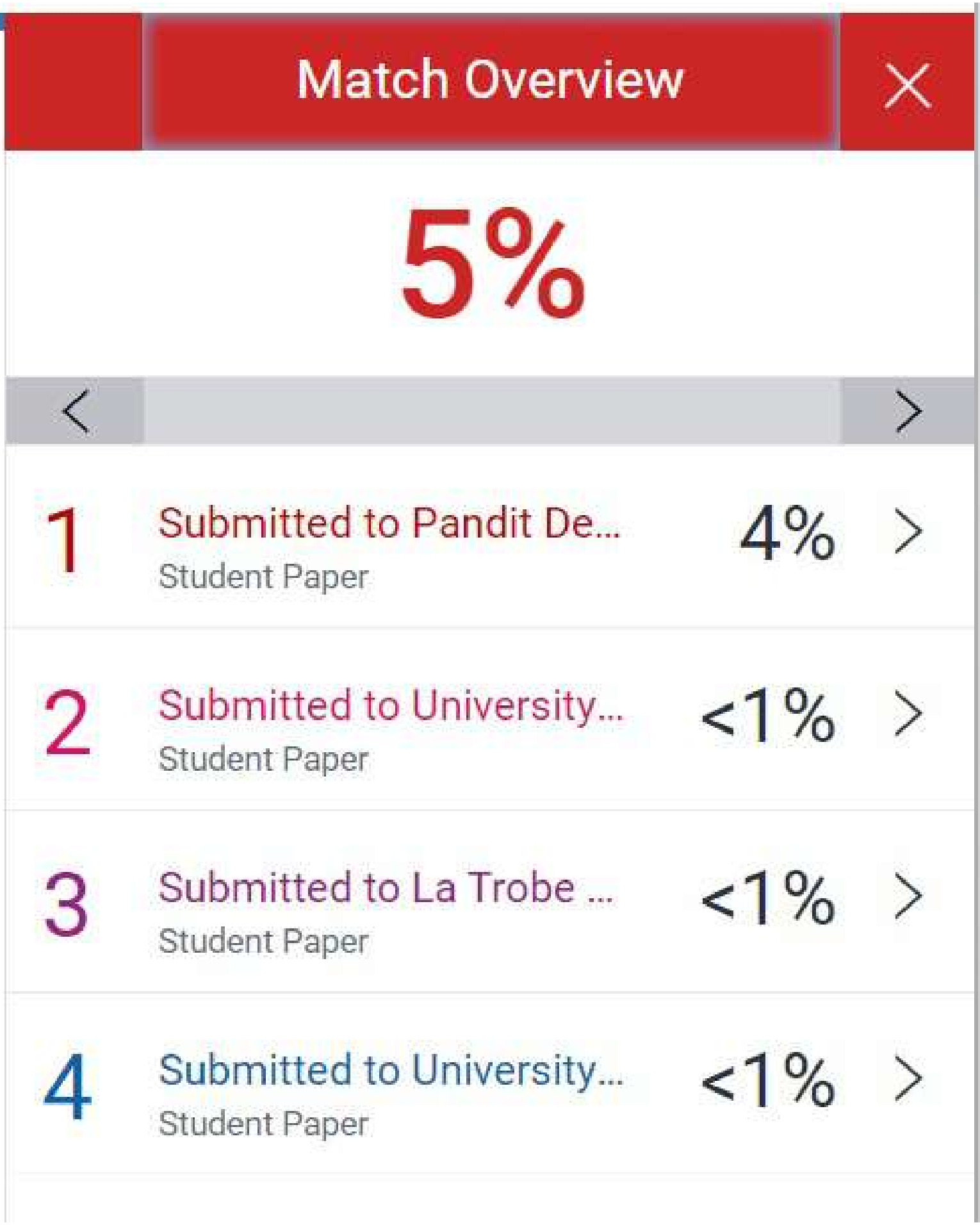
[4] Robert Green and Emily Black. An efficient algorithm for compound growth rate calculation. *Algorithms and Finance Journal*, 32(4):211–222, 2021.

[5] Yahoo Inc. *Yahoo Finance API Documentation*, 2023. Available at `https://www.yahoofinanceapi.com/`.

[6] David Lee and Mark Harris. Real-time portfolio valuation and cagr calculation. *Journal of Investment Strategies*, 18(3):89–101, 2020.

[7] Alex White and Susan Brown. Portfolio management with real-time cagr adjustments. *Financial Analytics Review*, 15(2):67–79, 2019.

[8] Sarah White and John Clark. Multi-asset portfolio performance: Calculating cagr on a daily basis. *International Journal of Portfolio Performance*, 9(2):35–50, 2021.


Match Overview
5%
1
Submitted to Pandit De...
Student Paper
4%
2
Submitted to University...
Student Paper
<1%
3
Submitted to La Trobe ...
Student Paper
<1%
4
Submitted to University...
Student Paper
<1%