**Software Requirements**

**Specification**

**FZM Crypto Wallet Application**

**Version 3.0**

**Submitted by:**

**Fasih Ahmad (20F-1036)**

**Abdul Moqeet (20F-0303)**

**Zain Ul Abideen (20F-0412)**

Contents

[1. Introduction 8](#_Toc134890953)

[1.1 Purpose: 8](#_Toc134890954)

[1.2 Scope: 8](#_Toc134890955)

[1.3 Definition: 9](#_Toc134890956)

[1.4 References: 9](#_Toc134890957)

[2. Overall Description 9](#_Toc134890958)

[2.1 Product perspective: 9](#_Toc134890959)

[2.2 User interface: 9](#_Toc134890961)

[2.3 Administrative Interface 10](#_Toc134890962)

[2.4 Product function: 11](#_Toc134890963)

[2.5 User Characteristics 11](#_Toc134890964)

[2.6 Constraints: 12](#_Toc134890965)

[2.7 Assumptions and Dependencies: 13](#_Toc134890966)

[3. Core Functional Requirements 13](#_Toc134890967)

[4. Other Nonfunctional Requirements 16](#_Toc134890974)

[5. Process Model 19](#_Toc134890975)

[6. Use-Case Diagram 20](#_Toc134890976)

[7. Use-Case Extended 21](#_Toc134890977)

[8. Class Diagram 32](#_Toc134890984)

[9. Activity Diagram 33](#_Toc134890986)

[10. System Sequence Diagram 36](#_Toc134890991)

[11. Sequence Diagram 39](#_Toc134890995)

[12. Dataflow Diagram 41](#_Toc134890999)

[13. Component Diagram 43](#_Toc134891003)

[14. Deployment Diagram 43](#_Toc134891004)

[15. Architecture Diagram 44](#_Toc134891005)

[16. Architectural Style 44](#_Toc134891006)

[17. Test Cases 46](#_Toc134891007)

|  |  |
| --- | --- |
| **Modifications Added** | **Page Number** |
| Functional Requirement Refinement | 7-10 |
| Non-Functional Req. Refinement | 11-13 |
| Process Model | 13 |
| Use Case Diagram | 14 |
| Use Case Extend | 15-24 |
| Activity Diagram | 26- 28 |
| System Sequence Diagram | 29-31 |
| Sequence Diagram | 32-34 |
| Data Flow Diagram | 34-36 |
| Component Diagram | 36 |
| Deployment Diagram | 36 |
| Architecture Diagram | 37 |
| Architectural Style | 37 |
| Test Cases | 38-43 |

# 1. Introduction

# Purpose:

The purpose of FZM Crypto Wallet Application is to provide users with a secure and efficient platform to manage their cryptocurrencies. The project aims to address the growing need for a reliable cryptocurrency wallet application that can provide users with the necessary tools to store, manage, and transact with their digital assets. The application will use advanced AI algorithms to provide personalized recommendations, fraud detection, and integration with popular cryptocurrency exchanges.

# Scope:

The FZM Crypto Wallet Application will be used by individuals and businesses who own or transact in cryptocurrencies. The application will provide users with a secure and efficient platform to manage their digital assets, including the ability to store, send, and receive cryptocurrencies.

For cryptocurrency traders, investors, and aficionados who require a dependable and user-friendly platform to manage their digital assets, the project will be very helpful. Businesses who take cryptocurrencies as payment will also profit from the program because it will give them a safe and effective way to handle their bitcoin transactions.

# Definition:

# 

**SRS**  – Software requirement specification

# References:

"Integrating Artificial Intelligence with Blockchain Technology for Developing Crypto Wallets" by R. Sathish Kumar and S. Sivasubramanian, published in the International Journal of Advanced Science and Technology.

* 1. **Overview:**

The FZM Crypto Wallet Application aims to develop a software platform that will allow individuals and businesses to securely manage their cryptocurrencies using a combination of advanced AI algorithms and blockchain technology. The application will provide users with a user-friendly and efficient way to store, send, and receive cryptocurrencies, while also providing them with personalized recommendations and fraud detection features to help them make informed decisions when managing their digital assets.

## 2. Overall Description

# Product perspective:

# 

# The FZM Crypto Wallet will provide users with a comprehensive platform for managing their cryptocurrencies. The application will serve as a digital wallet, allowing users to securely store their cryptocurrencies and manage their transactions. The project will work on the following interfaces:

* User interface
* Administrative interface
* Blockchain Interface

# 2.2 User interface:

The Application’s user interface (UI) will be made with the user's ease of use and intuitiveness in mind, giving them a seamless and effective way to manage their cryptocurrency.

The following screens are part of the UI:

* **Login Screen:** Users can safely connect into the program from this screen by entering their username and password.
* **Dashboard:** This screen will act as the users' primary interface, giving them a summary of their account balances, recent transactions, and other crucial data.
* **Digital Wallet Screen:** Users can examine their cryptocurrency holdings on this screen and add or withdraw cryptocurrencies from their digital wallets.
* **Transaction History:** Users can check their transaction history on this screen, which includes information about the sum of bitcoin transferred, the wallet address of the recipient, and the date and time of the transaction.
* **Buy/Sell Cryptocurrency:** Using well-known cryptocurrency exchanges, this screen enables users to purchase and sell cryptocurrencies from within the application.

2.3 Administrative Interface**:**

FZM Crypto Wallet Application's administrative interface is made to allow administrators to handle user accounts, track transactions, and carry out other administrative duties pertaining to the program.

The following screens will be included in the administration interface:

* **Dashboard:** This page will give administrators a summary of the performance of the program, including the quantity of active users, total volume of transactions, and other important indicators.
* **User Management:** Administrators can manage user accounts on this panel by adding new accounts, editing pre-existing accounts, and deactivating or deleting accounts as necessary.
* **Transaction Management:** With this page, administrators may observe and keep track of all transactions made through the program, as well as specifics like volume, frequency, and history of transactions.
* **Analytics:** With this page, administrators will have access to comprehensive statistics and reporting on the performance of the application, including information on user demographics, transaction volumes, and other important indicators.
* **Security Settings:** With this panel, administrators can modify the security options for the application, including configuring two-factor authentication, password policies, and fraud detection and prevention tools.

# Product function:

The main product function of an AI-based Crypto Wallet application is to provide users with a secure and efficient way to store, manage and transfer their cryptocurrencies. By integrating artificial intelligence technology, such applications can offer advanced features and functionalities that enhance the user experience and improve the overall security of the wallet.

Some of the key features that an AI-based Crypto Wallet application may offer include:

* Smart Transaction Management: The wallet can analyze the user's transaction history and suggest the best time and fees for sending transactions to ensure fast and cost-effective transfers.
* Predictive Asset Management: The wallet can provide users with recommendations on which cryptocurrencies to buy or sell based on market trends and user preferences.
* Advanced Security: The wallet can use AI algorithms to detect and prevent fraudulent activities, such as phishing attacks, hacking attempts, and unauthorized access.
* Personalized User Experience: The wallet can personalize the user experience by providing tailored content, such as news and educational resources, based on the user's interests and investment portfolio.

Overall, an AI-based Crypto Wallet application can help users manage their digital assets more efficiently and securely, while also providing advanced features and functionalities that make the experience more user-friendly and personalized.

# User Characteristics

The user can vary depending on the specific target audience and market segment. However, some general characteristics of users who may be interested in using such an application are:

* **Cryptocurrency Investors and Traders:** These users are actively engaged in buying, selling, and trading cryptocurrencies and are likely to have a diversified portfolio of digital assets.
* **Tech-Savvy Users:** Users who are familiar with blockchain technology and have experience using cryptocurrency exchanges and wallets.
* **Security-Conscious Users:** Users who prioritize the security of their digital assets and are willing to use advanced security features, such as multi-factor authentication, biometric authentication, and encryption.
* **Mobile Users:** Users who prefer to manage their digital assets on-the-go and want to access their crypto wallet from their mobile device.
* **Experienced Users:** Users who are familiar with using artificial intelligence technology and are comfortable with the idea of having an AI-powered crypto wallet.

Users are likely to be cryptocurrency investors and traders who are looking for a secure and efficient way to manage their digital assets while also benefiting from advanced features and functionalities powered by artificial intelligence technology.

# Constraints:

* **Regulatory Compliance:** Cryptocurrencies and their associated technologies are often subject to a complex and evolving regulatory landscape, which can pose legal and compliance challenges for developers and users of crypto wallets.
* **Security Risks:** While AI-based crypto wallets can offer advanced security features, they are not immune to security breaches and cyber-attacks. Developers need to ensure that the wallet is secure and resilient to attacks, which can be time-consuming and resource intensive.
* **Scalability:** As the number of users and transactions on a crypto wallet platform grows, developers need to ensure that the platform can scale to accommodate the increased demand for processing power and storage capacity.
* **Interoperability:** With so many different cryptocurrencies and blockchain networks available, ensuring interoperability between different wallets and platforms can be a challenge.
* **User Adoption:** Convincing users to switch to a new crypto wallet application can be difficult, especially if they are already comfortable with their current wallet and are not convinced of the benefits of an AI-powered solution.

# Assumptions and Dependencies:

**Assumptions**:

* Cryptocurrency adoption will continue to increase, and there will be a growing demand for advanced crypto wallet solutions.
* AI technology will continue to evolve, and new techniques and algorithms will be developed that can enhance the capabilities of the crypto wallet.
* The project team will have access to the necessary hardware, software, and infrastructure to develop and launch the crypto wallet application.
* The development team will have sufficient expertise in both blockchain technology and artificial intelligence to develop and maintain the application.
* The project budget and timeline will be sufficient to complete all development, testing, and launch activities.

**Dependencies:**

* The project may be dependent on the availability and reliability of third-party APIs, software libraries, and development tools that are used in the project.
* The project may be dependent on the stability and security of the underlying blockchain network(s) that the crypto wallet supports.
* The project may be dependent on the regulatory environment for cryptocurrencies and blockchain technology, which can impact the design and functionality of the crypto wallet.
* The success of the project may be dependent on the ability to attract and retain a critical mass of users who are interested in using the AI-powered crypto wallet.
* The project may be dependent on the ability to maintain a high level of security and privacy for users' digital assets and personal information.

# 

# 3. Core Functional Requirements

# 3.1 User Registration and Login:

# The Crypto Wallet Application should provide a secure registration and login process for users to access their wallets.

* Uniquely identified: Yes, this need may be uniquely identified.
* Complete: It seems to have covered every aspect of why users require a safe registration and login process.
* Consistent and unambiguous: The necessity for a secure registration and login process is stated in a clear and straightforward manner in the requirement.
* Atomic: This need may be tested and implemented independently of other requirements; hence the answer is yes.
* Traceable: This criterion can be linked to the demand for safe user access to wallets.
* Prioritized: Given the significance of this demand, it should be given top priority.
* Testable: This need can be tested to confirm and verify it.

# 3.2 Transaction Processing:

# The Crypto Wallet Application should allow users to send and receive cryptocurrencies securely, with the ability to track and cancel transactions as needed.

* Uniquely identified: Yes, this need may be uniquely identified.
* Complete: In outlining the need for safe transaction processing with tracking and cancellation options, it seems to be done.
* Consistent and unambiguous: The criterion defines the need for secure transaction processing in a clear and concise manner.
* Atomic: This need may be tested and implemented independently of other requirements; hence the answer is yes.
* Traceable: This criterion might be linked to the necessity for safe cryptocurrency transmission and receiving.
* Prioritized: Given the significance of this demand, it should be given top priority.
* Testable: This need can be tested to confirm and verify it.

# AI-based Fraud Detection:

# The application should use AI algorithms to detect fraudulent transactions and flag them for further investigation.

* Yes, this criterion may be uniquely identified.
* Complete: The necessity for AI-based fraud detection to spot and flag fraudulent transactions appears to be fully defined.
* Consistent and unambiguous: The criterion defines the need for AI-based fraud detection in a clear and concise manner.
* Atomic: This need may be tested and implemented independently of other requirements; hence the answer is yes.
* Traceable: This criterion might be linked to the demand for high-tech security measures to shield customers from fictitious transactions.
* Prioritized: Given the significance of this demand, it should be given top priority.
* Testable: This need can be tested to confirm and verify it.
  1. **Search Specific Cryptocurrency:**

The system allows the user to search for specific cryptocurrencies.

* Uniquely identified: searching for Bitcoin should return only the cryptocurrency with the symbol "BTC."
* Complete: search function should scan all available wallets and exchanges, and return results that include the user's current holdings, price information, and trading volumes.
* Consistent and unambiguous: search results should be presented in a consistent format, with clear labels and symbols, and without any unnecessary or redundant information.
* Atomic: each search query should return a separate set of results, without affecting other search queries or actions.
* Traceable: search results should include links to the source of the information, such as the blockchain or exchange platform, to allow users to verify the information.
* Prioritized: search results should be sorted by factors such as price, volume, or market capitalization, to help users identify the most valuable or relevant cryptocurrencies.
* Testable: search function should be tested with a representative sample of users, across different devices and networks, to ensure that it works effectively and efficiently.
  1. **UX Customization**

The system allows the user to Customize the UI based on what they want to see on their homepage.

* Uniquely identified: each user's custom UI settings should be uniquely identified and associated with their user account to ensure that their preferred settings are correctly applied every time they use the wallet.
* Complete: All relevant UI elements, such as font size, color scheme, or preferred display options, should be included in the customization options.
* Consistent and unambiguous: The customization options should be labeled clearly and presented in a consistent format, with clear instructions or tooltips to guide users.
* Atomic: The atomic trait means that each customization option should be discrete and self-contained, without any unnecessary dependencies or interactions with other options. Each UI element should have its own customization option, and changing one option should not affect others.
* Traceable: The traceable trait means that any changes made to the customized UI should be traceable and reversible, allowing users to track their changes and revert them if necessary. Users should be able to see their previous UI settings and revert to them at any time.
* Prioritized: The prioritized trait means that the customization options should be prioritized based on their importance and relevance to the user's needs. The most used or essential options should be presented first or highlighted, while less important options can be hidden or grouped under advanced settings.
* Testable: The testable trait means that the customized UI should be testable and validated for usability, functionality, and compatibility with different devices or browsers. The customization options should be tested with a representative sample of users to ensure they are effective, accessible, and error-free.
  1. **AI-powered Smart Trading:**

The Crypto Wallet Application uses AI algorithms to analyze market trends and patterns to offer users predictive trading options, enabling them to maximize their profits.

* Uniquely identified: The specification of the specific set of functionalities that must be AI-powered makes the need unique.
* Complete: The specification has all the intended AI-powered features that the wallet should have, hence it is complete.
* Consistent and explicit: The need is clear in stating that AI would be used to provide cutting-edge functionality for wallets, making it both consistent and clear.
* Atomic: Since it only refers to a single class of AI-powered characteristics, the requirement is atomic.
* Prioritized: Although no explicit priority level for the AI-powered features is mentioned in the requirement, it may be inferred that they are ranked according to importance and viability.
* Testable: The criterion is testable since the AI-powered features' efficiency and correctness can be formally assessed.

# 4. Other Nonfunctional Requirements

* 1. **Security and Reliability:**

The wallet must be secure and reliable, with a high level of protection against hacking attempts, cyber-attacks, and other security threats.

* Yes, this need may be uniquely identified.
* To guarantee the security and dependability of user accounts and assets, it is essential that the wallet has a high level of protection against hacking attempts, cyberattacks, and other security risks.
* Consistent and unmistakable: The specification clearly states that the wallet must have a high level of security and dependability.
* Atomic: It is possible to implement and test this need independently of other requirements.
* Traceable: The need for the wallet to offer customers a secure and dependable platform to store and manage their digital assets can be used to explain this criterion.
* Prioritized: This condition must be met to guarantee the dependability and security of the wallet; hence it must be given top priority.
* Testable: To determine whether the installed security measures effectively guard against hacking attempts, cyberattacks, and other security risks, it is necessary to assess their effectiveness.
  1. **Performance and Scalability:**

The wallet must be able to handle many users and transactions, with fast and efficient performance even during peak usage periods.

* Yes, this need may be uniquely identified.
* Complete: This criterion defines the necessity for the wallet to be able to handle multiple users and transactions with quick and effective performance even during peak usage periods to deliver a smooth user experience.
* Consistent and unambiguous: The necessity for the wallet to have strong performance and scalability is clearly defined in the requirement.
* Atomic: It is possible to implement and test this need independently of other requirements.
* Traceable: The need for the wallet to offer consumers a quick and effective platform for sending and receiving transactions, especially during times of high usage, can be used to explain this criterion.
* Prioritized: This criterion should be prioritized to ensure the usability and user experience of the wallet.
* Testable: This criterion may be tested by looking at how the wallet performs in various usage scenarios and making sure that it can handle a lot of users and transactions with quick and effective performance.
  1. **Usability and User Experience:**

The wallet must be user-friendly and intuitive, with a clean and easy-to-use interface that provides a seamless user experience.

* Yes, this need may be uniquely identified.
* Complete: This criterion appears to define the necessity for the wallet to be user-friendly and intuitive, with a clear and simple interface that offers a smooth user experience.
* Consistent and unambiguous: The need for the wallet to have strong usability and user experience is clearly defined in the requirement.
* Atomic: It is possible to implement and test this need independently of other requirements.
* Traceable: The necessity for a flawless user experience in the wallet, which is essential for ensuring user happiness and adoption, can be linked to this need.
* Prioritized: This need must be given top priority to ensure the usability and user experience of the wallet.
* Testable: Yes, this requirement can be tested by conducting user testing and gathering feedback on the usability and user experience of the wallet, and ensuring that the interface is clean, intuitive, and easy-to-use.
  1. **Accessibility and Availability:**

The wallet must always be accessible and available to users, with minimal downtime or disruption to service.

* Absolutely, this need may be uniquely identified.
* Complete: This need defines the necessity for the wallet to be always accessible and available to users, with little service interruption or downtime.
* Consistent and unambiguous: The need for the wallet to have high accessibility and availability is clearly defined in the requirement.
* Atomic: It is possible to implement and test this need independently of other requirements.
* Traceable: To ensure customer happiness and trust in the wallet, this criterion can be directly linked to the necessity for the wallet to offer users a dependable service.
* Prioritized: This need must be given top priority to ensure the wallet's availability and accessibility.
* Testable: Yes, this requirement can be tested by monitoring the uptime and downtime of the wallet, measuring the time it takes for users to access the wallet, and ensuring that the wallet can handle many users and transactions without experiencing significant downtime or disruption to service.
  1. **Compatibility and Interoperability:**

The wallet must be compatible with different devices and platforms, and support interoperability with other crypto wallets and blockchain networks.

* Complete: This criterion appears to define the need for the wallet to facilitate interoperability with other crypto wallets and blockchain networks, as well as being compatible with various devices and platforms.
* Consistent and unambiguous: The specification clearly states that the wallet must operate without issue across a variety of platforms, devices, and other wallets and networks.
* Atomic: It is possible to implement and test this need independently of other requirements.
* Traceable: To ensure that users can quickly transfer their digital assets to and from the wallet regardless of the devices and platforms they use, this criterion can be linked to the need for the wallet to be accessible and simple to use for all users.
* Prioritized: This need should be given the appropriate priority to ensure that the wallet is user-friendly and available to a variety of users.
* Testable: Yes, this criterion may be checked by examining how well the wallet functions across a variety of hardware and software platforms, as well as how well it communicates with other wallets and blockchain networks. The ability of the wallet to handle diverse file formats, protocols, and API interfaces, as well as the ability to move digital assets across various wallets and networks, may all be tested as part of compatibility and interoperability testing.
  1. **User Authentication and Authorization**:

The wallet must provide secure and user-friendly authentication and authorization methods, including multi-factor authentication and biometric authentication.

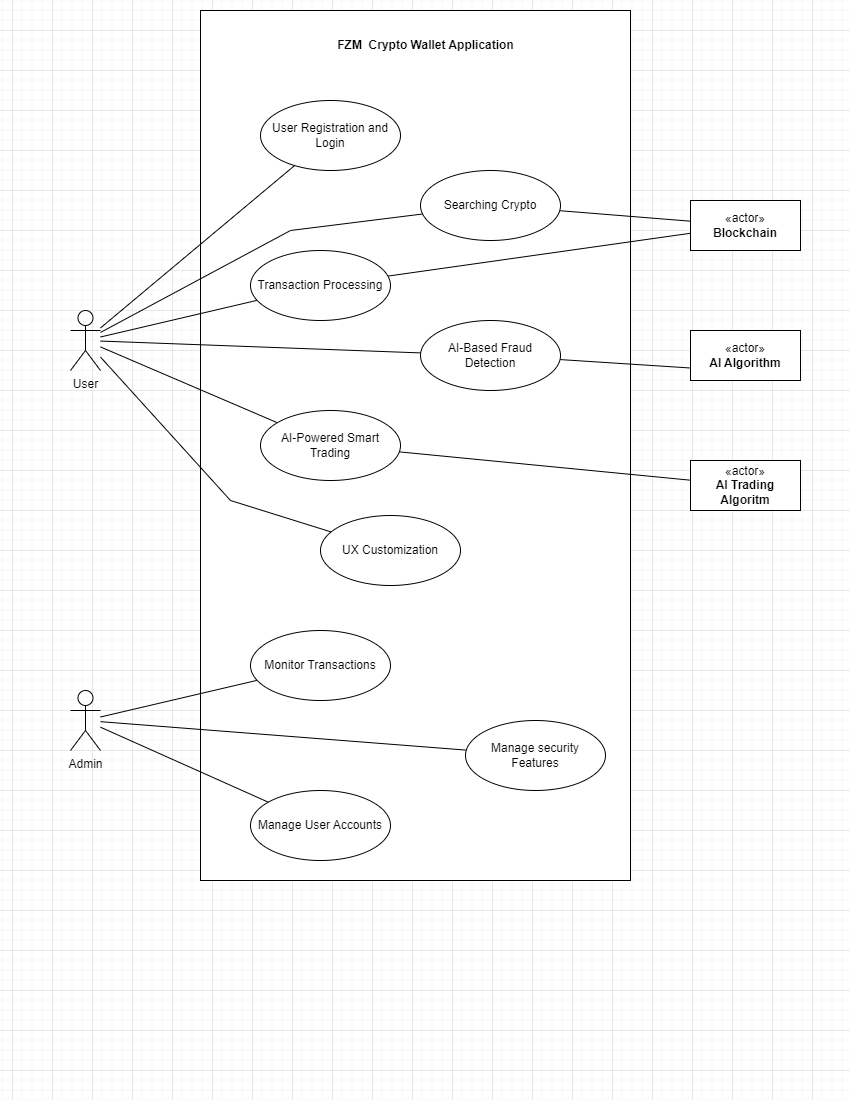
* Uniquely identified: Certain functionality that the wallet must have, such as offering safe and convenient authentication and authorization methods, including multi-factor authentication and biometric authentication, is what makes the criterion distinctive.
* Complete: The requirement satisfies all elements of user authentication and authorization, including security and usability, hence it is complete.
* Consistent and explicit: The requirement makes it obvious that secure and user-friendly authentication and authorization mechanisms, including multi-factor authentication and biometric authentication, are required. This makes the requirement plain and consistent.
* Atomic: The criterion is atomic since it only defines one feature that the wallet must have, namely the availability of safe and convenient mechanisms for authentication and authorization.
* Traceable: The necessity for maintaining the security of user accounts and transactions can be linked to the requirement.
* Prioritized: The need may be ranked in accordance with the value placed on preserving user accounts' and transactions' security. Before the wallet is released, this high-priority criteria must be fulfilled.
* Testable: The need is testable because it can be put to the test to determine whether the authentication and authorization techniques offered are safe, easy to use, and incorporate biometric and multi-factor authentication.

# 5. Process Model

The Process model we selected for this Project is Agile Development.

Agile methodologies, such as Scrum or K, are iterative and incremental approaches that emphasize flexibility, adaptability, and collaboration. They are well-suited for projects that require frequent changes and updates and can be particularly useful for developing AI-based applications where requirements may evolve over time. Agile development involves iterative cycles of planning, development, testing, and feedback, allowing for continuous improvement and rapid delivery of working software.

# 6. Use-Case Diagram



# 7. Use-Case Extended

# User Registration and Login

|  |  |  |
| --- | --- | --- |
| *Use case Name* | *User Registration and Login* | |
| *Primary Actors* | *Business User,*  *Non-Business User* | |
| *Description* | *This use case describes the process of a user registering and logging into their Crypto Wallet Application account to access their digital wallets.* | |
| *Precondition* | * *The Crypto Wallet Application is installed on the user's device.* * *The user has a valid email address.* | |
| *Postcondition* | * *The user is registered and logged into their Crypto Wallet Application account.* * *The user can access their digital wallets.* | |
| *Main Success Scenario* | *Step No.* | *Action* |
| *1.* | *The user opens the Application and is displayed the Login and Registration options.* |
| *2.* | *User selects the registration option and is prompted to enter their email and set a strong password.* |
| *3.* | *The application validates email and password and send a verification link on user’s email.* |
| *4.* | *Upon Verification, the application creates the user’s account and sends a confirmation message.* |
| *5.* | *The user enters their email and password on login page.* |
| *6.* | *The application validates the user’s credentials and logs them into their account.* |
| *7.* | *The application then displays the user’s digital assets.* |
| *Alternative Flow* | *1.* | *Application displays an error message for invalid email address and/or weak password* |
| *2.* | *Application displays an error message and prompts the user to verify their account if it is not verified yet.* |
| *Extension* | * *The application must validate email addresses and passwords for security purposes.* * *The application must send a verification email to the user's email address to confirm their account.* * *The application must encrypt user data to ensure privacy and security.* | |

# Transaction Processing

|  |  |  |
| --- | --- | --- |
| *Use case Name* | *Transaction Processing* | |
| *Primary Actors* | *Business User,*  *Non-Business User,*  *Blockchain Network* | |
| *Description* | *The Crypto Wallet Application allows users to securely send and receive cryptocurrencies with the ability to track and cancel transactions as needed* | |
| *Precondition* | * *The user must have a Crypto Wallet Application account and be logged in.* * *The user must have sufficient funds in their wallet to initiate a transaction.* * *The user must know the recipient's wallet address.* | |
| *Postcondition* | * *The user's wallet balance is updated based on the transaction status.* * *The user can view the transaction status, including confirmation or cancellation, using the transaction hash.* * *The user can initiate a new transaction or cancel a pending transaction as needed.* | |
| *Main Success Scenario* | *Step No.* | *Action* |
| *1.* | *The user selects the cryptocurrency that they want to sell or buy and its amount along with the transaction fee.* |
| *2.* | *System prompts the user to confirm transaction before proceeding* |
| *3.* | *After confirmation, system generates a unique transaction hash and sends the request to the blockchain network* |
| *4.* | *The blockchain network validates the transaction request, verifies the user's wallet balance, and confirms the transaction* |
| *5.* | *The system updates the user's wallet balance and displays a confirmation message with the transaction hash.* |
| *6.* | *The user can track the transaction status by checking the transaction hash on the blockchain explorer. The system also provides a link to the blockchain explorer for easy access.* |
| *7.* | *If the user needs to cancel the transaction, they can navigate to the "Transaction History" section of the wallet, select the pending transaction, and click on the "Cancel" button.* |
| *8.* | *If the user confirms the cancellation, the system sends a cancellation request to the blockchain network.* |
| *9.* | *The blockchain network verifies the cancellation request and cancels the transaction.* |
| *Alternative Flow* | *1.* | *If the blockchain network rejects the transaction request, the system displays an error message, and the user can try again.* |
| *Extension* | * *The user's wallet balance is updated based on the transaction status.* * *The user can view the transaction status, including confirmation or cancellation, using the transaction hash.* * *The user can initiate a new transaction or cancel a pending transaction as needed.* | |

# AI-Based Fraud Detection

|  |  |  |
| --- | --- | --- |
| *Use case Name* | *AI-Based Fraud Detection* | |
| *Primary Actors* | *Business User,*  *Non-Business User,*  *AI Algorithm* | |
| *Description* | *The Crypto Wallet Application uses AI algorithms to detect fraudulent transactions and flag them for further investigation.* | |
| *Precondition* | * *The user must have a Crypto Wallet Application account and be logged in.* * *The user must have at least one transaction history in their wallet.* | |
| *Postcondition* | * *The system flags potential fraudulent transactions for further investigation.* * *The security team reviews and investigates flagged transactions.* * *The user can view the transaction status, including confirmation or cancellation, using the transaction hash.* | |
| *Main Success Scenario* | *Step No.* | *Action* |
| *1.* | *The user initiates a transaction by sending or receiving cryptocurrencies.* |
| *2.* | *The system records the transaction details and passes them to the AI algorithm for analysis.* |
| *3.* | *The AI algorithm analyzes the transaction details and compares them to known fraud patterns and behaviors.* |
| *4.* | *If the AI algorithm detects potential fraud, it flags the transaction for further investigation.* |
| *5.* | *The system displays a message to the user indicating that the transaction is under review.* |
| *6.* | *The system notifies the security team to investigate the flagged transaction.* |
| *7.* | *The security team reviews the transaction and determines whether it is fraudulent or legitimate.* |
| *8.* | *The system updates the transaction status based on the security team's findings.* |
| *9.* | *The user can view the transaction status in the "Transaction History" section of the wallet.* |
| *Alternative Flow* | *1.* | *If the AI algorithm does not detect potential fraud, the system allows the transaction to proceed as normal.* |
| *2.* | *If the AI algorithm generates a false positive, the security team investigates and determines that the transaction is legitimate. The system updates the transaction status accordingly.* |
| *Extension* | * *If the AI algorithm is unable to analyze the transaction details, the system does not flag the transaction and allows it to proceed as normal.* * *If the security team is unable to investigate the flagged transaction, the system displays a message asking the user to try again later.* | |

# Searching Cryptocurrency

|  |  |  |
| --- | --- | --- |
| *Use case Name* | *Search Specific Cryptocurrency* | |
| *Primary Actors* | *Business User,*  *Non-Business User* | |
| *Description* | *The user wants to find their digital assets effectively by searching for specific cryptocurrencies in their AI-based crypto wallet.* | |
| *Precondition* | * *The user has a registered account in the AI-based crypto wallet.* * *The user has a stable internet connection and a compatible device to access the wallet* | |
| *Postcondition* | * *The user can view the search results for the specified cryptocurrency.* * *The user can manage their digital assets effectively using the search function.* | |
| *Main Success Scenario* | *Step No.* | *Action* |
| *1.* | *The user opens the AI-based crypto wallet and logs in to their account* |
| *2.* | *The user navigates to the search function in the wallet.* |
| *3.* | *The user enters the name or ticker code of the cryptocurrency they want to search for.* |
| *4.* | *The system searches for the cryptocurrency across all available wallets and exchanges connected to the user's account.* |
| *5.* | *The system displays the search results, including the name, ticker code, current price, trading volume, and other relevant information about the cryptocurrency.* |
| *6.* | *The user can select a specific cryptocurrency from the search results to view its detailed information.* |
| *7.* | *The user can manage their digital assets effectively by buying, selling, or transferring the selected cryptocurrency.* |
| *Alternative Flow* | *1.* | *If the user enters an invalid or non-existent cryptocurrency name or ticker code, the system displays an error message and prompts the user to enter a valid name or code.* |
| *Extension* | * *If the system encounters technical issues or errors during the search process, it displays an error message and prompts the user to try again later or contact customer support.* | |

# UX Customization

|  |  |  |
| --- | --- | --- |
| *Use case Name* | *UX Customization* | |
| *Primary Actors* | *Business User,*  *Non-Business User* | |
| *Stakeholders and interests* | *Crypto Users* | *Change the appearance of the UI according to their needs* |
| *Precondition* | *Users must log in to a crypto wallet.* | |
| *Postcondition* | *After the change, the setting user must save the setting.* | |
| *Main Success Scenario* | *Step No.* | *Action* |
| *1.* | *The user opens the setting and selects the homepage setting.* |
| *2.* | *Modify the homepage setting.* |
| *3.* | *Then after modification users must save the setting.* |
| *Extension* | *If the setting is not saved, then it will not change the homepage permanently.* | |

# AI-Powered Smart Trading

|  |  |  |
| --- | --- | --- |
| *Use case Name* | *AI-Powered Smart Trading* | |
| *Primary Actors* | *Business User,*  *Non-Business User,*  *AI Trading Algorithm* | |
| *Description* | *The Crypto Wallet Application uses AI algorithms to analyze market trends and patterns to offer users predictive trading options, enabling them to maximize their profits.* | |
| *Precondition* | * *The user must have a Crypto Wallet Application account and be logged in.* * *The Crypto Wallet Application must have integrated an AI-powered trading system.* | |
| *Postcondition* | * *The user's digital assets are securely traded based on market predictions to maximize their profits.* * *The AI-powered trading system analyzes market trends and patterns to offer users predictive trading options.* * *The user can view the details of their past trades, including the cryptocurrency traded, the trading option selected, and the resulting profit or loss.* | |
| *Main Success Scenario* | *Step No.* | *Action* |
| *1.* | *The user selects the "Smart Trading" section of the Crypto Wallet Application.* |
| *2.* | *The system displays the user's current cryptocurrency balances and the available options for predictive trading based on market trends and patterns.* |
| *3.* | *The user selects a cryptocurrency to trade and selects an appropriate trading option.* |
| *4.* | *The AI trading system analyzes the market data and provides a prediction of whether the value of the cryptocurrency will increase or decrease.* |
| *5.* | *The system displays the predicted outcome and the potential profit or loss of the trade.* |
| *6.* | *The user confirms the trade by entering the necessary details, such as the amount of cryptocurrency to trade and the target price.* |
| *7.* | *The AI trading system automatically executes the trade when the target price is reached, or the user cancels the trade.* |
| *Alternative Flow* | *1.* | *If the AI trading system is unable to provide a prediction, the system displays an error message and prompts the user to try a different trading option.* |
| *2.* | *If the predicted outcome changes before the trade is executed, the system notifies the user and prompts them to confirm or cancel the trade with the updated information.* |
| *Extension* | * *If the user experiences a technical issue while executing a trade, the system may prompt the user to retry or cancel the trade.* * *If the user experiences a significant loss due to a trade, the Crypto Wallet Application is not responsible for the financial loss incurred.* * *If the AI trading system provides inaccurate predictions, the Crypto Wallet Application is not responsible for any financial loss incurred.* | |

# 8. Class Diagram

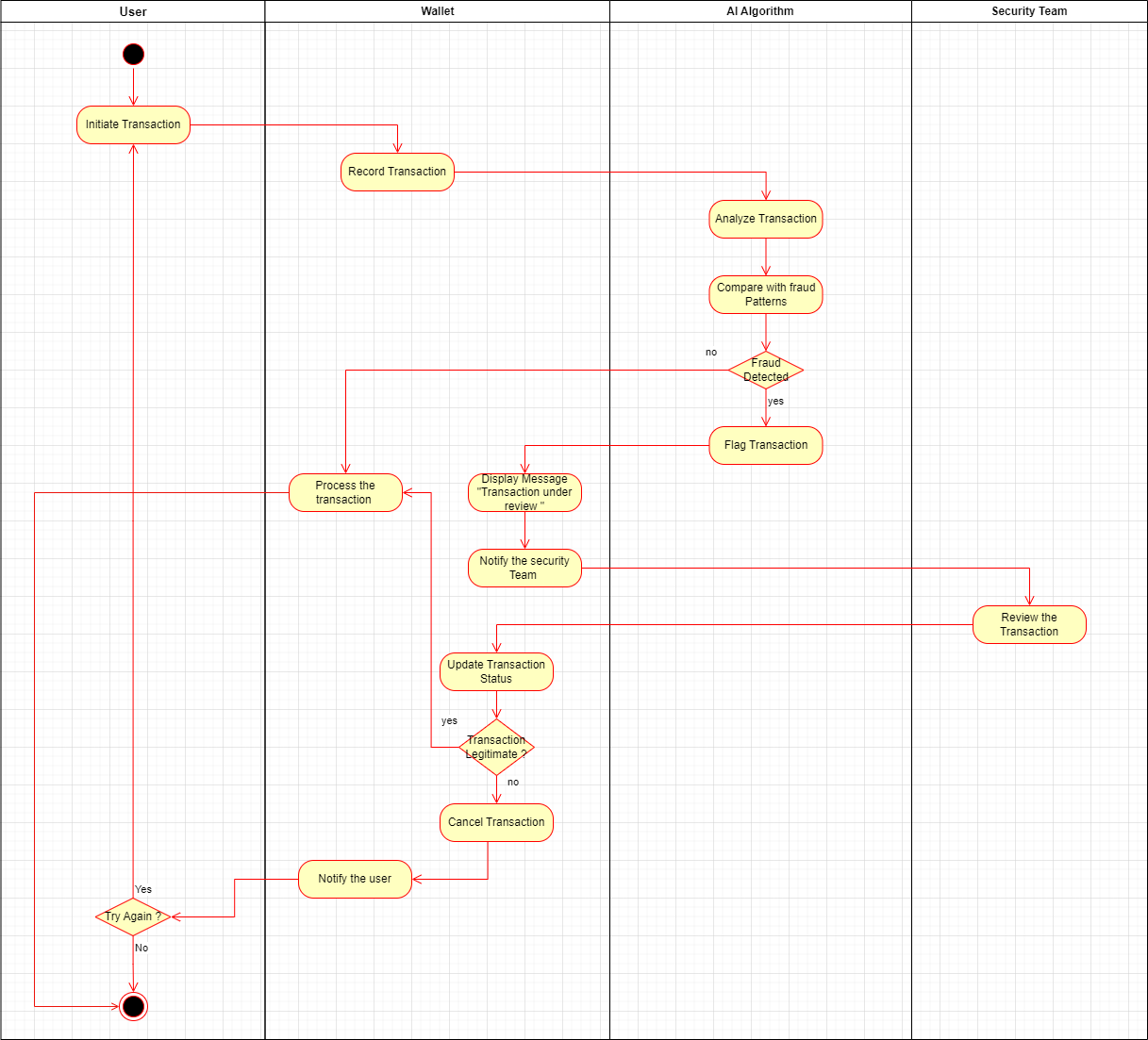
# 

# 9. Activity Diagram

# Transaction Processing

# 

# AI-Based Fraud Detection



# AI-Based Smart Trading

Diagram, schematic

Description automatically generated

# 10. System Sequence Diagram

**User Registration and Login**

# 

**Transaction Processing**

# 

**Searching Cryptocurrency**

# A screenshot of a diagram Description automatically generated with low confidence

# 11. Sequence Diagram

**AI-Based Fraud Detection**

# 

**AI-Powered Smart Trading**

# 

**UX Customization**

# Diagram Description automatically generated

# 12. Dataflow Diagram

Context Level

# 

Level 1.0

# 

Level 2.0

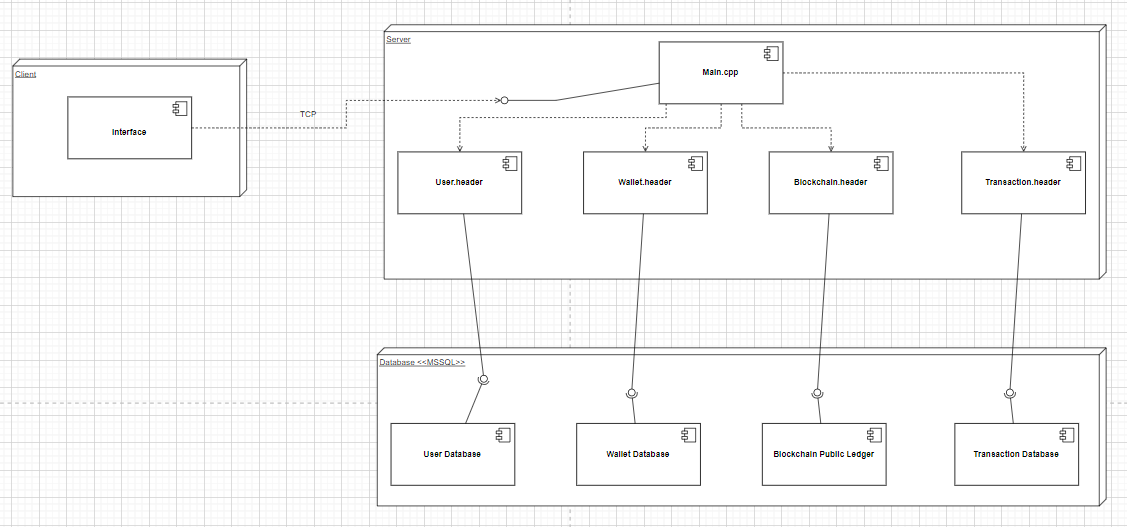
# Diagram Description automatically generated

# 13. Component Diagram

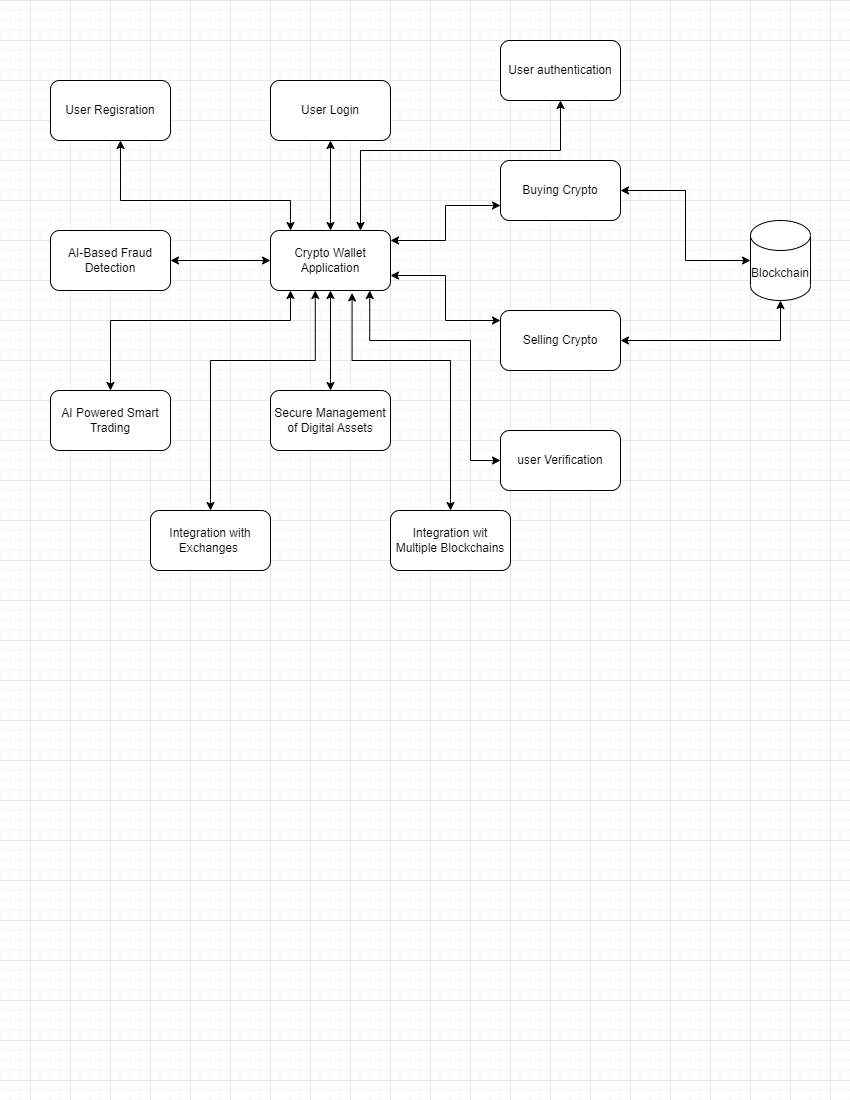
Diagram

Description automatically generated

# 14. Deployment Diagram



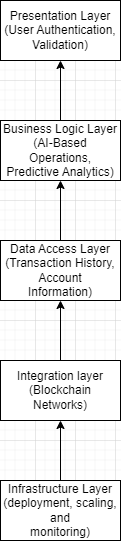
# 15. Architecture Diagram



# 16. Architectural Style

For our given project we are going to use Layer Architecture.

This traditional architectural style involves breaking the application into different layers, such as presentation, business logic, and data access. Each layer has a specific responsibility and communicates with other layers through defined interfaces. This architecture can be suitable for a crypto wallet application as it provides separation of concerns and allows for flexibility in changing or adding components without affecting other parts of the system.



# 17. Test Cases

**Test Case 1:**

|  |  |
| --- | --- |
| Project Name: | FZM Crypto Wallet |
| Module Name: | User Registration |
| Reference Document: |  |
| Created By: | Zain Ul Abideen |
| Date of Creation: | 13-05-2023 |
| Date of Review: | 13-05-2023 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Test Case Scenario | Test Case | Pre-Condition | Test Steps | Test Data | Expected Result | Post Condition |
| TC\_Reg\_001 | Verify the registration of new user | Enter valid email and Strong Password | The Crypto Wallet Application is installed on the user's device | 1.Enter email | davidmiller303@gmail.com | Successful Registration | A verification link is sent to user’s email address |
| 2. Enter Password | Hiitvbbvyr776994!@@$ |
| 3. Click on register button |  |
| TC\_Reg\_001 | Verify the registration of new user | Enter valid email and weak password | The Crypto Wallet Application is installed on the user's device | 1.Enter email | davidmiller303@gmail.com | A message “Password is too weak” is shown |  |
| 2. Enter Password | Helloworld123 |
| 3. Click on register button |  |
| TC\_Reg\_001 | Verify the registration of new user | Enter Invalid email and strong password | The Crypto Wallet Application is installed on the user's device | 1.Enter email | Davidmiller909@gmail.io.uk.edu | A message “Invalid email” is shown |  |
| 2. Enter Password | Hiitvbbvyr776994!@@$ |
| 3. Click on register button |  |
| TC\_Reg\_001 | Verify the registration of new user | Enter invalid email and weak password | The Crypto Wallet Application is installed on the user's device | 1.Enter email | Davidmiller909@gmail.io.uk.edu | A message “invalid email and weak password is shown |  |
| 2. Enter Password | Helloworld123 |
| 3. Click on register button |  |

**Test Case 2:**

|  |  |
| --- | --- |
| Project Name: | FZM Crypto Wallet |
| Module Name: | User Login |
| Reference Document: |  |
| Created By: | Zain Ul Abideen |
| Date of Creation: | 13-05-2023 |
| Date of Review: | 13-05-2023 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Test Case Scenario | Test Case | Pre-Condition | Test Steps | Test Data | Expected Result | Post Condition |
| TC\_Login\_002 | Verify the Login of user | Enter valid email and valid Password | Users must have verified their account. | 1.Enter email | davidmiller303@gmail.com | Successful Registration | Application dashboard is shown |
| 2. Enter Password | Hiuubrdsabvyr776994!@@$ |
| 3. Click on Login button |  |
| TC\_Login\_002 | Verify the Login of user | Enter valid email and invalid password | User must have verified their account | 1.Enter email | davidmiller303@gmail.com | A message “email and password invalid” is shown |  |
| 2. Enter Password | 87989786rld123 |
| 3. Click on Login button |  |
| TC\_Login\_002 | Verify the Login of user | Enter Invalid email and valid password | User must have verified their account | 1.Enter email | davidmiller303@outlook.com | A message “email and password invalid” is shown |  |
| 2. Enter Password | Hiuubrdsabvyr776994!@@$ |
| 3. Click on Login button |  |
| TC\_Login\_002 | Verify the Login of user | Enter invalid email and invalid password | User must have verified their account | 1.Enter email | davidmiller303@outlook.com | A message “email and password invalid” is shown |  |
| 2. Enter Password | Hellookkokiji123 |
| 3. Click on Login button |  |

**Test Case 3:**

|  |  |
| --- | --- |
| Project Name: | FZM Crypto Wallet |
| Module Name: | Transaction Amount verification |
| Reference Document: |  |
| Created By: | Fasih Ahmad |
| Date of Creation: | 13-05-2023 |
| Date of Review: | 13-05-2023 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Scenario | Test Case | Pre-Condition | Test Steps | Test Data | Expected Result | Post Condition |
| TC\_TAV\_007 | Verify the user’s Transaction when buying Crypto | Enter valid Purchasing amount | User must have a verified account | Enter amount of crypto to purchase | Crypto price = $300  Wallet Funds = $600 | Transaction Successful | User’s wallet balance updated |
| TC\_TAV\_007 | Verify the user’s Transaction when buying Crypto | Enter valid Purchasing amount | User must have a verified account | Enter amount of crypto to purchase | Crypto price = $600  Wallet Funds = $600 | Transaction Successful | User’s wallet balance updated |
| TC\_TP\_007 | Verify the user’s Transaction when buying crypto | Enter invalid Purchasing amount | User must have a verified account | Enter amount of crypto to purchase | Crypto Price = $900  Wallet Funds = $600 | Show Message “Insufficient funds to perform transaction” |  |
| TC\_TP\_008 | Verify the user’s Transaction when selling Crypto | Enter valid selling quantity | User must have a verified account | Enter quantity of crypto to sell | Selected quantity = 10 BTC  Total = 20 BTC | Transaction Successful | User’s wallet balance updated |
| TC\_TP\_008 | Verify the user’s Transaction when selling Crypto | Enter valid selling quantity | User must have a verified account | Enter quantity of crypto to sell | Selected quantity = 20 BTC  Total = 20 BTC | Transaction Successful | User’s wallet balance updated |
| TC\_TP\_008 | Verify the user’s Transaction when selling Crypto | Enter invalid selling quantity | User must have a verified account | Enter quantity of crypto to sell | Selected quantity = 21 BTC  Total = 20 BTC | Show message “Incorrect crypto amount entered” |  |

**Test Case 4:**

|  |  |
| --- | --- |
| Project Name: | FZM Crypto Wallet |
| Module Name: | AI Based Fraud Detection |
| Reference Document: |  |
| Created By: | Fasih Ahmad |
| Date of Creation: | 13-05-2023 |
| Date of Review: | 13-05-2023 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Scenario | Test Case | Pre-Condition | Test Steps | Test Data | Expected Result | Post Condition |
| TC\_AI\_FD\_004 | Detect Fraudulent transactions | Enter acceptable amount and known Wallet Address | User must have at least one transaction history | 1.Intiate transaction | 30 BTC or $1400 | Do not Flag Transaction | Transaction not interrupted |
| 2.Record Transaction (wallet address) | mukpjPudK8pwwCAqdzdqTfFjaXXdgYRJCu |
| 3.Pass them to AI algorithm |  |
| TC\_AI\_FD\_004 | Detect Fraudulent transactions | Enter unacceptable amount and known Wallet Address | User must have at least one transaction history | 1.Intiate transaction | 20000 BTC or $100000000000000000000 | Forward the transaction to security team for investigation |  |
| 2.Record Transaction (wallet address) | mukpjPudK8pwwCAqdzdqTfFjaXXdgYRJCu |
| 3.Pass them to AI algorithm |  |
| TC\_AI\_FD\_004 | Detect Fraudulent transactions | Enter acceptable amount and unknown Wallet Address | User must have at least one transaction history | 1.Intiate transaction | 30 BTC or $1400 | Forward the transaction to security team for investigation |  |
| 2.Record Transaction (wallet address) | 0x0D00E28C99133ED5DEEF63C30777ABE89C2D9453 |
| 3.Pass them to AI algorithm |  |
| TC\_AI\_FD\_004 | Detect Fraudulent transactions | Enter unacceptable amount and unknown Wallet Address | User must have at least one transaction history | 1.Intiate transaction | 20000 BTC or $100000000000000000000 | Flag the transaction | Transaction declined |
| 2.Record Transaction (wallet address) | 0x0D00E28C99133ED5DEEF63C30777ABE89C2D9453 |
| 3.Pass them to AI algorithm |  |

**Test Case 5:**

|  |  |
| --- | --- |
| Project Name: | FZM Crypto Wallet |
| Module Name: | Cryptocurrency Search |
| Reference Document: |  |
| Created By: | Abdul Moqeet |
| Date of Creation: | 13-05-2023 |
| Date of Review: | 13-05-2023 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test case ID | Test Case Scenario | Test Case | Pre-Condition | Test Steps | Test Data | Expected Result | Post Condition |
| TC\_CS\_004 | Verify the search results of the user | Enter Keywords for crypto currencies that are real and in search database | User must have a verified account | 1. Select the search box | Bitcoin or BTC  Ethereum or ETH | Display the search results |  |
| 2. Type in Keywords |
| 3. Click the “Search” Button |
| TC\_CS\_004 | Verify the search results of the user | Enter Keywords for non-existing Crypto currencies and that are not in search Database | User must have at least one transaction history | 1. Select the search box | FASTcoin  GDCccoin  FZMcoin | Show Message “The typed crypto does not exist |  |
| 2. Type in Keywords |
| 3. Click the “Search” Button |
| TC\_CS\_004 | Verify the search results of the user | Enter Keywords for crypto currencies that are real but not in search database | User must have at least one transaction history | 1. Select the search box | Dogecoin  Dash | Show Messege “The given Crypto blockchain is not supported yet” |  |
| 2. Type in Keywords |
| 3. Click the “Search” Button |