

BULLRUN
A PROJECT REPORT
for
Mini Project (KCA353)
Session (2024-25)

Submitted by

Om Singhal
University Roll No:2300290140111

Piyush Chaturvedi
University Roll No:2300290140115

Ritakshi Singh
University Roll No:2300290140141

Ritika
University Roll No:2300290140145

Submitted in partial fulfilment of the Requirements for the Degree of

MASTER OF COMPUTER APPLICATION
Under the Supervision of
Dr Monika Kansal (Assistant Professor)



Submitted to
DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad Uttar Pradesh-201206

December 2024

CERTIFICATE

Certified that **Om Singhal (2300290140111), Ritakshi Singh (2300290140141), Piyush Chaturvedi (2300290140115), Ritika (2300290140145)** have carried out the project work having “**BULLRUN**” (**Mini-Project-KCA353**) for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date:

Om Singhal (2300290140111)
Ritakshi Singh (2300290140141)
Piyush Chaturvedi (2300290140115)
Ritika (2300290140145)

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

Monika Kansal
Associate Professor
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

Dr. Arun Tripathi
Head
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

ABSTRACT

BULLRUN is an innovative web-based platform focused on simplifying the investment experience for individuals in the stock market. The platform is designed to provide users with key information and tools to effectively monitor and manage their investments, ensuring they stay updated on relevant market opportunities and trends. BULLRUN offers a seamless way to track portfolios, giving users a clear overview of their investment performance and financial growth over time.

One of the key features of the platform is its provision of weekly investment suggestions, which helps users make informed decisions by offering insights into potential market movements and opportunities. Additionally, BULLRUN provides comprehensive information on upcoming and on-going Initial Public Offerings (IPOs), allowing investors to stay informed about new stock listings and market entrants.

Targeted at both beginners and experienced investors, BULLRUN is designed to enhance the investment experience without overwhelming users with complex analytical tools. Instead, it focuses on providing relevant, easily accessible information to help investors make sound decisions and achieve long-term financial success. By prioritizing user-friendly features such as investment tracking, IPO information, and weekly guidance, BULLRUN empowers users to navigate the stock market with confidence.

ACKNOWLEDGEMENTS

Success in life is never attained single-handedly. My deepest gratitude goes to my project supervisor, **Monika Kansal** for her guidance, help, and encouragement throughout my project work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to Dr. Arun Kumar Tripathi, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help on various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me with moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Om Singhal

Ritakshi Singh

Piyush Chaturvedi

Ritika

TABLE OF CONTENTS

Certificate	ii
Abstract	iii
Acknowledgements	iv
Table of Contents	v-vi
List of Tables	vii
List of Figures	viii
1 Introduction	1-4
1.1 Background	1
1.2 Objective	2
1.3 Key Feature	2
1.4 Scope	4
2 Problem Identification and Feasibility Study	5-9
2.1 Problem Identification	5
2.2 Feasibility Study	6
2.2.1 Technical feasibility Study	7
2.2.2 Operational Feasibility Study	8
2.2.3 Economical Feasibility Study	9
3 Requirement Analysis	10-12
3.1 Introduction	10
3.2 Functional Requirements	10
3.2.1 User Authentication	10
3.2.2 Portfolio Management	10
3.2.3 Stock market Insights	11
3.3 Non-Functional Requirements	12
3.3.1 Performance	12
3.3.2 Security	12
3.3.3 Scalability	12
3.3.4 Usability	12
4 Project Planning and Scheduling	13-16
4.1 Gantt Chart	13
5 Hardware and Software Specification	14
6 Choice of Tools and Technology	15
6.1 React	15
6.2 Mongo Db	16

TABLE OF CONTENTS

7	Project Flow	17-27
7.1	Data Flow Diagram	17
7.1.1	O level DFD	17
7.1.2	1 level DFD	18
7.2	ER Diagram	21
7.3	Class Diagram	25
8	Database	27
9	Design	28-32
1.1	Login	28
1.2	Signup	29
1.3	Home Page	30
1.4	Stocks	31
1.5	Stock Watchlist	32
10	Testing	33
11	Conclusion	34
12	References	35
13	Bibliography	36

LIST OF TABLES

Table No.	Name of Table	Page
4.1	Gantt Chart	32

LIST OF FIGURES

Figure No.	Name of Figure	Page No.
7.1	O Level DFD	10
7.2	1 Level DFD	12
7.3	ER Diagram	14
7.4	Class Diagram	17
10.1	Login	36
10.2	Signup	37
10.3	Home Page	37
10.4	Home Page	38
10.5	Stocks	38
10.6	Stocks Watchlist	39

Chapter 1

INTRODUCTION

1.1. Background

BullRun is a stock trading platform built using the MERN stack, which includes MongoDB, Express.js, React.js, and Node.js. The platform integrates real-time stock data, offering users an intuitive and seamless experience for tracking, analyzing, and trading stocks. By utilizing a **real-time data integration**, BullRun ensures that users receive up-to-the-minute stock prices, news, and performance insights, enabling them to make informed decisions quickly.

Designed with a **user-friendly interface** and **responsive design**, BullRun prioritizes accessibility and ease of use, catering to both beginner and experienced traders. The platform's **clean layout** and efficient navigation allow users to manage their portfolios, monitor stock performance, and execute trades with minimal effort. The use of **React.js** ensures that the interface remains dynamic, offering real-time updates and interactive elements without compromising performance.

BullRun aims to replicate the core functionalities of a modern stock brokerage platform while maintaining **aesthetic appeal** and ensuring that users have a streamlined experience. The platform incorporates key features like portfolio management, stock analysis tools, and customizable watchlists to enhance user engagement.

This project demonstrates **full-stack development expertise**, with a focus on **API integration** for financial data and **real-time data streaming**. It showcases a comprehensive understanding of how to build finance-centric web applications that are both functional and user-friendly.

Additionally, BullRun incorporates advanced features such as **portfolio tracking**, **real-time alerts**, and **market analysis tools**, allowing users to stay updated on stock price movements and trends. The platform also offers **customizable watchlists**, enabling users to monitor their favorite stocks and receive notifications on significant price changes or news events.

1.2. Objective

The objective of Bullrun is to provide a feature-rich and user-friendly platform that simplifies the process of stock trading and investment for users. By integrating real-time stock data, intuitive design, and essential trading functionalities, Bullrun aims to replicate the experience of professional brokerage platforms . It serves as a learning project to demonstrate expertise in full-stack development, real-time API integration, and responsive web design, while also highlighting the practical application of technology in the financial sector.

The primary goal of Bullrun is to empower users with a comprehensive stock trading platform that combines functionality with an intuitive and visually appealing interface. It aims to provide real-time market data, seamless trade execution, and detailed portfolio management tools, ensuring an engaging and efficient trading experience. Additionally, Bullrun seeks to bridge the gap between complex trading systems and user accessibility, making it suitable for both novice and experienced traders. The project also reflects a deep understanding of financial technology and web development, showcasing a commitment to building scalable, responsive, and robust applications.

1.3. Key Feature

BullRun offers a comprehensive suite of features designed to help users efficiently manage and optimize their investments. At its core, the platform provides **real-time tracking** of stock prices, allowing users to stay updated on market fluctuations and make timely decisions. The integration of **data analysis tools** enables users to assess stock performance, monitor trends, and evaluate potential opportunities with detailed insights and visualizations.

BullRun's **intuitive user interface** ensures that both novice and experienced traders can navigate the platform easily, providing access to key functions such as **portfolio management**, **watchlists**, and **custom alerts**. Users can organize their investments, track performance, and receive notifications for significant market movements, ensuring they remain informed at all times.

These features work together to create a seamless, user-friendly experience that empowers individuals to make data-driven, confident decisions while managing their investments effectively.

1.3.1. Real-Time Portfolio Tracking

This feature allows users to effortlessly input their stock purchases and track the performance of their portfolio in real time. By automatically calculating **gains and losses** over time, it provides users with an up-to-date view of their investment's performance. The platform displays detailed metrics such as **Return on Investment (ROI)**, enabling users to evaluate the success of their trades and overall portfolio. These insights help users make informed, data-driven decisions, whether it's to hold, sell, or buy additional stocks. With this feature, users can better understand their investment strategy, track their progress, and adjust their portfolios accordingly for optimal returns.

1.3.2. Live Stock Market Data

BullRun delivers **real-time stock prices** and **market trends**, ensuring users have immediate access to the latest market data. This up-to-the-minute information empowers users to track stock movements, make timely decisions, and stay ahead of market changes. The platform features **visual tools** like interactive **charts and graphs**, which display stock performance over different timeframes, helping users easily identify trends and price fluctuations.

Additionally, BullRun incorporates key **technical indicators**, such as moving averages and Relative Strength Index (RSI), which assist traders in making informed decisions based on real-time analysis. These indicators help users assess market conditions, identify potential entry and exit points, and optimize their trading strategies.

By combining real-time data with visualizations and technical analysis tools, BullRun offers a comprehensive solution that supports traders in navigating the market with confidence, making it easier to analyze, track, and respond to stock performance effectively.

1.3.3. Investment Analytics

This functionality allows users to review **historical stock data**, giving them a deeper understanding of their portfolio's performance over time. By analyzing past trends, users can assess how different stocks have performed, identify patterns, and gauge the effectiveness of their investment strategy. The feature also provides valuable insights into **portfolio diversification**, helping users see if their investments are spread across different sectors or assets, reducing risk.

Additionally, it highlights **risk levels**, offering an analysis of how volatile the portfolio is, based on historical price movements. This information empowers users to make adjustments to align with their risk tolerance and financial goals.

The platform also suggests **potential strategies** for optimizing returns, such as rebalancing the portfolio or exploring new sectors or stocks. With these insights, users are equipped to

make smarter, data-driven investment choices, improving their chances of achieving long-term financial success.

1.3.4. User-Friendly Interface

The frontend of **BullRun** is built with **React.js**, which ensures a **dynamic** and **responsive** design. React's component-based architecture allows for a seamless user experience, with quick updates and real-time data integration. The platform's interface is optimized for performance, ensuring that users can easily navigate through the site without delays, regardless of their device or screen size.

BullRun's user interface is designed with simplicity and clarity in mind, making it accessible to users of all experience levels. Whether users are **beginners** just starting to invest or **seasoned traders** familiar with stock analysis, the platform offers **intuitive controls** that make it easy to manage portfolios, track stock performance, and execute trades.

The responsive design ensures smooth navigation across various devices, whether desktop or mobile, providing a consistent experience. This focus on user-centric design makes BullRun an ideal choice for anyone looking to manage their investments efficiently.

1.4. Scope of the project

The scope of BullRun extends to the following:

1.4.1. User Interface and Experience:

The UI/UX scope of Bullrun focuses on creating a responsive, user-friendly, and visually appealing platform.

It includes intuitive navigation, real-time data visualization with interactive charts, and seamless workflows for trading and portfolio management.

Personalized dashboards, modern design aesthetics, and accessibility features ensure the platform caters to both beginners and experienced traders, delivering a smooth and engaging user experience.

The UI/UX scope of Bullrun emphasizes simplicity, efficiency, and visual appeal. It includes responsive design for all devices, clear navigation, and real-time stock data presentation. Features like easy trade execution, customizable dashboards, and modern aesthetics ensure the platform delivers an intuitive and engaging experience for users of all expertise levels.

CHAPTER 2

PROBLEM IDENTIFICATION & FEASIBILITY STUDY

2.1. Problem Identification

The primary problem addressed by BullRun is the complexity and inaccessibility of traditional stock trading platforms, which present significant barriers for both new and experienced users. Many existing platforms are built with outdated designs and lack intuitive navigation, creating a steep learning curve for beginners who are unfamiliar with stock trading concepts. Their cluttered interfaces often overwhelm users with excessive information presented in disorganized ways, leaving new traders confused and hesitant to engage fully with the market.

For advanced traders, traditional platforms frequently fail to offer sufficient customization options to meet their specific needs. Many lack the ability to personalize dashboards, filter data efficiently, or set custom alerts for important market events. This limits the ability of experienced users to optimize their trading strategies and make timely decisions. Additionally, the fragmented nature of many platforms—requiring users to toggle between separate tools for data analysis, portfolio tracking, and trade execution—adds unnecessary complexity to the process.

Another significant pain point is the lack of efficient access to real-time stock data. Delays in updates, coupled with the need to navigate cumbersome systems, can result in missed opportunities in a fast-paced trading environment. For users seeking a seamless experience, these inefficiencies can be frustrating and detrimental to their overall trading performance.

BullRun directly addresses these challenges by providing a streamlined, user-friendly solution. Its interface is designed with simplicity and responsiveness in mind, making it easy for beginners to navigate while offering advanced tools for seasoned traders. Real-time data integration ensures users have access to up-to-date stock information at their fingertips, empowering them to act quickly on market changes. Customization options, such as personalized watchlists, performance alerts, and intuitive data visualization tools, cater to the unique preferences of individual users.

By combining ease of use with powerful functionality, BullRun bridges the gap between accessibility and efficiency in stock trading. It reduces barriers for beginners while delivering the advanced features that experienced traders demand, creating a platform that serves all levels of users. With BullRun, stock trading becomes more intuitive, engaging, and effective, empowering users to take control of their financial futures.

2.2. Feasibility Study

The feasibility study of BullRun demonstrates that the project is both technically and economically viable, with strong potential for successful implementation and market adoption. By leveraging the MERN stack (MongoDB, Express.js, React, Node.js), BullRun is built on a robust, modern, and highly scalable framework. This technology stack ensures the platform can handle the complexities of real-time stock data processing, user interactions, and seamless trade executions. React.js provides an interactive and responsive user interface, while Node.js and Express.js manage server-side functionality efficiently. MongoDB, with its flexible schema and scalability, ensures smooth storage and retrieval of large datasets, such as user profiles, watchlists, and historical market data.

The integration of financial APIs for real-time market data is both technically achievable and secure. APIs from trusted providers can be implemented to deliver up-to-date stock prices, news, and analytics to users, ensuring accurate and timely information for decision-making. Modern security protocols, including data encryption and secure API communication, further enhance the platform's reliability and user trust. The system's architecture can also be optimized to scale efficiently as the user base grows, ensuring the platform's technical sustainability over time.

From an economic standpoint, BullRun is a cost-effective project. By utilizing open-source technologies like React, Node.js, and MongoDB, the development costs are significantly reduced. These technologies eliminate the need for expensive proprietary software while offering flexibility and scalability. The cloud-based deployment of the platform also reduces infrastructure expenses, making the project more economically sustainable.

Additionally, the market demand for user-friendly, accessible trading platforms is steadily increasing. The shift toward digital finance and the growing popularity of self-managed investments present a prime opportunity for BullRun. Its focus on simplifying trading processes, providing real-time data, and ensuring a seamless user experience makes it well-positioned to attract both novice and experienced traders. The combination of a scalable technical foundation, cost-effective development, and strong market demand confirms the feasibility of BullRun as a commercially viable project with the potential to make a significant impact in the financial technology space.

2.2.1. Technical Feasibility

The technical feasibility of BullRun is exceptionally high, thanks to its utilization of the MERN stack, which is a widely adopted and robust technology stack for web development. This stack is highly regarded for its flexibility, scalability, and ability to support dynamic, data-intensive applications like a stock trading platform.

At the core of the MERN stack, **MongoDB** provides efficient and scalable data storage, making it ideal for managing large volumes of data generated by users, stocks, and real-time market updates. The document-oriented nature of MongoDB enables fast data retrieval and flexible schema design, which is crucial for storing diverse financial information and user profiles. This also allows the platform to scale easily as the amount of data and user activity increases over time.

Node.js and **Express.js** play a vital role in handling server-side logic and ensuring smooth API integration. Node.js is a fast, non-blocking JavaScript runtime environment that allows for handling numerous simultaneous user requests, essential for a real-time stock trading platform. Express.js, a lightweight web framework built on Node.js, facilitates seamless routing, middleware integration, and API management, ensuring efficient communication between the front end and back end.

React.js is used to create a responsive and dynamic user interface, which is critical for an application where users interact with complex real-time data. React's component-based architecture allows for a modular approach to building the interface, making it easy to update and maintain individual components. Its reactivity ensures that users receive immediate feedback, such as real-time stock price changes, making the trading experience seamless and user-friendly.

The integration of **third-party financial APIs** ensures that BullRun can provide timely, accurate, and real-time stock market data. APIs from reputable financial data providers allow for continuous market updates, delivering essential price movements, trading volumes, and other market indicators. This ensures that users always have access to the most up-to-date information for making informed trading decisions.

Overall, the MERN stack provides the perfect foundation for building a secure, scalable, and dynamic trading platform. With its combination of efficient data handling, real-time functionality, and user-friendly interfaces, BullRun is well-positioned to offer a seamless trading experience for both novice and advanced users. The technical feasibility of the project is not only high but also sustainable, ensuring that the platform can evolve with future demands and advancements in the financial technology space.

2.2.2. Operational Feasibility

The operational feasibility of BullRun is exceptionally strong, primarily due to its user-centric design and the robust technologies that power its functionality. One of the core strengths of the platform lies in its **user-friendly interface**, which is crafted to ensure intuitive navigation for both beginners and experienced traders. This simplicity minimizes the learning curve often associated with traditional stock trading platforms, making it accessible to users with varying levels of financial knowledge. By presenting market data in a clean and organized manner, BullRun enables users to quickly grasp essential information and make informed decisions, fostering a positive and efficient trading experience.

The platform's **responsive design** further enhances its operational feasibility by ensuring smooth functionality across a wide range of devices, including desktops, tablets, and smartphones. This cross-platform compatibility ensures that users can access their portfolios, monitor stock prices, and execute trades on-the-go, creating a seamless experience regardless of the device they are using. The use of **React.js** guarantees that the platform dynamically updates in real-time, providing users with the most current stock data and price movements without the need for page reloads, which is crucial for maintaining an efficient trading environment.

BullRun also stands out for its **easy maintenance** and the ability to scale seamlessly as user demand increases. The modular architecture provided by the MERN stack allows for quick updates and additions of new features without disrupting the core functionality of the platform. Whether adding new financial tools, incorporating advanced charting capabilities, or integrating additional APIs for market data, the platform can evolve over time to meet growing user expectations. Furthermore, the integration of existing **support systems**, such as user authentication, payment gateways, and communication channels, simplifies operational management by consolidating essential services into a cohesive system.

The use of well-established technologies like MongoDB, Express.js, React, and Node.js ensures that BullRun remains easy to manage and maintain. These technologies are supported by large, active communities, providing access to resources, documentation, and tools that streamline development and troubleshooting. Additionally, as the user base grows, BullRun can be easily scaled to accommodate higher traffic and data volumes, ensuring that performance remains optimal even during periods of increased demand.

Overall, BullRun's operational feasibility is high, thanks to its intuitive design, scalability, ease of maintenance, and integration capabilities, making it a reliable and efficient platform for users and administrators alike.

2.2.3. Economic Feasibility

The economical feasibility of BullRun is highly favorable due to its strategic use of open-source technologies and cost-effective integration methods. The project leverages the **MERN stack** (MongoDB, Express.js, React.js, and Node.js), all of which are open-source and widely supported. By relying on these technologies, BullRun significantly reduces development and licensing costs. The open-source nature of the MERN stack eliminates the need for expensive proprietary software or licensing fees, making the platform both budget-friendly and flexible. Furthermore, these technologies are well-documented and supported by active communities, which reduces development time and the need for costly technical expertise, enhancing cost efficiency.

For real-time stock data, BullRun can integrate with third-party financial **APIs**. Many financial data providers operate on a subscription-based model, allowing BullRun to access up-to-date stock prices and market data at a predictable cost. This approach eliminates the need for building and maintaining an expensive data infrastructure in-house, reducing the financial burden on the project. Additionally, subscription-based APIs allow for scalability in terms of data access, providing flexibility as the platform grows. As the user base increases, BullRun can adjust its subscription plans to accommodate higher demand for data, ensuring a cost-effective solution that adapts to evolving needs.

BullRun's **scalability** also plays a key role in its economic feasibility. The platform is designed to grow with demand without requiring significant reinvestments in infrastructure. The use of cloud-based services for hosting, paired with the flexibility of the MERN stack, ensures that the platform can scale efficiently. As user numbers rise, additional resources can be allocated seamlessly, avoiding the need for costly hardware or major infrastructure changes. This scalability ensures that BullRun remains financially viable even as it expands its user base.

From a commercial perspective, the growing demand for **user-friendly fintech solutions** in the stock trading space presents a significant opportunity. BullRun's focus on accessibility, intuitive design, and real-time data makes it an attractive solution for a wide range of users. The platform can generate revenue through **premium features** such as advanced analytics, customized stock alerts, and additional data feeds. Partnerships with financial institutions or brokers could also provide additional revenue streams.

In summary, BullRun offers a **cost-effective solution** by utilizing open-source technologies, integrating affordable third-party APIs, and providing scalable infrastructure. This makes the platform economically viable both in terms of development and future growth, with strong potential for profitability through commercial applications and premium offerings in the expanding fintech market.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1. Introduction

Bullrun, it involves gathering detailed information about the functional and non-functional requirements, including the features needed for the stock trading platform, such as real-time data integration, user authentication, trade execution, and portfolio management. This phase also includes analyzing performance, security, scalability, and usability requirements to ensure the platform meets both user and business goals. A thorough requirement analysis forms the foundation for designing and developing a successful, efficient, and user-centered application.

3.2. FUNCTIONAL REQUIREMENTS

3.2.1. User Authentication

Allow secure access to the platform by verifying the identity of users before granting them access to sensitive information and features. It includes features such as user registration, login, password recovery, and role-based access control. The authentication process is implemented using secure technologies like JWT (JSON Web Tokens) for session management, ensuring that user data is protected. Additionally, the system supports secure password hashing to safeguard against unauthorized access and enhances overall security for both individual traders and the platform.

3.2.2. Portfolio Management

It is a key feature of BullRun that empowers users to efficiently track, manage, and analyze their stock investments in real time. This module provides users with a comprehensive overview of their current holdings, displaying relevant information such as stock quantity, purchase price, and current market value. Users can easily monitor the performance of their investments, as the portfolio is continuously updated with real-time stock data, ensuring they have accurate, up-to-the-minute information.

Additionally, the module offers tools to calculate **gains and losses**, allowing users to assess the profitability of their investments over different periods. This feature helps traders understand how their portfolio is performing, highlighting potential areas for improvement or adjustment. A **financial summary** is provided, which gives users a clear snapshot of their overall financial status, including net worth, portfolio diversity, and performance trends.

The real-time integration with market data ensures that portfolio values reflect the latest price fluctuations, allowing users to make informed, data-driven decisions about their trades. The system also allows for easy rebalancing of the portfolio, helping users optimize their investment strategy according to market conditions and personal goals. With this intuitive and organized view, users are equipped to manage their investments more effectively and confidently, all in one place.

3.2.3. Stock Market Insights

The **Package Module** in BullRun delivers essential real-time market data and analytical tools designed to help users make informed trading decisions. Key features of this module include **live stock tickers**, which display up-to-the-minute stock prices and market movements, ensuring users stay updated with the latest data. Interactive **charts** allow for dynamic visual analysis of stock trends, enabling users to zoom in on specific time frames and identify patterns in price fluctuations.

Additionally, the module offers **historical performance analysis**, helping users evaluate past trends and make data-driven predictions for future investments. **Market news and trends** are also integrated, providing users with relevant updates on the broader financial landscape, including economic events and industry developments that may impact their investments.

The module aggregates data from various sources, offering insights into important market metrics like stock price movements, trading volume, volatility, and market sentiment. These insights are presented through clear, easy-to-understand **visualizations**, helping users make sense of complex data and enabling them to take timely actions.

By offering a comprehensive overview of the market and stock performance, the Package Module empowers users to stay ahead of market conditions, identify opportunities, and make strategic investment choices. It equips traders with the tools necessary to navigate the fast-paced stock market environment effectively and confidently.

3.3. NON-FUNCTIONAL REQUIREMENTS

3.3.1. PERFORMANCE

- **Response Time:** Ensure that pages load quickly to provide a seamless user experience, especially during more data access times.
- **Scalability:** The website should be able to handle increasing user loads without significant degradation in performance.
- **Reliability:** Minimize downtime and ensure high availability to prevent disruptions in service.

3.3.2. SECURITY

- **Data Encryption:** Implement secure communication protocols (e.g., HTTPS) to encrypt data transmission between the user's browser and the server.
- **Authentication and Authorization:** Ensure secure user authentication mechanisms and role-based access control to protect user accounts and sensitive information.
- **Data Protection:** Adhere to data protection regulations such as GDPR (General Data Protection Regulation) to safeguard user data and privacy.

3.3.3. SCALABILITY

- **Horizontal Scalability:** Design the website architecture to scale horizontally by adding more servers or resources to handle increased traffic.
- **Vertical Scalability:** Ensure that the website can scale vertically by optimizing code, databases, and server configurations to handle increased load on existing resources.

3.3.4. USABILITY

- **Intuitive Interface:** The website should be easy to navigate, with clear menus, intuitive search functionalities, and user-friendly design elements.
- **Accessibility:** Ensure that the website is accessible to users with disabilities, complying with accessibility standards such as WCAG (Web Content Accessibility Guidelines).
- **Multilingual Support:** Provide support for multiple languages to cater to a diverse user base.

CHAPTER 4

PROJECT PLANNING AND SCHEDULING

4.1. Gantt Chart

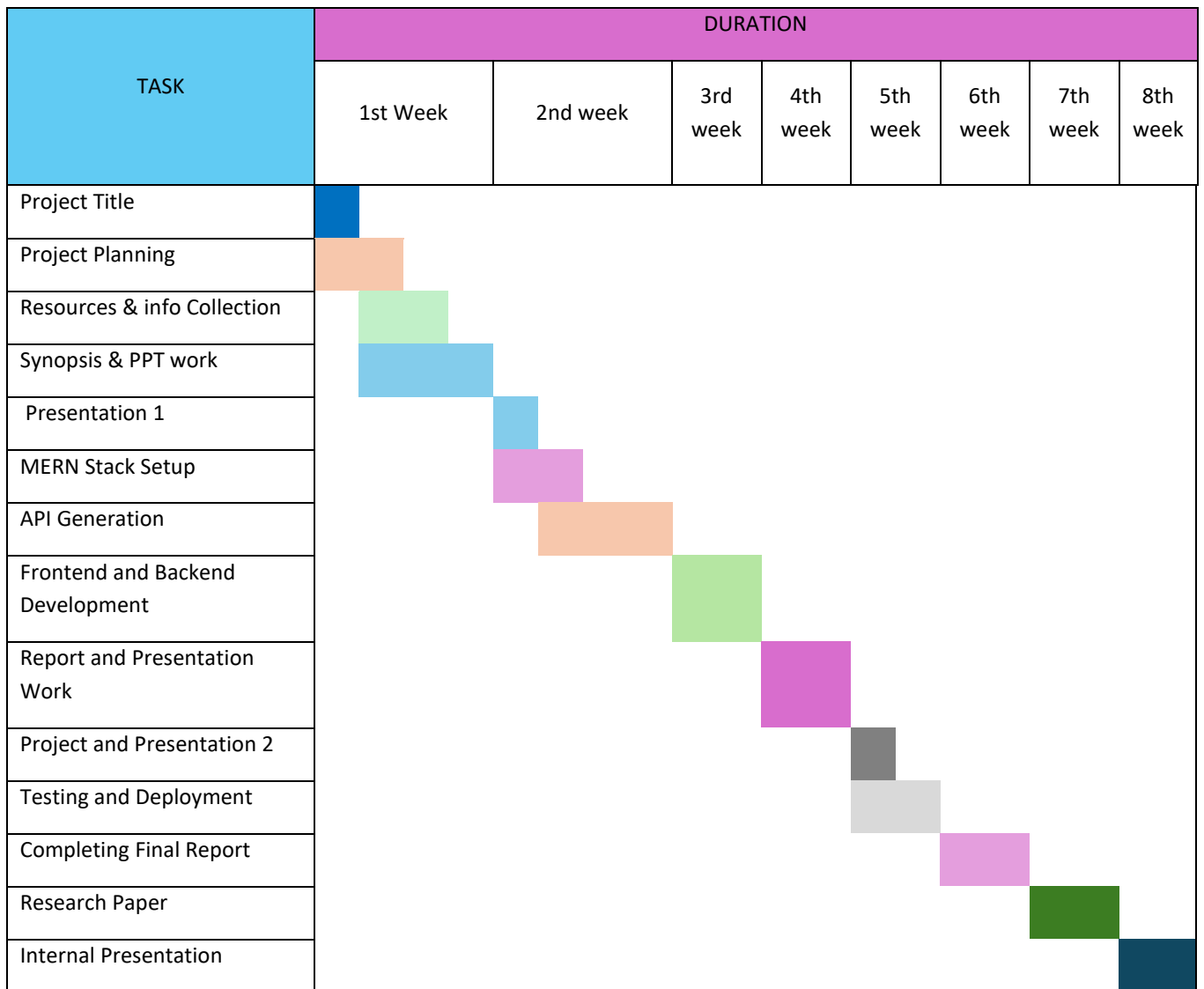


Fig.4.1

CHAPTER 5

HARDWARE & SOFTWARE SPECIFICATION

5.1. Hardware Specification

The BullRun will be developed and deployed on a hardware infrastructure that ensures optimal performance and reliability. The recommended hardware specifications are as follows:

5.1.1. Server:

- **Processor:** Intel Core i5 or equivalent RAM: 8 GB or higher
- **Storage:** 256 GB SSD or higher

5.1.2. Database Server:

- **Processor:** Intel Core i5 or equivalent RAM: 8 GB or higher
- **Storage:** 256 GB SSD or higher Network Interface: Gigabit Ethernet

5.1.3. RAM: 4 GB or higher

5.1.4. Storage: 128 GB SSD or higher

5.1.5. Network Interface: 100 Mbps Ethernet or Wi-Fi

5.2. Software Specification

The Stock trading website's BullRun will be developed using a combination of server-side and client-side technologies within the MERN (MongoDB, Express.js, React.js, Node.js) stack. The software specifications for the traveling website encompass:

5.2.1. Server-Side Technologies:

- **Operating System:** Windows Server 2016 or later Web Server: Node.js
- **Database Management System:** MongoDB
- **Server-Side Scripting Language:** JavaScript (Node.js)

5.2.2. Client-Side Technologies:

- **Web Browser:** Latest versions of Chrome, Firefox, Safari, or Edge Client-Side Scripting: JavaScript

5.2.3. Development Tool- Integrated Development Environment (IDE): Visual Studio Code

5.2.4. Version Control: Git: Version control for collaborative development

5.2.5. Security:

- **SSL/TLS:** Ensure secure data transmission over the network Firewall: Implement firewall rules to restrict unauthorized access

CHAPTER 6

CHOICE OF TOOLS & TECHNOLOGY

6.1 React

React is a powerful JavaScript library developed by Facebook for building user interfaces, particularly for single-page applications where a dynamic, fast, and interactive experience is essential. React allows developers to describe how the UI should appear at any given point in time. When data changes, React automatically updates and re-renders only the necessary components, ensuring that the UI stays in sync with the underlying data without manual intervention.

One of the core features of React is its **component-based architecture**. The UI is broken down into small, reusable components, each responsible for rendering a part of the user interface. This modular structure makes it easier to manage, maintain, and scale complex applications. Components in React can be both **stateful** (containing their own internal state) or **stateless** (relying solely on the data passed to them). By breaking the UI into independent, self-contained pieces, React simplifies development and testing, allowing developers to focus on small parts of the application at a time.

React utilizes a **virtual DOM**, a lightweight copy of the actual DOM, to improve performance. When a component's state or props change, React first updates the virtual DOM, then compares it with the previous version of the virtual DOM. This process, known as **reconciliation**, helps determine the most efficient way to update the actual DOM by minimizing unnecessary re-renders. This mechanism significantly enhances performance, especially in large applications where frequent updates are required.

Another key feature of React is **JSX** (JavaScript XML), a syntax extension that allows developers to describe the structure of UI components in a way that resembles HTML or XML. JSX enables developers to write components more concisely and in a readable format, mixing JavaScript logic with HTML-like structure. React components can also receive data through **props** (short for properties), allowing for dynamic and interactive user interfaces. With **state** and **props**, React provides a flexible mechanism for managing data and rendering dynamic content based on user interactions.

Overall, React's component-based structure, virtual DOM, JSX syntax, and efficient state management make it an excellent choice for building modern, interactive web applications. It enables developers to create high-performance, maintainable UIs while ensuring a seamless user experience.

6.2 MongoDB

MongoDB is a NoSQL database management system that excels in storing and managing large volumes of unstructured or semi-structured data. Unlike traditional relational databases that rely on predefined schemas, MongoDB uses a flexible, JSON-like format called BSON (Binary JSON). This structure allows developers to store data in a way that closely mirrors real-world objects, enabling dynamic and adaptable data models.

The BSON format supports complex data types, nested fields, and arrays, making it ideal for applications where data structures evolve over time. For example, an e-commerce application might store product details in varying formats, such as adding new fields for specific categories without disrupting existing data or requiring database migrations.

MongoDB's schema-less nature allows developers to iterate quickly, adapt to changing requirements, and handle heterogeneous datasets. It also supports horizontal scaling through sharding, enabling distributed storage and high availability across multiple servers. This makes MongoDB suitable for applications requiring scalability, such as social media platforms, real-time analytics, and IoT systems.

Additionally, MongoDB offers robust querying capabilities, indexing, and aggregation frameworks, ensuring efficient data retrieval. Its flexibility and performance make it a popular choice for modern applications that demand agility and scalability in managing diverse and dynamic datasets.

CHAPTER 7

PROJECT FLOW

7.1 Data Flow Diagram

A Data Flow Diagram (DFD) is a visual tool used to map out the flow of data within a system, illustrating how information is input, processed, stored, and output. It is a type of flowchart designed to represent the movement of data, highlighting the interactions between various system components. DFDs are widely used in system design to help stakeholders and developers understand the structure and operation of a system.

A DFD typically consists of four key elements: external entities, processes, data stores, and data flows. **External entities** represent sources or destinations of data outside the system, such as users or external systems. **Processes** depict operations or transformations applied to data, such as calculations or decision-making steps. **Data stores** show where data is stored, such as a database or file. **Data flows** are arrows that illustrate the movement of data between entities, processes, and stores.

For instance, in an online shopping system, a DFD might show how customer orders are received (external entity), processed for payment and inventory updates (processes), stored in a database (data store), and sent as order confirmations (data flow). This visual clarity helps identify inefficiencies, redundancies, or missing components in the system.

7.1.1. Zero Level DFD

It represents a high-level view of a system focused on stock transactions and user interactions.

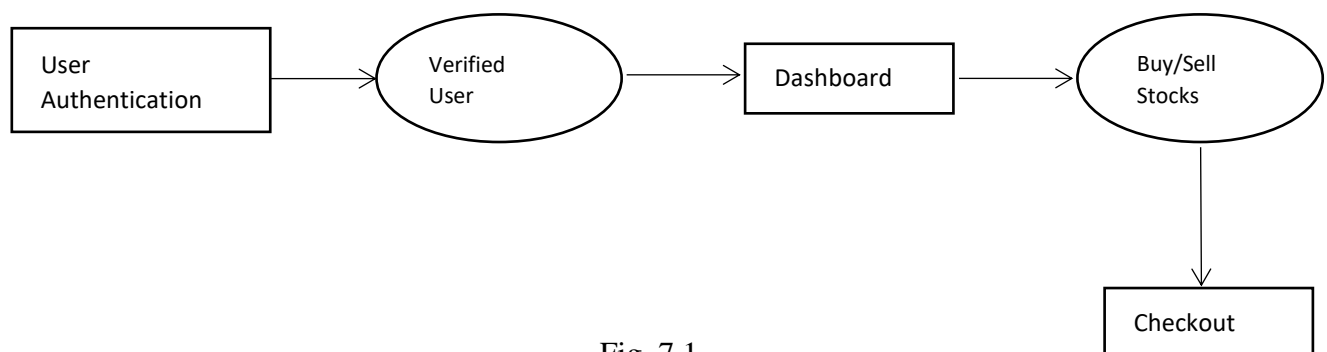


Fig. 7.1

1. **User Authentication:** This process handles the authentication of users when they log into the system or sign up.
 - **Input:** The system takes in User Data, which includes credentials such as username, email, and password.
 - **Output:** After verifying the user credentials, the system will output:
 - **Verified User:** A verified user is allowed to access further system functionality.
 - **Not Verified:** If the credentials are incorrect or not found, the user is not verified and cannot proceed.
 - If it's the signup process, the system will output a Signup Successful notification.
2. **Verified User:** After authentication, only verified users can proceed to access the main features of the application.
 - **Input:** This is the output of the User Authentication process, representing users who have successfully logged in or signed up.
 - **Output:**
 - A dashboard containing an overview of stock information, portfolio, watchlist, and transaction history is shown to the user.
 - The verified user also gains access to various functionalities, such as Buy/Sell Stocks.
3. **Dashboard:** The dashboard is the main interface after the user is verified. It provides a central location for various operations, such as stock analysis, viewing a portfolio, and accessing other features.
 - **Input:** The Verified User accesses this after authentication.
 - **Output:** The dashboard provides options to the user, including:
 - The user can initiate buying or selling of stocks from the dashboard.
 - It also serves as the interface for checking details like the user's portfolio, stock analysis, and related information.
4. **Buy/Sell Stocks:** This process allows the user to perform stock trading activities like buying and selling.
 - **Input:** The user interacts with the Dashboard to initiate stock trading actions.
 - **Output:** The system sends a request for buying/selling to be processed. If successful, the transaction is confirmed, and the user's balance or portfolio is updated.
5. **Checkout:** The checkout process handles the final steps of a transaction, such as confirming and processing payment for the stocks bought or sold.
 - **Input:** The Buy/Sell Stocks process leads into this step, where the system confirms the completion of the transaction.
 - **Output:** This results in the finalization of the stock transaction, reflecting the purchase or sale of stocks in the user's portfolio.

7.1.2. One level DFD

Data Flow of the project goes as follow:

1. User Registration and login:

- The process begins with users signing up for the platform or logging in if they already have an account.
- During sign-up, users provide their data, which is verified to ensure authenticity. If the user is not verified, they cannot access the platform.

2. Access to Dashboard:

- Upon successful login and verification, users are directed to the dashboard.
- The dashboard acts as the central hub, giving users access to their portfolio, watchlist, and the available stocks.

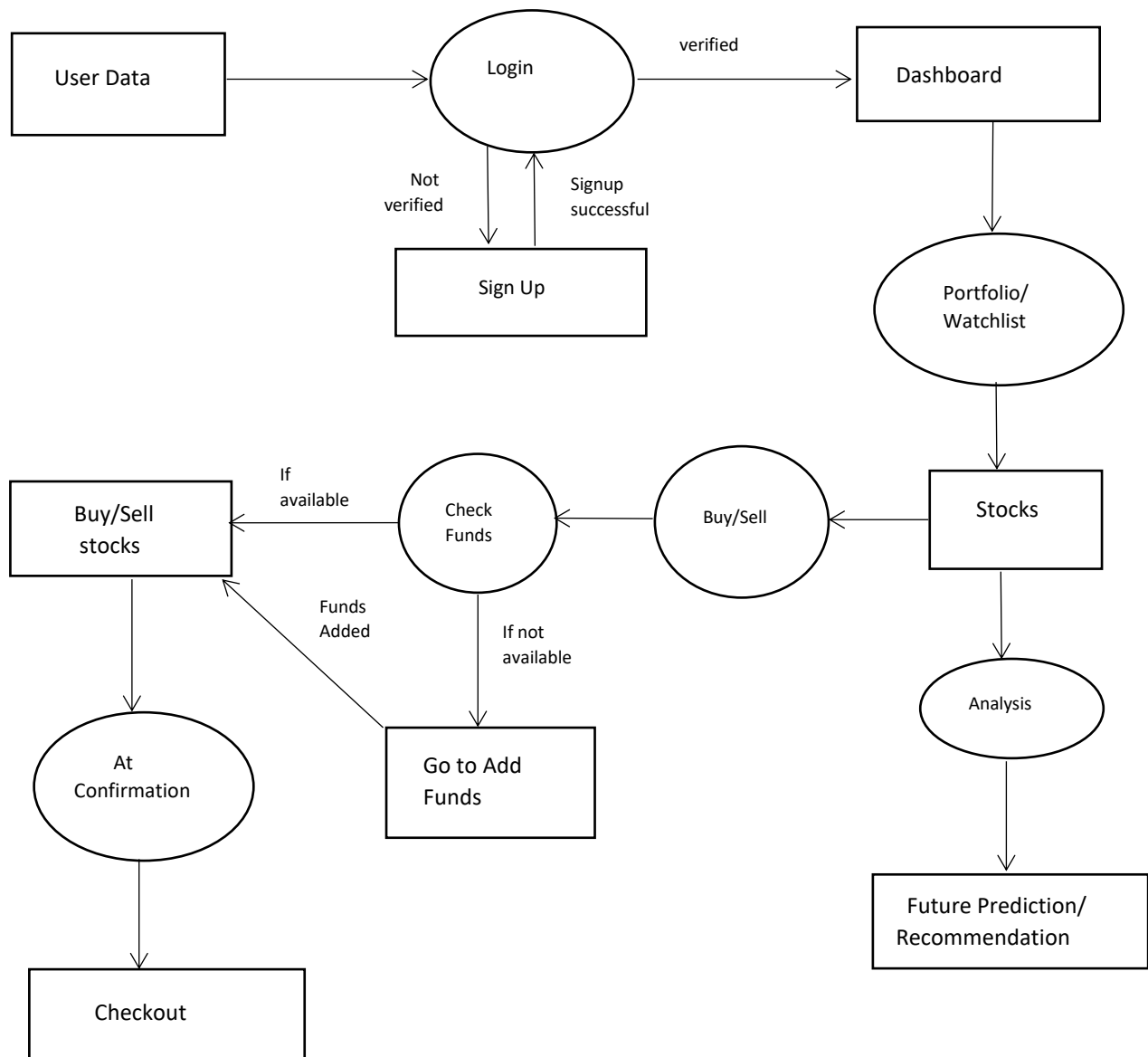


Fig. 7.2

3. Buying and Selling Stocks:

- Users can choose to buy or sell stocks from the available list.
- Before completing the transaction, the system checks the availability of funds in the user's account.

4. Fund Management:

- If sufficient funds are available, the transaction moves forward.
- If funds are insufficient, users are redirected to a section where they can add more funds.
- Once the funds are successfully added, the user can resume and complete their transaction.

5. Transaction Confirmation and Checkout:

- After selecting the stocks and ensuring funds are available, the transaction is confirmed.
- The final step is the checkout, where all details of the transaction are reviewed and completed.

6. Stock Analysis and Recommendations:

- To assist users in making better decisions, the system provides detailed stock analysis.
- Future predictions and recommendations are also available, enabling users to strategize their investments effectively.

7.2. ER Diagram

ER stands for “Entity-Relationship” and it is a data modelling technique used to represent the relationships between different entities in a database. ER diagrams are used to visually represent the structure of a database, including entities, attributes, and relationships between entities. The diagram would show how each entity is related to others and how data is exchanged between them.

Entities and Attributes

1. User: The User is a key part of the stock management system, representing every person who interacts with the platform. It plays an important role in ensuring the system works smoothly by connecting different functions and providing personalized experiences. Whether the users are investors, analysts, or casual viewers, the User entity ensures their interactions are safe, efficient, and tailored to their needs.

1.1. Attributes:

- **User ID:** A unique identifier that acts as the primary key for the User entity. It ensures that every user is uniquely distinguishable, which is critical when managing large-scale databases.
- **Name:** It refers to the full name of the user. The name attribute ensures user-friendliness and personalization across the system.
- **Contact:** Includes email addresses and/or phone numbers. Contact information is also essential for account recovery processes and two-factor authentication (2FA).
- **Account:** Represents the user’s financial or investment profile. Tracks key data points such as total balance, transaction history, and current holdings.

1.2. Purpose:

- To integrate user-specific data seamlessly into the system for efficient management and personalization.
- To offer tailored insights, such as investment recommendations or notifications, based on user preferences.

- To enhance the system's usability by tracking individual behaviours and enabling better engagement.

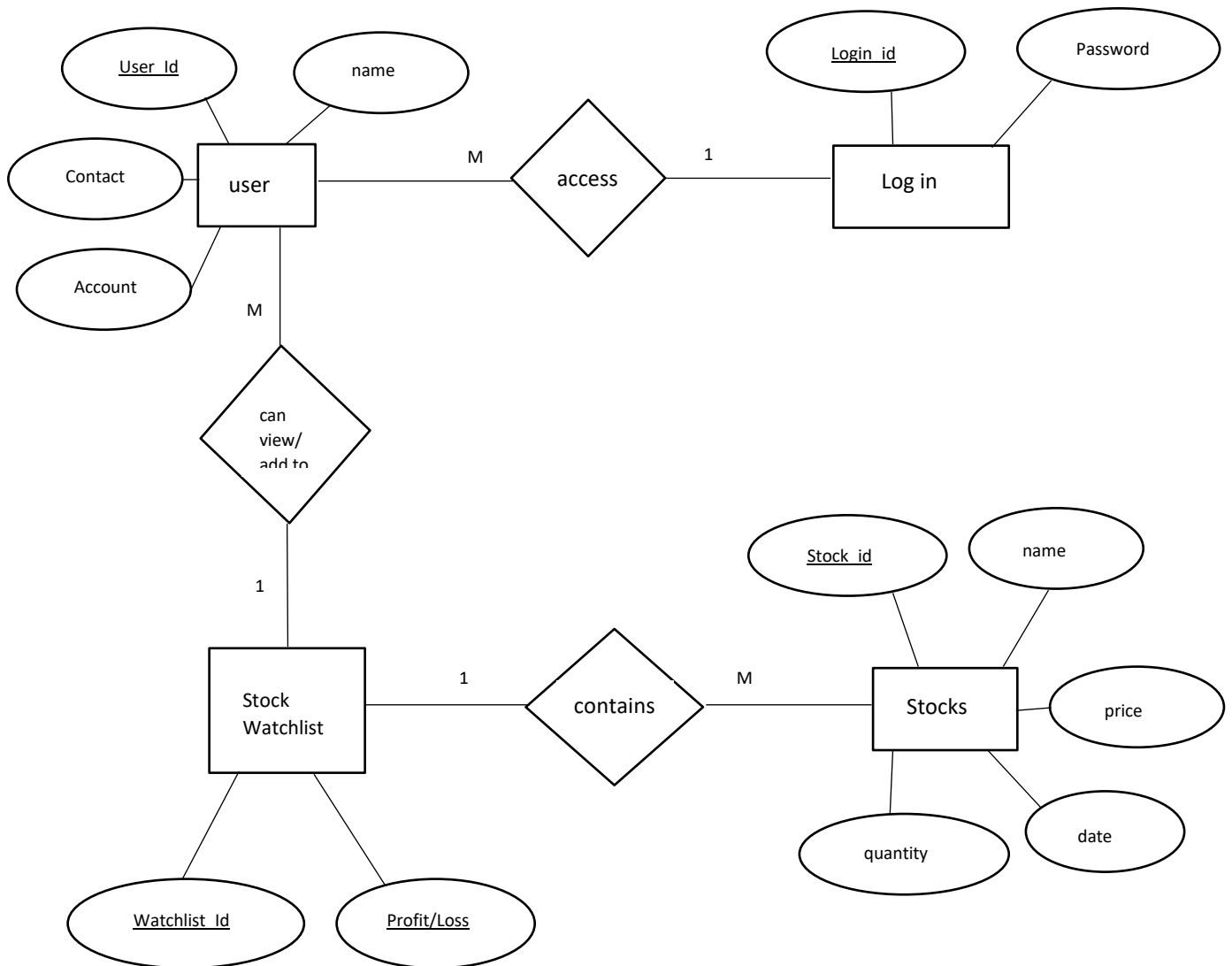


Fig. 7.3

2. **Log in:** This entity ensures secure and authenticated access to the stock management platform. It is a critical layer in safeguarding sensitive data and preventing unauthorized access.

2.1. Attributes:

- **Login ID:** A unique string, often the user's email address or a username. This ID helps identify which user is accessing the system at any given time.
- **Password:** A confidential, encrypted string used to authenticate the user. This ensures data protection and compliance with cybersecurity best practices.

2.2. Purpose:

- Enhances system security by allowing only authorized individuals to access user-specific data.
- Tracks user sessions to provide seamless multi-device experiences.
- Plays a critical role in securing sensitive data, such as financial portfolios or transaction histories.

3. Stock Watchlist: The Stock Watchlist entity allows users to maintain a curated list of stocks they are interested in tracking. It acts as a dynamic, user-specific portfolio designed for active monitoring and decision-making.

3.1. Attributes:

- **Watchlist ID:** A unique identifier for each watchlist.
- **Profit/Loss:** Represents the net gain or loss for stocks in the watchlist. Calculated as the difference between the stock's purchase price and its current market value.

3.2. Purpose:

- Provides a structured and user-friendly way to track stocks based on categories or preferences.
- Helps users assess their investment strategies and modify their portfolios accordingly.

4. Stocks: This entity captures detailed information about individual stocks that users are monitoring or trading. It is essential for presenting accurate, up-to-date market data.

4.1. Attributes:

- **Stock ID:** A unique alphanumeric identifier for each stock, such as a ticker symbol.
- **Name:** The full name of the stock or company, providing clarity to users. Example: "Apple Inc." or "Infosys Limited."
- **Price:** Refers to the real-time market price of the stock.
- **Quantity:** Indicates the number of shares of a specific stock that a user owns or is monitoring. Example: A user might own 10 shares of TCS and monitor 5 shares of Wipro.
- **Date:** Represents the date on which the stock was added to the watchlist or purchased. Example: "2024-12-01" signifies when the stock was first included.

4.2. Purpose:

- Helps users analyse market trends, monitor specific stocks, and make data-driven investment decisions.
- Consolidates all key details into one entity, simplifying data management for the system.

Relationships and Functionalities

1. User and Log in Relationship: 1:M (One-to-Many)

A single user can have multiple login sessions, enabling flexibility and convenience. Each login session is recorded for tracking and auditing purposes.

1.1. Functionalities:

- **Authentication:** Ensures valid credentials are used to access the system. Example: Incorrect login attempts are limited to prevent brute-force attacks.
- **Session Management:** Allows users to remain logged in across multiple devices. Example: A user logged in on their laptop can simultaneously access the system on their smartphone.
- **Security Enhancements:** Multi-factor authentication (MFA) ensures that even if a password is compromised, the account remains secure.

2. User and Stock Watchlist Relationship: M:1 (Many-to-One)

Users can create and manage multiple watchlists. Each watchlist is tied to a single user but may support collaborative access.

2.1. Functionalities:

- **Custom Watchlists:** Users can organize stocks into categories, such as "Short-term Gains" or "Retirement Portfolio." Example: A user might track high-risk stocks in one watchlist and blue-chip stocks in another.
- **Collaboration:** Enables users to share watchlists with others, such as family members or financial advisors. Example: A shared watchlist between a parent and child allows them to plan investments together.
- **Performance Tracking:** Calculates profit/loss for each watchlist, helping users make quick decisions.

3. Stock Watchlist and Stocks Relationship: 1:M (One-to-Many)

A single watchlist can contain multiple stocks, allowing for diverse investment tracking.

3.1. Functionalities:

- **Portfolio Management:** Organizes stocks in a structured manner, enabling efficient tracking. Example: Stocks in a "Tech Portfolio" can include companies like Microsoft and Google.

- **Real-time Updates:** Ensures that stock prices and other metrics are updated in real-time. Example: A stock in the watchlist is flagged when its price exceeds a pre-set target.
- **Decision Support:** Highlights underperforming stocks to alert users about potential losses. Example: If a stock's value drops by 10%, the system sends a notification for action.

7.3. Class Diagram

7.3.1. Authentication:

- Classification: Weak Class
- Description: Represents user authentication details, including username and password. This class is responsible for user login functionality.

7.3.2. User:

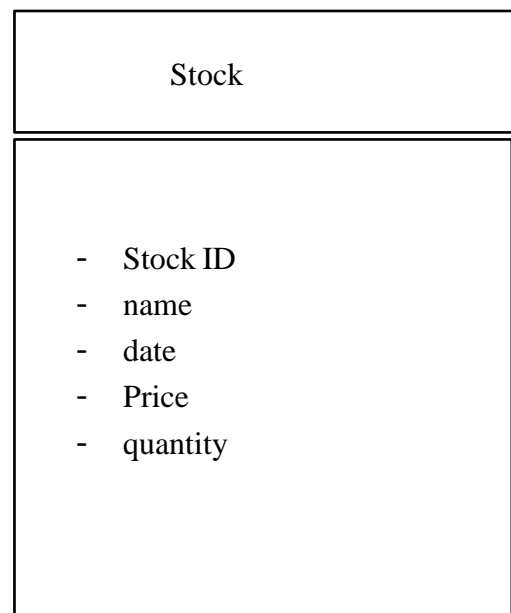
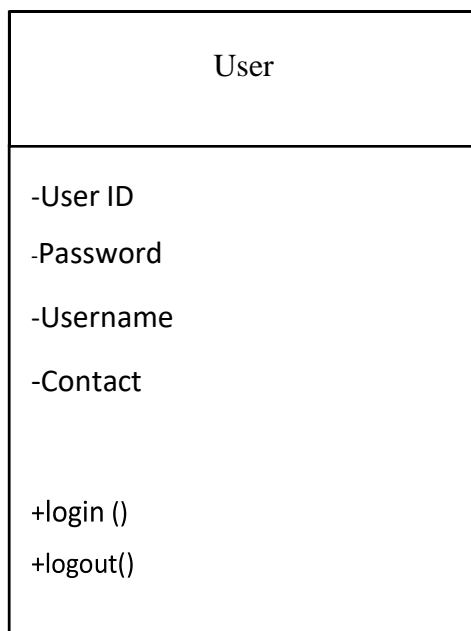
- Classification: Strong Class
- Description: Represents the users of the system, including administrators and employees.

7.3.3. Package:

- Classification: Strong Class
- Description: Represents the stocks details including Stock ID ,name, price and quantity.

7.3.4. Packages:

- Classification: Strong Class
- Description: Represents the watchlist details such as watchlist_id, profit/loss.



Authentication
<ul style="list-style-type: none"> -User_id -password

Stock Watchlist
<ul style="list-style-type: none"> - watchlist_Id - profit/loss

Fig. 7.4

CHAPTER 8

DATABASE

9.1. Admin

- **Email:** This attribute stores the email address of an individual. It is a unique identifier and is commonly used for user authentication and communication.
- **Password:** The "password" attribute stores a securely hashed or encrypted version of the user's password. It is a critical attribute for user authentication, ensuring secure access to the system.

9.2. User Credentials

- **Id:** The "Id" attribute is typically a unique identifier assigned to each individual in the database. It serves as a primary key, ensuring that each record can be uniquely identified and referenced.
- **Name:** The "Name" attribute stores the full name of an individual. It is a fundamental piece of personal information and is often used for identification and communication purposes.
- **Email:** The "Email" attribute stores the email address of an individual. It serves as a Unique identifier for user accounts and is commonly used for communication and login credentials.
- **Password:** The "Password" attribute stores a securely hashed or encrypted version of the user's password. It is a critical attribute for user authentication, ensuring the security of user accounts by protecting access to sensitive information.

9.3. Stocks

- **stock_Id:** The "Id" attribute typically serves as a unique identifier for each record in the database table. It is a primary key that ensures each entry can be uniquely identified and referenced.
- **name:** The "Description" attribute provides additional details or a brief description of the product. This field allows for a more comprehensive understanding of the product's characteristics or features.
- **date:** The date refers to the starting time of a buying or selling,
- **Quantity:** The Quantity refers to the number of stock that you buy or sell. This field allows for a more comprehensive understanding of the product's characteristics or features.
- **Price:** The price refers to the cost of the stock on which a stock can be buy or sell. This field allows for a more comprehensive understanding of the product's characteristics or features.

CHAPTER 9

DESIGN

10.1. Login

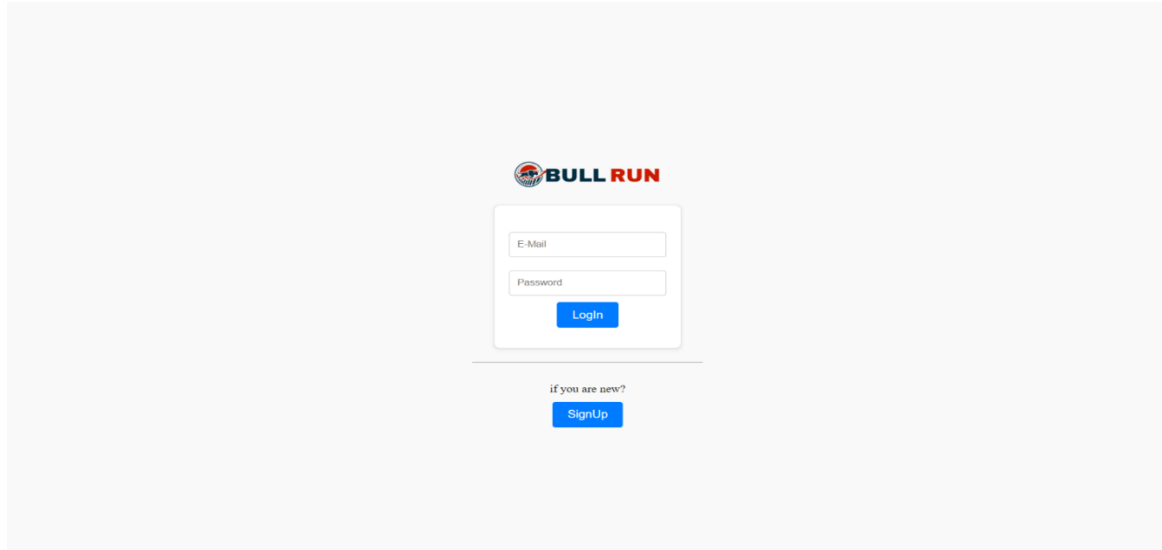


Fig. 10.1

Welcome to your personalized dashboard, where smarter investing begins. Log in to access a suite of tools and insights designed to help you take control of your financial journey. With just a few clicks, you can track your investments, monitor performance, and stay updated with real-time stock market data—all in one secure and user-friendly platform.

Your dashboard is tailored to meet your unique financial goals. Whether you're an experienced investor or just starting out, our platform empowers you to make informed decisions. Create and manage your personalized watchlist to keep an eye on your favorite stocks, industries, or trends. Use advanced analytics tools to explore historical data, compare performance metrics, and identify emerging opportunities in the market.

Stay ahead with real-time updates on stock prices, market indices, and breaking financial news. Our insights are curated to give you a competitive edge, helping you respond swiftly to market changes. From analyzing trends to identifying growth opportunities, your dashboard is your gateway to building a robust investment strategy.

Logging in is simple and secure. Enter your credentials below to unlock a world of possibilities. Our state-of-the-art encryption ensures that your data remains private and protected, allowing you to focus on what matters most—growing your portfolio.

Together, let's navigate the complexities of the financial world and achieve your investment goals. With personalized tools, actionable insights, and a seamless experience,

you have everything you need to take your investments to the next level. Log in now and step confidently into the future of smarter, more empowered investing. The journey to financial success starts here!

10.2. Signup

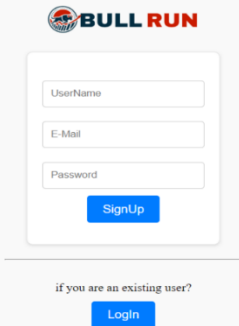
The image shows a web form for signing up on the BULLRUN platform. At the top center is the BULLRUN logo, which consists of a circular icon with a bull's head and the text "BULL RUN" in a bold, sans-serif font. Below the logo is a white rectangular box containing three input fields: "UserName", "E-Mail", and "Password". Each field has a small placeholder text. Below the "Password" field is a blue button with the text "SignUp" in white. Below the white box is a horizontal line, and below that is the text "if you are an existing user?". Below this text is another blue button with the text "Login" in white.

Fig.10.2

The signup page on BULLRUN is designed to provide a seamless registration experience for users eager to become part of our dynamic community. With a few simple steps, visitors can create personalized accounts and unlock access to a world of opportunities in stock trading and financial growth.

To get started, users are required to provide basic details such as their name, email address, and a secure password. Optional fields allow users to share their interests, helping us tailor their experience and enrich their profiles. This personalized touch ensures that every user feels valued and catered to right from the start.

To guarantee account security and authenticity, our signup process includes an email verification step. This ensures that all accounts are genuine, offering a safe environment for buying, selling, and managing stocks.

Joining BULLRUN opens the door to a vibrant community of like-minded individuals. Connect with fellow investors, stay informed with real-time market insights, and embark on exciting new adventures in the world of trading. Whether you're a seasoned investor or just starting your journey, BULLRUN is the perfect platform to achieve your financial goals. Sign up today and take the first step toward smarter, more empowered investing!

10.3. Home Page

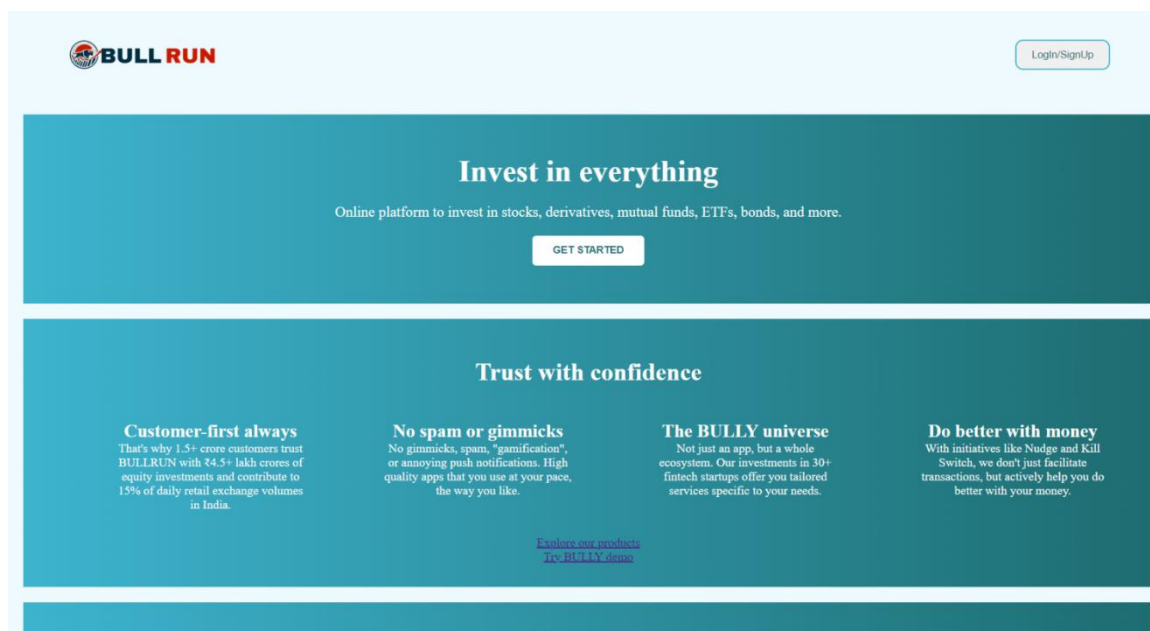


Fig. 10.3

Your personalized investing hub is here. Access your stock watchlist, manage your portfolio, and stay ahead with the latest market trends, news, and insights tailored just for you. Dive deeper into detailed analytics, set goals, and track your progress toward financial success.

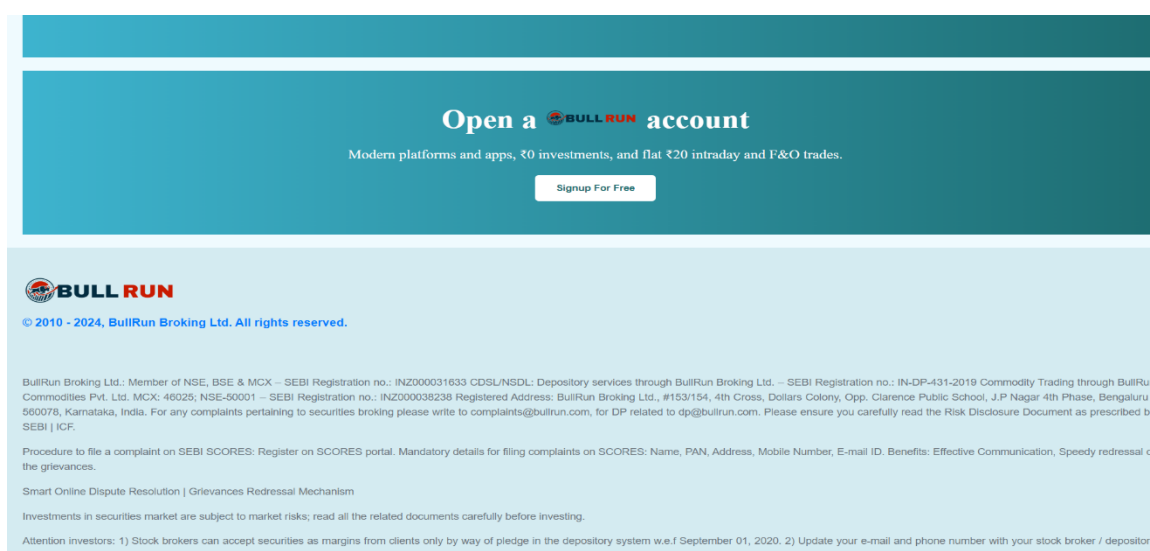


Fig. 10.4

Easily navigate through stocks, IPOs, and market updates with a user-friendly interface designed to enhance your experience. Whether you're monitoring your investments or exploring new opportunities, BullRun ensures you have everything you need at your fingertips. Let's take the next step in your investment journey.

10.3.1. Stocks

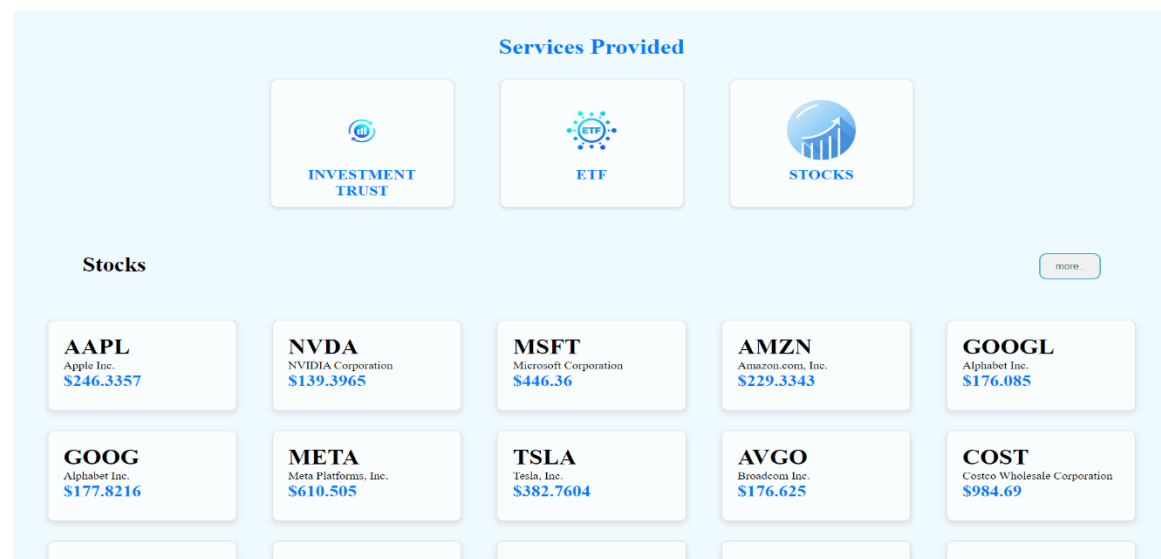


Fig.10.5

Discover a world of investment opportunities with BullRun, where you can explore a diverse range of stocks across various industries. Our intuitive platform is designed to empower investors of all levels, providing the tools and insights needed to navigate the dynamic stock market confidently.

Stay ahead with real-time updates on stock prices and movements, ensuring you're always informed about market trends. BullRun's platform offers detailed stock data, from historical performance to key financial metrics, enabling you to make well-informed decisions. Track price changes, compare trends, and identify growth opportunities with ease.

Whether you're a seasoned investor or just beginning your journey, BullRun caters to your needs with a user-friendly interface and advanced analytics. Create and manage a personalized watchlist to monitor your favorite stocks, set alerts, and seize opportunities as they arise. Our platform simplifies the complexities of investing, making it accessible and straightforward for everyone.

Dive into the stock market with BullRun and take control of your financial future. Explore, analyze, and manage your investments all in one place. With BullRun, the tools to succeed are at your fingertips. Start exploring today and make smarter, more confident investment decisions!

10.3.2. Stock Watchlist

LargeCap				
AAL American Airlines Group Inc. 17.1082 11242328574	AAON AAON, Inc. 130.52 10608613392	AAPL Apple Inc. 246.3 3723021540000	ABMD Allomid, Inc. 381.02 17180649405	ABNB Airbnb, Inc. 137.18 86968227929
ACGL Arch Capital Group Ltd. 96.1 36156856200	ACGLO Arch Capital Group Ltd. 22.65 36645951737	ACGLP Arch Capital Group Ltd. 24.99 15165357055	ADBE Adobe Inc. 548.85 241603770000	ADI Analog Devices, Inc. 220.245 109306932765
ADP	ADSK	AEP	AERM	AKAM
SmallCap				
AACG ATA Creativity Global 0.8399 26846988	AACI Armada Acquisition Corp. I 9.75 69490103	AACIU Armada Acquisition Corp. I 11 69490102	AACIW Armada Acquisition Corp. I 0.1299 932872	AACQU Arctis Acquisition Inc. 11.05 1581255000

Fig.10.6

Stay connected to the stocks that matter most with BullRun's personalized Watchlist. Designed to keep you informed and in control, the Watchlist is your go-to tool for tracking and managing your favorite investments effortlessly.

With BullRun, you can monitor real-time price changes, ensuring you never miss a market movement. Set custom alerts to stay updated on critical price thresholds or performance milestones, so you're always ready to act. Whether you're analyzing short-term trends or keeping an eye on long-term growth, the Watchlist provides a seamless way to organize and stay informed.

The intuitive interface allows you to add, remove, and prioritize stocks based on your interests and goals. Instant updates and insights make it easier to identify opportunities and make timely decisions, helping you stay ahead in a dynamic market.

Whether you're a seasoned trader or just starting to explore the market, BullRun's Watchlist simplifies the process, giving you a clear, consolidated view of your chosen stocks. Take the first step toward smarter investing by creating your personalized Watchlist today. Stay organized, stay informed, and stay ahead with BullRun.

CHAPTER 11

TESTING

11.1. Introduction

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionalities of components, sub-assemblies, and/or a finished product it is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

11.2. Types of Testing

11.2.1. Unit Testing

Unit testing focuses verification effort on the smallest unit of software design, the module. The unit testing, we have is white box oriented and some modules the steps are conducted in parallel.

11.2.2. Integration Testing

Testing is done for each module. After testing all the modules, the modules are integrated and testing of the final system is done with the test data, specially designed to show that the system will operate successfully in all its aspects conditions. Thus, the system testing is a confirmation that all is correct and an opportunity to show the user that the system works.

The purpose of integration testing is to verify functional, performance and reliability requirements placed on major design items. These "design items", i.e. assemblages (or groups of units), are exercised through their interfaces using black box testing, success and error cases being simulated via appropriate parameter and data inputs.

11.2.3. System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

CONCLUSION

The BullRun project is an innovative, comprehensive platform that simulates stock market operations and enables users to participate in important financial activities like portfolio management, stock trading, and mutual fund investing. BullRun successfully closes the gap between theoretical understanding and practical financial operations by addressing the fundamental functions of a stock market. This project offers customers a useful and instructive tool to understand market dynamics and make wise investing choices.

By including features like a live portfolio tracker, which shows all owned stocks together with their individual prices, the platform guarantees an easy-to-use user experience. Users can monitor their financial holdings and their current market worth in one place due to this feature. The user experience is further enhanced by the addition of specific stock pages, which provide in-depth information on individual stocks. These sections facilitate the analysis and planning of investment plans by providing users with information such as stock prices, historical performance trends, and company-specific statistics.

In a technical sense, BullRun serves as confirmation of the developer's skill in software development and coding. The project shows a solid understanding of database administration, user interface design, and computer languages. It ensures dynamic data handling and an attractive layout by connecting frontend responsiveness with backend functionalities. This technical stability guarantees that BullRun runs well and gives users accurate and trustworthy information.

In conclusion, the BullRun project successfully achieves its objectives by delivering a versatile, educational, and interactive platform that replicates stock market functionalities. It highlights the developer's ability to combine technical skills, analytical thinking, and innovative design to address a complex problem. This project not only showcases expertise in software development but also serves as a stepping stone for future endeavors in the rapidly evolving field of financial technology. By offering users an engaging and practical tool, BullRun paves the way for further exploration and innovation in the intersection of technology and finance.

REFERENCES

1. <http://www.investing.com/rates-bonds/india-10-year-bond-yield-advanced-chart>
2. <http://economictimes.indiatimes.com/markets/stocks/stock-quotes>
3. <https://in.finance.yahoo.com/q/hp?s=TECHM.NS&a=02&b=31&c=2010&d=02&e=31&f=2014&g=d>
4. Stern Stewart and Company, "Why EVA works", <http://eva.com/>
5. www.investopedia.com/terms/c/cva.asp
6. www.valuebasedmanagement.net/methods_cva.html
7. www.ripublication.com/gjfm-spl/gjfmv6n2_16.pdf
8. www.iosrjournals.org/iosr-jef/papers/icsc/volume-1/8.pdf
9. <http://www.acmeintellects.org/images/AIJRMSST/Jan2015/10-1-15.pdf>
10. https://www.ici.org/pdf/rpt_risk.pdf.

BIBLIOGRAPHY

BOOKS AND ARTICLES

1. Shroff, M. (2020). *Full-Stack React Projects: Modern Web Development Using React, Node, Express, and MongoDB*. Packt Publishing.
2. Duckett, J. (2014). *JavaScript and JQuery: Interactive Front-End Web Development*. Wiley.
3. Ritchie, J. (2021). *MongoDB: The Definitive Guide*. O'Reilly Media.

WEB RESOURCES

4. Parth Thakkar (2022). *Stock Market Prediction with LSTM*. Retrieved from [GitHub](#).
5. GeeksforGeeks. *Stock Market Portfolio App Using MERN Stack*. Retrieved from GeeksforGeeks.
6. Medium. (2023). *Stock Market Dashboard Using MERN Stack*. Retrieved from Medium.
7. Ali K. *StockApp - Stock Watch List and Simulator*. Retrieved from AliK604 GitHub.

TUTORIALS AND BLOGS

8. CCBP. (2023). *15 Best MERN Stack Project Ideas*. Retrieved from CCBP Blog.
9. Net Ninja. (2022). *MERN Stack Crash Course for Beginners*. Retrieved from [YouTube](#).

SOFTWARE AND FRAMEWORKS

10. Facebook. (2013). *React.js: A JavaScript Library for Building User Interfaces*. Retrieved from [React.js Official Site](#).
11. MongoDB Inc. (2009). *MongoDB Database Documentation*. Retrieved from [MongoDB Official Site](#).