SOFTWARE GROUP PROJECT REPORT

On

Invest Control Crypto Website

Submitted by

Kewal Suthar (IU2241221189)
Sumit Panchal (IU2241221188)
Vedant Trivedi (IU2241221186)
Dev Amdavadi (IU2241221194)

In partial fulfillment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

In

Information Technology

INSTITUTE OF TECHNOLOGY AND ENGINEERING

INDUS UNIVERSITY CAMPUS, RANCHARDA ,VIA-THALTEJ

AHMEDABAD-382115, GUJARAT, INDIA,

WEB: www.indusuni.ac.in

OCT-2024

SOFTWARE GROUP PROJECTREPORT

On

Invest Control Crypto Website

At

In partial fulfillment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

In

Information Technology

PREPARED BY

Kewal Suthar (IU2241221189)
Sumit Panchal (IU2241221188)
Vedant Trivedi (IU2241221186)
Dev Amdavadi (IU2241221194)

UNDER GUIDANCE OF

Internal Guide
Jaya Shukla,
Department of
Information Technology,
I.T.E, Indus University,
Ahmedabad

SUBMITTED TO

INSTITUTE OF TECHNOLOGY AND ENGINEERING
INDUS UNIVERSITY CAMPUS, RANCHARDA, VIA-THALTEJ
AHMEDABAD-382115, GUJARAT, INDIA,
WEB: www.indusuni.ac.in
OCT-2024

CANDIDATE'S DECLARARTION

I declare that final semester report entitled "**Invest Control Crypto Website**" is my own work conducted under the supervision of the guide **Jaya Shukla**

I further declare that to the best of my knowledge, the report for B. Tech final semester does not contain part of the work which has been submitted for the award of B.Tech Degree therein this university or any other university without proper citation.

Candidate's Signature
Kewal Suthar (IU2241221189)

Guide: Dr. Shruti B Yagnik

Associate Professor,

Department of Information Technology,

Indus Institute of Technology and Engineering

INDUSUNIVERSITY–Ahmedabad,

State: Gujarat

CANDIDATE'S DECLARARTION

I declare that final semester report entitled "**Invest Control Crypto Website**" is my own work conducted under the supervision of the guide **Jaya Shukla**

I further declare that to the best of my knowledge, the report for B. Tech final semester does not contain part of the work which has been submitted for the award of B.Tech Degree therein this university or any other university without proper citation.

_____

Candidate's Signature
Sumit Panchal (IU2241221188)

_____

Guide: Dr. Shruti B Yagnik

Associate Professor,

Department of Information Technology,

Indus Institute of Technology and Engineering

INDUSUNIVERSITY–Ahmedabad,

State: Gujarat

CANDIDATE'S DECLARARTION

I declare that final semester report entitled "**Invest Control Crypto Website**" is my own work conducted under the supervision of the guide **Jaya Shukla**

I further declare that to the best of my knowledge, the report for B. Tech final semester does not contain part of the work which has been submitted for the award of B.Tech Degree therein this university or any other university without proper citation.

_____

Candidate's Signature
Vedant Trivedi (IU2241221186)

_____

Guide: Dr. Shruti B Yagnik

Associate Professor,

Department of Information Technology,

Indus Institute of Technology and Engineering

INDUSUNIVERSITY–Ahmedabad,

State: Gujarat

CANDIDATE'S DECLARARTION

I declare that final semester report entitled "**Invest Control Crypto Website**" is my own work conducted under the supervision of the guide **Jaya Shukla**

I further declare that to the best of my knowledge, the report for B. Tech final semester does not contain part of the work which has been submitted for the award of B.Tech Degree therein this university or any other university without proper citation.

_____

Candidate's Signature
Dev Amdavadi (IU2241221194)

_____

Guide: Dr. Shruti B Yagnik

Associate Professor,

Department of Information Technology,

Indus Institute of Technology and Engineering

INDUSUNIVERSITY–Ahmedabad,

State: Gujarat

INDUS INSTITUTE OF TECHNOLOGY AND ENGINEERING

INFORMATION TECHNOLOGY
2024-2025

CERTIFICATE

Date:25/04/2023

This is to certify that the project work entitled "**Invest Control Crypto Website"** has been carried out by **Kewal, Sumit, Vedant and Dev** under my guidance in partial fulfillment of degree of Bachelor of Technology in the department of **Information Technology** of Indus University, Ahmedabad during the academic year 2024– 2025.

Ms. Jaya Shukla	Dr. Shruti B Yagnik
Assistant Professor,	Head of the Department,
Department of Information Technology ,	Department of IT,
I.T.E, Indus University	I.T.E, Indus University
	Ahmedabad

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to all those who have contributed to the successful completion of my Bachelor of Technology Project.

Firstly, I would like to thank my project guide, prof. Shruti B Yagnik, for their guidance and support throughout the project. Their valuable insights and suggestions have been instrumental in shaping my work.

I am also grateful to the faculty members of the department of information technology for their support and encouragement. Their feedback and suggestions have been in valuable in improving the quality of my work.

I would like to express my gratitude to my family and friends for the irun wavering support and encouragement throughout the project. Their constant motivation has been a source of inspiration for me.

Lastly, I would like to thank the university for providing me with the necessary resources and facilities to complete my project successfully.

Thank you all for your support and guidance.

TABLE OF CONTENT

ABSTRACT

This project report covers the development of **Invest Control**, a crypto and stock trading website designed to offer users a smooth trading experience. The platform lets users track stocks and crypto prices in real-time and practice trades before making real-life investments. We focused on making the site simple, secure, and responsive, so it works seamlessly on desktops, tablets, and smartphones.

The Crypto platform features a range of functionalities commonly seen in modern online trading platform , including user authentication (sign-up and login), real time crypto/ stocks prices, stocks management, and crypto history tracking. Additionally, users can use mutual funds calculator for their investment plan that helps future customers in their money control decisions. The platform also offers real-time interaction between the front end and back end through a well-structured API, ensuring that user data, stock and crypto prices, and other details are processed efficiently.

The **front end** was built using **Jinja**, a fast, expressive, and extensible templating engine. It allows us to write code with Python-like syntax in the templates. This helped in creating a dynamic and interactive user experience. We used **HTML** and **CSS** to structure and style the platform's pages, ensuring a responsive design across all devices.For the **back end**, we used **Python** and **Flask** to handle server-side operations like managing user data, processing requests, and transactions. The system runs on **SQLite** for database management, allowing scalable storage of user and product data. We also integrated a **RESTful API** to ensure smooth communication between the client and server.

The project tackles several important challenges, like ensuring secure user authentication, providing real-time prices, maintaining a responsive design, and enabling smooth interaction between the front end and back end. We've built in strong security measures, especially when it comes to managing user data and transactions. In the next phase, we'll be conducting unit testing, integration testing, and user acceptance testing (UAT) to make sure everything works smoothly under different conditions.

Key features of the platform include real-time price quotes, order history, a mutual funds calculator, and the ability to buy and sell stocks and crypto, all within a fully responsive design. Invest Control offers more than just a typical trading website, paving the way for future expansion into other stock market services. As the platform grows, we can add more features like personalized stock recommendations, IPO tracking, and broader payment options.

The development of the platform was a team effort, with each member contributing to specific areas of the project. **Kewal** focused on the **back-end**, ensuring smooth server-side operations. **Sumit** worked on the user interface design and front-end styling, using **UI/UX** to create a visually appealing and user-friendly experience. **Vedant** took charge of **integrating APIs** and making sure the platform's features worked correctly. **Dev** brought the design to life, turning it into a fully functional **front-end** application using pure HTML and CSS. The project's progress and success can be credited to the team's use of modern web technologies and their commitment to delivering a great user experience.

This report outlines the design process, the technologies used, the challenges faced during development, and the future steps required to complete the project. It also highlights the potential for future expansion, including the integration of additional products, services, and features to enhance user satisfaction.

LIST OF FIGURES

ABBREVIATION

Abbreviations used throughout this whole document are:

- **API:** Application Programming Interface

- **UI:** User Interface

- **UX:** User Experience

- **HTML:** HyperText Markup Language

- **JS:** JavaScript

- **CSS:** Cascading Style Sheets

- **DBMS:** Database Management System

- **UAT:** User Acceptance Testing

CHAPTER 1: INTRODUCTION

1.1 Overview

The trading platform Invest Control, crypto and stock trading, was create   to make trading easier for users. Thanks to such tools, one can see the real-time prices of stocks and crypto and can perform practice trades before putting up real money. Invest Control is also available on desktops, tablets or smartphones. Users are assured that it will be easy for all of them to use.

The project included also some of such key phases like UI/UX design, front end and back end development, API integration, and testing. That platform has basic functions for user authentication, surveillance of the current price and car transactions. The structure is built with responsiveness and flexibility that helps to meet the growth of many users using the application after some time.

1.2 Purpose / Problem Statement

The purpose of this project is to address the following problems:

- **Lack of a dedicated platform for crypto and stock trading**:

While trading platforms exist, there's a gap when it comes to platforms that focus on both crypto and stock investments in one place. Invest Control addresses this by offering a dedicated space for users to track prices, manage stocks, and trade crypto, all in real-time.

- **Need for a smooth and secure user experience**:

Users expect a fast, seamless, and secure experience when trading or managing investments online. Our platform ensures this by providing a responsive, user-friendly interface, efficient transaction processes, and secure authentication.

1.3 Scope

The scope of this project is extensive, aiming to create a scalable and secure platform that supports all key aspects of trading stocks and crypto. Key elements of the project scope include:

- **User Registration and Authentication**: Enabling users to create accounts, log in securely,

and manage their profiles and personal details.

- **Stock and Crypto Management**: Displaying real-time crypto and stock prices, along with management tools for users to track their investments and view detailed transaction data.

- **Trading and Payment Integration**: Offering an easy-to-use interface for buying and selling stocks and crypto, along with secure checkout processes through integrated payment systems.

- **Order History and Tracking**: Users can review their trade history, track ongoing investments, and receive insights based on their past trades.

- **API Integration**: Facilitating smooth interaction between the front end and back end using REST APIs to ensure real-time data synchronization and efficient functionality.

Looking ahead, the platform can expand to offer additional features like personalized stock recommendations, IPO listings, loyalty programs, and more diverse payment options.

1.4 Objectives

The objectives of this project are as follows:

1. **Create a dedicated trading platform for stocks and crypto:** To offer a seamless experience for users looking to buy and sell crypto and stocks, focusing on real-time trading capabilities.

2. **Develop a responsive and intuitive user interface**: Ensuring that the platform is accessible and user-friendly across various devices, providing a consistent and enjoyable experience.

3. **Implement secure user authentication and data protection:** Establishing strong authentication methods to safeguard user information and secure the transaction process.

4. **Integrate a comprehensive trading and payment system**: Enabling users to efficiently manage their trades and make payments through secure gateways.

5. **Enable order history and tracking features:** Allowing users to view their trading history, track ongoing trades, and share feedback on their experiences.

6. **Facilitate real-time data integration through APIs:** Ensuring smooth communication between the front end and back end for accurate product information, portfolio management, and order processing.

These objectives shaped the development of the platform, ensuring it meets both business needs and user expectations.

1.5 Technology Overview

This project utilizes a modern tech stack, incorporating a variety of technologies ensure the platform is fast,secure and scalable.

Frontend Technologies:

- **JavaScript**: A versatile programming language used for building dynamic user interfaces and enhancing interactivity on the platform.

- **HTML and CSS**: Standard languages utilized to structure and style the web pages. HTML provides the layout of the web page, while CSS ensures a visually appealing and responsive design across all devices.

- **Jinja**: A fast and expressive templating engine for Python that allows us to generate dynamic HTML content, making it easier to create interactive and data-driven user interfaces.

- **Backend Technologies**:

  - **Python**: The primary programming language used for server-side logic, enabling efficient handling of user requests and data processing.

  - **Flask**: A lightweight web application framework for Python that is used to build RESTful APIs, facilitating communication between the client and server.

  - **SQLite**: A relational database management system used to store user data, investment details, and transaction records. SQLite offers scalability and ease of use, making it a great choice for managing data in our trading platform.

- **API Integration:**

  - **RESTful APIs**: These are implemented to ensure smooth communication between the front end and back end, allowing real-time processing of user information, stock prices, and trading activities.

- **Additional Tools:**

  - **Figma:** Used for UI/UX design, enabling the team to create wireframes, prototypes, and final designs that align with the project's goals.

  - **Postman:** Used for testing APIs to ensure that data exchange between the client and server is smooth and error-free.

  - **GitHub:** A version control platform used to manage the project's codebase, track changes, and collaborate effectively among team members.

This combination of technologies ensures that the platform is scalable, efficient, and user-friendly, providing a solid foundation for future development and expanse.
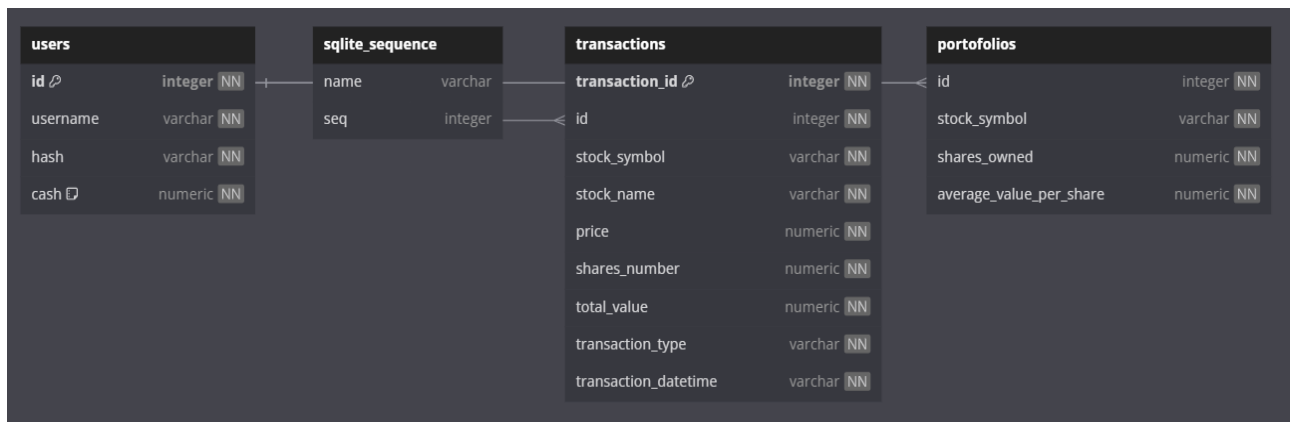
Figure 1: System Architecture Flowchart

CHAPTER 2: SYSTEM REQUIREMENTS / LITERATURE SURVEY

2.1 System Requirements

The development of the Invest Control crypto platform required specific system configurations, software, and hardware to ensure smooth functionality, scalability, and security. The system requirements can be categorized into Functional and Non-Functional requirements, as well as hardware and software prerequisites.

2.1.1 Functional Requirements

The functional requirements outline the essential functions that the Invest Control platform must support to enable users to complete their trading tasks efficiently. The key functional requirements include:

1. **User Registration and Authentication:**
   o Users must be able to securely sign up and log in to the platform.
   o The system should provide users with the option to reset their passwords.
   o Future enhancements may include OAuth login integration (e.g., Google or Facebook login).
2. **Real-Time Price Tracking:**
   o The platform should display real-time prices for stocks and cryptocurrencies.
   o Users must be able to view price fluctuations and trends over time.
3. **Trade Management:**
   o Users should have the ability to practice trades before committing real investments.
   o The system must allow users to buy and sell stocks and cryptocurrencies seamlessly.
4. **Mutual Funds Calculator:**
   o A mutual funds calculator must be integrated to assist users in planning their investments.
   o Users should be able to input their investment parameters and receive calculated outcomes.
5. **Order History and Tracking:**
   o Users must be able to track their trading history and view past transactions, including details of buys and sells.
6. **User Dashboard:**
   o Users should have access to a personalized dashboard where they can manage their investments, view stock and crypto prices, and access tools like the mutual funds calculator.
7. **API Integration for Real-Time Data:**

o The platform should utilize RESTful APIs to facilitate real-time communication between the front end and back end, ensuring efficient processing of user data and price updates.

2.1.2 Non-Functional Requirements

1. **Performance:**

- The platform should load quickly and run smoothly, even with many users at once.
- It must handle many requests without crashing or slowing down.

2. **Scalability:**

- The platform should grow easily to support more users and features in the future.
- The database and server should be able to handle increased data volume.

3. **Security:**

- User data, including payment information, must be encrypted.
- The platform should follow best security practices to prevent common threats (like SQL injection and XSS).

4. **Usability:**

- The platform should be easy to use with a simple design for browsing and trading.
- It should be accessible to all users, including those with disabilities.

5. **Availability:**

- The system should be available 99% of the time, allowing users to access it whenever needed.
- Any downtime for maintenance should be planned and kept to a minimum.

6. **Responsiveness:**

- The platform must work well on different devices, including desktops, tablets, and smartphones.

7. **Maintainability:**

- The code should be organized for easy updates and new features without breaking existing functions.
- Continuous Integration (CI) and Continuous Deployment (CD) practices should be in place for smooth updates and testing.

2.1.3 Hardware Requirements

To ensure the system operates effectively, the following hardware requirements are necessary for both development and deployment:

- **Development Machine Requirements:**
- o **Processor:** Intel i5 or above (or equivalent AMD)
- o **RAM:** 8 GB or higher
- o **Storage:** 250 GB SSD or more
- o **Operating System:** Windows, macOS, or Linux (Ubuntu)
- o **Tools:** Sqlite, web development tools, Postman (for API testing), and Figma (for UI/UX design).
- **Server-Side Requirements:**
- o **Database Server:** MySql, sqlite
- o **Processor:** Multi-core CPU (minimum 2 cores).
- o **Memory:** Minimum 4 GB RAM, recommended 8 GB RAM for higher traffic.
- o **Storage:** SSD storage of at least 100 GB.

2.1.4 Software Requirements

The software stack used for the development of the NFC card platform includes:

- **Frontend**:
- o **Jinja**: Templating engine for dynamic content.
- o **HTML5 and CSS3**: For web page structure and styling.
- o **JavaScript**: For interactive features and user interface enhancements.
- **Backend**:
- o **Python**: Programming language for server-side logic.
- o **Flask**: Web framework for building RESTful APIs.
- o **SQLite**: Lightweight database for storing user, stock, and crypto data.
- **APIs and Tools**:
- o **Postman**: For testing API endpoints.

- RESTful API: For client-server communication.

- GitHub: For version control and project collaboration.
- Figma: For UI/UX design and prototyping.

- **Version Control**:
- GitHub: For version control, collaboration, and tracking changes.

2.2 Literature Survey

In developing the **Invest Control** platform, we reviewed existing literature and research on crypto-currency trading, online trading platforms, and modern web development technologies. Below are the key areas explored to ensure that the project aligns with the latest industry trends and best practices:

2.2.1 Crptocurrency and Blockchain Technology

Cryptocurrency technology, particularly blockchain, plays a pivotal role in secure digital transactions. Blockchain enables decentralized, transparent, and tamper-proof record-keeping, which is crucial for trading platforms. The adoption of cryptocurrencies has surged, with various sectors integrating blockchain solutions to enhance security and efficiency in transactions.

Research Findings:

- Blockchain technology ensures secure and immutable records, making it suitable for trading platforms like Invest Control..
- Cryptocurrencies facilitate instant transactions, allowing users to buy and sell assets in real-time without traditional banking delays..
- The potential for integrating other financial services and products on the blockchain, such as decentralized finance (DeFi) applications and smart contracts, supports the platform's growth

---

2.2.2 Trading Platforms

The online trading industry has experienced substantial growth over the past decade, with platforms

like Robinhood, Coinbase, and eToro at the forefront of innovation. Key trends in online trading include a strong emphasis on mobile-first designs, the integration of AI and machine learning for personalized trading recommendations, and the adoption of secure payment gateways to protect user data..

Research Findings:

- User experience (UX) plays a critical role in the success of an trading platform, with a focus on responsive design, easy navigation, and a smooth checkout process.
- Modern trtading platforms rely heavily on RESTful APIs to ensure real-time communication between the client and server.

2.2.3 Web Development Technologies

The rapid advancement of web development frameworks has facilitated the creation of scalable, responsive, and high-performance trading applications. Frameworks such as Jinja for templating and Flask for server-side programming have gained prominence for building dynamic user interfaces and managing real-time interactions

Research Findings:

- Jinja and Flask**:** Jinja allows developers to create reusable templates, leading to faster development and a more maintainable codebase. Flask provides a lightweight framework for handling server-side operations, making it ideal for applications requiring quick response times.
- RESTful APIs: These APIs enable efficient communication between the front end and back end, ensuring real-time updates of stock and cryptocurrency prices.
- SQLite**:** As a lightweight, serverless database, SQLite is suitable for managing structured data in trading applications while allowing for easy scalability as user demand grows.

Conclusion of Literature Survey:

The findings from the literature survey indicate that cryptocurrency technology and online trading platforms are rapidly evolving fields with significant potential for expansion. RESTful APIs, and re

sponsive design can create a robust and user-friendly trading platform. These advancements ensure that **Invest Control** is aligned with industry standards and is well-positioned to adapt to future technological innovations and user needs.

2.3 Database Design and Architecture

The **Invest Control** platform is built on SQLite, a lightweight and efficient database management system. SQLite is known for its simplicity and ease of use, making it an ideal choice for managing structured data in a trading application. Unlike traditional relational databases, SQLite operates in a serverless environment, allowing for seamless integration with the application while minimizing overhead. Its flexibility enables the platform to efficiently handle user data, trading history, and real-time market information, accommodating future growth as the platform evolves.

2.3.1 Database Structure

In the Sqlite model, data is structured into the following collections:

1. **Users Collection:**
o This collection stores documents for each user, capturing personal information and account details.
o **Document Structure:**
  ▪ **_id:** A unique identifier for each user (MongoDB ObjectId).
  ▪ **Username:** Basic user credentials and personal details.
  ▪ **Hash:** Hashed password for secure authentication.
  ▪ **Cash:** Amount of cash user holds.
2. **Sqlite_sequence Collection:**
o This collection holds the details of NFC card products offered in the platform.
o **Document Structure:**
  ▪ **User:** Unique identifier for each user.
  ▪ **Transaction:** Total number of transaction user have.
3. **Transaction Collection:**
o Stores user-generated reviews for products.

- o **Document Structure:**
- **Transaction_id:** Unique identifier for each review.

- **_id:** References to the user and the user who purchases the stock.
- **Stock_symbol:** Symbol representing the stock or cryptocurrency being reviewed.
- **Stock_name:** The Stock name.
- **price:** The Stock current price.
- **Shares_number:** Total number of shares of crypto and stocks
- **Total_value:** Total values of prices of stocks and crypto
- **Transaction_type:** BUY or SELL
- **Transaction_datetime:** Transaction time and date
4. **Portofolis Collection:**
- o Manages user shopping carts and stores products added for purchase.
- o **Document Structure:**
- **_id:** Unique identifier for the portfolio entry.
- **Stock_symbol:** Reference to the Stock symbol.
- **Shares_owned:** Total number of shares hold by specific user.
- **Average_value_per_share:** Average value per share and cypto.

2.3.2 Entity-Relationship in SQLite

In our SQLite database, relationships between tables are maintained using foreign keys (e.g., id in transactions and portfolios referencing users.id). This structure allows for robust relationships between users, their transactions, and their portfolio holdings, ensuring data consistency and integrity across the platform.

Below is an Entity-Relationship Diagram (ERD) that visualizes how the tables are connected. This diagram shows how Users, Transactions, and Portfolios interact:

- **Users ↔ Transactions**: Users can have multiple transactions. Each transaction records details about stock trades, including stock symbol, name, price, number of shares, total value, and the type of transaction (buy/sell). The id in the transactions table references the users.id.
- **Users ↔ Portfolios**: Users have a portfolio that holds information about the stocks they own. The portfolio tracks the number of shares owned for each stock and the average value per share. The id in the portfolios table references the users.id.

- **Transactions**: The transactions table keeps a history of all trades made by the users. This includes data on stock symbols, names, share prices, number of shares, total value of the trade, type of transaction (buy/sell), and the transaction date and time.
- **Portfolios**: The portfolios table provides a snapshot of the stocks a user owns. It records the stock symbol, the number of shares owned, and the average value per share.
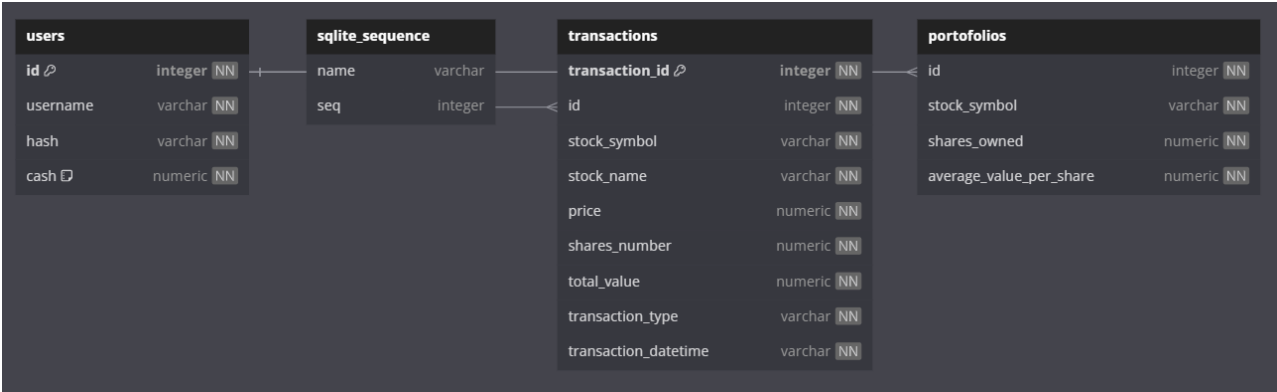

Figure 2: Entity-Relationship Diagram

2.3.3 Data Flow and Integration in SQLite

In our SQLite architecture, tables interact dynamically via foreign keys, ensuring data consistency and integrity across the platform. Here's how data flows between different tables:

- **User Transactions**: When a user initiates a transaction (buy/sell), the transactions table logs the details (stock symbol, price, number of shares, transaction type, etc.). The transactions table references the users table via the id foreign key, ensuring the transaction is associated with the correct user. This way, every user's trade history is accurately maintained.
- **Portfolio Updates**: After a transaction is processed, the portfolios table is updated to reflect the current number of shares owned by the user. If a user buys shares, the table adds the new shares to the existing ones. If they sell shares, the number is reduced accordingly. The portfolios table is linked to the users table, ensuring that each portfolio is correctly tied to the respective user.

CHAPTER 3: DETAILED MODULE DESCRIPTION

The Invest Control platform is designed to deliver a user-friendly and efficient trading experience, using a modular architecture to ensure functionality, scalability, and security. Below is a detailed description of each module in the system, outlining its features and role within the overall platform:

3.1 Sign Up and Sign In Module

The Sign-Up and Sign-In module is the entry point to the platform, allowing users to create an account or log in with their existing credentials. This module handles user authentication securely, ensuring the privacy of user data and preventing unauthorized access.

- **Key Features:**
- **Sign-Up:** New users can create an account by providing personal details such as name, email, and password. An email verification system is in place to confirm the authenticity of the registered email.
- **Sign-In:** Existing users can log in using their email and password. The platform uses secure hashing techniques to protect passwords.
- **Technologies Used:**
- Figma for UI
- Python Flask for backend authentication
- Sqlite for storing user credentials securely
- flask_session tokens for session management
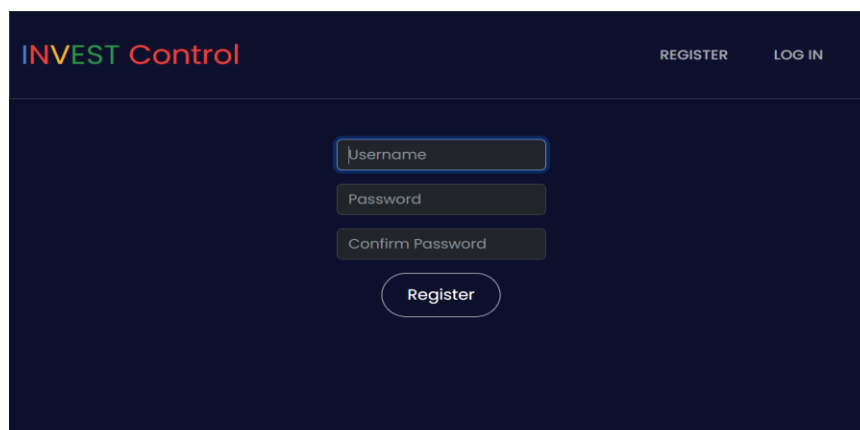
Figure 3: Sign Up Page

Figure 4: Sign In Page

- **Screenshot Explanation:**
  - Screenshot 1: Sign-Up Page with fields for Name, Password, and Confirm Password.
  - Screenshot 2: Sign-In Page with fields for Email and Password.

3.2 Home Page Module

The Home Page serves as the central landing page for users once they log in. It introduces users to the platform's offerings and presents the most popular cryptocurrencies for trading. The home page focuses on providing a user-friendly experience with visually appealing content to engage users right away.

Key Features:

- **Hero Section**: A bold headline promoting the platform's main benefit — a secure and easy way to trade cryptocurrency. This section includes a call-to-action for contacting support and a promotional video button.
- **Carousel**: Displays popular cryptocurrencies like Bitcoin, Ethereum, and Tether, with their real-time prices and discounts. Each item includes a "Buy & Trade" button, encouraging users to start trading directly from the home page.

- **Chain Process**: Highlights the simplicity and transparency of trading and buying cryptocurrencies on the platform. Users are reassured about the lack of fees and the security of their transactions.
- **Markets at Finger**: Emphasizes the wide array of trading options available on the platform, including buying, trading, and supporting cryptocurrencies.
- **Dashboard Promotion**: Encourages users to explore the trading dashboard, highlighting the quick and seamless experience of trading crypto in seconds.
- **Fund Control Section**: Informs users about the variety of funds available and emphasizes the support for all currencies, the use of blockchain technology, and crypto card integration.
- **Newsletter Signup**: Allows users to stay updated on the latest news by subscribing to the platform's newsletter.
- **Footer Section**: Offers links to explore more about the platform, including services like mining, security, and data control, along with quick access to social media profiles.
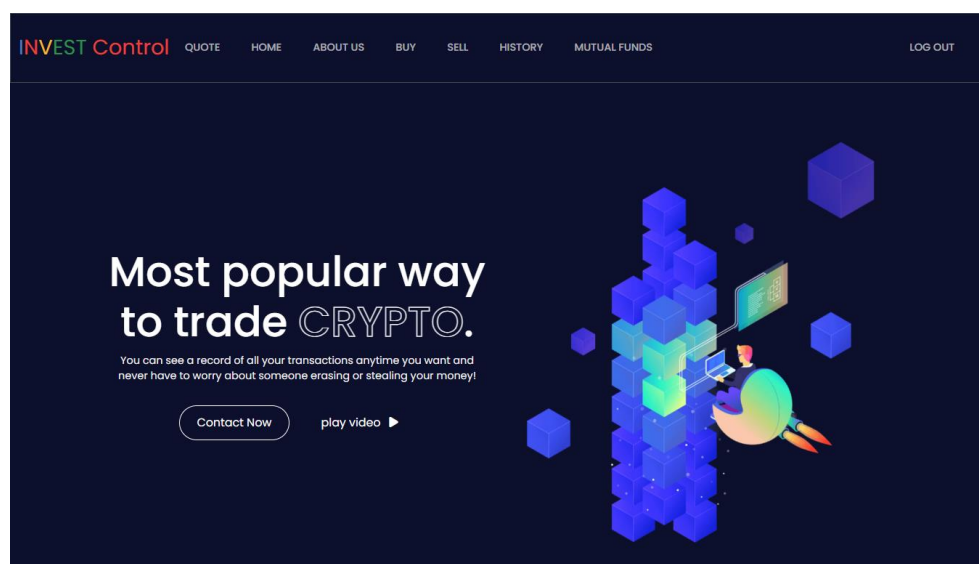
Figure 5: Hero Section
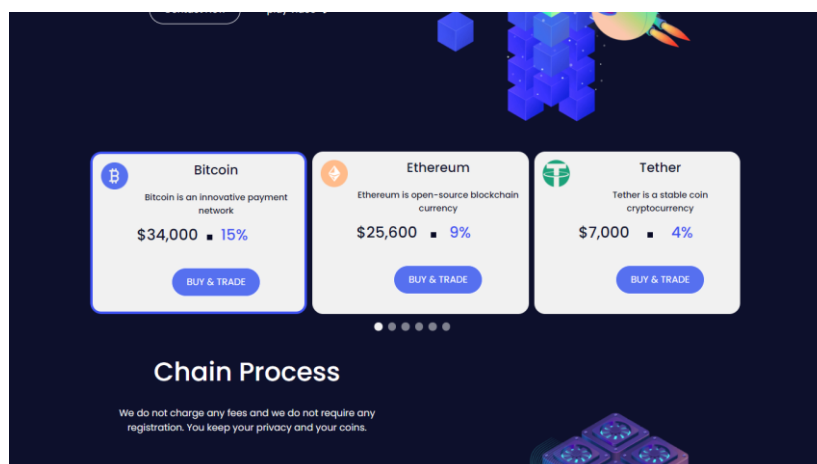
Figure 6: Carousel Section
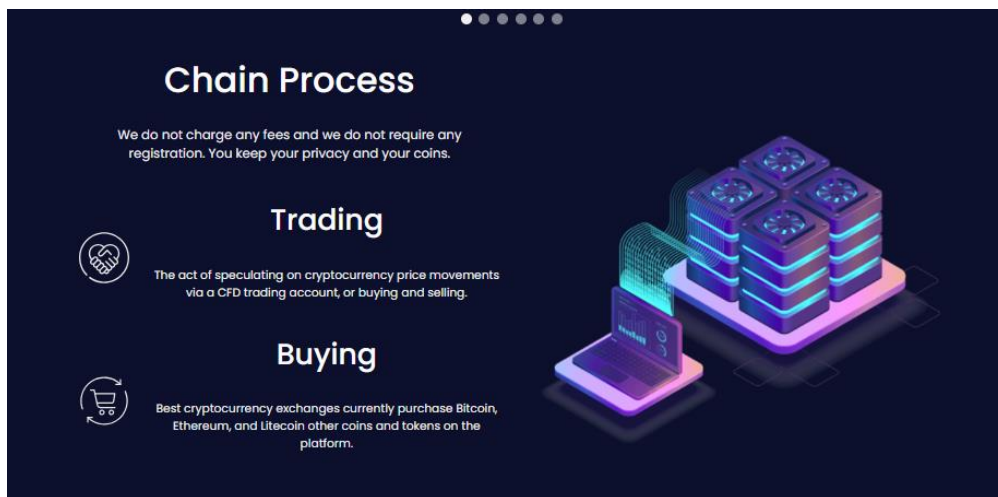
Figure 7: Chain Process Section
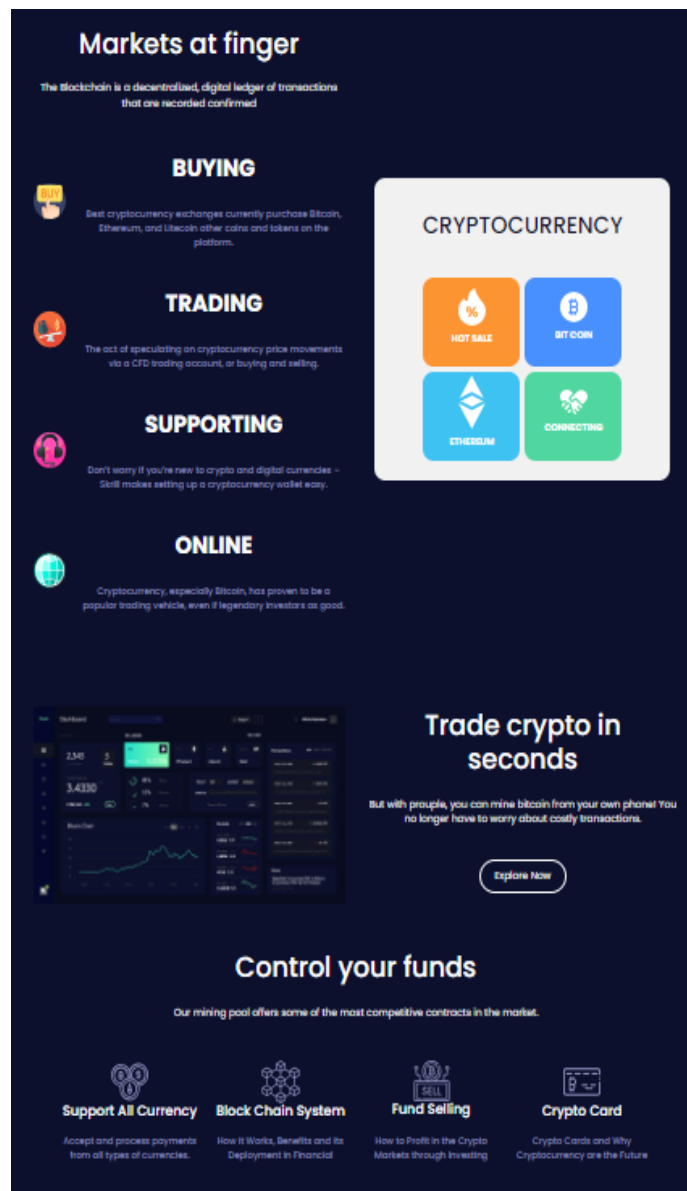
Figure 8: Why Us Section

- **Screenshot Explanation:**
    - Screenshot 1: Hero Section with a promotional banner and CTA button.
    - Screenshot 2: Key Features Section highlighting the advantages of NFC cards.
    - Screenshot 3: Explore our Chain process we deliver.
    - Screenshot 4: Why Us highlights advantage of using Us over Competitors .
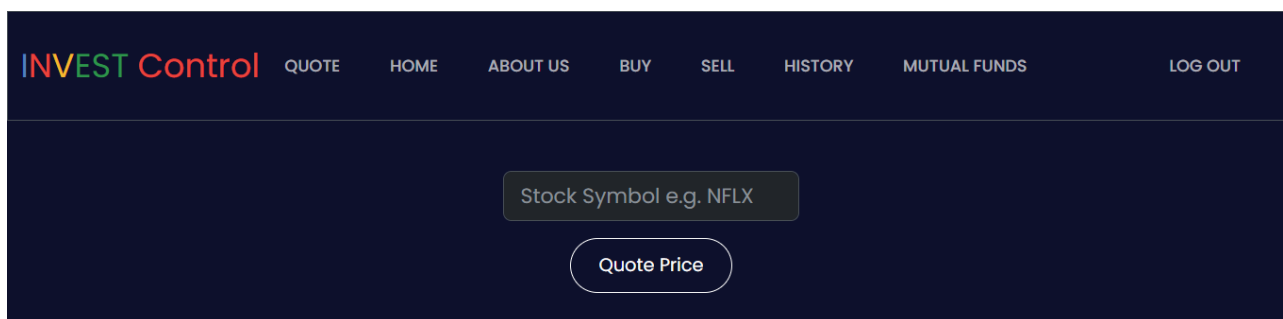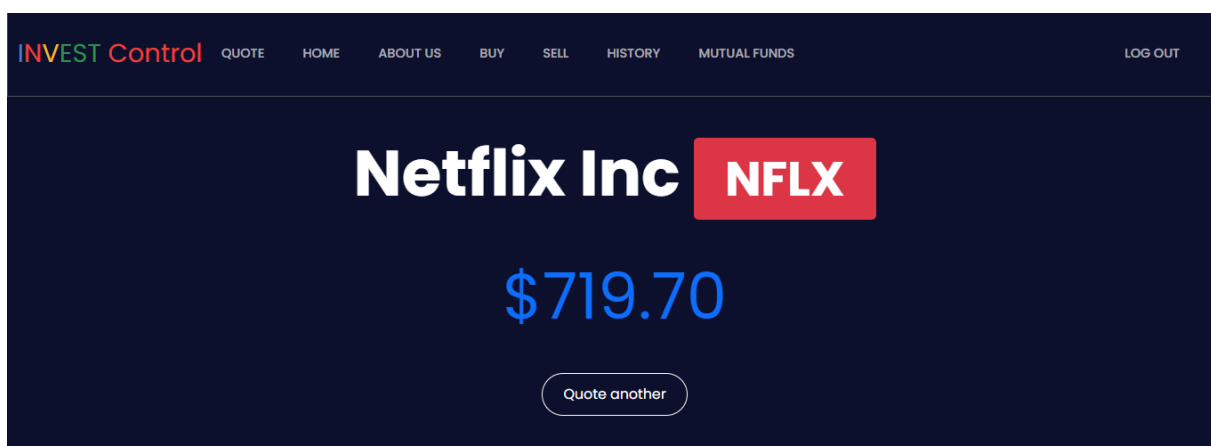

3.3 Stock and Crypt Quote Module

The Stock and Crypto Quote Module enables users to fetch real-time price quotes for stocks and cryptocurrencies by entering a symbol. This user-friendly interface allows users to quickly get the information they need.

- **Key Features:**
- **Quote Input Form:** Users can enter a stock or cryptocurrency symbol to request a quote.
- **Display Quote:** Shows the name and current price of the stock or cryptocurrency once the quote is retrieved.
- **Navigation Button:** Users can easily request another quote without needing to refresh the page.

Figure 9: Stock Quote Page

- **Screenshot Explanation:**
  - Screenshot 1: A search bar , where you have to enter your requested Stock's quote.
  - Screenshot 2: Output is shown with the real-time prices globally, and you can make next quote request using "Quote another" button.
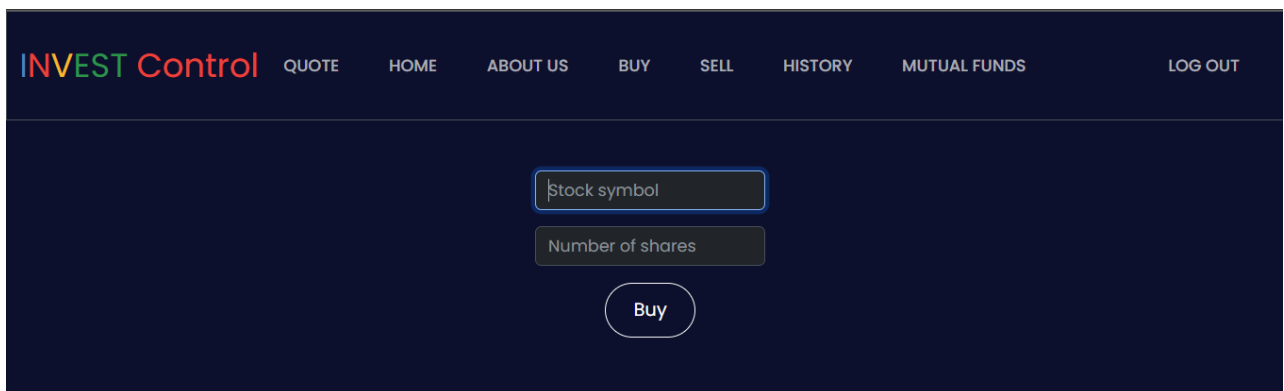

3.4 Stock and Crypt Buy Module

The Stock and Crypto Buy Module provides users with a straightforward way to buy shares of stocks or cryptocurrencies. Users can enter the symbol of the asset and the number of shares they wish to purchase, facilitating seamless transactions.

- **Key Features:**
  - **Purchase Form:** Users can enter a stock or cryptocurrency symbol and the number of shares they want to buy.
  - **Input Validation:** Ensures that valid symbols and share quantities are provided before processing the purchase.
  - **Transaction Processing:** Automatically deducts the total cost from the user's cash balance and records the transaction.
  - **Portfolio Management:** Updates the user's portfolio with the newly purchased shares or creates a new entry if the asset is not already owned.

Figure 10: Stock Buy Overview Page

- **Screenshot Explanation:**
  - Screenshot 1: Users can enter the symbol of the asset and the number of shares they wish to purchase, facilitating seamless transactions.

3.5 Stock and Crypt Sell Module

The Stock and Crypto Sell Module allows users to easily sell their holdings by selecting the desired asset from their portfolio and specifying the number of shares. The system validates the transaction and updates both the user's portfolio and cash balance accordingly.

- **Key Features:**
- o **Sell Form:** Users can select which stock or cryptocurrency they want to sell and enter the number of shares.
- o **Input Validation:** The module checks if the user has sufficient shares to sell and validates the selected symbol and quantity.
- o **Transaction Processing:** Updates the transaction history and modifies the user's cash balance accordingly.
- o **Portfolio Management:** Automatically updates the user's portfolio, removing the asset if all shares are sold.

Figure 11: Sell Page

- **Screenshot Explanation:**
- o Screenshot 1: Show us the Sell stock/Crypt interface with a quantity to enter and a button to sell

o Screenshot 2: Show us the list of the stocks/crypto that the specific user has purchased and want to see.

3.6 History Module

The History Module provides users with a comprehensive overview of their past transactions, enabling them to track their buying and selling activities in an organized manner.

 Key Features:

o **Transaction Table:** Displays transaction type, stock symbol, number of shares, price, and date/time for each transaction.

o **Conditional Rendering:** Informs users if they have no transaction history, encouraging them to make a purchase.

o **Responsive Design:** Ensures a user-friendly experience across devices with a clear and structured layout for transaction data.

Transaction	Symbol	Shares	Price	Date and Time
BUY	GOOGL	12	$161.49	2024-09-26 12:56:29
SELL	GOOGL	1	$161.49	2024-09-26 12:56:57
SELL	GOOGL	2	$162.73	2024-09-27 04:22:47
BUY	NFLX	2	$707.35	2024-09-29 12:27:10
SELL	GOOGL	5	$163.95	2024-09-29 12:27:39
BUY	NVDA	8	$124.92	2024-10-06 10:18:20

Figure 12: Stock History Page

- **Screenshot Explanation:**
o Screenshot 1: History Page showing user history of buying and selling the stocks and crypto.

3.7 Mutual Funds Calculator Integration

The Mutual Funds Calculator Integration allows users to estimate their potential returns from mutual fund investments based on their inputs.

Key Features:

- **User-Friendly Interface**: The calculator features a clean and easy-to-navigate design, allowing users to input their investment amount, duration, expected return rate, tax percentage, and investment type effortlessly.

- **Real-Time Calculation**: As users enter their data, they can instantly calculate estimated returns, including total investment, returns before tax, tax on gains, and final returns after tax.

- **Visual Results**: The results are displayed clearly, helping users understand their potential profits and enabling informed investment decisions.

Figure 13: Mutual Funds Calculator

- **Screenshot Explanation:**
  - Screenshot 1: Mutual Funds Calculator , where you have to write your requirements entry to get results.
  - Screenshot 2: An example of how the result will be shown.

CHAPTER 4: LIMITATIONS AND FUTURE ENHANCEMENTS

4.1 Limitations

While Invest Control has been developed to fulfill essential functional and non-functional requirements, several limitations exist that impact the current project scope and present challenges for future enhancements.

- **Limited Asset Coverage:**
  o Currently supports only stocks and cryptocurrencies. Future expansion could include mutual funds and ETFs.

- **Real-Time Data Constraints:**
  o Occasional discrepancies in real-time price tracking. Improvement needed in data accuracy and update latency.

- **Basic Transaction History Feature:**
  o Lacks detailed analytics and visual insights. Future enhancements could include performance graphs.

- **Scalability Concerns:**
  o Designed for moderate traffic; may face bottlenecks during peak times. Optimization needed for large-scale operations.

- **User Authentication Vulnerabilities:**
  o Current system lacks two-factor authentication (2FA).Additional security features could enhance account protection.

- **Mobile Responsiveness Issues:**
  o Some elements may not display optimally on all devices. Further optimization required for consistent user experience.

- **Limited Payment Options:**
  o Currently supports only bank transfers and credit/debit cards. Integration of more payment methods (e.g., PayPal, cryptocurrencies) needed.

- **Testing and Quality Assurance:**
  o Limited testing across various devices and scenarios. Comprehensive testing phases required for reliability.

4.2 Future Enhancements

Future enhancements for Invest Control will focus on expanding asset options, improving data accuracy, enhancing security, and adding diverse payment methods.

- **Expansion of Asset Options:**
  o We aim to include additional asset types, such as ETFs and commodities, to give users a wider range of investment opportunities and further diversify their portfolios.

- **AI-Powered Trading Insights:**
  o By implementing AI algorithms, we can provide personalized trading insights and recommendations based on user behavior and market trends, helping users make more informed decisions.

- **Enhanced User Interface for Mobile Devices:**
  o To ensure an optimal trading experience, we will focus on enhancing the mobile interface, making it even more user-friendly and responsive for traders on the go.

- **Integration of Multiple Payment Gateways:**
  o Expanding our payment options to include various gateways, such as PayPal, Google Pay, and cryptocurrency payments, will provide users with greater flexibility and convenience during transactions.

- **User Profile and Transaction History Pages:**
  o Once we enhance our data management capabilities, we will implement user profile and transaction history pages. Users will be able to view and manage their investment profiles, track their transaction history, and access detailed reports, which will enhance user satisfaction and engagement.

CHAPTER 5: CONCLUSION

5.1 Conclusion

The journey of developing **Invest Control** has been a rewarding endeavor, resulting in a dynamic online trading platform that combines **cryptocurrency** and **stock trading** in a user-friendly way. Our aim was to create a seamless experience for users, and I believe we've achieved that by allowing **them to track real-time market prices, practice trades, and manage their investments efficiently**—all backed by a solid **API**.

By leveraging **Jinja** for the **front end** and using **Python with Flask for the back end**, we built a flexible and high-performing architecture. We put a strong emphasis on user experience, ensuring the platform is intuitive and accessible, which is essential for both new and experienced traders. With **features** like **user authentication, real-time price tracking, a mutual funds calculator, and transaction history management,** Invest Control stands out as a comprehensive trading solution.

Of course, we faced some challenges along the way. There are areas for **improvement**, such as broadening the **range of assets available**, enhancing our **security** measures, and offering more **payment options**. But these challenges are also exciting opportunities for future enhancements that can take the platform to the next level.

Looking ahead, we plan to expand our asset offerings, introduce **AI-driven trading insights**, and **support multiple languages** and **currencies**. Enhancing scalability and security will also be a priority, and we're considering developing a dedicated **mobile app** to make trading even more accessible.

In summary, this project highlights how modern web technologies can effectively meet the evolving needs of today's traders**. Invest Control** is well-positioned for growth, with the potential to adapt to market changes and user expectations. I'm excited about what the future holds for this platform, and I look forward to seeing it grow and thrive in the trading landscape.