IT-314 Software Engineering

Project: Functional / Non-Functional Requirements, User Stories & Use Cases

Group Members and Student IDs

Member Name	Student Id
DHAMECHA AYUSH VIJAYBHAI	202301204
MODI GAUTAM KIRANKUMAR	202301214
SHAH JENIL SMITESHBHAI	202301249
DODIYA VIVEK JODHABHAI	202301250
VADODARIYA JENISH KIRTIKUMAR	202301202
VORA VANSH VAIBHAVKUMAR	202301266
DODIYA CHIRAYU SHANTILAL	202301236
PATEL KRISH PIYUSHKUMAR	202301264
FARHAN ANSARI	202301256
SHETHWALA MOHAMMED NAUMAN ZAKIR	202301237
PATEL MUKUNDBHAI RAJESHBHAI	202301234

Table of Content: -

- The report contains the following things:

1. Functional Requirements
2. Non - Functional Requirements
3. Domain Requirements
4. User Stories
5. Use Cases

Git repo link: https://github.com/Ayush-Dhamecha-11/InsightStox---Portfolio-analyzer-tracker-and-console.git

1. Functional Requirements(FRs)

From elicitation done, we got an idea about what features should/must/can be present in our platform. The **FRs** are the services the system must provide which are listed below: • **User Authentication:** Register, Login, Logout

Elicitation Techniques Used:

1. Analysis of Existing Systems
2. Brainstorming

How Identified

A fundamental requirement of the platform is secure user authentication, identified through analysis of existing systems, documentation, and brainstorming. This functionality is crucial to ensure account security and personalized access for users.

Additionally, brainstorming within the project team confirmed that without authentication, users cannot have a secure, personalized experience.

Thus, **Register, Login, and Logout were recognized as core features** and are essential for both functional security and for delivering a reliable user experience.

- **Stock Search**: Search by Stock name.

Elicitation Techniques Used:

1. Analysis of Existing Systems
2. Brainstorming

How Identified

We checked existing platforms similar to our project. All of them let users search by stock name, so we saw that it is a common and important feature.

In our team discussion, we agreed that search by stock name makes the system simple for all users and serves as a useful feature.

- **Real-time Price Tracking**: Fetch live market data and update.

Elicitation Techniques Used:
1. Analysis of Existing Systems

2. Interviewing
3. Questionnaires
4. Brainstorming

How Identified

By reviewing other stock platforms, we noticed that all of them provide real-time updates on stock prices and portfolio values. This shows that users expect live information to make timely investment decisions.

Users mentioned that they want their portfolio to update automatically with current market prices, so they can track gains and losses without checking each stock manually.

During team discussions, we agreed that real-time tracking would make the platform more interactive and useful for users, helping them respond quickly to market changes.

Example: a user wants to see the updated value of their portfolio every few minutes during market hours. This scenario confirmed that real-time tracking is an essential feature.

- **Watchlist Creation & Management**: Create and maintain watchlists.

Elicitation Techniques Used:

1. Analysis of Existing Systems
2. Interviewing
3. Questionnaires
4. Brainstorming

How Identified

By reviewing other stock-tracking platforms, we observed that watchlists are a common feature. They allow users to keep an eye on potential investments and stocks separately from their main portfolios, showing that this feature is widely expected.

Users shared that they like to monitor stocks they are interested in before making a purchase. They emphasized that watchlists help them follow price changes, news, and performance without affecting their actual holdings.

During team discussions, we concluded that creating and maintaining watchlists is an essential feature to complement the portfolio, helping users make better investment decisions.

- **Portfolio Creation & Management**: Add/Remove stocks, track holdings

Elicitation Techniques Used:

1. Analysis of Existing Systems
2. Interviewing
3. Questionnaires
4. Brainstorming

How Identified

By reviewing other investment platforms, we noticed that most of them provide portfolio management as a main feature. This shows that users find it useful to keep all their stock details in one place, and it has become a standard expectation in such systems.

In surveys and interviews, users highlighted the need for a feature that allows them to add or remove stocks and track their holdings in a single place.

The above three ways of requirement elicitation concluded our discussion (brainstorming session) to keep this functionality.

- **Visual Dashboard**: Interactive charts, graphs, performance information

Elicitation Techniques Used:

1. Analysis of Existing Systems
2. Interviewing
3. Questionnaires
4. Brainstorming

How Identified

By reviewing other stock platforms, we observed that dashboards with charts and performance information are common. Users expect to see visual summaries of their portfolio and stock performance.
31% of people expressed that understanding detailed analysis is difficult and 45% of people agreed that a good and clear dashboard will help them to visualize their

investment properly, which led us to understand that visually appealing and easy-to-read graphs are more suitable for presenting results.

In team discussions, we concluded that an interactive dashboard would improve usability and make the platform more engaging.

Example: a user wants to view a chart showing portfolio growth over the past month. This confirmed the need for a visual dashboard to support quick insights.

- **Compare stocks**: Compare stocks based on valid parameters

  Elicitation Techniques Used:

  1. Analysis of Existing Systems
  2. Brainstorming

  How Identified

  During team discussions, we concluded that including a stock comparison module would help users analyze investments efficiently and make informed choices.

  By Exploring the existing platforms, we got to know that this is also a common feature which generally helps users to compare and decide before investing.

  For example, a user may want to compare some stocks they are thinking of investing in. These scenarios confirmed that a comparison feature would be good functionality to provide.

- **AI-based portfolio suggestions:** Recommendations, Risk alerts, and Portfolio optimization.

  Elicitation Techniques Used:

  1. Analysis of Existing Systems
  2. Interviewing
  3. Questionnaires
  4. Brainstorming

  How Identified

By reviewing other investment platforms, we noticed that some of them provide portfolio optimization features. These show that automated guidance is valuable for investors.

Articles and guides on investment strategies highlight the benefits of recommendations and risk alerts to improve decision-making and minimize losses. 45% of people expressed that they struggle due to lack of clarity in investment decisions, 61% of people from surveys and all the people from interviews expressed interest in getting suggestions for their portfolio, risk alerts, and advice to optimize their investments.

During team discussions, we focused on the fact that in this world where AI is evolving everywhere, it would be a great idea to safely and legally employ this AI utility as helpers for investments and thus, we concluded that AI-based portfolio suggestions would make the platform more helpful, interactive, and user-friendly.

Example: a user receives a recommendation to rebalance their portfolio or an alert when a stock's risk level changes. This confirmed the need for AI-based guidance.

2. Non-Functional Requirements

These describe the **quality attributes** of the system:

1. **real-time updates (low latency for live data):** the platform must deliver live stock price and portfolio valuation updates with minimal (feasible) latency during market hours to ensure users always view current market conditions.

Elicitation Techniques Used:

1. *Questionnaires*
2. *Interviewing*
3. *Brainstorming*
4. *Analysis of Existing Systems*

How Identified

Through analysis of existing stock platforms, the team observed that investors rely on instant updates to make timely decisions. Questionnaires and interviews with potential users confirmed that delays in stock prices, portfolio values, or watchlist data would negatively affect decision-making.

During brainstorming sessions, the team concluded that low-latency real-time updates are necessary to meet user expectations and provide a responsive experience.

2. **Scalability & Reliability:** The platform must scale to support growth in users, portfolios, and historical data without impacting performance. It should be reliable to cover consistent data processing, error handling, and recovery from failures, so that users can trust the system to deliver good performance.

   Elicitation Techniques Used:

   1. *Brainstorming*

   How Identified

   These are two of the important pillars which help the platform to win user experience and these are necessary things to consider when building the platform which we discussed in the brainstorming sessions.

   Reliability refers to the operational capabilities of the platform and scalability refers to the maintainability with growing number of users and complexities. The goal should be to achieve them to the level which helps to build long term user adoption.

3. **Availability**: The system must remain accessible and functional during NSE/BSE trading hours. Other than that, it must be available the entire day with minimal maintenance time.

   Elicitation Techniques Used:

   1. *Questionnaires*
   2. *Interviewing*
   3. *Analysis of Existing Systems*

   How Identified

   System analysis showed that investors need constant access to their portfolios, watchlists, and live market data to make timely decisions.

   Surveys and interviews confirmed that users expect the platform to be available most of the time, especially during trading hours, with minimal disruptions. This requirement directly affects the performance of the platform and that is why It is one of the necessary requirements to ensure.

4. **Security**: The platform must ensure that all user data is protected from unauthorized access. It must allow users to securely delete or withdraw their account and portfolio information (along with the consent they have provided) at any time.

Elicitation Techniques Used:

1. *Questionnaires*
2. *Interviewing*
3. *Brainstorming*
4. *Analysis of Existing Systems*

How Identified

Response from the questionnaires and interviews showed that the most common concern every user has is security of their data. 67% of people from the survey emphasized that data security is a must.

We refer to the regulatory and data privacy documentation relevant here which also suggested that privacy and security of the user data must be the most important factor in the platform's integration process.

5. **Usability**: The platform should have an intuitive interface that allows users to navigate easily. Clear UI, visualizations and guided explanations should minimize confusion and enhance the overall user experience.

Elicitation Techniques Used:

1. *Questionnaires*
2. *Interviewing*
3. *Brainstorming*

How Identified

Many investors are not very experienced with technology and need a clear and simple interface to understand financial information. Surveys and interviews showed that presenting data with easy-to-read charts and visuals helps users to make better decisions.

Team discussions also revealed that complicated interfaces make it hard for users to know what actions to take. A simple and visual design improves usability, reduces confusion, and makes the platform more comfortable and effective for everyone.

6. **Accuracy:** The platform must ensure that all stock market data is sourced from reliable APIs and is accurate to use. AI-driven results should be accurate, consistent and free from errors to support informed decision-making.

Elicitation Techniques Used:

1. *Questionnaires*
2. *Interviewing*
3. *Brainstorming*

How Identified

Questionnaires and interviews showed that 63% of people demanded that accuracy of the data sources and AI suggestions must be very high as this platform is going to provide important suggestions to the users and lack of accuracy can result in false information which can worsen the investment of users.

This way we got to know that it directly affects user's trust, platform's credibility and reputation. That is why it is one of the most necessary requirements to maintain.

3. Domain and Business Rules

1. Real-Time Stock Data Integration & Compliance with API rules:

The system must integrate with NSE-approved data APIs to provide near real-time stock prices, and historical data which compiles stock exchange API rules.

Elicitation Techniques Used:

1. Analysis of Existing Systems/Documentation.

How Identified

By reviewing existing stock platforms and their documentation, the team observed that APIs from stock exchanges have usage rules, including rate limits and licensing requirements. Analysis of these systems showed that exceeding limits can block access or violate compliance.

During team discussions, it was decided that our platform must follow these API rules to ensure uninterrupted access to market data. This confirmed the need to include API compliance as a domain requirement.

2. **Adherence with financial standards:** All computations and performance metrics must follow recognized financial and legal standards.

Elicitation Techniques Used:

1. *Questionnaires*
2. *Interviewing*
3. *Brainstorming*
4. *Analysis of Existing Systems/Documentation.*

How Identified

Through brainstorming and reviewing similar platforms, the team noted that investors rely on common financial ratios and indicators to evaluate stocks.

Questionnaires and interviews with potential users confirmed that showing these metrics is critical for informed decision-making. Analysis of existing platforms demonstrated how standard indicators like P/E ratio, ROI, and market cap are commonly presented. Combining these observations, the team identified that supporting standard financial ratios and indicators is essential for the platform.

3. **Compliance with Regulatory Bodies:** The platform must comply with rules and regulations set by financial authorities (e.g., SEBI) to ensure lawful and legal execution. This includes following disclosure norms, data handling policies, and restrictions on advisory practices.

Elicitation Techniques Used:

1. *Interviewing*
2. *Brainstorming*
3. *Analysis of Existing Systems/Documentation.*

How Identified

Interviews showed that user need the platform to follow the standard rules and norms announced by financial regulatory bodies like SEBI to ensure that what they are using is authorized and safe to use.

By exploring the documentation of these regulatory bodies, we got to know that this requirement is also directly connected with legality and reputation of the platform that is the main factor behind user trust. Therefore, it concluded that this must be one of the important requirements.

4. User Stories

User Story 1

Front of Card	Back of Card
As an unregistered user, I want to register on the platform, So that I can use the utilities of the application.	Success: - Fetch the details from the user and send a verification OTP. - Validate the OTP given by the user. - Valid user registered and referred to Login Page. Failure: - Display user already exists for existing user. - Display “OTP invalid” in case of user entering invalid OTP. - Display “OTP expired” in case the OTP is expired.

User Story 2

Front of Card	Back of Card

As an unregistered user, I want to register directly from google registration, So that I can register in a faster way.	Success: - Fetch the details from google authentication and validate it. - Valid user registered and referred to Login Page. Failure: - Display user already exists for existing user.

User Story 3

Front of Card	Back of Card
As a registered user, I want to login into the application, So that I can access the application utilities.	Success: - Verify the credentials provided by the user. - A valid user gets logged in and referred to the home page. Failure: - Display message according to the type of failure: - "Incorrect Credentials." - "Unregistered user." - "Incorrect Password."

User Story 4

Front of Card	Back of Card
As a registered user, I want to login directly with Google login, So that I can access the application utilities in directly 1 click.	Success: - Verify the credentials directly by google auth. - A valid user gets logged in and referred to the home page. Failure: - Display message "Unregistered user" in case of user not registered.

User Story

5

Front of Card	Back of Card
As a registered user who has forgotten my password, I want to request a password reset, So that I can set a new password and regain access to my account.	Success: - Take email from the user and send OTP. - Validate OTP and set the new Password provided by the user. Failure: - Display “User unregistered” if the user is not registered. - Display “Invalid OTP” if the user has entered invalid OTP. - Display “OTP expired” in case the OTP is expired.

User Story 6

Front of Card	Back of Card
As a registered user, I want to request a password reset, So that I can set a new password to keep my password safe by updating.	Success: - Reset the Password and refer user to homepage Failure: - Display “old Password incorrect”.

User Story

7

Front of Card	Back of Card
As a registered user, I want to securely log out from the platform, So that I can ensure my account is safe when I am not using it.	Success: - Redirect the user to the login page. - Display a confirmation message *"You have been logged out successfully."* Failure: - If session termination fails, display *"Logout unsuccessful. Please try again."*

User Story 8

Front of Card	Back of Card
As an Investor, I want the platform to reflect real-time stock data, so that I can make timely investment decisions.	Success: - Successful Real-Time Data Display (Market is Open) - Final Closing Data Display (Market is Closed) Failure: - Display the last data available in case of data extraction failure with stating that it's not the real time price and retry to fetch data.

User Story

9

Front of Card	Back of Card
As a registered user, I want to search the stocks I want, So that I can quickly find a specific stock to view its detailed information or add it to my portfolio.	Success: - displays a list of relevant stocks matching the user's query. Failure: - Display message “No results found” in case of result not found. - In case of internal error try retrying the search on a given query.

User Story 10

Front of Card	Back of Card
As a user, I want to maintain a watchlist of stocks I wanted, So that I can track the stocks in which I am more interested.	Success: - Add the stocks the user has selected in the Watchlist. - Display the stocks the user has in its watchlist to the user. Failure: - In case of empty watchlist access display “Watchlist is empty” and provide a search option. - Already added stock should not allow add to watchlist option.

User Story

11

Front of Card	Back of Card
As a user, I want to manage the stocks in my portfolio (add, edit, delete), So that my performance analytics accurately reflect my real-world holdings.	Success: - The user can add, edit or delete stocks in his/her portfolio. - On updating the portfolio its statistics metrics should also be updated. - Prompt the user to confirm the stocks selected to delete. Failure: - Display “Invalid data” In case of anything that is not possible such as negative shares or any other problematic fields. - Save the data in case of any network error. - Should not be allowed to remove stock if its quantity is 0.

User Story 12

Front of Card	Back of Card

User Story

As a user, I want to see a consolidated overview of my most important investment data on a single dashboard, So that I can quickly assess my financial position and access key features without confusion.	Success: - Display the total portfolio value and easy to read visual analysis. - There should be obvious and intuitive navigation to all major sections of the application. Failure: - The Dashboard should not crash in case of failure in loading some specific section, only that specific section should display error. - New Users should not be shown an empty dashboard, It should show the functions to add stocks or portfolio.

13

Front of Card	Back of Card
As a user, I want to select multiple stocks and compare their key performance metrics side-by-side So that I can make effective, data-driven investment decisions.	Success: - The User can select up to 3 stocks to compare on a dedicated comparison page. Failure: - If users try to navigate to comparison with less than 2 stocks then the system should show a clear message and give a search option. - If the entered stock name is incorrect then accordingly a failure message must be displayed.

User Story 14

Front of Card	Back of Card

User Story

As a user, I want to receive suggestions and guidance on my stock allocation, So that I can make strategic adjustments to better align my portfolio with my financial goals.	Success: - Display the user's current asset allocation (e.g., 80% Tech, 20% Healthcare) directly beside a target allocation model that matches their risk profile. - Display easy to understand suggestions such as: Your portfolio is heavily concentrated in the Technology sector. To align with a 'Moderate' profile, consider diversifying into other sectors like Financials or Industrials. To reach their specific goal. Failure: - When a user's risk profile is not available the system should prompt the user to Complete the risk Profile. - When the user portfolio is empty it should display the user to add portfolio holdings to receive allocation insights.

15

Front of Card	Back of Card
As a user, I want to see a visual breakdown of my portfolio's diversification by sector, weightage etc.. and get suggestions to concentrate risks and rebalance my portfolio. So that I can focus on risks and improve my portfolio growth.	Success: - Display clear and interactive charts displaying portfolio allocation according to different categories. - Display suggestions on concentrating the diversified portfolio to rebalance the portfolio. - In the case of a well diversified portfolio display message to acknowledge the user about it. Failure: - In case of an empty portfolio it should prompt the user to add portfolio to analyze diversification. - When Stock's diversification data is unavailable it should be added to an uncategorized part of the chart.

User Story

User Story 16

Front of Card	Back of Card
As a user, I want to get a clear assessment of my portfolio's overall risk and volatility, with data-driven suggestions for improvement, So that I can make informed decisions to optimize my returns while managing my risk exposure.	Success: - The user is presented with a dedicated "Risk Analysis" dashboard that successfully visualizes risk and volatility assessment. - The user is also presented with suggestions on actions to take. Failure: - If the user's portfolio is empty then the display prompts the user to build a portfolio. - When data service for a specific metric fails, still load the dashboard with data unavailable messages.

17

Front of Card	Back of Card
As a user, I want to see detailed performance of my portfolio, So that I can understand how my investments are doing.	Success: - Detailed metrics (returns, gains/losses, trends) are shown. - Data is clear and easy to understand. Failure: - If data is not available, show users “insufficient data” for proper performance analysis.

User Story 18

Front of Card	Back of Card

User Story

As a user, I want to know if my portfolio is stable over short and long periods, So that I can get suggestions if it is not stable.	Success: - Checks on portfolio performance over short and long periods are shown. - Shows stability status clearly. - Gives suggestions to improve if unstable. Failure: - Show error message: “Unable to calculate stability” and then guide for next step to be taken like if data is not available then ask for proper data.

19

Front of Card	Back of Card
As a user, I want the app to calculate taxation on stock sales (e.g. Short-Term Capital Gains (STCG) and Long-Term Capital Gains (LTCG), etc), So that I know the tax for immediate sale or future sale.	Success: - Shows tax for immediate sale and for sale after chosen period. Failure: - Show error if tax cannot be calculated due to missing data and ask the user to enter proper data.

User Story 20

Front of Card	Back of Card

User Story

As a regulatory authority, I want the platform to provide disclaimers, So that users understand investment risks.	Success: - Disclaimer shown clearly before users begin interacting with the platform. - Users acknowledge disclaimer and agree to all terms and conditions before proceeding. Failure: - Show warning and error if the user skips disclaimer acknowledgement or does not agree to terms and conditions.

21

Front of Card	Back of Card

User Story

As a data provider, I want the platform to respect API rate limits, So that my servers are not overloaded.	Success: - Requests stay within allowed limits. - All functionalities will work as they are supposed to work. Failure: - If a limit is crossed, the provider blocks or rejects requests. - On reaching the limit, the currently running process will stop and users will be instructed to stop for a specific cooldown period due to overload of data. After that period of time as data can be fetched again, the user can start again.

User Story 22

Front of Card	Back of Card
As a user, I want to see the specific data, sources, and reasoning behind every AI-generated recommendation, So that I can verify the information, understand the context, and make a confident and well-informed decision.	Success: - Display the reference and proof of all the recommendations that user got by AI Failure: - The system must not present a recommendation without an explanation or resources.

User Story

Front of Card	Back of Card
As a user, I want to filter the entire market of stocks based on a specific set of financial criteria, So that I can discover new companies that match my personal investment strategy.	Success: - The system should display stocks according to a given query with some constraints such as in case of price fluctuates every second so the result should state that the list is as per a given time range so the user doesn't make mistakes due to incorrect data. Failure: - When no stock matches the filter criteria then the system should prompt that there are no stock with given criteria.

5. Use Cases

UC ID	Use Case Name
UC01	User Registration
UC02	Google OAuth Registration
UC03	Login
UC04	Google Login

User Story

UC05	Reset Password
UC06	Logout

UC07	Search Stock
UC08	View Real-Time Stock Data
UC09	Create/Manage Watchlist
UC10	Create/Manage Portfolio
UC11	View Dashboard
UC12	Compare Stocks
UC13	View AI Suggestions
UC14	View Diversification Insights
UC15	View Risk Analysis
UC16	View Portfolio Performance
UC17	Portfolio Stability Check
UC18	Tax Calculation
UC19	View Mandatory Disclaimer
UC20	API Rate Limit Handling

UC21	Explainable AI
UC22	Market Filter Tool

Use Cases Description:- UC01

- User Registration

Primary Actor: User
Secondary Actor: Authentication Service
Preconditions:

- User is not registered
- System is running normally

Postconditions:

- A new verified account is created

Main Flow

1. User opens the registration page
2. User enters name, email, password
3. System validates the input fields
4. System sends an OTP to the user's email
5. User enters OTP
6. System verifies OTP
7. System creates the user account
8. System redirects the user to the login page

Alternate / Exception Flows

- A1: Invalid email → System shows "Invalid email format"
- A2: Weak password → System prompts for stronger password
- A3: OTP incorrect → System asks user to retry
- A4: OTP expired → System allows sending a new OTP
- A5: Email already exists → System shows "User already registered"

UC02 - Google OAuth Registration

Primary Actor: User
Secondary Actor: Google OAuth Service
Preconditions:

- User has a Google account

Postconditions:

- User account created using Google credentials

Main Flow

1. User selects “Register with Google”
2. System redirects to Google OAuth
3. User approves authorization
4. Google returns verified user identity
5. System checks if user already exists
6. If not, system creates a new account
7. User is redirected to dashboard or login

Alternate Flows

● A1: OAuth denied → System cancels registration ● A2: User already exists → System shows error

UC03 - Login

Primary Actor: User
Preconditions:

● User is registered **Postconditions:**

● User session is created

Main Flow

1. User enters email + password
2. System validates credentials
3. System creates a session
4. User is taken to dashboard

Alternate Flows

- A1: Wrong password → Show error
- A2: Email not registered → Prompt to register
- A3: Server unavailable → Show retry message

UC04 - Google Login

Primary Actor: User
Secondary Actor: Google OAuth

Main Flow

1. User selects “Login with Google”
2. OAuth screen opens
3. User grants permission
4. Google sends identity token
5. System logs in user

Alternate

- A1: OAuth cancelled → Return to login

UC05 - Reset Password

Primary Actor: User

Main Flow

1. User clicks “Forgot Password”
2. System asks for registered email
3. System sends OTP / reset link
4. User enters OTP
5. User sets new password
6. System confirms password reset **Alternate**

- A1: Email not found → Show error
- A2: OTP invalid
- A3: OTP expired

UC06 - Logout

Primary Actor: User

Main Flow

1. User clicks “Logout”
2. System terminates session
3. User redirected to login page

Alternate

- None

UC07 - Search Stock

Primary Actor: User
Secondary Actor: Stock API

Main Flow

1. User types stock name in search bar
2. System queries stock API
3. System displays list of matching stocks
4. User selects a stock **Alternate**

- A1: No results found
- A2: API failure → Show cached data ● A3: Network error → Show retry

UC08 - View Real-Time Stock Data

Primary Actor: User
Secondary Actor: Market Data API

Main Flow

1. User selects a stock
2. System fetches live price, volume, charts
3. System displays updated metrics
4. System refreshes data periodically **Alternate**

● A1: API delay → Show last updated value ●
A2: No internet → Show offline notice

UC09 - Manage Watchlist

Primary Actor: User

Main Flow (Add Stock)

1. User opens watchlist
2. User searches for a stock

3. User clicks "Add to Watchlist"
4. System saves stock to watchlist

Main Flow (Remove Stock)

1. User views watchlist
2. User selects a stock
3. User clicks "Remove"
4. System updates the list

Alternate

● A1: Duplicate entry → Show “Already in watchlist” ● A2: Empty watchlist → Show empty state

UC10 - Manage Portfolio

Primary Actor: User

Main Flow (Add Stock)

1. User opens portfolio
2. User clicks “Add Holdings”
3. User enters stock, quantity, buy price
4. System validates input
5. System adds to portfolio
6. System recalculates metrics

Main Flow (Edit Stock)

1. User selects an existing stock
2. User edits details
3. System updates metrics

Main Flow (Delete Stock)

1. User clicks “Delete”
2. System removes entry

Alternate

- A1: Negative quantity → Show error
- A2: Missing details → Reject submission

UC11 - View Dashboard

Primary Actor: User

Main Flow

1. User logs in
2. System loads dashboard
3. Display portfolio value, insights, charts, shortcuts

Alternate

- A1: New user → Show onboarding cards
- A2: No portfolio → Show empty state

UC12 - Compare Stocks

Primary Actor: User

Main Flow

1. User opens compare tool
2. User selects 2–3 stocks
3. System fetches data
4. System displays comparison table/graph **Alternate**

● A1: <2 stocks selected → Show message ●
A2: API failure → Partial data shown

UC13 - View AI Suggestions

Primary Actor: User
Secondary Actor: AI Engine

Main Flow

1. User navigates to AI Insights
2. System fetches AI model output
3. Show suggestions (buy/sell, risk alerts, diversification)

Alternate

- A1: Portfolio empty → Cannot show insights
- A2: AI engine error → Show fallback message

UC14 - View Diversification Insights

Primary Actor: User

Main Flow

1. User opens diversification screen
2. System analyzes sectors, weights
3. Show pie chart + recommendations **Alternate**

- A1: Insufficient data → Show warning

UC15 - View Risk Analysis

Primary Actor: User

Main Flow

1. User opens “Risk Overview”
2. System calculates volatility, drawdowns
3. Display risk score
4. Show risk warnings **Alternate**

- A1: AI engine unavailable ●
A2: Portfolio too small

UC16 - View Portfolio Performance

Primary Actor: User

Main Flow

1. User opens performance tab
2. System loads profit/loss, returns, timeline charts
3. Show summary metrics

Alternate

- A1: Missing inputs → Show placeholder

UC17 - Portfolio Stability Check

Main Flow

1. System evaluates past volatility & performance

2. System shows stability indicator (e.g., Stable / Volatile)

Alternate

- A1: Not enough history

UC18 - Tax Calculation

Main Flow

1. User opens tax insights
2. System identifies STCG, LTCG
3. Show tax summary **Alternate**

- A1: Missing buy/sell price

UC19 - View Mandatory Disclaimer

Main Flow

1. User accesses disclaimers
2. System shows terms, legal notes

Alternate: None

UC20 - API Rate Limit Handling

Main Flow

1. System requests data
2. API returns "rate limit exceeded"
3. System switches to cached data
4. Shows timestamp of last update **Alternate**

- A1: No cache → Show error

UC21 - Explainable AI

Main Flow

1. User asks “Why this suggestion?”

2. System displays explanation:
    - Data source
    - Metrics
    - Model reasoning
3. User understands AI logic

Alternate

- A1: Explanation unavailable → Show generic reasoning

UC22 - Market Filter Tool

Main Flow

1. User opens market filter
2. User sets criteria (P/E, volume, market cap, etc.)
3. System applies filters
4. Show list of matching stocks **Alternate**

- A1: No matches → Show message
- A2: Invalid filter input