### **Functional Requirements**

1. **User Authentication and Authorization**
   * Sign up: Users should be able to create accounts using email, username, and password.
   * Login: Users should be able to log in using their credentials.
   * Password recovery: Users should be able to reset their passwords if they forget them.
   * Role management: Admin roles for managing the system and user roles for trading.
2. **User Dashboard**
   * Overview of the user's portfolio including current balance, stock holdings, and recent transactions.
   * Display key metrics such as total profit/loss, available balance, and investment value.
3. **Market Data**
   * Real-time stock prices: Fetch and display real-time prices for various stocks.
   * Historical data: Provide historical price data for stocks for analysis.
   * Stock details: Show detailed information about each stock, including price charts, company information, and news.
4. **Trading Functionality**
   * Buy/Sell stocks: Users should be able to place buy and sell orders for stocks.
   * Order types: Support for market orders, limit orders, and stop orders.
   * Order history: Display a history of all past transactions and order statuses.
   * Portfolio management: Users can view and manage their stock portfolio.

### **Non-Functional Requirements**

1. **Performance**
   * The application should handle a large number of concurrent users and transactions without significant performance degradation.
   * Real-time data updates should be processed with minimal latency.
2. **Scalability**
   * The system should be scalable to accommodate growth in user base and trading activity.
   * Utilize scalable infrastructure, such as cloud-based services, to ensure high availability.
3. **Security**
   * Ensure secure storage of user data and transactions.
   * Use encryption for data transmission and storage.
   * Implement measures to prevent unauthorized access and ensure data integrity.
4. **Usability**
   * User-friendly interface: Ensure the UI is intuitive and easy to navigate.
   * Responsive design: The app should be accessible on various devices, including desktops, tablets, and smartphones.
5. **Reliability**
   * High availability: Ensure the app is available with minimal downtime.
   * Backup and recovery: Implement regular data backups and a recovery plan in case of data loss or system failure.
6. **Maintainability**
   * Codebase should be well-documented and modular to facilitate easy updates and maintenance.
   * Implement version control and continuous integration/continuous deployment (CI/CD) practices.
7. **Compliance**
   * Ensure compliance with relevant regulations and standards, such as GDPR for data protection.
   * Adhere to financial regulations and guidelines for stock trading simulators.

### **Additional Features (Optional)**

1. **API Integration**
   * Integrate with financial data providers for real-time stock prices and news.
   * Provide API access for users who want to connect their own trading algorithms.

By following these requirements, you can build a comprehensive and effective stock trading simulator app that caters to the needs of both novice and experienced traders.