Money Machine

Requested from:

**Document Information**

|  |  |
| --- | --- |
| **Client** | All Clients |
| **Subject** | Money Machine |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Released** | **Description of Change** | **Version** | **Author** |
| 24 August 2024 | Initial Document | 1.0 | Jia Lin |

Table of Contents

[Process Flow Diagram 4](#_Toc176091286)

[Use Case Diagram 5](#_Toc176091287)

[Introduction 6](#_Toc176091288)

[Objective 6](#_Toc176091289)

[**1.1 Registration** 7](#_Toc176091290)

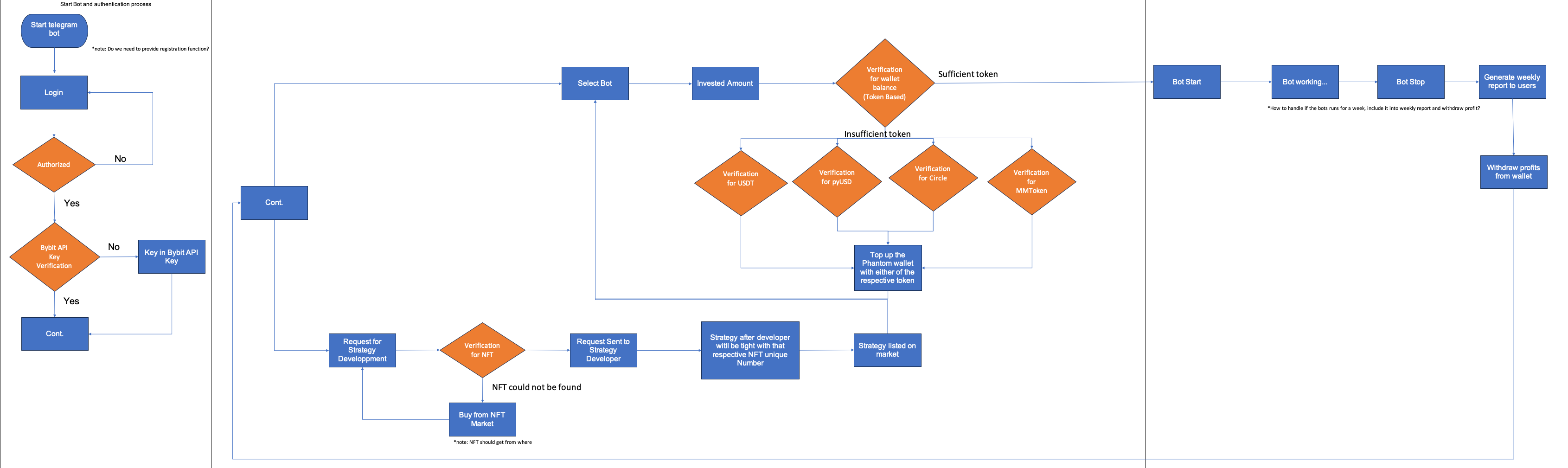
[**2.1 Login** 8](#_Toc176091291)

[**3.1 Link Bybit API** 9](#_Toc176091292)

[**4.1 Bot Selection** 10](#_Toc176091293)

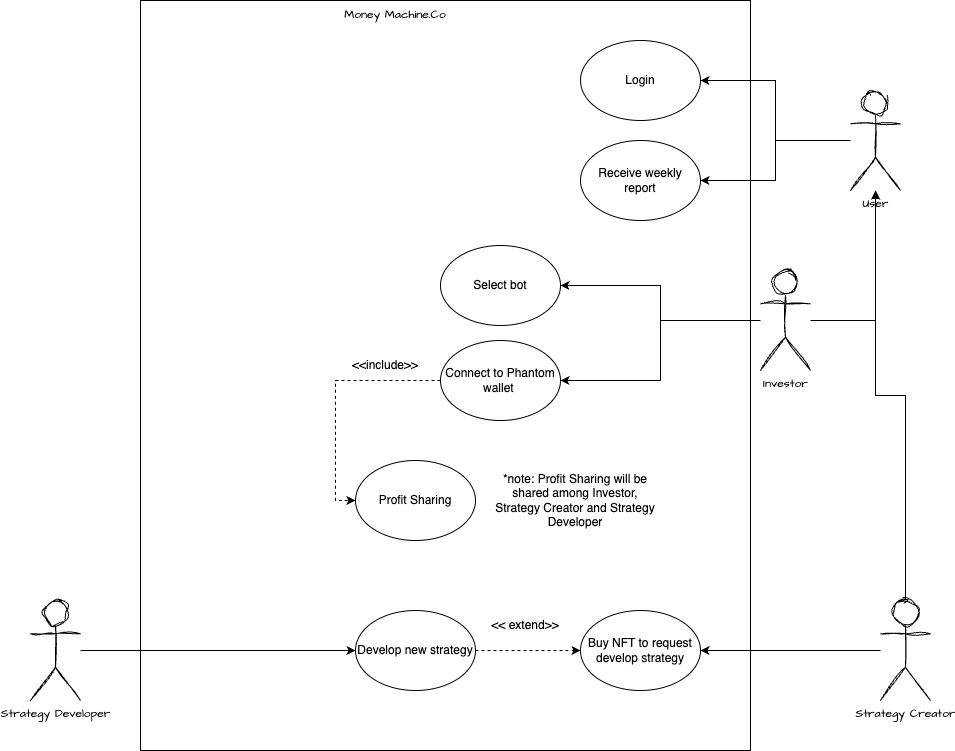
[**5.1 Invested Amount** 11](#_Toc176091294)

**Process Flow Diagram**



****

**Use Case Diagram**



**Introduction**

Money Machine is an innovative solution designed to simplify and streamline the process of automated cryptocurrency trading through a user-friendly Telegram bot. As the digital asset market continues to evolve, the demand for accessible and efficient trading tools has grown significantly. Money Machine bridges the gap between complex trading platforms and everyday users by providing a seamless interface that integrates directly with existing bot trading websites and the Bybit API.

This project is developed with a. focus on usability, security, and performance, ensuring that users can easily register, log in, and manage their trading activities without needing in-depth technical knowledge. By leveraging the power of the Solana blockchain and the flexibility of the Bybit API, Money Machine offers a robust platform for users to optimize their investment strategies with minimal effort.

**Objective**

The primary objective of the Money Machine project is to create a fully functional Telegram bot that empowers users to participate in automated cryptocurrency trading with ease and confidence. Specifically, the project aims to:

1. Simplify User Onboarding: Develop an intuitive registration and login system that allows users to quickly access the bot’s features without unnecessary complexity.
2. Automated Bot Selection and Investment: Provide a straightforward process for users to select trading bots and allocate their investments, making automated trading accessible to a broader audience.
3. Enhance User Experience: Ensure the bot delivers a seamless and engaging experience, with real-time feedback and easy-to-navigate commands.
4. Promote Adoption of Automated Trading: Encourage wider adoption of automated trading strategies by lowering the barriers to entry and providing a reliable, user-friendly tool for managing investments
5. **Registration**

|  |  |
| --- | --- |
| **1.1 Registration** | |
| **Description** | The registration module is designed to onboard new users to the Money Machine platform via a simple and secure process |
| **Path** | - |
| **Business Rules** | **Start Registration**   * When a user interacts with the Telegram bot for the first time, they are prompted to begin the registration process.   **User Input Collection**   * **Username**: The bot requests the user to provide a unique username * **Password**: The bot prompts the user to set a password. This password should be stored securely, preferable hashed. * **Email Verification**: The bot will be sending an OTP (One-Time-Password) for verification purposes. An OTP will be sent to the email, which the user must enter in the bot to complete the process.   **User Validation**   * The bot checks the availability of the username and validates the password for complexity.   **Confirmation**   * Upon successful registration, the bot confirms the user’s registration and prompts them to log in.   **Error Handling**   * If username is already taken, the bot prompts the user to choose another. * If the password does not meet the complexity requirements, the bot will ask the user to enter a stronger password. * If email verification fails, the bot will allow the user to resend the OTP. |

1. **Login**

|  |  |
| --- | --- |
| **2.1 Login** | |
| **Description** | The login module allows registered users to securely access their accounts and manage their activities through the Telegram bot. |
| **Path** | - |
| **Business Rules** | **Login Prompt:**   * Users are prompted to log in by providing their username and password.   **Authentication:**   * The bot verifies the username and password against the stored credentials in database.   **Session Management:**   * Upon successful authentication, the bot initiates a user session, allowing them to access the bot’s features without repeatedly logging in. * Sessions may expire after a certain period of inactivity, requiring re-authentication   **Error Handling**   * If the username or password is incorrect, the bot provides an error message and prompts the user to retry. * After multiple failed login attempts, the bot may lock the account temporarily for security reasons. |

1. **Link Bybit API**

|  |  |
| --- | --- |
| **3.1 Link Bybit API** | |
| **Description** | The link Bybit API module allows users to securely connect their Bybit trading account to the Money Machine bot. |
| **Path** | - |
| **Business Rules** | **API Key Collection:**   * The bot prompts the user to enter their Bybit API Key and API Secret. * The user is also informed about the required (e.g., trading permission and related permissions) that the API key should have.   **API Key Validation:**   * The bot sends a test request to the Bybit API to verify the validity and permission of the API Key.   **Secure Storage:**   * The API key and secret are stored securely, ensuring they are not exposed or misused (Key will be encrypted)   **Confirmation:**   * Once linked successfully, the bot confirms the connection and provides access to the bot trading features**.**   **Error Handling**   * If the API key is invalid or permissions are insufficient, the bot will notify the user and prompt them to re-enter the correct information. * Bot will be handling network or API errors gracefully, informing the user and allowing them to retry. |

1. **Bot Selection**

|  |  |
| --- | --- |
| **4.1 Bot Selection** | |
| **Description** | The module allows users to select a trading bot from the available options and specify the amount they wish to invest. |
| **Path** | - |
| **Business Rules** | **Bot Listing:**   * The bot prompts retrieve and display a list of available trading bots from the connected bot trading website.   **Bot Selection:**   * The user selects a bot from the list by interacting with the bot’s menu or typing the bot’s identifier * The bot confirms the selected trading bot and provides a brief summary or its details.   **Phantom Wallet Connection:**   * After bot selection, the user is prompted to connect their Phantom wallet. * The bot initiates a connection to the user’s Phantom wallet to verify the token balance   **Token Balance Verification:**   * The bot checks the user’s token balance in the Phantom wallet to ensure it is sufficient to cover the required investment. * If the balance is insufficient, the bot proceeds with the next steps. * If the balance is insufficient, the bot notifies the user and offers option such as selecting a different bot or adding funds to their wallets.   **Error Handling**   * If the selected bot is unavailable or invalid, the bot informs the user and prompts them to select another. * If there is an issue connecting to the Phantom wallet, the bot provides an error message and allows the user to try. * If the token balance is insufficient, the bot notifies the user and provides instructions on how to resolve the issue. |

1. **Invested Amount**

|  |  |
| --- | --- |
| **5.1 Invested Amount** | |
| **Description** | The module allows users to specify the amount they wish to invest in the previously selected trading bot |
| **Path** | - |
| **Business Rules** | **Investment Prompt:**   * After selecting a bot, the user is prompted to enter the amount they wish to invest.   **Amount Validation:**   * The bot validates the entered amount against the user’s available balance, minimum investment requirements, and any other relevant constraints (e.g., maximum allowed investment)   **Confirmation:**   * Once the investment amount is validated, the bot confirms the entered amount and provides a summary of the investment details, including the selected bot and the amount.   **Execution:**   * Upon user confirmation, the bot triggers the necessary API calls to execute the investment. * The bot confirms the successful initiation of the investment and provides a transaction summary.   **Error Handling**   * If the investment amount is insufficient or exceeds available funds, the bot notifies the user and requests a valid amount. * The bot should handle API errors (e.g., insufficient balance, API downtime) and provide appropriate feedback to the user. |