## 



T-GUILD (Blockchain based pre-sale)

05.12.2022

**Syed Jehanzaib Faseeh 2019192**

**Syed Muhammad Shahzeb Abbas 2019373**

**Muhammad Umais Chaudhary 2019386**

**Taha Ahmed Siddiqui 2019509**

**Supervisor: Mr. Ahsan Shah**

### Revision History:

|  |  |  |
| --- | --- | --- |
| ***Revision History*** | ***Date*** | ***Comments*** |
| 1.00 | 20-11-2022 | Functional Requirements not defined properly. |
| 2.00 | 25-11-2022 | Use cases are not defined properly. |

*Document Approval:*

The following document has been accepted and approved by the following:

|  |  |  |
| --- | --- | --- |
| ***Signature*** | ***Date*** | ***Name*** |
|  |  |  |
|  |  |  |

## **List of Contents**

[**LIST OF FIGURES**](#_30j0zll) **4**

[**LIST OF TABLES**](#_30j0zll) **5**

* 1. [**INTRODUCTION 6**](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.1fob9te)
     1. [PURPOSE 6](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.3znysh7)
     2. [PRODUCT SCOPE 6](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.2et92p0)
  2. [**OVERVIEW 7**](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.tyjcwt)
  3. [THE OVERALL DESCRIPTION 7](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.3dy6vkm)
  4. [PROBLEM STATEMENT 7](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.1t3h5sf)
  5. [PRODUCT PERSPECTIVE 8](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.4d34og8)
  6. [PRODUCT FUNCTIONS 8](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.2s8eyo1)
  7. [USER CHARACTERISTICS 9](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.17dp8vu)
  8. [CONSTRAINTS 9](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.3rdcrjn)
  9. [ASSUMPTIONS AND DEPENDENCIES 9](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.26in1rg)
  10. [**STATE OF THE ART 9**](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.lnxbz9)
  11. [LITERATURE REVIEW 9](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.35nkun2)
  12. [EXISTING SYSTEM 10](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.1ksv4uv)
  13. [**USER/SYSTEM REQUIREMENTS 11**](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.44sinio)
  14. [EXTERNAL INTERFACE REQUIREMENTS 11](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.2jxsxqh)
      1. [*User Interfaces 11*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.z337ya)
      2. [*Hardware Interfaces 11*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.3j2qqm3)
      3. [*Software Interfaces 11*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.1y810tw)
      4. [*Communication Interfaces 11*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.4i7ojhp)
  15. [**FUNCTIONAL REQUIREMENTS 12**](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.2xcytpi)

[5.1 FUNCTIONAL REQUIREMENTS WITH TRACEABILITY INFORMATION 12](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.1ci93xb)

* 1. [**NONFUNCTIONAL REQUIREMENTS & SOFTWARE SYSTEM ATTRIBUTES 24**](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.3whwml4)
     1. [*Security 24*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.2bn6wsx)
     2. [*Scalability 24*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.qsh70q)
     3. [*Availability 24*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.3as4poj)
     4. [*Accuracy 24*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.1pxezwc)
     5. [*Usability 24*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.49x2ik5)
     6. [*Data Integrity 24*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.2p2csry)
     7. [*Maintainable 24*](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.147n2zr)

[6.2 PERFORMANCE REQUIREMENTS 25](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.3o7alnk)

* 1. [**PROJECT DESIGN/ARCHITECTURE 26**](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.23ckvvd)
  2. [USE CASE VIEW 26](https://docs.google.com/document/d/1qq-d891Gxw_dSvbbczVEbCwVag6HhWxL/edit#heading=h.ihv636)
  3. LOGICAL VIEW 27
  4. DEPLOYMENT VIEW 28
  5. PROCESS VIEW 29
  6. PHYSICAL VIEW 30
  7. SEQUENCE DIAGRAM 31
  8. USER INTERFACE 35

## **List of Figures**

Figure 1 Use Case View 26

Figure 2 Logical View - Class Diagram 27

Figure 3 Deployment View - Component Diagram 28

Figure 4 Process View 29

Figure 5 Physical View 30

Figure 6 Sequence Diagram for Wallet Connection 31

Figure 7 Sequence Diagram for Token Creation 32

Figure 8 Sequence Diagram for Pre-Sale Creation 33

Figure 9 Sequence Diagram for Pre-Sale Browsing 34

# List of Tables

Table 1: Terms used in this document and their description 07

Table 2: Table for Functional Requirement 1 12

Table 3: Table for Functional Requirement 2 13

Table 4: Table for Functional Requirement 3 13

Table 5: Table for Functional Requirement 4 14

Table 6: Table for Functional Requirement 5 15

Table 7: Table for Functional Requirement 6 16

Table 8: Table for Functional Requirement 7 17

Table 9: Table for Functional Requirement 8 18

Table 10: Table for Functional Requirement 9 19

Table 11: Table for Functional Requirement 10 20

Table 12: Table for Functional Requirement 11 21

Table 13: Table for Functional Requirement 12 22

Table 14: Table for Functional Requirement 13 23

1. **INTRODUCTION**

There has been a lot going on the Blockchain in recent years. Many have described it as the most disruptive technology of the decade. In particular, the financial markets could be the hardest hit.

The technology is being applied in many industries such as healthcare, medicine, insurance, smart properties, automotive and even governments.

One of the great advances of blockchain technology is that digital items have certain physical properties, such as not being replicable or usable only in one place at a time. This means you can create virtual entities that can be traded digitally while still retaining their value in physical form. These objects are called tokens.

Creating a personal token is a task that cannot be achieved by a layman.

T-guild is the most convenient and hassle-free pre-sale platform that offers user to create their custom tokens without any blockchain development expertise.

It will offer an easy-to-use method to create and launch their own tokens greater efficiency, convenience, and, most importantly with a single click that ensures your personalized token has created.

Conventionally, people hire a blockchain expert to create a personal token because of the intricacies of the blockchain technology but now T-guild is a one stop solution to all of this.

* 1. **PURPOSE**

The primary goal of the T-guild is to create custom tokens for the user who does not have any prior experience or without any blockchain development expertise.

By supplying suitable credentials at the time of creation, to offer an authenticated and approved platform for users such as blockchain enthusiasts.

The proposed platform will let users create personal tokens, it will also provide users with token ICO (initial coin offering), Users will have options to have different pairing tokens to add liquidity with pre-sale token.

* 1. **PRODUCT SCOPE**

Blockchain is a relatively new technology that is rapidly expanding, bringing with it a slew of issues. This project provides a simple yet effective solution to a problem encountered by many enthusiasts interested in creating their own tokens.

Pakistan has a lot of potential and unemployment ratio is rising every year. One of the aims behind the making of this platform is to provide more work opportunities to the increasing demand of Blockchain market and building a community leading to new economic market.

**Table 1: Terms used in this document and their description**

|  |  |
| --- | --- |
| **Name** | **Description** |
| **SRS** | Software Requirement Specifications |
| **UC** | Use case |
| **SQ** | Sequence Diagram |
| **UI** | User Interface |
| **FAQ** | Frequently asked questions |
| **API** | Application Programming Interface |
| **REST** | Representational State Transfer |
| **HTTP** | Hypertext Transfer Protocol |
| **FR** | Functional Requirements |
| **ICO** | initial coin offering |

## **OVERVIEW**

## **THE OVERALL DESCRIPTION**

A free of cost platform which provide a hassle-free token creation where a user will come and connect his wallet and will enter his credentials and name of token and within a few minutes he/she will be able to create his own personalized token. From this token they will be able to make their own community known by their token name.

From the user’s end, it benefits them based on better prices, time management, experts in the field, recommendations from the community, and hassle-free environment.

Hence our project focuses to provide an equal opportunity for the users who lack prior experience of token creation and previously they must pay blockchain experts a subtle amount of money to create their token.

Hence our project focuses on providing a user-friendly platform for people to create and launch their own tokens and facilitating blockchain enthusiasts to further the usage of blockchain technology.

* 1. **PROBLEM STATEMENT**

The growth in blockchain technology has expanded the call for customized token advent; but people want a blockchain expert for this due to complexities in blockchain era, and there is continually an issue of launching their tokens.

* 1. **PRODUCT PERSPECTIVE**

This Product is supposed to be a blockchain based web 3.0 web application. Since it is a web-based it will be implementing a client-server model.

The following are the main features that are included in T-Guild:

* Wallet Connection: The website allows the user to connect his wallet to the website and the smart contract coded here will recognize the user from the blockchain database.
* Number of users being supported by the system: Though the number of users is not precisely mentioned but the system is able to support many users at a single instance.
* Token Presale: The dashboard will contain all the tokens available for the presale, the tokens will be sold at their set prices.
* Search: Search will be the basic local search engine based on keywords which will find specific tokens from the presale dashboard.
* Token Creation: Token creation will be done when the user wants to create his own token and give all the details related to its hardcap, softcap and valuation.

## **PRODUCT FUNCTIONS**

* The user should be able to connect his wallet e.g., Metamask, Ronin.
* The user should select the correct for creating a token. (bsc)
* The user should be able to create token after giving few details about the token and its presale.
* The user should be able to choose token to create presale.
* The user should be able to create presale with the selected token.
* The user should be able to view presales in dashboard
* The user should be able to see detailed view of a presale present in the dashboard.
* The user should be able to buy presale token once the presale is started.
* The owner of the token should be able to finalise presale, conditions: (if hard cap is reached or (presale time ended and soft cap is reached))
* The user should be able to claim presale tokens, after presale is ended
* The owner of the presale should be able to cancel presales.
* The user should be able to withdraw their funds if presale is cancelled by the owner of that token.
* The owner should be able to view presale contributors (people who bought presale token)

## **USER CHARACTERISTICS**

Users for this system are divided into three categories:

1. Users who want to create their token
2. Users who want to create a presale of their token
3. Users who will buy presale tokens

All users from the categories mentioned above should own a Meta-Mask wallet, have general knowledge of using a personal computer or a smartphone, and be able to understand the basic English language.

## **CONSTRAINTS**

Web3 has evolved into a movement that anticipates a fundamentally different era of the internet, with a concentration on a more decentralized and democratized internet. But due to blockchain being a relatively new technology, this system is still dependent on some centralized platforms that include BSC scan and Meta-Mask. Furthermore, this system will have a poor UX/UI, as there is a tradeoff between the system being more secure and less convenient.

## **ASSUMPTIONS AND DEPENDENCIES**

* Assumptions:
  + Users should have enough funds in their wallets to perform transactions
  + The software uses Graphical User Interfaces
* Dependencies:
  + The Basic hardware and software to run the Web3 app

## **STATES OF THE ART**

## **Literature Review**

T-GUILD is a platform based on the blockchain technology to provide people with the option of creating their own personalized and custom tokens; ultimately putting it on the pre-sale market. The platform removes the conventional way of hiring a blockchain developer to create token due to complexity of the technology. It also gives people the solution to launch their tokens in the wider market and crypto community.

Conventionally people have been hiring blockchain developers to create their tokens. However, this is a highly expensive process with developers charging hefty amounts. It has always been a time-consuming process, both in terms of making the smart contracts behind the development of the token as well as finding a competent developer to fulfill these needs.

There is also an issue of how to launch the token in the blockchain community, especially for a beginner who has no understanding of the workings behind the market.

We offer a simple yet efficient solution to create tokens by entering some information, without any prior blockchain development knowledge. Moreover, it will be a place to launch it as a pre-sale to the wider community.

## **Existing System**

It is a blockchain based platform that facilitates the creation of customized tokens for anyone who wants it, making token creation easy and accessible for people with no prior development and programming knowledge.

* **TokenMaker**

TokenMaker is a token creation platform, based on blockchain technology, making use of open-source smart contracts which are secure and reliable. You can choose the total number of tokens you can create, enforce the hard cap etc.

* **MoonDeploy**

It is a blockchain based platform that facilitates the creation of customized tokens for anyone who wants it, making token creation easy and accessible for people with no prior development and programming knowledge.

* **PinkSale**

PinkSale offers token creation by navigating through a simple and user-friendly terminal, with just a few clicks of a button. It offers the opportunity to list your token on a pre-sale market as well.

**Differences:**

* T-Guild offers custom token creation
* T-Guild offers listing personalized tokens on pre-sale
* T-Guild offers different pairing tokens to add liquidity for pre-sale

1. **USER/SYSTEM REQUIREMENTS:**
   1. **USER INTERFACES:**

We will be having a UI which will be user-friendly and easy to navigate for a novice who wants to create custom tokens. There will be a starting landing page which will lead to a page where you will be able to connect to your crypto wallet. It will further ask you about the information needed for creating the token and its relevant specifications.

UI also consists of a browsing page for going through various tokens listed on the pre-sale to buy them. One can also view the details of the respective tokens as well.

**4.2. HARDWARE INTERFACES**

This Application will not have any hardware interfaces. Devices running both iOS and Android are supported. The web app will also be supporting all other OS including Windows, Mac, Linux.

**4.3. SOFTWARE INTERFACES**

The web page first urges you to connect your wallet so that you can connect your metamask or ronin wallet to the website and access your funds to buy or list tokens on the presale dashboard. Once the wallet is connected the user can go and buy the tokens from the presale dashboard. If the user wants to create and list his token for the presale, he can go and create his token from the create token button on the dashboard and enter certain details about their token, these details include presale rate, listing rate, softcap, hardcap, min buy, max buy, presale start date, presale end date. Once the token is created the user can view his token listed on the dashboard with the other listed tokens. The user can buy the tokens as well buy funding them through their connected wallet.

**4.4. COMMUNICATION INTERFACES**

Users communicate with software using a pointing device e.g., a mouse and keyboard. The main communications are always held between the app and the decentralized ledger of the blockchain. This will further include sending and receiving transactions requests on the crypto networks e.g., Blockchain or Ethereum. Further the distribution will be done on the blockchain ledger.

## **FUNCTIONAL REQUIREMENTS**

**5.1 FUNCTIONAL REQUIREMENTS WITH TRACEABILITY INFORMATION**.

**5.1.1 USER SHOULD BE ABLE TO CONNECT HIS WALLET (FR1)**

When the user arrives on our website, he should be able to connect his wallet, e.g., Meta mask, ronin. Connecting his wallet will give him access to his funds from which he can invest in the presale tokens listed by other users.

**Table 1: Table for Functional Requirement 1**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR1 | | | **Requirement Type** | | | Functional | | **Use Case #** | |  |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | N/A | | | | | | | | | | |
| **Description** | Connecting the wallet will grant the user access to his funds | | | | | | | | | | |
| **Rationale** | Prevention against illegal transaction | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | User can access to dashboard | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

**5.1.2. CORRECT NETWORK SHOULD BE SELECTED. (BSC)**

As a next step, choose the network on which you wish to create the token presale (**Note**: the network should be the same as the one the token will be created ).

**Network will only be Binance(BSC) as smart contract is on Binance Blockchain.**

**Table 2: Table for Functional Requirement 2**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR2 | | | **Requirement Type** | | | Functional | | **Use Case #** | |  |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR1 | | | | | | | | | | |
| **Description** | Prior to creating a token, the client must first select the network. | | | | | | | | | | |
| **Rationale** | Prevention against unidentified network transactions. | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | NA | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

**5.1.3. USER SHOULD BE ABLE TO CREATE TOKEN(FR3)**

A token may be made. Users need to define its token properties. The properties of the token include mentioning their network type, their value which the owner wants the token to be sold on, a hard cap and a soft cap.

**Table 3: Table for Functional Requirement 3**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR3 | | | **Requirement Type** | | | Functional | | **Use Case #** | |  |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR2 | | | | | | | | | | |
| **Description** | System will create your personalized token along with their estimated prices. | | | | | | | | | | |
| **Rationale** | Prevention from unnecessary hindrance of making a token. | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | Users can refer to the YouTube tutorials. | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

**5.1.4. USER SHOULD BE ABLE TO CHOOSE TOKEN TO CREATE PRESALE(FR4)**

· Step 1: Input token sale contract address.

· Step 2: Review & edit token presale parameters.

· Step 3: Add addresses to whitelist investors.

· Step 4: Review & manage whitelist addresses.

· Step 5: Set withdrawals parameters if applicable.

· Step 6: Liquidity & Listing Rate.

· Step 7: Presale Start and End time.

**Table 4: Table for Functional Requirement 4**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR4 | | | **Requirement Type** | | | Functional | | **Use Case #** | | 1 |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR3 | | | | | | | | | | |
| **Description** | System will provide a list of settings/steps to create presales along with their estimated prices. | | | | | | | | | | |
| **Rationale** | Prevention from unnecessary tokens. | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | Users can view the you tube tutorial. | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

**5.1.5. USER SHOULD BE ABLE TO CREATE PRESALE WITH THE SELECTED TOKEN(FR5)**

A presale is created for every specific token a user has recently created. It will have token Address, Name, Symbols, Token Hardcap, Token Softcap, Token Supply, Presale start and end dates and Presale currency which will be Binance(BNB).

**Table 5: Table for Functional Requirement 5**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR5 | | | **Requirement Type** | | | Functional | | **Use Case #** | |  |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR4 | | | | | | | | | | |
| **Description** | A presale is created for every specific token a user has recently. | | | | | | | | | | |
| **Rationale** | NA | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | Users can view the you tube tutorial. | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

**5.1.6. USER SHOULD BE ABLE TO VIEW PRESALES IN DASHBOARD(FR6)**

It will show all necessary presale info about presale goal, presale start time , presale end time, presale type, minimum and maximum buy, and show an option to claim my token.

**Table 6: Table for Functional Requirement 6**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR6 | | | **Requirement Type** | | | Functional | | **Use Case #** | | 1 |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR5 | | | | | | | | | | |
| **Description** | System will provide all necessary information of presale to the customer. | | | | | | | | | | |
| **Rationale** | Prevention from unnecessary Info of presale. | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | NA | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

**5.1.7. USER SHOULD BE ABLE TO SEE A DETAILED VIEW OF A PRESALE (FR7)**

The detailed view of the presale would mostly consist of all the tokens currently present for the presale, it would also be containing the value of the tokens on which the tokens are being sold, Moreover, the presale status would also be present there showing which presale is live now and which has expired and crypto network on which the token was created would also be there.

**Table 7: Table for Functional Requirement 7**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR7 | | | **Requirement Type** | | | Functional | | **Use Case #** | |  |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR1 | | | | | | | | | | |
| **Description** | The system will be providing a detailed view of the presale, once the client connects his wallet and goes to the presale center | | | | | | | | | | |
| **Rationale** | Prevention against the hassle of finding presale tokens. | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | Users can view the tokens on the presale | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

**5.1.8. USER SHOULD BE ABLE TO BUY PRESALE TOKENS, ONCE PRESALE IS LIVE (FR8)**

Once the user can view the detailed presale table, he will be able to buy the tokens whose presale is live and set a value for the token when released in the decentralized market. The user will also be able to view other presales which are live, and he can buy multiple tokens.

**Table 8: Table for Functional Requirement 8**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR8 | | | **Requirement Type** | | | Functional | | **Use Case #** | |  |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR7 | | | | | | | | | | |
| **Description** | User can buy the tokens whose presale is live from the detailed view of presale | | | | | | | | | | |
| **Rationale** | Prevention against the buying of tokens on an expensive price when they are on sale. | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | User can buy tokens on presale. | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

**5.1.9. OWNER SHOULD BE ABLE TO FINALISE PRESALE, CONDITIONS: (IF HARDCAP IS REACHED OR (PRESALE TIME ENDED AND SOFTCAP IS REACHED)) (FR9)**

A token listed on the presale platform is only successful in qualifying for the sale of that token when either the hard cap is reached, or the soft cap requirement is fulfilled, and the presale ends otherwise the collected funds are refunded to the participants wallets.

**Table 9: Table for Functional Requirement 9**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR9 | | | **Requirement Type** | | | Functional | | **Use Case #** | |  |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR8 | | | | | | | | | | |
| **Description** | User will be given an option to list his token when either the hard cap is reached, or the presale time has ended. | | | | | | | | | | |
| **Rationale** | This is done to prevent the owner on losing potential participants if the hard cap is reached. | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | Owners can list their tokens for sales, analyzing the presales | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

**5.1.10. USER SHOULD BE ABLE TO CLAIM PRESALE TOKENS, AFTER PRESALE IS ENDED (FR10)**

When the soft cap/hard cap has been reached and the owner decides to finalize his presale, you can claim your tokens once the “Claim token” button is available. The amount of tokens that you can claim will also be shown next to the “Claim token” button.

**Table 10: Table for Functional Requirement 10**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR10 | | | **Requirement Type** | | | Functional | | **Use Case #** | | 1 |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR9 | | | | | | | | | | |
| **Description** | System will provide you with a claim token button once the presale is ended. | | | | | | | | | | |
| **Rationale** | Prevention from unnecessary loss of tokens when presale ends. | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | User can view the you tube tutorial. | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |

**5.1.11. OWNER OF PRESALE SHOULD BE ABLE TO CANCEL PRESALE(FR11)**

The owner of the presale always has the option to cancel the presale at any moment, a cancel button will always be available and once the owner cancel’s the presale, all the participants will receive their funds back in their wallets.

**Table 11: Table for Functional Requirement 11**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR11 | | | **Requirement Type** | | | Functional | | **Use Case #** | |  |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR5 | | | | | | | | | | |
| **Description** | A presale can be cancelled whenever the owner wants to cancel it. | | | | | | | | | | |
| **Rationale** | It is to prevent the sale of any unwanted token | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** |  | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |

**5.1.12. USER SHOULD BE ABLE TO WITHDRAW THEIR FUNDS, IF PRESALE IS CANCELLED (FR12)**

Once the owner of the presale cancels the presale, the users who have already bought the token on the presale will be returned with their investment in their respective wallet.

**Table 12: Table for Functional Requirement 12**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR12 | | | **Requirement Type** | | | Functional | | **Use Case #** | | 1 |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR11 | | | | | | | | | | |
| **Description** | System will automatically refund the users with their investments if the owner decides to cancel the presale | | | | | | | | | | |
| **Rationale** | Prevention from unnecessary loss of funds of the user. | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | NA | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

**5.1.13. OWNER SHOULD BE ABLE TO VIEW PRESALE CONTRIBUTORS (FR13)**

Once the presale of a certain token goes live and users start to buy the tokens in the presale, the owner of the token can view the list of contributors by clicking on the details button available on the presale dashboard against that specific token.

**Table 13: Table for Functional Requirement 13**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Requirement ID** | FR13 | | | **Requirement Type** | | | Functional | | **Use Case #** | | 1 |
| **Status** | ***New*** | X | ***Agreed-to*** | | | - | ***Baselined*** | - | ***Rejected*** | - |  |
| **Parent Requirement #** | FR8 | | | | | | | | | | |
| **Description** | Once the user buys presale tokens, the owner can view the contributor from the details of the token | | | | | | | | | | |
| **Rationale** | Prevention from the loss of track of sales of the token. | | | | | | | | | | |
| **Source** |  | | | | | | **Source Document** | | - | | |
| **Acceptance/Fit Criteria** | NA | | | | | | | | | | |
| **Dependencies** |  | | | | | | | | | | |
| **Priority** | ***Essential*** | | X | | ***Conditional*** | | - | ***Optional*** | - |  | |
| **Change History** |  | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |

## **6 NONFUNCTIONAL REQUIREMENTS & SOFTWARE SYSTEM ATTRIBUTES**

Non-functional requirements or NFRs are a set of specifications that describe the system’s operation capabilities and constraints and attempt to improve its functionality. These are basically the requirements that outline how well it will operate including things like speed, security, reliability, data integrity, etc.

## **Security**

All data inside the system or its part should be protected against malware attacks or unauthorized access. Since it is a blockchain based application the security breach risk is very low since every transaction is verified by every other node present on the blockchain ledger

## **Scalability**

## Scalability refers to a network's capacity to sustain larger transaction throughput and is a key criterion in blockchain networks. As a result, increasing scalability would be increasing the number of computations per second since transactions can vary in complexity and cost.

## **Availability**

The servers should be up 100% of the time (24/7). Data availability is the guarantee that the block proposer published all transaction data for a block and that the transaction data is available to other network participants. Hence our website might go down in some unforeseen system crashes otherwise the blockchain network will always be available.

## **Accuracy**

Using a blockchain, however, does not ensure data accuracy of the entered data on-chain, by design. Nevertheless, blockchain specifically protect against manipulation of data, which is immutable once it goes on the shared ledger.

## **Usability**

The webpage’s interface must be user-friendly and easy to use. The app should be accessed easily through:

* Windows / Mac OS / Linux
* Android/ IOS

## **Data Integrity**

Once any transaction is done on any blockchain network including ours, the transaction is immutable and available to every node present on the blockchain ledger.

## **Maintainable**

The automated services may be unavailable for up to three hours if they become unavailable.

## **6.2 Performance Requirements**

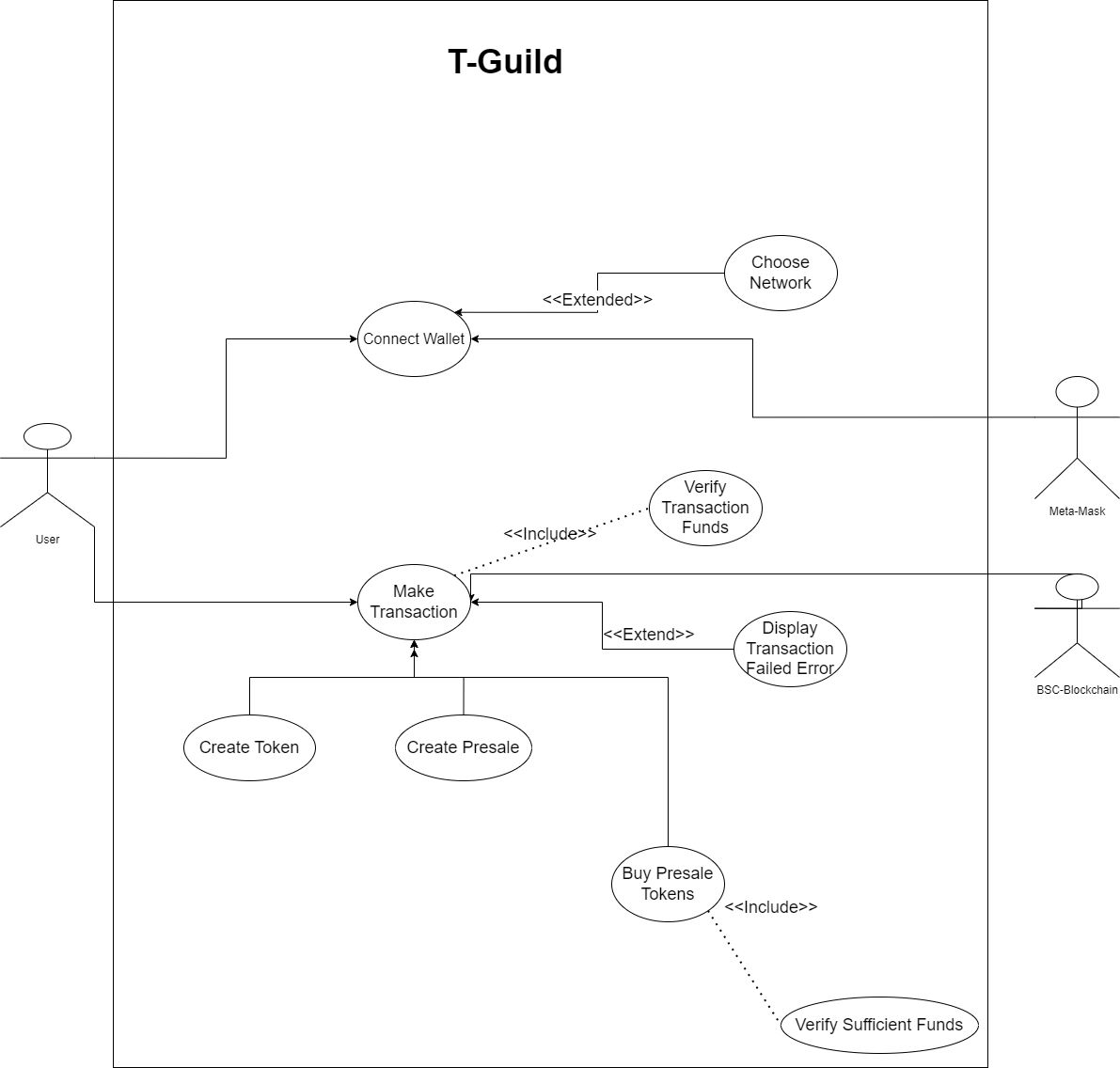
The product must be based on the web and run on a web server. The product will take some time to load at first, depending on the strength of the internet connection and the media from which it is run. This app will have a user-friendly interface for easy accessibility. We want to accomplish access in Pakistan in less than 6 seconds over a 4G network and an access time of fewer than 12 seconds for 3G (or any internet connection with a local telecommunication network's average speed). Since it is a blockchain based application taking too long connect might cause troubles in the transactions, so having a good internet is the key here. The performance will be determined by the client's/hardware customer's components.

# 7 Project Design/Architecture

* 4+1 ARCHITECTURE VIEW MODEL (mandatory for Software Projects)
  + Use Case View
  + Logical View:
  + Development View
  + Process View
  + Physical View
  + User Interface Design

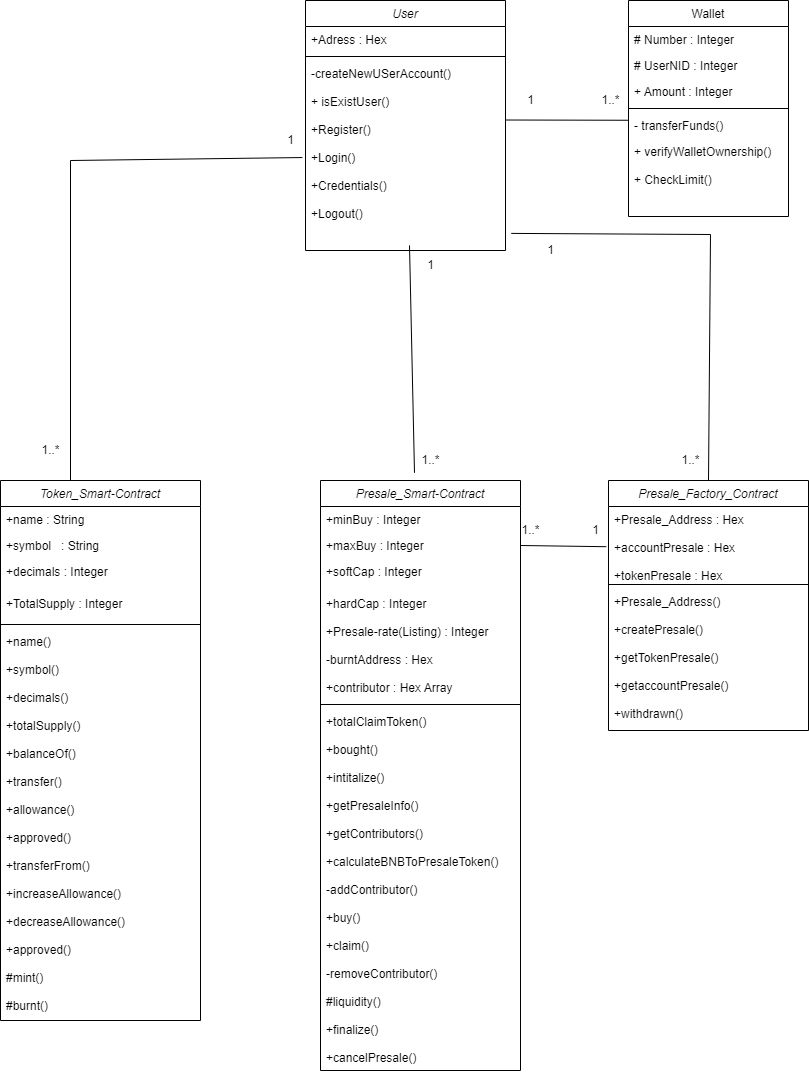
## **Use Case View**

* + 1. **UC-1:**



## **Logical View**

**1. Class Diagram**



## **Deployment View**

**1. Component Diagram**

Diagram

Description automatically generated

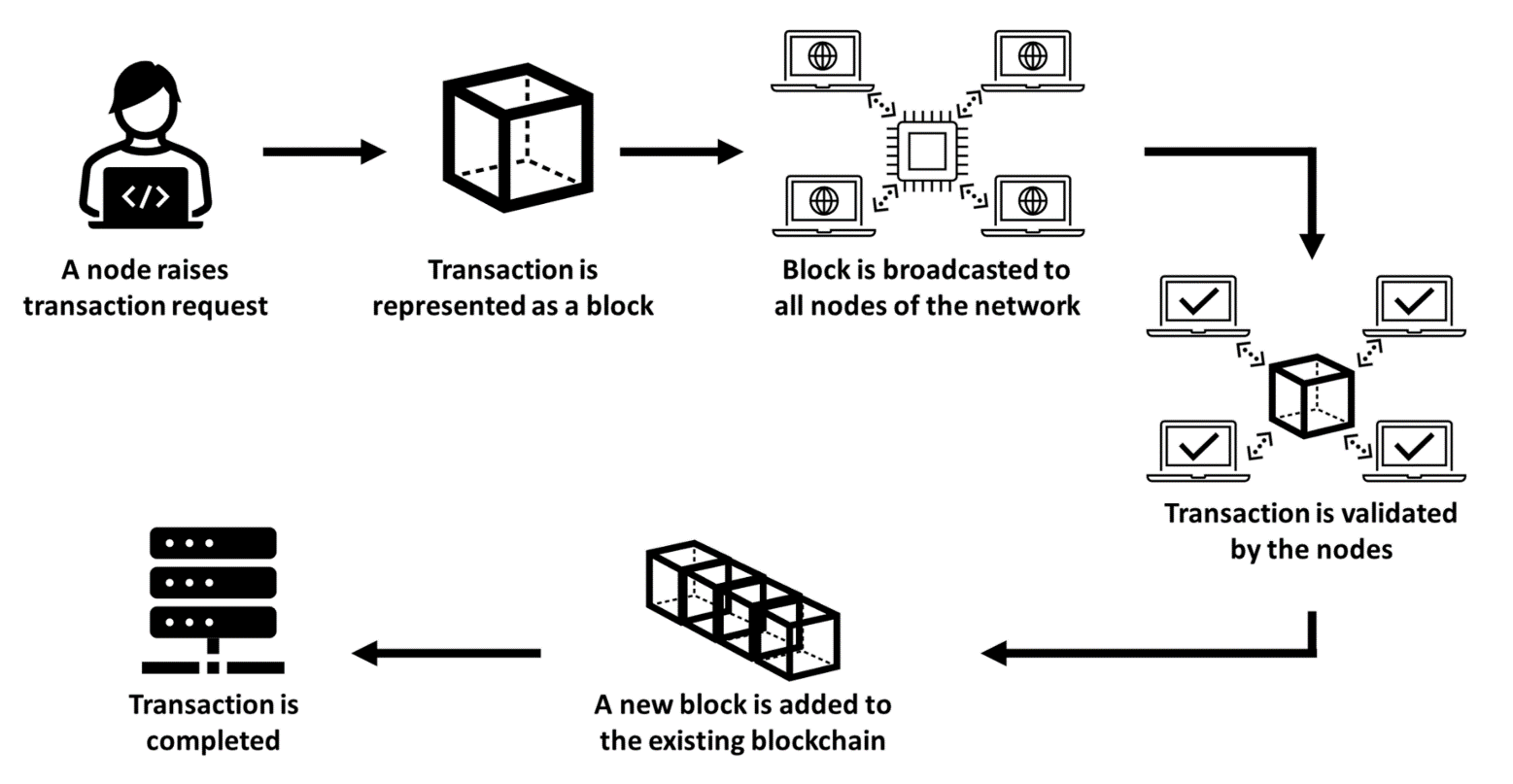
## **Process View**

**AC-1:**

A screenshot of a computer

Description automatically generated

## **Physical View**



## **Sequence Diagram**

## **Sequence Diagram for Wallet Connection**

Diagram

Description automatically generated

## **Sequence diagram for token Creation**

Diagram

Description automatically generated

## **Sequence Diagram for Pre-Sale Creation**

Diagram

Description automatically generated

## **Sequence Diagram for Browsing Pre-Sales**

Diagram

Description automatically generated

* 1. **User Interface**

***Main Landing Page Token Creation Information***

**Text

Description automatically generated Graphical user interface, application

Description automatically generated**

**Pre-Sale Creation UI**

***Token Pre-Sale Creation Information Token Pre-Sale Creation Information (2)***

**Graphical user interface, text, application, email

Description automatically generated** Text, letter

Description automatically generated

**Buying Pre-Sale UI**

***Browsing Token Pre-Sales Viewing Details of Token Pre-Sales***

**Graphical user interface, text, application, chat or text message

Description automatically generated** Graphical user interface, text, application

Description automatically generated